Logic Programs with Ordered Disjunction: First-order Semantics and Expressiveness

Yan Zhang
Artificial Intelligence Research Group
University of Western Sydney, Australia
The Key Laboratory of Computer Science
Institute of Software, Chinese Academy of Sciences, China

Vernon Asuncion
Intech Solutions Pty Ltd, Australia

Heng Zhang
College of Computer Science and Technology
Huazhong University of Science and Technology, China

Abstract

Logic programs with ordered disjunction (LPODs) generalize normal logic programs by combining alternative and ranked options in the heads of rules. It has been showed that LPODs are useful in a number of areas including game theory, policy languages, planning and argumentation. In this paper, we extend propositional LPODs to the first-order case. We study the first-order (FO) LPODs from both semantics and expressiveness perspectives. We first provide two equivalent semantics: one is defined via a second-order formula as a counterpart of the second-order definition of the general stable model theory; and the other is defined based on a progression that naturally represents the reasoning procedure of the underlying program. We then consider how LPODs may be translated into first-order sentences/theories, and argue that results on this aspect will provide an important basis for computing stable models of LPODs. To understand the essential expressiveness of LPODs, we further prove that almost positive LPODs precisely capture first-order normal logic programs, which indicates that ordered disjunction itself and constraints are sufficient to represent negation as failure.
1. Introduction

Logic programs with ordered disjunction (LPODs) generalize normal logic programs by combining features of qualitative choice logic so that alternative and ranked options may be explicitly expressed in the heads of rules (Brewka, Benferhat, & Berre, 2002; Brewka, 2002). An LPOD contains a finite set of rules of the form

\[
\alpha_1 \times \cdots \times \alpha_k \leftarrow \beta_1, \ldots, \beta_l, \not\gamma_1, \ldots, \gamma_m, \tag{1}
\]

where \(\alpha_i, \beta_j, \gamma_h\ (1 \leq i \leq k, 1 \leq j \leq l, 1 \leq h \leq m)\) are propositional atoms. Intuitively, rule (1) says that when the body is satisfied, then whenever it is possible, derive \(\alpha_1\), otherwise if possible, derive \(\alpha_2\), and so on. The semantics of an LPOD is defined in terms of the stable models of so-called split programs of the underlying LPOD.

Let us consider a simple program \(\Pi_1\) from (Brewka, 2002):

\[
\begin{align*}
A \times B & \leftarrow \not C, \\
B \times C & \leftarrow \not D.
\end{align*}
\]

\(\Pi_1\) has the following four split programs:

\[
\begin{align*}
A & \leftarrow \not C & A & \leftarrow \not C \\
B & \leftarrow \not D & C & \leftarrow \not D, \not B \\
B & \leftarrow \not C, \not A & B & \leftarrow \not C, \not A \\
B & \leftarrow \not D & C & \leftarrow \not D, \not B.
\end{align*}
\]

Then the class of stable models of \(\Pi\) consists of all stable models of these four split programs, which is \(\{\{A, B\}, \{B\}, \{C\}\}\). Then by integrating proper preference relation among these stable models, the preferred stable models can be obtained for an LPOD.

There have been several extensions of LPODs: Karger et al (2008) extended LPODs by allowing both ordered and unordered disjunction in the heads of rules; Confalonieri et al (2010) recently defined a possibilistic semantics for LPODs in order to handle uncertainty; and Cabalar (2011) also proposed a direct translation from LPODs to normal logic programs via the logic of Here-and-There. It has been argued that LPODs provide a natural way to deal with preference in reasoning that are useful in various applications such as game theory, policy languages, planning and argumentation (Brewka, 2002; Cabalar, 2011; Confalonieri et al., 2010).

On the other hand, in recent years, Answer Set Programming (ASP) has been generalized to the first-order case so as to deal with various reasoning problems beyond Herbrand domains as well as to integrate aggregates (Ferraris, Lee, & Lifschitz, 2011). One challenging research along this direction is to establish proper logical and computational foundations for promoting useful functionalities in existing ASP paradigm to the first-order level. A number of topics in this aspect have been investigated, e.g., (Asuncion, Lin, Zhang, & Zhou, 2012; Asuncion, Zhang, & Zhou, 2014b; Lee & Meng, 2011; Babb & Lee, 2012). The major advantage of first-order ASP is that it provides a succinct declarative language, in which the underlying problem constraints (rules) may be completely separated from concrete problem instances, and hence more flexible for problem representation and modeling (Lin & Zhou, 2011).
In this paper, we study the semantics and expressiveness of LPODs on the first-order level. We make the following main contributions towards this topic:

1. Following the style of general stable model semantics (Ferraris et al., 2011), we define the stable model semantics for first-order LPODs via a classical second-order sentence, and show that this semantics is a correct lifting of Brewka’s original LPODs semantics to the first-order level. We then develop a progression semantics that is equivalent to the second-order based semantics but naturally represents the reasoning procedure of LPODs;

2. We propose two translations from LPODs to first-order sentences/theories on finite structures. The first translation is an extension of ordered completion for first-order normal logic programs (Asuncion et al., 2012), while the second one is defined based on first-order loop-formulas (Chen, Lin, Wang, & Zhang, 2006). We argue that study on such translations are important as they may be served as a computational basis for developing an efficient LPOD solver.

3. We address the complexity and expressiveness issues of LPODs. We show that LPODs data complexity is NP-complete, that remains true even if for positive LPODs. We also prove that almost positive LPODs precisely capture the full class of first-order normal logic programs, which indicates that ordered disjunction itself and constraints are sufficient to represent negation as failure.

4. We further consider preferred LPODs, and provide a logic characterization of such preference semantics, which again, generalizes the corresponding preferred propositional LPODs. We show that under certain conditions, Both LPODs and preferred LPODs satisfy a modular feature via splitting, which is a desirable property for computing stable models of a logic program.

The rest of this paper is organized as follows. In section 2 we focus on the syntax and semantics of first-order LPODs. In section 3 we present a progression semantics for LPODs, which we believe is more intuitive to reflect the underlying reasoning procedure for LPODs. We also prove that this progression semantics is equivalent to the earlier stable model semantics we define in section 2. In section 4 we propose an extended ordered completion that translates from LPODs to first-order sentences on finite structures, so that the problem of computing stable models of an LPOD can be viewed as a SAT solving problem. With the same motivation, in section 5, we further extend the notion of first-order loop formulas to LPODs. In particular, we show that under finite structures, an LPOD can be exactly represented by the conjunction of the completion of this program and its FO loop formulas. We then study the complexity and expressiveness issues in section 6. In section 7 we consider the preferred stable models for LPODs, where certain preference relation among the class of stable models of a given LPOD is taken into account. We present a logic characterization on the preferred semantics of LPODs that we call preferred LPODs, and We show that the stable model existence problem for preferred LPODs is co-NP-complete. In section 8, we further extend the splitting theorem for general stable model theory (Ferraris, Lee, Lifschitz, & Palla, 2009) to both LPODs and preferred LPODs respectively. Such results are useful for computing stable models. Finally, we conclude the paper with some remarks in section 9.

1. Some results presented in this paper were published in (Asuncion, Zhang, & Zhang, 2014a).
2. First-order LPODs: Syntax and Semantics

We assume readers are familiar with basic concepts and notations of classical first-order logic. Here we consider a second-order logic language without function symbols but with equality. In particular, an atom is called an equality atom if it is of the form \( t_1 = t_2 \), and a proper atom otherwise. A signature \( \tau \) is a set of symbols of the form \( \{ c_1, \ldots, c_m, P_1, \ldots, P_n \} \) such that \( c_i \) (1 \( \leq i \leq m \)) are constant symbols and \( P_j \) (1 \( \leq j \leq n \)) are predicate symbols. A structure \( M \) of signature \( \tau \) (or simply called a \( \tau \)-structure) is a tuple of the form

\[
(M, c_1^M, \ldots, c_m^M, P_1^M, \ldots, P_n^M),
\]

where \( M \) is the domain of \( M \), which we usually denote as \( \text{Dom}(M) \), and \( c_i^M \) and \( P_j^M \) are the interpretations of the constant and predicate symbols in \( M \) respectively.

Given two signatures \( \tau \) and \( \tau' \) such that \( \tau' \subseteq \tau \). Let \( M \) and \( M' \) be a \( \tau \)-structure and a \( \tau' \)-structure respectively. We say that \( M' \) is a restriction of \( M \) to the signature of \( \tau' \), denoted by \( M|_{\tau'} \), i.e., \( M' = M|_{\tau'} \), if (1) \( \text{Dom}(M') = \text{Dom}(M) \); (2) \( c_i^{M'} = c_i^M \) for each constant symbol \( c \) of \( \tau' \); and (3) \( P_i^{M'} = P_i^M \) for each predicate symbol \( P \) of \( \tau' \). Symmetrically, we call \( M \) an expansion of \( M' \) to the signature \( \tau \).

2.1 The Syntax

An ordered disjunction rule is a construct of the form:

\[
\alpha_1 \times \ldots \times \alpha_k \leftarrow \beta_1, \ldots, \beta_l, \text{not } \gamma_1, \ldots, \text{not } \gamma_m,
\]

where:

- Each \( \alpha_i \) (for 1 \( \leq i \leq k \)) is a proper atom \( P(x) \) for some predicate \( P \) and tuple of terms (variables or constants) \( x \). If \( k = 0 \), then (3) is called a constraint.

- Each \( \beta_i \) (1 \( \leq i \leq l \)) and \( \gamma_i \) (1 \( \leq i \leq m \)) are either a proper or an equality atom.

- By \( \text{Head}(r) \), we denote the ordered expression \( \alpha_1 \times \ldots \times \alpha_k \), that corresponds to the head of \( r \). We also denote \( \text{Head}(r)_i = \alpha_i \).

- As usual, by \( \text{Body}(r) \), we denote the set of literals \( \{ \beta_1, \ldots, \beta_l, \text{not } \gamma_1, \ldots, \text{not } \gamma_m \} \).

- Similarly, by \( \text{Pos}(r) \) and \( \text{Neg}(r) \), we denote the sets of atoms \( \{ \beta_1, \ldots, \beta_l \} \) and \( \{ \gamma_1, \ldots, \gamma_m \} \), respectively.

A first-order logic program with ordered disjunction, FO LPOD in short or simply LPOD, \( \Pi \), is a finite set of ordered disjunction rules of the form (3). If for all rules \( r \in \Pi \) we have that \( k \leq 1 \) in \( \text{Head}(r) = \alpha_1 \times \ldots \times \alpha_k \), then \( \Pi \) becomes a normal logic program. It is obvious that this LPOD syntax naturally extends Brewka’s original propositional logic programs with ordered disjunction (Brewka, 2002) to the first-order case.

For a given LPOD \( \Pi \), a predicate is called intensional if it occurs at least once in the head of some rule in \( \Pi \). All other predicates are called extensional. For convenience, we denote by \( \tau(\Pi) \), \( \text{Pext}(\Pi) \) and \( \text{Pint}(\Pi) \), the signature of \( \Pi \), the set of extensional predicates and the set of intensional predicates, respectively. Furthermore, we also use \( \tau_{\text{ext}}(\Pi) \) and \( \tau_{\text{int}}(\Pi) \) to denote the sub-signatures of \( \tau(\Pi) \) which only contain extensional together with all constant symbols of \( \tau(\Pi) \), and intensional predicates, respectively. When there is no confusion in the context, we often omit the parameter \( \Pi \) in these notions.

4
2.2 The Stable Model Semantics

Now we give a stable model semantics definition of LPODs. For a rule of the form (3), we denote by \( \text{Body}(r) \) the formula \( \beta_1 \land \cdots \land \beta_l \land \neg \gamma_1 \land \cdots \land \gamma_m \), and \( \text{Head}(r)^{FO} \), the following formula

\[
\alpha_1 \lor (\neg \alpha_1 \land \alpha_2) \lor (\neg \alpha_1 \land \neg \alpha_2 \land \alpha_3) \lor \cdots \lor (\neg \alpha_1 \land \neg \alpha_2 \land \cdots \land \neg \alpha_{k-1} \land \alpha_k).
\]

Then for an LPOD \( \Pi \), by \( \widehat{\Pi} \), we denote the conjunction

\[
\bigwedge_{r \in \Pi} (\text{Body}(r) \rightarrow \text{Head}(r)^{FO}).
\]  

(4)

**Definition 1** Let \( \Pi \) be an LPOD with tuple of distinct intensional predicates \( P = P_1 P_2 \ldots P_n \) and \( M \) a \( \tau(\Pi) \)-structure. We say that \( M \) is a stable model of \( \Pi \) iff it is a model of the following second-order sentence:

\[
\text{SM}(\Pi, P) = \widehat{\Pi} \land \neg \exists U (U < P \land \widehat{\Pi}(U)^*),
\]  

(5)

where:

- \( U = U_1 U_2 \ldots U_n \) is a fresh tuple of distinct predicates matching the corresponding predicate arities in \( P \);
- \( U < P \) denotes the formula

\[
\bigwedge_{1 \leq i \leq n} \forall x(U_i(x) \rightarrow P_i(x)) \land \neg \bigwedge_{1 \leq i \leq n} \forall x(P_i(x) \rightarrow U_i(x));
\]
- \( \widehat{\Pi}(U)^* \) denotes the formula

\[
\bigwedge_{r \in \Pi} \forall x_r (\beta_1^* \land \cdots \land \beta_l^* \land \neg \gamma_1 \land \cdots \land \neg \gamma_m \rightarrow (\alpha_1 \times \cdots \times \alpha_k)^*),
\]

where \( (\alpha_1 \times \cdots \times \alpha_k)^* \) denotes the formula:

\[
\alpha_1^* \lor (\neg \alpha_1 \land \alpha_2^*) \lor (\neg \alpha_1 \land \neg \alpha_2 \land \alpha_3^*) \lor \cdots \lor (\neg \alpha_1 \land \neg \alpha_2 \land \cdots \land \neg \alpha_{k-1} \land \alpha_k^*),
\]  

(6)

and such that \( P_i(x)^* = U_i(x) \) for any atomic formula \( P_i(x) \) with \( 1 \leq i \leq n \), and \( Q(y)^* = Q(y) \) otherwise.

In is not difficult to see that Definition 1 is a counterpart of the second-order definition for the general stable model semantics (Ferraris et al., 2011), with the additional operation (6). In the following, we will show that Definition 1 is also a generalization of the semantics for the original propositional LPODs, and hence, our first-order LPOD stable model semantics is a proper uplifting from the propositional level.

Let \( \tau \) be a signature of \( \Pi \) with the sets \( \mathcal{P}_{ext} \) and \( \mathcal{P}_{int} \) of extensional and intensional predicates respectively, and let \( M \) be a \( \tau \)-structure. Then we define \( I_{\mathcal{M}} \) as follows:

\[
I_{\mathcal{M}} := \{ P(a) \mid P \in (\mathcal{P}_{ext} \cup \mathcal{P}_{int}) \text{ and } a \in P^{\mathcal{M}} \},
\]
i.e., the set of propositional atoms corresponding to the extents of the predicates of \( \tau \) under \( \mathcal{M} \). On the other hand, by \( Ext_\mathcal{M} \), we define the set of rules (which are facts) as follows:

\[
Ext_\mathcal{M} := \{ Q(a) \leftarrow Q \in \mathcal{P}_{ext} \text{ and } a \in Q^\mathcal{M} \},
\]

i.e., the set of “facts” corresponding to the extensional database under \( \mathcal{M} \).

In addition to the two previous notions, we now introduce the **grounding** of a program. Let \( \Pi \) be an LPOD and \( \mathcal{M} \) a \( \tau(\Pi) \)-structure. Then by \( \Pi |_{Dom(\mathcal{M})} \), we denote the propositional program obtained from \( \Pi \) by (1) simultaneously replacing the variables by domain elements from \( Dom(\mathcal{M}) \) in all possible ways, and the constant symbols by their corresponding interpretation in \( Dom(\mathcal{M}) \) under \( \mathcal{M} \); and (2) removing all rules that contain false instantiations of equality predicates and deleting those true instantiations of equalities from the remaining rules.

**Theorem 1** Let \( \Pi \) be an LPOD and \( \mathcal{M} \) a \( \tau(\Pi) \)-structure. Then \( \mathcal{M} \) is a stable model of \( \Pi \) iff \( I_\mathcal{M} \) is a stable model of some split program of \( \Pi |_{Dom(\mathcal{M})} \cup Ext_\mathcal{M} \).

**Proof:** \( \mathcal{M} \) is a stable model of \( \Pi \) iff \( I_\mathcal{M} \) is a stable model of \( \widehat{\Pi} |_{Dom(\mathcal{M})} \cup Ext_\mathcal{M} \) (in the sense of (Cabalar, 2011)) and where \( \widehat{\Pi} |_{Dom(\mathcal{M})} \) denotes the grounding of \( \Pi \) under the domain \( Dom(\mathcal{M}) \) iff \( I_\mathcal{M} \) is a stable model of some split program of \( \Pi |_{Dom(\mathcal{M})} \cup Ext_\mathcal{M} \) (via Theorem 4 in (Cabalar, 2011)). \( \square \)

### 3. Progression Semantics: An Alternative

As we observe that Definition 1 hardly reveals the rule based reasoning feature of an underlying LPOD. In this section, we provide an alternative semantics - the progression semantics, which is an extension of Zhang and Zhou’s progression semantics for first-order normal logic program (Zhang & Zhou, 2010). As we will see, the important feature of such semantics is that it naturally represents the reasoning procedure through the progression stages.

**Definition 2** Let \( \Pi \) be an LPOD and \( \mathcal{M} \) a \( \tau(\Pi) \)-structure

\[
(Dom(\mathcal{M}), c_1^\mathcal{M}, \ldots, c_r^\mathcal{M}, Q_1^\mathcal{M}, \ldots, Q_s^\mathcal{M}, P_1^\mathcal{M}, \ldots, P_n^\mathcal{M}),
\]

such that \( c_i \) (for \( 1 \leq i \leq r \)), \( Q_i \) (for \( 1 \leq i \leq s \)), and \( P_i \) (for \( 1 \leq i \leq n \)) are its constant symbols, extensional and intensional predicate symbols, respectively. We define a sequence of \( \tau(\Pi) \)-structures \( \mathcal{M}^0, \ldots, \mathcal{M}^t, \mathcal{M}^{t+1}(\Pi), \ldots \), inductively as follows:

\[
\mathcal{M}^0(\Pi) = (Dom(\mathcal{M}), c_1^{\mathcal{M}^0(\Pi)}, \ldots, c_r^{\mathcal{M}^0(\Pi)}, Q_1^{\mathcal{M}^0(\Pi)}, \ldots, Q_s^{\mathcal{M}^0(\Pi)}, P_1^{\mathcal{M}^0(\Pi)}, \ldots, P_n^{\mathcal{M}^0(\Pi)}), \quad (7)
\]

where: \( c_i^{\mathcal{M}^0(\Pi)} = c_1^\mathcal{M} \) (for \( 1 \leq i \leq r \)), \( Q_i^{\mathcal{M}^0(\Pi)} = Q_1^\mathcal{M} \) (for \( 1 \leq i \leq s \)), and \( P_i^{\mathcal{M}^0(\Pi)} = \emptyset \) (for \( 1 \leq i \leq n \));

\[
\mathcal{M}^{t+1}(\Pi) = \mathcal{M}^t(\Pi) \cup \{ \alpha_i \eta \mid \exists r \in \Pi \text{ with } Head(r) = \alpha_i \times \ldots \times \alpha_i \times \ldots \times \alpha_k \text{ and } \eta \text{ an assignment such that:}
\]

1. \( Pos(r) \eta \subseteq \mathcal{M}^t(\Pi) \) and \( Neg(r) \eta \cap \mathcal{M} = \emptyset \);
2. \( i \) (for \( 1 \leq i \leq k \)) is the largest \( i \) such that \( \{ \alpha_1 \eta, \ldots, \alpha_{i-1} \eta \} \cap \mathcal{M} = \emptyset \). \( \square \)
Then by $M^\infty (\Pi )$, we denote $\bigcup _{0 \leq t \leq \infty } M^t (\Pi )$. 

Since $M^t (\Pi ) \subseteq M^{t+1} (\Pi )$, we know that the sequence $M^0, M^1, \cdots$ is monotonic and $M^t (\Pi )$ always converges to its fixpoint. Also note that when $\text{Head}(r) = \alpha _1$ (i.e., $r$ is a normal rule), then we will have in Item 2 in (8) that $i = 1$ will automatically be the largest $i$ such that $\{ \alpha _1 \eta , \ldots , \alpha _{i-1} \eta \} \cap M = \emptyset$. From this fact, the following proposition shows that the progression characterization captures the stable models when $\Pi$ is a normal logic program.

**Proposition 1** Let $\Pi$ be a normal logic program and $M$ a $\tau (\Pi )$-structure. Then $M$ is a stable model of $\Pi$ iff $M^\infty (\Pi ) = M$.

**Proof:** If $\Pi$ is a normal program, then we will have by default in Item 2 in (8) of Definition 2 that $i = 1$ will be the largest $i$ such that $\{ \alpha _1 \eta , \ldots , \alpha _{i-1} \eta \} \cap M = \emptyset$ because $\{ \alpha _1 , \ldots , \alpha _{i-1} \} = \emptyset$ in this case. Therefore, we can omit Item 2 in (8) so that $M^{t+1} (\Pi )$ will be as originally defined in (Zhang & Zhou, 2010) for normal programs. □

Theorem 2 shows that the progression semantics coincides with the second-order stable model semantics of LPODs as defined in Definition 1. We first present the following lemma which is needed in the proof of Theorem 2.

**Lemma 1** If $M^\infty (\Pi ) = M$ then $M \models \bar{\Pi}$. 

**Proof:** Assume that $M \models \text{Body}(r) \eta$ for some rule $r \in \Pi$ and we show that $M \models \text{Head}(r)^{FO} \eta$ as well. Indeed, since $M \models \text{Body}(r) \eta$, then we have that $\text{Pos}(r) \eta \subseteq M$ and $\text{Neg}(r) \eta \cap M = \emptyset$. Then since $M^\infty (\Pi ) = M$, there exists some certain $t$ such that $\text{Pos}(r) \eta \subseteq M^t (\Pi ) \subseteq M^\infty (\Pi ) = M$. Then since, if we assume that $\text{Head}(r) = \alpha _1 \times \ldots \times \alpha _i \times \ldots \times \alpha _k$, there will exist a largest $i$ (for $1 \leq i \leq k$) such that $\{ \alpha _1 \eta , \ldots , \alpha _{i-1} \eta \} \cap M = \emptyset$, then by the definition of $M^{t+1} (\Pi )$, we will have that $\alpha _i \eta \in M^{t+1} (\Pi ) \subseteq M^\infty (\Pi ) = M$, which implies that $M \models (\neg \alpha _1 \land \ldots \land \neg \alpha _{i-1} \land \alpha _i) \eta$. Then it follows that $M \models \text{Head}(r)^{FO} \eta$ as well. This completes the proof of Lemma 1. □

**Theorem 2** Let $\Pi$ be an LPOD and $M$ a $\tau (\Pi )$-structure. Then $M$ is a stable model of $\Pi$ iff $M^\infty (\Pi ) = M$.

**Proof:** ($\Rightarrow $) First we show that $M^\infty \subseteq M$ by showing $M^t \subseteq M$ holds for all $t \geq 0$ by induction. Clearly, $M^0 (\Pi ) \subseteq M$ holds by the definition of $M^0 (\Pi )$ since all the intensional relations are set to empty. Now assume that $M^t (\Pi ) \subseteq M$ holds for all $0 \leq t \leq t$ and let $P(\alpha ) \in M^{t+1} (\Pi )$ such that $P(\alpha ) \notin M^t (\Pi )$. Then by the definition of $M^{t+1} (\Pi )$, we have that there exists a rule $r \in \Pi$ with $\text{Head}(r) = \alpha _1 \times \ldots \times \alpha _i \times \ldots \times \alpha _k$ and an assignment $\eta$ such that:

1. $P(\alpha ) = \alpha _i \eta$;
2. $\text{Pos}(r) \eta \subseteq M^t (\Pi )$ and $\text{Neg}(r) \cap M = \emptyset$;
3. $i$ (for $1 \leq i \leq k$) is the largest $i$ such that $\{ \alpha _1 \eta , \ldots , \alpha _{i-1} \eta \} \cap M = \emptyset$. 


\begin{align*}
\begin{aligned}
\text{Lemma 1:} & \quad \text{If } M^\infty (\Pi ) = M \text{ then } M \models \bar{\Pi}.
\text{Proof:} & \quad \text{Assume that } M \models \text{Body}(r) \eta \text{ for some rule } r \in \Pi \text{ and we show that } M \models \text{Head}(r)^{FO} \eta \text{ as well. Indeed, since } M \models \text{Body}(r) \eta, \text{ then we have that } \text{Pos}(r) \eta \subseteq M \text{ and } \text{Neg}(r) \eta \cap M = \emptyset. \text{ Then since } M^\infty (\Pi ) = M, \text{ there exists some certain } t \text{ such that } \text{Pos}(r) \eta \subseteq M^t (\Pi ) \subseteq M^\infty (\Pi ) = M. \text{ Then since, if we assume that } \text{Head}(r) = \alpha _1 \times \ldots \times \alpha _i \times \ldots \times \alpha _k, \text{ there will exist a largest } i \text{ (for } 1 \leq i \leq k) \text{ such that } \{ \alpha _1 \eta , \ldots , \alpha _{i-1} \eta \} \cap M = \emptyset, \text{ then by the definition of } M^{t+1} (\Pi ), \text{ we will have that } \alpha _i \eta \in M^{t+1} (\Pi ) \subseteq M^\infty (\Pi ) = M, \text{ which implies that } M \models (\neg \alpha _1 \land \ldots \land \neg \alpha _{i-1} \land \alpha _i) \eta. \text{ Then it follows that } M \models \text{Head}(r)^{FO} \eta \text{ as well. This completes the proof of Lemma 1. □}
\end{aligned}
\end{align*}
Then since $Pos(r)\eta \subseteq M^t(\Pi)$ and $Neg(r) \cap M = \emptyset$, and where $M^t(\Pi) \subseteq M$, it follows that $M \models Body(r)\eta$. Thus, since $M \models \tilde{\Pi}$ (because $M \models SM(\Pi, P)$), then it follows that $M \models Head(r)^{FO} = (\neg \alpha_1 \land \ldots \land \neg \alpha_{i-1} \land \alpha_i)\eta$, which implies that $\alpha_i \eta = P(a) \in M$. Therefore, we had shown that $M^\infty(\Pi) \subseteq M$.

Now let us assume on the contrary that it is also the case that $M^\infty(\Pi) \subseteq M$ and we show that this will lead to a contradiction. We define a $\tau(\Pi) \cup \{U_1, \ldots, U_n\}$-structure $U$ as follows:

- $c^U = c^M$ for every constant $c$ of $\tau(\Pi)$;
- $Q^U = Q^M$ for every extensional predicate of $\tau(\Pi)$;
- $P^U_i = P_i^M$ for every intensional predicate $P_i$ of $\tau(\Pi)$, such that $1 \leq i \leq n$.
- $U^U_i = P_i^M(\Pi)$ for every predicate $U_i$ in $\{U_1, \ldots, U_n\}$, with $1 \leq i \leq n$.

Now we will show that $U \models U < P$ and $U \models \tilde{\Pi}(U)^*$. The first part $U \models U < P$ follows from the fact that $M^\infty(\Pi) \subseteq M$ and by the way the $\tau(\Pi) \cup \{U_1, \ldots, U_n\}$-structure $U$ is constructed from $M$ and $M^\infty(\Pi)$ above. Now we show the second part. So assume for some rule $r$ in $\Pi$ of the form (3) and some assignment $\eta$ that $U \models Body(r)^*\eta$. Then since $U \models U < P$ and by the definition of $U$, it follows that $Pos(r)\eta \subseteq M^\infty(\Pi)$, which implies that $Pos(r)\eta \subseteq M^t(\Pi) \subseteq M^\infty(\Pi) \subseteq M$ for some $t \geq 0$. Then since $Pos(r)\eta \subseteq M$, then $M \models Body(r)^*\eta$ as well. Then since $M \models \tilde{\Pi}$, we have that $M \models Head(r)^{FO} \eta$, which further implies that $M \models \neg \alpha_1 \land \ldots \land \neg \alpha_{i-1} \land \alpha_i$ for some $1 \leq i \leq k$. Then by Item 2 of (8), this implies that $\alpha_i \eta \in M^{t+1}(\Pi) \subseteq M^\infty(\Pi)$ since $i$ will be the largest $i$ such that $\{\alpha_1 \eta, \ldots, \alpha_i \eta\} \cap M = \emptyset$, which further implies that $U \models Head(r)^*\eta$ by the way $U$ was constructed from $M^\infty(\Pi)$ and since $M^t(\Pi) \subseteq M^\infty(\Pi)$. Therefore, because we have shown that $U \models \tilde{\Pi}(U)^*$, then we now have a contradiction since we initially assumed that $M \models SM(\Pi, P)$, and where this implies that $M \models \neg \exists U(U < P \land \tilde{\Pi}(U)^*)$ (which is contradicted by the existence of the structure $U$).

(\Leftrightarrow) Now we assume that $M \not\models SM(\Pi, P)$. Since by Lemma 1 we have that $M \models \tilde{\Pi}$, then this implies that $M \models \neg \exists U(U < P \land \tilde{\Pi}(U)^*)$. Thus, assume that $U$ is a $\tau(\Pi) \cup \{U_1, \ldots, U_n\}$-structure such that $U \models U < P \land \tilde{\Pi}(U)^*$ and $M \models SM(\Pi, P)$, where by induction on $t$ that $P_i^{M^t(\Pi)} \subseteq U_i^U$, for $1 \leq i \leq n$. Therefore, since $U \models U < P$ (i.e., $U$ is “strictly smaller” than $P$), this contradicts the assumption that $M^\infty(\Pi) = M$ because $M^\infty(\Pi)$ would be “strictly less” than $M$ in this case.

The base case for $M^0(\Pi)$ clearly holds since $P_1^{M^0(\Pi)} = \emptyset$, for $1 \leq i \leq n$. So let us assume that $P_i^{M^t(\Pi)} \subseteq U_i^U$, for $1 \leq i \leq n$, and consider $M^{t+1}(\Pi)$. Indeed, let $P(a) \in M^{t+1}(\Pi) \setminus M^t(\Pi)$. Then by the definition of $M^{t+1}(\Pi)$ as we find in (8) of Definition (2), there exists a rule $r \in \Pi$ with $Head(r) = \alpha_1 \times \ldots \times \alpha_i \times \ldots \times \alpha_k$ such that:

1. $Pos(r)\eta \subseteq M^t(\Pi)$ and $Neg(r)\eta \cap M = \emptyset$;
2. $i$ (for $1 \leq i \leq k$) is the largest $i$ such that $\{\alpha_1 \eta, \ldots, \alpha_i \eta\} \cap M = \emptyset$.

and where $P_i(a) = \alpha_i \eta$. Then it is not difficult to see that $U \models Body(r)^*\eta$ as well since $P_i^{M^t(\Pi)} \subseteq U_i^U$, for $1 \leq i \leq n$. Then since $U \models \tilde{\Pi}(U)^*$, we have that $U \models Head(r)^*$, and where in particular, $U \models (\neg \alpha_1 \land \ldots \land \neg \alpha_{i-1} \land \alpha_i)^\eta$, so that $\alpha_i^\eta = P(a) \in U$. This completes the proof of Theorem 2. \qed
**Example 1** Let $\Pi$ be the following LPOD with the two rules:

\begin{align}
    & r_1: P(x) \times Q(x) \times R(x) \leftarrow S(x), \\
    & r_2: T(x) \times U(x) \leftarrow Q(x),
\end{align}

such that $S$ is the only extensional predicate. Now let $M$ be a $\tau(\Pi)$-structure such that:

$$M = \{(a, b), S^M = \{a, b\}, P^M = \{a\}, Q^M = \{b\}, R^M = \emptyset, T^M = \emptyset, U^M = \{b\}\}.$$  

Now based on (7) of Definition 2, we have that

$$M^0(\Pi) = \{(a, b), S^M = \{a, b\}, P^M = \emptyset, Q^M = \emptyset, R^M = \emptyset, T^M = \emptyset, U^M = \emptyset\},$$

i.e., all the intensional relations are initially set to empty. Now let us compute $M^1(\Pi)$. From (8) of Definition 2, we have that $P(a) \in M^1(\Pi)$ since $\alpha_{i=1} = P(a)$ (with $\eta : x \rightarrow a$) is the largest $i$ such that $\{\alpha_{i=1}, \ldots, \alpha_{i-1}\} = \emptyset \cap M = \emptyset$. Similarly, we also have that $Q(b) \in M^1(\Pi)$ since in this case, with $\eta : x \rightarrow b$, we have that $\alpha_{i=2} = Q(b)$ is the largest $i$ such that $\{\alpha_{i=2}, \ldots, \alpha_{i-1}\} = \{P(b)\} \cap M = \emptyset$ (i.e., since $P(b) \notin M$), so that we now have

$$M^1(\Pi) = \{(a, b), S^M = \{a, b\}, P^M = \{a\}, Q^M = \{b\}, R^M = \emptyset, T^M = \emptyset, U^M = \emptyset\}.$$  

Similarly, we will have $M^3(\Pi)$ as follows:

$$\{(a, b), S^M = \{a, b\}, P^M = \{a\}, Q^M = \{b\}, R^M = \emptyset, T^M = \emptyset, U^M = \{b\}\},$$

and further, we can obtain $M^t(\Pi) = M^t(\Pi)$ for all $t \geq 4$, that is $M^\infty(\Pi) = M^3(\Pi)$. Therefore, since $M^\infty(\Pi) = M$, we have by Theorem 2 that $M$ is a stable model of $\Pi$. □

4. From LPODs to First-order Formulas

While Definition 2 provides an alternative semantics for LPODs, which represents the program reasoning feature through progression process, it, however, still does not reveal much information about how a stable model of a given LPOD may be computed. In this section, we show how a variant of the ordered completion (Asuncion et al., 2012) for normal logic programs can capture the stable models of first-order LPODs, which, like demonstrated in (Asuncion et al., 2012), may be viewed as a computational basis for an LPOD solver development.

For this purpose, for a given pair of predicates $(P, Q)$ (where $P$ and $Q$ can be the same), by $\leq_{PQ}$, we denote a new predicate such that its arity is the sum of the arities of $P$ and $Q$. We refer to such predicates as the *comparison predicates*. The intuitive meaning of atom $\leq_{PQ}(x, y)$ is that: $Q(y)$ is true only if $P(x)$ is true.

**Definition 3** Let $\Pi$ be an LPOD with a tuple of distinct intensional predicates $P$. By $\text{MCOMP}(\Pi, P)$, we denote the following first-order sentence:

\[ \bigwedge_{r \in \Pi} \forall x_r (\text{Body}(r) \rightarrow \text{Head}(r)^{FO}) \]

\[ \land \bigwedge_{P \in \Pi} \forall x (P(x) \rightarrow \bigvee_{r \in \Pi, \text{Head}(r) = \alpha_1 \times \ldots \times \alpha_k} \bigvee_{1 \leq t \leq k} \exists x_r (x = y \land \text{Body}(r) \land \text{Pos}(r) < P(x) \land \bigwedge_{1 \leq j \leq i-1} \neg \alpha_j)), \]

where:

\[ \text{Body}(r) = \alpha_1 \times \ldots \times \alpha_k \land \alpha_i = P(y) \]
• For a rule \( r \in \Pi, x \) in (11) and (12) is the tuple of distinct variables of \( r \);
• We assume that \( x \) in (12) is a tuple of fresh distinct variables not mentioning those from \( x \);
• For two tuples \( x = x_1 x_2 \ldots x_s \) and \( y = y_1 y_2 \ldots y_s \), \( x = y \) in (12) denotes the formula
  \[ (x_1 = y_1) \land (x_2 = y_2) \land \ldots \land (x_s = y_s); \]
• \( Pos(r) < P(x) \) in (12) denotes the formula
  \[ \bigwedge_{Q(z) \in Pos(r), Q \in P} (\leq_{QP} (z, x) \land \neg \leq_{PQ} (x, z)); \]
where the \( \leq_{QP} \) and \( \leq_{PQ} \) here are comparison predicates.

Then we further define

\[ OC(\Pi, P) = MComp(\Pi, P) \land Trans(\Pi, P), \]

where \( Trans(\Pi, P) \) denotes the following formula:

\[ \bigwedge_{P, Q, R \in P} \forall x y z (\leq_{PQ} (x, y) \land \leq_{QR} (y, z) \rightarrow \leq_{PR} (x, z)), \tag{13} \]

and each of \( x, y, \) and \( z \) are tuples of distinct variables.

Now we present the main result of this section as follows.

**Theorem 3** Let \( \Pi \) be an LPOD with tuple of distinct intensional predicates \( P \) and \( \mathcal{M} \) a finite \( \tau(\Pi) \)-structure. Then \( \mathcal{M} \) is a stable model of \( \Pi \) iff it can be expanded to a model of \( OC(\Pi, P) \).

**Proof:** \( \Rightarrow \) Then \( \mathcal{M} \models \hat{\Pi} \), so that \( \mathcal{M} \) is also a model of (11). Therefore, it is now sufficient to show that \( \mathcal{M} \) is also a model of (12) and (13). Indeed, by Theorem 2, we have that \( \mathcal{M}^x(\Pi) = \mathcal{M} \) as well since \( \mathcal{M} \) is a stable model of \( \Pi \) (by assumption). Then based on the progression stages \( \mathcal{M}^0(\Pi), \mathcal{M}^1(\Pi), \ldots, \mathcal{M}^k(\Pi), \mathcal{M}^{k+1}(\Pi), \ldots \), we now construct an expansion \( \mathcal{M}' \) of \( \mathcal{M} \) to the signature \( \tau(\Pi) \cup \{ \leq_{PQ} \mid P, Q \in P \} \). Thus, for \( t \geq 0 \), set \( \Delta^t(\Pi) \) as the \( \tau(\Pi) \)-structure defined inductively as follows: \( \Delta^0(\Pi) = \mathcal{M}^0(\Pi) \) and where for \( t \geq 1 \), \( \Delta^t(\Pi) = \mathcal{M}^t(\Pi) \setminus \mathcal{M}^{t-1}(\Pi) \). Intuitively, for \( t \geq 1 \), \( \Delta^t(\Pi) \) represents the difference of the intensional relations between the stages \( t \) and \( t - 1 \). Then we can prove the following Claim 1.

**Claim 1.** If \( P(a) \in \Delta^t(\Pi) \) then for all \( t' \neq t \), we have that \( P(a) \notin \Delta^{t'}(\Pi) \).

Now the expansion \( \mathcal{M}' \) of \( \mathcal{M} \) is constructed by setting:

\[ \leq_{PQ}^{\mathcal{M}'} = \{ ab \mid P(a) \in \Delta^t(P), Q(b) \in \Delta^{t_2}(\Pi), \text{ and } t_1 < t_2 \}, \]

for each pair of predicates \( P \) and \( Q \) (can be the same) of \( P \). Now we show that \( \mathcal{M} \) satisfies both (12) and (13). Indeed, let \( ab \in \leq_{PQ}^{\mathcal{M}'} \) and \( bc \in \leq_{QR}^{\mathcal{M}'} \) for some predicates \( P, Q, \) and \( R \) of \( P \). Then by the definitions of the interpretations \( \leq_{PQ}^{\mathcal{M}'} \) and \( \leq_{QR}^{\mathcal{M}'} \), we have that: \( P(a) \in \Delta^t(\Pi) \) and \( Q(b) \in \Delta^{t_2}(\Pi) \) with \( t_1 < t_2 \); and \( Q(b) \in \Delta^{t_1}(\Pi) \) and \( R(c) \in \Delta^{t_2}(\Pi) \) with \( t_1 < t_2 \). Then since
by Claim 1, $Q(b)$ can only be in one particular $t$ such that it is in $\Delta^t(\Pi)$, then we have that $t_2 = t_1'$, so that by transitivity, we have $t_1 < t_2$. Then by the definition of the interpretation $\leq_{PR}^{M'}$, we have that $ac \in \leq_{PR}^{M'}$, so that $M'$ satisfies (13). Now to show $M'$ satisfies (12), let $P(a) \in M'$ such that $P \in P$. Then $P(a) \in M_{PQ}^{\infty}(\Pi)$ since $M$ is a stable model of $\Pi$ and $M_{PQ}^{\infty}(\Pi) = M$ by Theorem 2. Then for some $t$, we have that $P(a) \in \Pi(t)$. Let us assume without loss of generality that $P(a) \notin \Pi(t)$ for all $0 \leq t' < t$, i.e., $t$ is the first stage that derives $P(a)$. Then by the definition of $M'(\Pi)$, there exists some rule $r \in \Pi$ with $\text{Head}(r) = \alpha_1 \times \ldots \times \alpha_i \times \ldots \times \alpha_k$ and assignment $\eta$ such that:

1. $\text{Pos}(r)\eta \subseteq M_{t-1}^{\Pi}(\Pi) \subseteq M$ (and where $M_{t-1}^{\Pi}(\Pi) \subseteq M$ since $M_{t-1}^{\Pi}(\Pi) \subseteq M_{PQ}^{\infty}(\Pi) = M$) and $\text{Neg}(r)\eta \cap M = \emptyset$;

2. $i$ (for $1 \leq i \leq k$) is the largest $i$ such that $\{\alpha_1\eta, \ldots, \alpha_{i-1}\eta\} \cap M = \emptyset$.

Then since $M'$ is simply the expansion of $M$ to include the interpretation of the comparison symbols in $\{\leq_{PR}^{P}, P, Q \in P\}$, we have that

$$M' \models (a = y \land \text{Body}(r) \land \bigwedge_{1 \leq j \leq i-1} \neg \alpha_j)\eta.$$ 

Therefore, it is only left to show that $M' \models (\text{Pos}(r) < P(a))\eta$ so that

$$M' \models (a = y \land \text{Body}(r) \land \text{Pos}(r) < P(a) \land \bigwedge_{1 \leq j \leq i-1} \neg \alpha_j)\eta$$

(14)
as well. Indeed, since $M_t^{\Pi}(\Pi)$ is the first stage that derives $P(a)$, then by the definition of $\Delta^t(\Pi)$, we have that $P(a) \in \Delta^t(\Pi)$. Now let $Q(b) \in \text{Pos}(r)\eta$ such that $Q \in P$. Then since $Q(b) \in M_{t-1}^{\Pi}(\Pi)$, there exists the least stage $t' \leq t - 1$ such that $Q(b) \in M_{t'}^{\Pi}(\Pi)$. Then this implies that $Q(b) \in \Delta^t(\Pi)$, and since $t' \leq t - 1 < t$, we further have by the definition of the interpretation $\leq_{PR}^{M'}$, that $\text{body}(a) \in \leq_{PR}^{M'}$. Now we show that $\text{body}(a) \not\in \leq_{PQ}^{M'}$. Otherwise, assume that $\text{body}(a) \in \leq_{PQ}^{M'}$. Then by the definition of $\leq_{PQ}^{M'}$, there exists $t_1$ and $t_2$, where $t_1 < t_2$, such that $P(a) \in \Delta^{t_1}(\Pi)$ and $Q(b) \in \Delta^{t_2}(\Pi)$. Then by Claim 1, we have that $t_2 = t'$ since $Q(b)$ is both in $\Delta^{t_2}(\Pi)$ and $\Delta^{t}(\Pi)$. Then because we already have that $t' < t$ and where $t_1 < t_2 = t' < t$, then we have that $t_1 < t$. Thus, since $P(a)$ is both in $\Delta^{t_1}(\Pi)$ and $\Delta^{t}(\Pi)$ where $t_1 \neq t$ (since $t_1 < t$), this is a contradiction by Claim 1. Therefore, it follows that $M' \models \text{Pos}(r) < P(a)$ so that (14) holds as well.

$(\Rightarrow)$ Assume that $M'$ is the expansion of $M$ such that $M' \models \text{OC}_P(\Pi)$. Then there exists a strict-partial order $\mathcal{P} = (\text{Dom}(\mathcal{P}), \lhd^P)$, where

$$\text{Dom}(\mathcal{P}) = \{P(a) \mid a \in P^M \text{ and } P \in P\},$$
such that $\mathcal{P}$ is induced by the interpretations of the comparison predicates $\leq_{PQ}$ (for $P, Q \in P$). Then by the Order-extension Theorem (Kaye & Macpherson, 1994), there exists a strict-total order $\mathcal{T} = (\text{Dom}(\mathcal{T}), \lhd^T)$, where $\text{Dom}(\mathcal{T}) = \text{Dom}(\mathcal{P})$, such that for all distinct elements $P(a)$ and $Q(b)$ in $\text{Dom}(\mathcal{T})$, either $P(a) \lhd^T Q(b)$ or $Q(b) \lhd^T P(a)$ holds. Now, since $M$ is finite, then $\text{Dom}(\mathcal{T})$ will be finite as well, so that a bottom element, denoted $\text{bot}(\mathcal{T})$, exists. Then for a $P(a) \in \text{Dom}(\mathcal{T})$, $\mathcal{T}^{P(a)}$ is inductively defined via the following:

1. $\mathcal{T}^{\text{bot}(\mathcal{T})} = \{\text{bot}(\mathcal{T})\}$;
where $\text{succ}(P(a))$ denotes the successor of $P(a)$ under the ordering $\precsim_T$. Then $T^{\text{top}(T)}$, where $\text{top}(T)$ is the greatest element under $\precsim_T$, is simply the collection of all the elements of $\text{Dom}(T)$, since $T$ is a strict-total ordering of $\text{Dom}(T)$. We will now show by induction on $P(a)$ that $T^{\text{top}(T)} \subseteq M^\infty(\Pi)$ for all $P(a) \in \text{Dom}(T)$. For the base case, assume $\text{bot}(T) = P(a)$. Then since $M'$ satisfies (12), there exists a rule $r \in \Pi$ with $\text{Head}(r) = \alpha_1 \times \ldots \times \alpha_k$ such that for some $1 \leq i \leq k$ and some assignment $\eta$, we have that

$$M' \models (a = y \land \text{Body}(r) \land \text{Pos}(r) \precsim P(a) \land \bigwedge_{1 \leq j \leq i-1} \neg \alpha_j)\eta,$$

and where $\alpha_i = P(y)$. Moreover, since $P(a)$ is the bottom element under the ordering $\precsim_T$, then $\text{Pos}(r)$ does not mention atoms of intensional predicates (since the strict-total order $T$ is induced by the interpretation of the comparison atoms). Therefore, since $P(a) = \alpha_\eta \in M$ (because $\text{Dom}(T) = \{P(a) \mid a \in P^M$ and $P \in P\}$), then it is not difficult to verify by Item 2 in (8) that $P(a) \in M^\infty(\Pi) \subseteq M^\infty(\Pi)$. Now assume $T^{Q(b)} \subseteq M^\infty(\Pi)$ and we will show that $T^{\text{succ}(Q(b))} \subseteq M^\infty(\Pi)$ as well. Thus, for convenience, assume that $\text{succ}(Q(b)) = P(a)$. Now, since $M'$ satisfies (12), then there exists some rule $r \in \Pi$ with $\text{Head}(r) = \alpha_1 \times \ldots \times \alpha_k$ and assignment $\eta$ such that

$$M' \models (a = y \land \text{Body}(r) \land \text{Pos}(r) \precsim P(a) \land \bigwedge_{1 \leq j \leq i-1} \neg \alpha_j)\eta,$$

where $\alpha_i = P(y)$ (the “ordered support” for $P(a)$). Then since $M' \models (\text{Pos}(r) \precsim P(a))\eta$ and $T$ is the total-order extension of the strict-partial order induced by the interpretation of the comparison atoms, we have that $\text{Pos}(r)\eta \subseteq T^{Q(b)}$ since $Q(b) <^T P(a)$. Moreover, since $M' \models \text{Body}(r)\eta$, then $\text{Neg}(r)\eta \cap M = \emptyset$ as well. Therefore, since $T^{Q(b)} \subseteq M^\infty(\Pi)$ for some $t$ (since $T^{Q(b)} \subseteq M^\infty(\Pi)$ by the inductive hypothesis), then we have that both $\text{Pos}(r)\eta \subseteq M^\infty(\Pi)$ and $\text{Neg}(r)\eta \cap M = \emptyset$ holds. In addition, since we also have from (12) that $M \models (\neg \alpha_1 \times \ldots \times \neg \alpha_{i-1})\eta$ and where $\alpha_i = P(a) \in M$, then it follows from (8) that $P(a) \in M^{t+1}(\Pi)$. Hence, since we have shown that $T^{\text{top}(T)} \subseteq M^\infty(\Pi)$, then since $M^0(\Pi) \subseteq M^\infty(\Pi)$, it follows that $(T^{\text{top}(T)} \cup M^0(\Pi)) = M \subseteq M^\infty(\Pi)$. Therefore, to show that $M = M^\infty(\Pi)$, it is left for us to show that $M^\infty(\Pi) \subseteq M$. We now show this fact by induction on $t$. For the base case, we clearly have that $M^\infty(\Pi) \subseteq M$ since for $M^0(\Pi)$, all interpretations of the comparison predicates are set to be empty. Now let us assume that for $0 \leq t' \leq t$, we have that $M^{t'}(\Pi) \subseteq M$ holds and we will now show that $M^{t+1}(\Pi) \subseteq M$ holds as well. Indeed, let $P(a) \in M^{t+1}(\Pi) \setminus M^t(\Pi)$. Then by the definition of $M^{t+1}(\Pi) \setminus M^t(\Pi)$, there exists a rules $r \in \Pi$ with $\text{Head}(r) = \alpha_1 \times \ldots \times \alpha_i \times \ldots \times \alpha_k$ and assignment $\eta$ such that

1. $P(a) = \alpha_i \eta$;
2. $\text{Pos}(r)\eta \subseteq M^t(\Pi)$ and $\text{Neg}(r)\eta \cap M = \emptyset$;
3. $i$ (for $1 \leq i \leq k$) is the largest $i$ such that $\{\alpha_1 \eta, \ldots, \alpha_{i-1} \eta\} \cap M = \emptyset$.

Then since $M^t(\Pi) \subseteq M$ and $M \models \hat{\Pi}$ (since $M$ satisfies (11)), then it follows that $P(a) \in M$. Therefore, since we had shown that $M^\infty(\Pi) = M$, then we have by Theorem 2 that $M$ is a stable
Theorem 3 has an important practical value towards an LPOD solver development. Basically, for a given LPOD, to compute its stable models, we can firstly translate this program into its corresponding ordered completion OC(\(\Pi, P\)), then by taking extensional databases as input to ground OC(\(\Pi, P\)) to a propositional formula, and finally compute its classical models by calling an SAT solver. The major advantages of this approach over other ASP solvers for normal logic programs have been explained and demonstrated in (Asuncion et al., 2012).

**Proposition 2** Let \(\Pi\) be an LPOD. Then \(OC(\Pi, P)\) can be computed in time \(O(|\Pi| \cdot N \cdot H \cdot |P| + |P|^3)\), where \(N\) and \(H\) are the maximum length of the rules and the maximum length of the ordered disjunctive heads in the rules of \(\Pi\), respectively.

5. Loop Formulas

As we argued in section 4, translating an LPOD into a first-order sentence makes it possible to compute stable models of the given LPOD via an SAT based solver. In fact, idea of translating a propositional ASP program into a classical propositional formula has been studied previously but via a notion so-called loop formulas (Lin & Zhao, 2004). This approach has then been further extended to the case of normal logic programs with variables (Chen et al., 2006; Lee & Meng, 2011). In this section, we show that on finite structures, an LPOD can also be translated into a first-order theory via the the notion of FO loop formulas of LPODs.

Firstly, we introduce the notion of positive dependency graphs for LPODs. In general, the positive dependency graph of an LPOD is a simple extension of that for normal logic programs with variables. Given an LPOD \(\Pi\), its positive dependency graph \(G^+_\Pi = (V, E^+_G)\) is an infinite graph with vertices:

\[
V = \{\alpha'_i | \exists r \in \Pi \text{ and assignment } \theta \text{ such that } Head(r) = \alpha_1 \times \ldots \times \alpha_k \text{ and } \alpha'_i = \alpha_i \theta \text{ for some } 1 \leq i \leq k\}
\]

and edges

\[
E^+_G = \{(\alpha'_i, \beta'_j) | \exists r \in \Pi \text{ and assignment } \theta \text{ such that } Head(r) = \alpha_1 \times \ldots \times \alpha_k, \ P_{int}(Pos(r)) = \{\beta_1, \ldots, \beta_l\}, \text{ and } (\alpha'_i, \beta'_j) = (\alpha_i \theta, \beta_j \theta) \text{ for some } 1 \leq i \leq k \text{ and } 1 \leq j \leq l\}.
\]

Note that \(G^+_\Pi\) is indeed infinite since there can be an infinite amount of assignments \(\theta\). We say that a finite subset \(L\) of the vertices \(V\) of \(G^+_\Pi = (V, E^+_G)\) is a loop of \(\Pi\) if \(L\) is a strongly connected component of \(G^+_\Pi\).

The following definition now defines the notion of an external support of a loop \(L\). For this purpose, given an LPOD \(\Pi\) and a loop \(L\) of it, we assume that \(\Pi\) does not mention variables from \(L\) by renaming the variables in \(\Pi\) appropriately.

**Definition 4 (Loop formulas for LPOD)** Let \(\Pi\) be an LPOD with tuple of intensional predicates \(P\) and \(L\) a loop of \(\Pi\). Then for some atom \(\alpha \in L\), the external support of \(\alpha\) under the program \(\Pi\)
and the loop \( L \), denoted \( ES(\alpha, L, \Pi, \mathbf{P}) \), is the following first-order formula:

\[
\bigvee_{\alpha \in L} \exists y_{r\theta}(\text{Body}(r\theta) \land \bigwedge_{1 \leq i \leq k, \alpha = \alpha_i \theta} t' \neq t \land \alpha_i \theta \land \bigwedge_{1 \leq j \leq i-1} \neg \alpha_j \theta),
\]

(15)

where \( r\theta \) denotes the rule obtained from \( r \) via the assignment \( \theta \) such that \( \alpha = \alpha_i \theta \). Then the external support of loop \( L \), denoted \( ES(L, \Pi, \mathbf{P}) \), is the universal closure of the first-order formula

\[
\bigvee_{\alpha \in L} \alpha \rightarrow \bigvee_{\alpha \in L} ES(\alpha, L, \Pi).
\]

Finally, the loop formulas of \( \Pi \), denoted \( LF(\Pi, \mathbf{P}) \), is the set of all formulas \( ES(L, \Pi, \mathbf{P}) \) where \( L \) is a loop of \( \Pi \).

Note that since \( \Pi \) may have an infinite number of “non-subsumed” loops, the set \( LF(\Pi, \mathbf{P}) \) may be infinite. A simple example of this case is a normal logic program with just one rule \( \{P(x) \leftarrow P(y)\} \), which has an infinite number of loops of the form \( \{P(x_1), \ldots, P(x_k)\} \) for arbitrary \( k \geq 1 \).

Also note that one main difference between the loop formulas for normal programs with variables (Chen et al., 2006) and that for LPOD is that we incorporate the meaning of the ordered disjunctions in the head into the negative body of the external support, i.e., we also need to include the conjunctions \( \alpha_i \theta \land \bigwedge_{1 \leq j \leq i-1} \neg \alpha_j \theta \) with a rule’s body in (15).

**Theorem 4** Let \( \Pi \) be an LPOD with intensional predicates \( \mathbf{P} \) and \( \mathcal{M} \) a finite \( \tau(\Pi) \)-structure. Then \( \mathcal{M} \) is a stable model of \( \Pi \) iff \( \mathcal{M} \) is a model of the first-order theory \( \{\text{Comp}(\Pi, \mathbf{P})\} \cup LF(\Pi, \mathbf{P}) \).

**Proof:** (\( \Rightarrow \)) By Theorem 3, we have that \( \mathcal{M}' \models OC(\Pi, \mathbf{P}) \) where \( \mathcal{M}' \) is the expansion of \( \mathcal{M} \) to the signature \( \tau(\Pi) \cup \{\leq_{PQ}, P, Q \in P_{\text{int}}(\Pi)\} \). Then since \( \mathcal{M}' \) satisfies (11) and (12) and \( \mathcal{M}' = \mathcal{M}|_{\tau(\Pi)} \), it follows that \( \mathcal{M} \models \text{Comp}(\Pi, \mathbf{P}) \).

Now we show that the restriction \( \mathcal{M} \) of \( \mathcal{M}' \) also satisfies all loop formulas in \( LF(\Pi, \mathbf{P}) \). Indeed, let \( L \) be a loop of \( \Pi \) such that \( \mathcal{M} \models (\bigvee_{\alpha \in L} ES(\alpha, L, \Pi, \mathbf{P}))\eta \) under some assignment \( \eta \). We show that \( \mathcal{M} \models (\bigvee_{\alpha \in L} ES(\alpha, L, \Pi, \mathbf{P}))\eta \). First, for convenience, let \( L\eta \) denote the following set: \( \{\alpha\eta \mid \alpha \in L\} \), i.e., the set of the atomic instances of the loop \( L \) under the assignment \( \eta \). In addition, similarly to \( L\eta \), let us also denote by \( L^{SAT} \eta \) as the set \( \{\alpha\eta \mid \alpha \in L, \mathcal{M} \models \alpha\eta\} \), i.e., the subset of atomic instances of \( L\eta \) that are satisfied by \( \mathcal{M} \). Now, based on the strict-partial order as induced by the comparison atoms of \( OC(\Pi, \mathbf{P}) \) (since \( \mathcal{M}' \models OC(\Pi, \mathbf{P}) \)), there is a strict-partial order \( \mathcal{P} = (\text{Dom}(\mathcal{P}), <^\mathcal{P}) \), where Dom(\mathcal{P}) = L^{SAT} \eta, such that \( <^\mathcal{P} \) is consistent with the interpretation of the comparison predicates of \( OC \mathcal{P}[\Pi] \). Then this implies that there is a bottom element \( \alpha\eta \in L^{SAT} \eta \) under \( <^\mathcal{P} \). Then since this bottom element \( \alpha\eta \) is supported in (12), it follows that \( \mathcal{M} \) satisfies

\[
\exists y_{r\theta}(\text{Body}(r\theta) \land \alpha_i \theta \land \bigwedge_{1 \leq j \leq i-1} \neg \alpha_j \theta)\eta
\]

for some rule \( r \in \Pi \) with \( \text{Head}(r) = \alpha_1 \times \ldots \times \alpha_i \times \ldots \times \alpha_k \) and such that \( \alpha_i \theta \eta = \alpha \eta \). Then this further implies that \( \mathcal{M} \) satisfies

\[
(\text{Body}(r\theta) \land \alpha_i \theta \land \bigwedge_{1 \leq j \leq i-1} \neg \alpha_j \theta)\eta'\]

14
for some extension \( \eta' \) of the assignment \( \eta \). Now we show that \( \mathcal{M} \) even satisfies

\[
(Body(r\theta) \land \bigwedge_{P(t') \in L, \ P(t) \in Pos(r\theta)} t' \neq t \land \alpha_i \theta \land \bigwedge_{1 \leq j \leq i-1} \neg \alpha_j \theta) \eta' \]

by showing that

\[
\mathcal{M} \models (\bigwedge_{P(t') \in L, \ P(t) \in Pos(r\theta)} t' \neq t) \eta' \tag{16}
\]

also holds. Indeed, since by (12), we have that all the intensional instances \( Pos(r\theta)\eta' \) are at a lower level than \( \alpha \eta \) (i.e., \( \beta \eta' <^P \alpha \eta \) for all intensional instances \( \beta \eta' \in Pos(r\theta)\eta' \)), then they cannot be in \( L^{SAT} \eta \) since under the ordering \( <^P \), we had that \( \alpha \eta \) is at a lower level than those in \( L^{SAT} \eta \setminus \{ \alpha \eta \} \) since \( \alpha \eta \) is a bottom element of \( L^{SAT} \eta \) under the partial ordering \( <^P \). Moreover, since \( \mathcal{M} \models Pos(r\theta)\eta' \), then each of those intensional instances in \( Pos(r\theta)\eta' \) also cannot be in \( L\eta \setminus L^{SAT} \eta \), since \( L\eta \setminus L^{SAT} \eta \) comprises the subset of \( L\eta \) that is false under \( \mathcal{M} \) (since recall that \( L^{SAT} \eta \) is the subset of \( L\eta \) that is true under \( \mathcal{M} \)). Therefore, since this implies that \( L\eta \cap Pos(r\theta)\eta' = \emptyset \), then it is now not difficult to see that (16) holds as well.

(\( \Leftarrow \)) We assume that \( \mathcal{M} \not\models SM(\Pi, P) \). Now, since \( \mathcal{M} \models COMP(\Pi, P) \), then it is not difficult to see that \( \mathcal{M} \models \bar{\Pi} \). Then since \( \mathcal{M} \not\models SM(\Pi, P) \) but where we have \( \mathcal{M} \models \bar{\Pi} \), it must be the case that \( \mathcal{M} \not\models \exists U(U < P \land \bar{\Pi}(U)^*) \). Thus, let \( \bar{U} \) be a structure such that \( \bar{U} \models U < P \land \bar{\Pi}(U)^* \). Now let \( L_{P \setminus U} \) be the set defined as follows: \( \{ R_i(a) \mid a \in P_i^l \setminus U_i^l, 1 \leq i \leq n \} \). Then \( L_{P \setminus U} \) corresponds to those intensional predicate instances that are not under the interpretation of the \( U \) predicates. Then we have the following lemma.

**Lemma 2** Let \( P(a) \in L_{P \setminus U} \). Then for each rule \( r \in \Pi \) with \( Head(r) = \alpha_1 \times \ldots \times \alpha_i \times \ldots \times \alpha_k \) and corresponding assignment \( \eta \) such that \( P(a) = \alpha \eta \) for some \( 1 \leq i \leq k \), we have that either:

\[
\mathcal{M} \not\models (Body(r) \land \alpha_i \land \bigwedge_{1 \leq j \leq i-1} \neg \alpha_j) \eta \tag{17}
\]

or

\[
P_{pos}(r) \eta \cap L_{P \setminus U} \neq \emptyset.
\]

That is, intuitively speaking, for each \( P(a) \in L_{P \setminus U} \), either \( P(a) \) is not “supported” or is circularly justified by some atomic instance from \( L_{P \setminus U} \).

**Proof:** Otherwise, assume that for some \( P(a) \in L_{P \setminus U} \), there exists a rule \( r \in \Pi \) with \( Head(r) = \alpha_1 \times \ldots \times \alpha_i \times \ldots \times \alpha_k \) and corresponding assignment \( \eta \) such that \( P(a) = \alpha_i \eta \) for some \( 1 \leq i \leq k \), and where we have that

\[
\mathcal{M} \models (Body(r) \land \alpha_i \land \bigwedge_{1 \leq j \leq i-1} \neg \alpha_j) \eta \tag{18}
\]

and

\[
P_{pos}(r) \eta \cap L_{P \setminus U} = \emptyset.
\]
Then this would contradict the assumption that \( \mathcal{U} \models \widehat{\Pi}(U)^* \). Indeed, since \( Pos(r) \eta \cap L_{P \setminus U} = \emptyset \), then \( \mathcal{U} \models Body(r)^* \eta \). But now, we will show that \( \mathcal{U} \not\models \text{Head}(r)^* \), which clearly contradicts the assumption that \( \mathcal{U} \models \widehat{\Pi}(U)^* \). Indeed, since \( \mathcal{M} \models \alpha_i \eta \) (i.e., by (18)), then \( \mathcal{U} \not\models (\neg \alpha_1 \times \ldots \times \neg \alpha_{j-1} \times \alpha_j^*) \eta \) for \( i < j \leq k \). Moreover, since \( \mathcal{M} \models (\neg \alpha_1 \times \ldots \times \neg \alpha_{j-1} \times \alpha_j^*) \eta \) (i.e., also by (18)) and \( \mathcal{U} \models U < P \) (i.e., since \( \mathcal{U} \models U < P \wedge \widehat{\Pi}(U)^* \) by assumption), then \( \mathcal{U} \not\models (\neg \alpha_1 \times \ldots \times \neg \alpha_{j-1} \times \alpha_j^*) \eta \) for \( 1 \leq j < i \) (since \( \alpha_j^* \to \alpha_j \equiv \neg \alpha_j \to \neg \alpha_j^* \) for \( 1 \leq j \leq k \) because \( \mathcal{U} \models U < P \)). Then it follows that we must have \( \mathcal{U} \models (\neg \alpha_1 \times \ldots \times \neg \alpha_{i-1} \times \alpha_i^*) \eta \) (i.e., the only possible “choice” left) because we need \( \mathcal{U} \models \text{Head}(r)^* \) to hold. But this is a contradiction since \( \mathcal{U} \not\models \alpha_i^* \eta \) because \( \alpha_i \eta = P(a) \in L_{P \setminus U} \). This completes the proof of Lemma 2. \( \square \)

**Lemma 3** For each \( P(a) \in L_{P \setminus U} \), there exists a rule \( r \in \Pi \) with \( \text{Head}(r) = \alpha_1 \times \ldots \times \alpha_i \times \ldots \times \alpha_k \) and corresponding assignment \( \eta \) such that:

(i) \( P(a) = \alpha_i \eta \);

(ii) \( \mathcal{M} \models (Body(r) \wedge \alpha_i \wedge \bigwedge_{1 \leq j \leq k} \neg \alpha_j) \eta \);

(iii) \( Pos(r) \eta \cap L_{P \setminus U} \neq \emptyset \).

**Proof:** (i) and (ii) follows from the fact that \( \mathcal{M} \models \text{COMP}(\Pi, vectP) \), and (iii) follows from Lemma 2 given that (ii) holds. This completes the proof of Lemma 3. \( \square \)

Then by Lemma 3 and by the finiteness of \( L_{P \setminus U} \), we have that \( L_{P \setminus U} \) must contain a loop (since each “support” for \( L_{P \setminus U} \) also depends on \( L_{P \setminus U} \)). Thus, let \( L \) be a maximal loop of \( L_{P \setminus U} \) and, without loss of generality, assume that \( L \) does not have outgoing edges to other maximal loops of \( L_{P \setminus U} \). Now there can only be two possibilities:

**Case 1:** \( L \) does not have outgoing edges to other elements of \( L_{P \setminus U} \).

Then since \( \mathcal{M} \models LF(P) \), it follows that \( L \) is externally supported from \( L_{P \setminus U} \). That is, there exists some \( P(a) \in L_{P \setminus U} \) and a rule \( r \in \Pi \) with \( \text{Head}(r) = \alpha_1 \times \ldots \times \alpha_i \times \ldots \times \alpha_k \) and corresponding assignment \( \eta \) such that:

(i) \( P(a) = \alpha_i \eta \);

(ii) \( \mathcal{M} \models (\text{Body}(r) \wedge \alpha_i \wedge \bigwedge_{1 \leq j \leq k} \neg \alpha_j) \eta \);

(iii) \( Pos(r) \eta \cap L_{P \setminus U} = \emptyset \).

Then similarly as in the proof of Lemma 2, it can be shown that this contradicts the assumption that \( \mathcal{U} \models \widehat{\Pi}(U)^* \) since again, it follows that \( \mathcal{U} \models Body(r)^* \eta \) but where \( \mathcal{U} \not\models \text{Head}(r)^* \eta \) since \( \mathcal{U} \not\models \alpha_i^* \eta \) because \( \alpha_i \eta = P(a) \in L_{P \setminus U} \).

**Case 2:** \( L \) has outgoing edges to other elements of \( L_{P \setminus U} \).

Let \( L' \) denote the set of all the elements of \( L_{P \setminus U} \) that are reachable from \( L \). Then since \( L \) is a loop without outgoing edges to other maximal loops of \( L_{P \setminus U} \), we have that \( L' \) does not contain a loop. Then similarly to above, since \( \mathcal{M} \models LF(P) \), there exists some \( P(a) \in L_{P \setminus U} \) and a rule \( r \in \Pi \) with \( \text{Head}(r) = \alpha_1 \times \ldots \times \alpha_i \times \ldots \times \alpha_k \) and corresponding assignment \( \eta \) such that:
(i) \( P(a) = \alpha_i \eta; \)

(ii) \( \mathcal{M} \models (\text{Body}(r) \land \alpha_i \land \bigwedge_{1 \leq j \leq i-1} \neg \alpha_j) \eta; \)

(iii) \( \text{Pos}(r) \eta \cap (L_{P\setminus U} \setminus L') = \emptyset. \)

Then since \( \mathcal{M} \models \text{COMP}(\Pi) \) and where \( L' \) does not contain a loop, it follows that there exists some \( Q(b) \in L' \) such that \( Q(b) \) is externally supported from \( L_{P\setminus U}. \)

\( \square \)

6. Complexity and Expressiveness

Now we investigate the complexity and expressiveness issues of LPODs. For the propositional case, we can prove that the stable model existence problem for LPODs is NP-complete. Indeed, since all normal programs are trivially LPODs (i.e., an LPOD program with just single selections on heads of rules), then it follows that propositional LPODs are NP-hard. On the other hand, since the stable models of propositional LPODs can be captured by their ordered completion (which is a SAT formula) by Theorem 3, then it follows that propositional LPODs are NP-complete.

In the following, we focus on FO LPODs. We first introduce some useful notions. Let \( \Pi_1 \) and \( \Pi_2 \) be two programs and \( \tau(\Pi_1) \subseteq \tau(\Pi_2). \) We say that \( \Pi_1 \) and \( \Pi_2 \) are equivalent under \( \tau(\Pi_1) \) if \( \Pi_1 \) and \( \Pi_2 \) have exactly the same stable models by restricting each of \( \Pi_2 \)'s stable models \( \mathcal{M} \) to \( \mathcal{M}|_{\tau(\Pi_1)}. \)

A program is positive if negation only occurs on atoms of extensional predicates. A normal logic program is local variable free if all the variables in the bodies of rules are also mentioned in their corresponding heads. The following proposition reveals an interesting relationship between normal logic programs and LPODs in their first-order cases.

**Proposition 3** Under finite structures, every local variable free normal logic program \( \Pi \) can be translated to a positive LPOD with auxiliary predicates such that these two programs are equivalent under \( \tau(\Pi). \)

**Proof:** Let \( \Pi^{\text{NORM}} \) be a local variable free FO normal program with intensional predicate symbols \( \mathcal{P}_{\text{int}}(\Pi^{\text{NORM}}). \) Then for each predicate \( P \in \mathcal{P}_{\text{int}}(\Pi^{\text{NORM}}), \) let us introduce a new predicate \( \overline{P} \) of the same arity as \( P. \) Roughly speaking, \( \overline{P} \) will encode the negative extents of \( P. \) Now we are ready to define the positive LPOD \( \Pi^{\text{LPOD}}. \)

Denote by \( (\Pi^{\text{NORM}})^{\text{pos}} \) as the following set of rules:

\[
\{ \alpha \leftarrow \beta_1, \ldots, \beta_l, \overline{\gamma_1}, \ldots, \overline{\gamma_m} \mid \alpha \leftarrow \beta_1, \ldots, \beta_l, \text{not } \gamma_1, \ldots, \text{not } \gamma_m \in \Pi^{\text{NORM}}, \\
\text{for } 1 \leq i \leq m, \text{ if } \gamma_i = P(t) \text{ and } P \in \mathcal{P}_{\text{int}}(\Pi^{\text{NORM}}) \text{ then } \overline{\gamma_i} = \overline{P}(t), \text{ otherwise } \overline{\gamma_i} = \gamma_i \}. \tag{19}
\]

Then clearly, \( (\Pi^{\text{NORM}})^{\text{pos}} \) is a positive LPOD of signature \( \tau(\Pi) \cup \{ \overline{P} \mid P \in \mathcal{P}_{\text{int}}(\Pi^{\text{NORM}}) \}. \)
Now let us define another program $\Pi'$ as follows:\footnote{Here $A_{ext}(S)$ and $A_{int}(S)$ denotes the set of extensional and intensional atoms in $S$, respectively.}

\[
\{Bd_r(x) \times \ldots \times Bd_r(x) \} \leftarrow P(x) \mid P \in \mathcal{P}_{int}(\Pi^{NORM}),
\]

$P$ has defining rules $r_1, \ldots, r_k \in \Pi^{NORM}$

\[
\cup \{ \beta \leftarrow Bd_r(x), \perp \leftarrow \neg \beta', Bd_r(x), \neg \beta' \leftarrow Bd_r(x), \perp \leftarrow \gamma', Bd_r(x) \mid r \in \Pi^{NORM}, \beta \in A_{int}(Pos(r)), \beta' \in A_{ext}(Pos(r)), \gamma \in A_{int}(Neg(r)), \gamma' \in A_{ext}(Neg(r))) \}
\]

\[
\cup \{ \langle QP (z, x) \leftarrow Bd_r(x) \mid r \in \Pi^{NORM}, Head(r) = P(x), Q(z) \in A_{int}(Pos(r)) \}
\]

\[
\cup \{ \langle PR (x, z) \leftarrow \langle PQ (x, y), \langle QR (y, z) \mid P, Q, R \in \mathcal{P}_{int}(\Pi^{NORM}) \}
\]

\[
\cup \{ \perp \leftarrow \langle PQ (x, y), \langle QP (y, x) \mid P, Q \in \mathcal{P}_{int}(\Pi^{NORM}) \}
\]

Lastly, further define one more program $\Pi''$ as follows:

\[
\{ \perp \leftarrow P(x), \overline{P}(x), \}
\]

\[
P(x) \times \overline{P}(x) \leftarrow | P \in \mathcal{P}_{int}(\Pi^{NORM}).
\]

Then we can now set $\Pi^{LPOD}$ as the union $(\Pi^{NORM})^{pos} \cup \Pi' \cup \Pi''$. Clearly, $\Pi^{LPOD}$ is a positive LPOD.

Intuitively, we have that (19)-(24) encodes the ordered completion of $\Pi^{NORM}$ within the LPOD $\Pi^{LPOD}$ itself. In particular, we note (21), which encodes the “supporting” atoms for the “body atoms” $Bd_r(x)$ (which corresponds to the body of the rule $r$), but in such a way that we avoid defining the extensional predicates in a head of a rule. On the other hand, (25) enforces that $(P(x) \rightarrow \neg \overline{P}(x)) \wedge (\overline{P}(x) \rightarrow \neg P(x))$ while (26) enforces that $(\neg P(x) \rightarrow \overline{P}(x)) \wedge (\neg \overline{P}(x) \rightarrow P(x))$, i.e., the “necessary” counterpart of (25) with respect to the extents of $\overline{P}$ being the symmetric negation of those in $P$. \qed

**Theorem 5** Let $\Pi$ be an LPOD and $M$ a finite $\tau_{ext}(\Pi)$-structure, i.e., an extensional or input database structure. Then the problem of determining if $M$ can be expanded to a stable model of $\Pi$ is NP-complete. This result remains true even if for positive LPODs.

**Proof:** (Membership) Given our reduction of $\Pi$ to the FO formula $OC(\Pi, P)$ via Theorem 3 (i.e., the reduction of $\Pi$ to its ordered completion), we have that determining if $M$ can be expanded to a model of $OC(\Pi, P)$ is the model expansion problem, which is in NP (model expansion is in fact NP-complete).

**(Hardness)** Consider the 3-color program $\Pi_{3\text{color}}$ as follows:

\[
C_1(x) \leftarrow \neg C_2(x), \neg C_3(x),
\]

\[
C_2(x) \leftarrow \neg C_1(x), \neg C_3(x),
\]

\[
C_3(x) \leftarrow \neg C_1(x), \neg C_2(x),
\]

\[
\leftarrow E(x, y), C_1(x), C_1(y),
\]

\[
\leftarrow E(x, y), C_2(x), C_2(y),
\]

\[
\leftarrow E(x, y), C_3(x), C_3(y).
\]

\[
\leftarrow E(x, y), C_3(x), C_3(y).
\]
Given a graph structure \( G = (\text{Dom}(G), V^G, E^G) \) such that \( V^G = \text{Dom}(G) \) (i.e., the vertices of \( G \)), \( G \) has a corresponding 3-coloring iff \( \Pi_{3\text{color}} \) has a stable model. It is well known that the problem of 3-coloring is NP-complete. On the other hand, we have by Proposition 3 that \( \Pi_{3\text{color}} \) can be reduced to a positive LPOD since \( \Pi_{3\text{color}} \) is a local variable free normal program. \( \square \)

An LPOD is called **almost positive** if each negated intensional atoms in the program only occurs in the bodies of some constraints. The following theorem states that almost positive LPODs precisely capture the full class of normal logic programs. That is, ordered disjunction and constraints are sufficient enough to represent negation as failure.

**Theorem 6** Every normal logic program \( \Pi \) can be translated to an almost positive LPODs with auxiliary predicates such that these two programs are equivalent under \( \tau(\Pi) \).

**Proof:** Let \( \Pi^{\text{NORM}} \) be an arbitrary normal logic program. We define a reduction from \( \Pi^{\text{NORM}} \) to an almost positive LPOD \( \Pi^{\text{LPOD}} \) by defining \( \Pi^{\text{LPOD}} \) as the following set of rules:

\[
\{ \alpha \leftarrow \beta_1, \ldots, \beta_i, \overline{x_1}, \ldots, \overline{x_m} \mid \alpha \leftarrow \beta_1, \ldots, \beta_i, \text{not } \gamma_1, \ldots, \text{not } \gamma_m \in \Pi^{\text{NORM}}, \\
\text{for } 1 \leq i \leq m, \text{if } \gamma_i = P(t) \text{ and } P \in \mathcal{P}_{\text{int}}(\Pi^{\text{NORM}}) \text{ then } \overline{x_i} = \overline{P(t)}, \text{ otherwise } \overline{x_i} = \gamma_i \} \tag{33}
\]

\[
\cup \{ \bot \leftarrow P(x), \overline{P(x)}, \mid P \in \mathcal{P}_{\text{int}}(\Pi^{\text{NORM}}) \} \tag{34}
\]

\[
\downarrow \leftarrow \text{not } P(x), \text{not } \overline{P(x)}, \tag{35}
\]

\[
\downarrow \leftarrow P'(x), \overline{P(x)}, \tag{36}
\]

\[
P'(x) \times \overline{P(x)} \leftarrow \mid P \in \mathcal{P}_{\text{int}}(\Pi^{\text{NORM}}) \}, \tag{37}
\]

where for each \( P \in \mathcal{P}_{\text{int}}(\Pi) \), we introduce two new predicate symbols \( \overline{P} \) and \( P' \). Then \( \Pi^{\text{LPOD}} \) is clearly an almost positive LPOD since negation only occurs in the constraints via (35).

Intuitively, the key here is the combination of (36) and (37), which enforces \( (\neg P(x) \rightarrow \overline{P(x)}) \wedge (\neg \overline{P(x)} \rightarrow P'(x)) \). This has the effect of “fixing” \( \overline{P(x)} \) since the “minimization” of \( \overline{P} \) implies the “expansion” of \( (P')^* \) (and vice versa). This simulates the condition that \( \overline{P(x)} \) (which encodes “not \( P(x) \)” will be fixed in the minimization of \( (P')^* \) since we also have \( (\neg P'(x)^* \rightarrow \overline{P(x)}^*) \wedge (\neg \overline{P(x)}^* \rightarrow P'(x)^*) \). On the other hand, (34) and (35) again enforces \( (P(x) \rightarrow \overline{P(x)}) \wedge (\overline{P(x)} \rightarrow \neg P(x)) \) and \( (\neg P'(x) \rightarrow \overline{P(x)}) \wedge (\overline{P(x)} \rightarrow P(x)) \), respectively. \( \square \)

**7. Preferred Stable Model Semantics**

As indicated by Brewka (2002, 2006), the stable models based on split programs are usually not sufficient to capture the preference semantics for a given LPOD, while a certain preference relation based on the degree of satisfaction has to be imposed on the stable models. Now we show how such preferred stable model semantics can be defined for our FO LPODs.

**Definition 5** Let \( r \) be a rule in a given LPOD \( \Pi \), \( M \) a \( \tau(\Pi) \)-structure, and \( \eta \) an assignment from the distinct tuple of variables \( x \) of \( r \) to \( \text{Dom}(M) \). The satisfaction degree of \( r \) under the structure
$\mathcal{M}$ and assignment $\eta$, denoted by $\text{Deg}_{\mathcal{M},\eta}(r)$, is defined as follows:

$$
\text{Deg}_{\mathcal{M},\eta}(r) :=
\begin{cases}
1 & \text{if } \mathcal{M} \not\models \text{Body}(r)\eta, \\
i & \text{otherwise}.
\end{cases}
$$

Similarly to the propositional case, if $(\mathcal{M}, \eta)$ does not satisfy the rule $r$’s body, then the default satisfaction degree is 1. Otherwise, the satisfaction degree is the “minimal” (i.e., most preferred) of the atoms in $\alpha_1 \times \ldots \times \alpha_k$ that $(\mathcal{M}, \alpha)$ satisfies. Again as in the propositional case, this encodes that we are paying for a penalty for the “least preferred” atoms in $\alpha_1 \times \ldots \times \alpha_k$ being satisfied in the sense that, the smaller the satisfaction degree, then the better it is.

**Definition 6** Let $\Pi$ be an LPOD and $\mathcal{M}_1$ and $\mathcal{M}_2$ be two stable models of $\Pi$ such that $\mathcal{M}_1|_{\tau_{ext}(\Pi)} = \mathcal{M}_2|_{\tau_{ext}(\Pi)}$. We say that $\mathcal{M}_1$ is Pareto-preferred to $\mathcal{M}_2$, denoted by $\mathcal{M}_1 > \mathcal{M}_2$, if the following two conditions hold:

1. For each rule $r \in \Pi$ and assignment $\eta$, we have that $\text{Deg}_{\mathcal{M}_1,\eta}(r) \leq \text{Deg}_{\mathcal{M}_2,\eta}(r)$;

2. There is a rule $r' \in \Pi$ and an assignment $\eta'$ such that $\text{Deg}_{\mathcal{M}_1,\eta'}(r') < \text{Deg}_{\mathcal{M}_2,\eta'}(r')$.

**Example 2** Let $\Pi$ be the following LPOD with a single rule $r$:

$$
r : \ P(x) \times Q(x) \times R(x) \leftarrow S(x)
$$

such that $S$ is the only extensional predicate. Now let $\mathcal{M}_1$ and $\mathcal{M}_2$ be two $\tau(\Pi)$-structures such that:

$$
\mathcal{M}_1 = (\{a, b\}, S^{\mathcal{M}_1} = \{a, b\}, P^{\mathcal{M}_1} = \{a\}, Q^{\mathcal{M}_1} = \{b\}, R^{\mathcal{M}_1} = \emptyset);
$$

$$
\mathcal{M}_2 = (\{a, b\}, S^{\mathcal{M}_2} = \{a, b\}, P^{\mathcal{M}_2} = \emptyset, Q^{\mathcal{M}_2} = \{b\}, R^{\mathcal{M}_2} = \{a\}).
$$

Then we have that $\mathcal{M}_1|_{\tau_{ext}(\Pi)} = \mathcal{M}_2|_{\tau_{ext}(\Pi)}$ and where the grounding of $\Pi$ under the domain $\{a, b\}$ will be the propositional program

$$
P(a) \times Q(a) \times R(a) \leftarrow S(a),
$$

$$
P(b) \times Q(b) \times R(b) \leftarrow S(b).
$$

Then based on the grounding of $\Pi$, it can be seen that both $\mathcal{M}_1$ and $\mathcal{M}_2$ are stable models of $\Pi$. In addition, since for assignments $\eta_1 : x \rightarrow a$ and $\eta_2 : x \rightarrow b$ of the variable $x$ to $\{a, b\}$, we have that $\text{Deg}_{\mathcal{M}_1,\eta_1}(r) < \text{Deg}_{\mathcal{M}_2,\eta_1}(r)$ and $\text{Deg}_{\mathcal{M}_1,\eta_2}(r) = \text{Deg}_{\mathcal{M}_2,\eta_2}(r)$. That is, then $\mathcal{M}_1 > \mathcal{M}_2$, i.e., $\mathcal{M}_1$ is Pareto-preferred to $\mathcal{M}_2$. □

Based on the notion of a Pareto-preferred stable models, we now define the notion of a preferred stable model of an LPOD.

**Definition 7** Let $\Pi$ be an LPOD and $\mathcal{M}$ its a stable model. $\mathcal{M}$ is called a preferred stable model of $\Pi$ if there does not exist another stable model $\mathcal{M}'$ of for which $\mathcal{M}' > \mathcal{M}$.
7.1 Logic Formalization of Preferred Stable Models

Now we provide a logical characterization of the preferred stable models. Let $P = P_1 \ldots P_n$ be the tuple of distinct intensional predicates in $\tau(\Pi)$ for a given program $\Pi$. and $P' = P'_1 \ldots P'_n$ the tuple of distinct new predicate symbols $P'_1 \ldots P'_n$ such that each $P'_i$'s arity matches that of $P_i$'s ($1 \leq i \leq n$). Consider a rule $r$ in $\Pi$ of the form (3). We use $\text{Deg}(r)^{P'} \leq \text{Deg}(r)^P$ to denote the following formula:

\[
\bigwedge_{1 \leq i \leq k} \forall \alpha_r ( (\text{Body}(r) \land \bigwedge_{1 \leq j \leq i} \neg \alpha_j)[P/P'] \rightarrow (\text{Body}(r) \land \bigwedge_{1 \leq j \leq i} \neg \alpha_j)), \tag{41}
\]

where $(\text{Body}(r) \land \bigwedge_{1 \leq j \leq i} \neg \alpha_j)[P/P']$ denotes the formula obtained from $(\text{Body}(r) \land \bigwedge_{1 \leq j \leq i} \neg \alpha_j)$ by replacing the occurrences of predicates from $P$ by those corresponding ones in $P'$. Roughly speaking, $\text{Deg}(r)^{P'} \leq \text{Deg}(r)^P$ is a formula that encodes all instances of $r$ under the interpretation of the predicates from $P'$ equals to or is less than the satisfaction degree than those ones from $P$.

We now extend this notion to an entire program $\Pi$ so that by $\text{Deg}(\Pi)^{P'} \leq \text{Deg}(\Pi)^P$, we denote the conjunction $\bigwedge_{r \in \Pi}(\text{Deg}(r)^{P'} \leq \text{Deg}(r)^P)$, and that we further denote by $\text{Deg}(\Pi)^{P'} < \text{Deg}(\Pi)^P$ as the conjunction

\[
(\text{Deg}(\Pi)^{P'} \leq \text{Deg}(\Pi)^P) \land \neg(\text{Deg}(\Pi)^P \leq \text{Deg}(\Pi)^{P'}). \tag{42}
\]

Clearly, $\text{Deg}(\Pi)^{P'} < \text{Deg}(\Pi)^P$ is a formula which encodes that the interpretations of $P'$ satisfy $\Pi$ in a more optimal manner, i.e., in the sense of Definition 6, than that of $\Pi$ under $P$'s interpretations. Finally, by $\text{PSM}(\Pi, P)$ (here, “PSM stands for “preferred stable model”), we denote the following second-order formula:

\[
\text{SM}(\Pi, P) \land \neg \exists P' (\text{SM}(\Pi, P)[P/P'] \land \text{Deg}(\Pi)^{P'} < \text{Deg}(\Pi)^P), \tag{43}
\]

where $\text{SM}(\Pi, P)[P/P']$ denotes the formula obtained from $\text{SM}(\Pi, P)$ by simultaneously replacing all predicates from $P$ by those corresponding ones from $P'$.

The following theorem now provides a classical logic characterization of preferred stable models of LPODs via a second-order sentence.

**Theorem 7** Let $\Pi$ be an LPOD with tuple of intensional predicates $P = P_1 \ldots P_n$ and $P' = P'_1 \ldots P'_n$ the fresh tuple of predicates matching $P$, and $M$ a $\tau(\Pi)$-structure. Then $M$ is a preferred stable model of $\Pi$ if and only if it satisfies $\text{PSM}(\Pi, P)$.

It can be showed that (43) also correctly represents Brewka’s preferred stable model semantics when we restrict it to the case of propositional LPODs.

**Theorem 8** Let $\Pi$ be an LPOD and $M$ a $\tau(\Pi)$-structure such that $M$ is a stable model of $\Pi$. Then determining if $M$ is also a preferred stable model of $\Pi$ is co-NP-complete.

**Proof:** (Hardness) Given a FO normal program $\Pi^{\text{NORM}}$ and an extensional structure $M_{\text{ext}}$ of it, we reduce the problem of determining if $M_{\text{ext}}$ cannot be expanded to a stable model of $\Pi$ to the problem of determining if a stable model of an LPOD program is an optimal stable model.
Thus, let $\Pi^{\text{NORM}}$ be any first-order normal program. Then we specify the set of rules $\Pi^{\text{NORM}}_{\text{Pos}}\cup\{P(a)\}$ such that

$$\Pi^{\text{NORM}}_{\text{Pos}}\cup\{P(a)\} := \{ \alpha \leftarrow P(a), \beta_1, \ldots, \beta_l, \text{not } \gamma_1, \ldots, \text{not } \gamma_m \mid \alpha \leftarrow \beta_1, \ldots, \beta_l, \text{not } \gamma_1, \ldots, \text{not } \gamma_m \in \Pi^{\text{NORM}} \},$$

where $P(a)$ is an atom with $P$ a new predicate symbol and $a$ a new constant symbol. Then $\Pi^{\text{NORM}}_{\text{Pos}}\cup\{P(a)\}$ is simply the program obtained from $\Pi^{\text{NORM}}$ by adding the atom $P(a)$ (i.e., propositional atom) into each of the positive bodies of the rules of $\Pi^{\text{NORM}}$. Now by $\Pi^{\text{LPOD}}$, we further specify the following set of rules such that:

$$\Pi^{\text{LPOD}} := \{ \quad P(a) \leftarrow \text{not } Q(a), \quad (44) \\
Q(a) \leftarrow \text{not } P(a), \quad (45) \\
R_1(a) \times R_2(a) \leftarrow Q(a), \quad (46) \\
\quad \leftarrow R_1(a) \quad (47)\},$$

where we assume here that $Q, R_1, \text{ and } R_2$ are new predicate symbols as well.

Intuitively, rules (44) and (45) are the defining rules for $P(a)$ and $Q(a)$, respectively, such that at least one but not both of them must be in a stable model of $\Pi^{\text{LPOD}}$. In addition, the rule (46) is an LPOD rule with $R_1(a)$ taking preference over $R_2(a)$. Finally, the constraint (47) simply enforces to choose $R_2(a)$ (which is lesser preferred over $R_1(a)$) over $R_1(a)$ so that the satisfaction degree of (46) will only be as low as 1 only if $Q(a)$ is false (since $Q(a)$ false makes the body of (46) to be false as well). Thus, if we want to get the most preferred (or optimal) stable model of $\Pi^{\text{LPOD}}$, then it must involve $Q(a)$ to be false.

**Lemma 4** Let $\mathcal{M}$ be a structure of an extension of the signature $\tau(\Pi^{\text{NORM}})$ such that it also includes the predicate symbols $P, Q, R_1, R_2$, the constant symbol $a$, and where $R^M = \emptyset$ for each $R \in P_{\text{int}}(\Pi)$ (i.e., all intensional predicates are set to empty) and $\mathcal{M} |_{\text{ext}}(\Pi) = M_{\text{ext}}$. In addition, let $P^\mathcal{M} = \emptyset$; $Q^\mathcal{M} = \{a^\mathcal{M}\}$; $R_1^\mathcal{M} = \emptyset$; and $R_2^\mathcal{M} = \{a^\mathcal{M}\}$, for each of the other predicate symbols. Then $\mathcal{M}$ is a stable model of the LPOD program $\Pi^{\text{NORM}}_{\text{Pos}}\cup\{P(a)\} \cup \Pi^{\text{LPOD}}$.

**Proof:** It is not difficult to check that $\mathcal{M}$ is a stable model of $\Pi^{\text{LPOD}}$. In addition since $P(a)$ is false, then all the bodies of the rules in $\Pi^{\text{NORM}}_{\text{Pos}}\cup\{P(a)\}$ would be false (i.e., these rules will be “deactivated”), so that adding $\Pi^{\text{NORM}}_{\text{Pos}}\cup\{P(a)\}$ to $\Pi^{\text{LPOD}}$ would still retain the condition of a stable model. This ends the proof of Lemma 4. \(\square\)

**Lemma 5** Let $\mathcal{M}$ be the structure as defined in Lemma 4. Then $\mathcal{M}$ is a preferred (or optimal) stable model of $\Pi^{\text{NORM}}_{\text{Pos}}\cup\{P(a)\} \cup \Pi^{\text{LPOD}}$ iff $\mathcal{M}_{\text{ext}}$ cannot be expanded to a model of $\Pi^{\text{NORM}}$.

**Proof:** (\(\Rightarrow\)) If $\mathcal{M}$ is not a preferred stable model of $\Pi^{\text{NORM}}_{\text{Pos}}\cup\{P(a)\} \cup \Pi^{\text{LPOD}}$, then since (46) is the only rule that has a satisfaction degree of 2 under $\mathcal{M}$, then the only way for another structure $\mathcal{M}$ to be more a preferred stable model of $\Pi^{\text{NORM}}_{\text{Pos}}\cup\{P(a)\} \cup \Pi^{\text{LPOD}}$ is to have (46) to be satisfied under degree 1. Then since we cannot choose $R_1(a)$ in the ordered disjunctive head $R_1(a) \times R_2(a)$ of (46) due to the constraint (47), then the only possible way is for the body of (46) to be false, which implies that
Q(a) must be false. Then since if Q(a) is false we have P(a) by (44), we clearly have that P(a) must be true. But then, if P(a) is true, then all the rules in Π_{Pos∪{P(a)}}^{NORM} will be “activated,” which corresponds to Π_{Pos∪{P(a)}}^{NORM} having a stable model, since there must be such a model of Π_{Pos∪{P(a)}}^{NORM} \cup Π_{LPOD} by the assumption that M is not a preferred stable model of Π_{Pos∪{P(a)}}^{NORM} \cup Π_{LPOD}.

(⇐) Let M′ be a τ(Π_{NORM})-structure expansion of M_{ext} that is a stable model of Π_{NORM}. Then we construct an expansion M" of M′ to the predicates in {Π, Q1, R1, R2} and the constant a by setting: P^M = \{a^M\}; Q^M = \emptyset; R_1^M = \emptyset; R_2^M = \emptyset; and where a^M can be some arbitrary domain element of M′. Then since Q(a) is false under M", we have that (46) will have a satisfaction degree of 1 under M", so that M" is a more Pareto-preferred stable model of Π_{Pos∪{P(a)}}^{NORM} \cup Π_{LPOD} compared to that of M. This completes the proof of Lemma 5. □

(Membership) To prove this part, we show that the problem of determining if a structure is an optimal stable model can be reduced to the problem of determining if an extensional structure has no expansion that is a stable model of some LPOD (which is a co-NP problem since it is the complement of its NP counterpart of determining the existence of a stable model). Thus, let Π be an LPOD with tuple of intensional predicates P = P_1 \ldots P_n and M a τ(Π)-structure that is a stable model of Π. Now let P' = P'_1 \ldots P'_n be a tuple of new predicates that matches P. Then define Π' as the following set of LPOD rules:

\[
\begin{align*}
\{ & (\text{Head}(r) \leftarrow \text{Body}(r))[P/P'] \mid r : \text{Head}(r) \leftarrow \text{Body}(r) \in \Pi \} \\
\cup & \{ \not \phi, (\text{Body}(r), \not \alpha_1, \ldots, \not \alpha_i)[P/P'] \mid r \in \Pi, \\
 & \phi \in (\text{Body}(r) \cup \{\not \alpha_1, \ldots, \not \alpha_i\}), \text{Head}(r) = \alpha_1 \times \ldots \times \alpha_i \times \ldots \times \alpha_k \} \\
\cup & \{ s \leftarrow (\text{Body}(r), \not \alpha_1, \ldots, \not \alpha_i)[P/P'], \text{Body}(r), \not \alpha_1, \ldots, \not \alpha_i \\
 & \mid r \in \Pi, \text{Head}(r) = \alpha_1 \times \ldots \times \alpha_i \times \ldots \times \alpha_k \} \\
\cup & \{ \not s \},
\end{align*}
\]

where:

- (Head(r) \leftarrow Body(r))[P/P'] in (48) denotes the rule obtained from r by simultaneously replacing the occurrences of predicates from P by those corresponding ones from P;
- For a φ ∈ Body(r) \cup \{not α_1, …, not α_i\}, the \not φ in (49) denotes the complement of φ;
- The (Body(r), not α_1, …, not α_i)[P/P'] also in (49) denotes the body atoms obtained by simultaneously replacing the occurrences of predicates from P by those corresponding ones from P'.

Now let us take a close look at the intuition behind the LPOD program Π'. In a nutshell, Π' is program that looks for a more optimal stable model by encoding this as the interpretation of the P predicates. Indeed, (48) is a set of rules that is exactly Π apart from where each occurrence of the predicates from P are replaced by those corresponding ones from P'. On the other hand, each rule in (49) encodes (41) since if (Body(r), not α_1, …, not α_i)[P/P'] holds, then it cannot be the case that any of the atoms φ ∈ (Body(r) \cup \{not α_1, …, not α_i\}) can be false (i.e., as encoded by the condition that the complement \not φ of φ cannot hold). In addition, we also have that (50) together with the single rule (51) encodes

\[\neg(Deg(Π)^P \leq Deg(Π)^{P'}),\]
since if it is the case that
\[(\text{Body}(r), \text{not } \alpha_1, \ldots, \alpha_i)[P/P'], \text{Body}(r), \text{not } \alpha_1, \ldots, \text{not } \alpha_i \text{Body}(r), \text{not } \alpha_1, \ldots, \text{not } \alpha_i\]
holds (which encodes that \(r\) under \(P\) has a lower satisfaction degree from that under \(P'\)), then we
must have \(s\), and where we must derive \(s\) by (51). Thus, this combination of (50) and (51) encodes
that there is at least one rule instance under \(P\) that is at a lower satisfaction degree than that under \(P'\).
Therefore, it can now be seen that the set of rules (49), (50), and (51) simply encodes the formula
\(\text{Deg}(\Pi)^P < \text{Deg}(\Pi)^{P'}\), i.e., as defined by (42).

Lemma 6  With the program \(\Pi'\) as defined above (i.e., the union of the set or rules (48), (49), (50),
and (51)), let us select our set of extensional predicates as \(P_{ext}(\Pi) \cup P_{int}(\Pi)\) and our intensional
predicates as \(\{P_1', \ldots, P_n'\}\), so that we can now view \(M\) as an extensional structure under \(\Pi\). Then
\(M\) is an optimal stable model of \(\Pi\) iff there is no expansion of \(M\) that is a stable model of \(\Pi\).
This ends the proof of Theorem 8. \(\square\)

Proposition 4  Let \(\Pi\) be an LPOD and \(M\) a corresponding extensional structure. The problem of
determining if \(M\) can be expanded to a preferred stable model of \(\Pi\) is NP-complete.
Proof: \(\Pi\) has a preferred stable model iff \(\Pi\) has a stable model and where the problem of determining if \(\Pi\) has a stable model is \(NP\)-complete. \(\square\)

8. Splitting LPODs

Splitting is an important and desirable property for logic programs, which makes possible to reduce
the computation of a large program’s stable models to the computations of smaller programs’ stable
models (Ferraris et al., 2009; Lifschitz & Turner, 1994; Zhang, 1999). In this section, we study this
property for our proposed LPODs. In particular, we consider under what conditions, an LPOD may
be split into smaller LPODs.

First, we define the notion of positive predicate dependency graph of an LPOD. Given an LPOD
\(\Pi\), its positive predicate dependency graph \(PG^{+}_{\Pi} = (V, E^{PG^{+}_{\Pi}})\) is specified as follows:
\[V = \{P \mid \exists r \in \Pi \text{ such that } Head(r) = \alpha_1 \times \ldots \times \alpha_k \text{ and } P(x) = \alpha_i \text{ for some } 1 \leq i \leq k\}\]
and edges
\[E^{PG^{+}_{\Pi}} = \{(P, Q) \mid \exists r \in \Pi \text{ with } Head(r) = \alpha_1 \times \ldots \times \alpha_k \text{ and } Pos(r) = \{\beta_1, \ldots, \beta_l\} \text{ such that } P(x) = \alpha_i \text{ and } Q(y) = \beta_j \text{ for some } 1 \leq i \leq k \text{ and } 1 \leq j \leq l\}\]
Then the following result is now a direct consequence of the splitting lemma from (Ferraris et al.,
2009).

Theorem 9  Let \(\Pi\) be an LPOD and \(P \subseteq P_{int}(\Pi)\) and \(Q \subseteq P_{int}(\Pi)\) be disjoint tuples of intensional
predicates. If each strongly connected component of \(PG^{+}_{\Pi}\) is either a subset of \(P\) or a subset of \(Q\),
then
\[\models \text{SM}(\Pi, PQ) \equiv \text{SM}(\Pi, P) \land \text{SM}(\Pi, Q)\]
Proof: This follows from the fact that our definition here of the positive dependency graph of an LPOD is the same as that in (Ferraris et al., 2009) for general first-order theories when restricted to the syntax of first-order sentences of the form $\bar{\Pi}$ (see formula (4)). □

Now our next result reveals that the splitting theorem also holds for the preferred stable models of LPODs, while its proof is not a trivial extension from that of Theorem 9.

Theorem 10 Let $\Pi$ be an LPOD and $P \subseteq \mathcal{P}_{\text{int}}(\Pi)$ and $Q \subseteq \mathcal{P}_{\text{int}}(\Pi)$ be disjoint tuples of intensional predicates. If each strongly connected component of $\mathcal{P}^+_Q$ is the same as that in (Ferraris et al., 2009) for general first-order theories when restricted to $P$ and $Q$, then

$$\models \text{PSM}(\Pi, PQ) \equiv \text{PSM}(\Pi, P) \land \text{PSM}(\Pi, Q).$$

Proof: ($\Rightarrow$) Assume $\mathcal{M} \models \text{PSM}(\Pi, PQ)$. Then by definition, we have that

$$\mathcal{M} \models \text{SM}(\Pi, PQ) \land \neg \exists P'Q'([\text{SM}(\Pi, PQ)]PQ/P'Q') \land \text{Deg}(\Pi)^{PQ} < \text{Deg}(\Pi)^{PQ}.$$ 

Since by Theorem 9 we have that

$$\models \text{SM}(\Pi, PQ) \equiv \text{SM}(\Pi, P) \land \text{SM}(\Pi, Q),$$

we further have

$$\mathcal{M} \models (\text{SM}(\Pi, P) \land \text{SM}(\Pi, Q)) \land \neg \exists P'Q'([\text{SM}(\Pi, P) \land \text{SM}(\Pi, Q)]PQ/P'Q') \land \text{Deg}(\Pi)^{PQ} < \text{Deg}(\Pi)^{PQ}.$$ 

Then it follows that

$$\mathcal{M} \models (\text{SM}(\Pi, P) \land \text{SM}(\Pi, Q)) \land \exists P'Q'([\text{SM}(\Pi, P) \land \text{SM}(\Pi, Q)]PQ/P'Q') \land \text{Deg}(\Pi)^{PQ} < \text{Deg}(\Pi)^{PQ},$$

and

$$\mathcal{M} \models (\text{SM}(\Pi, P) \land \text{SM}(\Pi, Q)) \land \forall P'Q'([\text{SM}(\Pi, P) \land \text{SM}(\Pi, Q)]PQ/P'Q') \rightarrow \neg \text{Deg}(\Pi)^{PQ} < \text{Deg}(\Pi)^{PQ}.$$ 

Now suppose $P = P_1 \ldots P_s$, $P' = P'_1 \ldots P'_s$, $Q = Q_1 \ldots Q_t$, and $Q' = Q'_1 \ldots Q'_t$, then by setting the interpretations on $P_i$ and $Q_j$ to be those of $P_i^M$ and $Q_j^M$ in (55) and (56), respectively, and where we also have that $\mathcal{M} \models \text{SM}(\Pi, P) \land \text{SM}(\Pi, Q)$, then it follows that:

$$\mathcal{M} \models \text{SM}(\Pi, P) \land \text{SM}(\Pi, Q) \land \forall P'[\text{SM}(\Pi, P)[P/P'] \rightarrow \neg \text{Deg}(\Pi)^{P'} < \text{Deg}(\Pi)^{P}],$$

$$\land \forall Q'[\text{SM}(\Pi, Q)[Q/Q'] \rightarrow \neg \text{Deg}(\Pi)^{Q'} < \text{Deg}(\Pi)^{Q}],$$

$$\mathcal{M} \models \text{SM}(\Pi, P) \land \forall P'[\text{SM}(\Pi, P)[P/P'] \rightarrow \neg \text{Deg}(\Pi)^{P'} < \text{Deg}(\Pi)^{P}],$$

$$\land \forall Q'[\text{SM}(\Pi, Q)[Q/Q'] \rightarrow \neg \text{Deg}(\Pi)^{Q'} < \text{Deg}(\Pi)^{Q}],$$

$$\mathcal{M} \models \text{PSM}(\Pi, P) \land \text{PSM}(\Pi, Q),$$

25
where (59) follows from (57) and (58) via the definitions of PSM(Π, P) and PSM(Π, Q), respectively.

(⇔) Now we assume $M \models \text{PSM}(Π, P) \land \text{PSM}(Π, Q)$. Again by the definition, we have that

$$M \models \text{SM}(Π, P) \land \lnot \exists P'[\text{SM}(Π, P)[P/P'] \land \text{Deg}(Π)^{P'} < \text{Deg}(Π)^P]$$

$$\land \text{SM}(Π, Q) \land \lnot \exists Q'[\text{SM}(Π, Q)[Q/Q'] \land \text{Deg}(Π)^Q < \text{Deg}(Π)^Q],$$

which further implies that

$$M \models \text{SM}(Π, P) \land \lnot \exists P'[\text{SM}(Π, P)[P/P'] \land \text{Deg}(Π)^P < \text{Deg}(Π)^P]$$

$$\land \text{SM}(Π, Q) \land \lnot \exists Q'[\text{SM}(Π, Q)[Q/Q'] \land \text{Deg}(Π)^Q < \text{Deg}(Π)^Q].$$

This follows that

$$M \models \text{SM}(Π, P) \land \text{SM}(Π, Q)$$

$$\land \forall P'Q'[\text{SM}(Π, P)[P/P'] \land \text{SM}(Π, Q)[Q/Q'] \rightarrow (\lnot \text{Deg}(Π)^P < \text{Deg}(Π)^P$$

$$\lor \lnot \text{Deg}(Π)^Q < \text{Deg}(Π)^Q)].$$

Then by Theorem 9 and by viewing $P'$ and $Q'$ as just the relabelling of $P$ and $Q$, respectively, then it follows that:

$$M \models \text{SM}(Π, PQ) \land \forall P'Q'[\text{SM}(Π, PQ)[PQ/P'Q'] \rightarrow (\lnot \text{Deg}(Π)^P < \text{Deg}(Π)^P$$

$$\lor \lnot \text{Deg}(Π)^Q < \text{Deg}(Π)^Q)],$$

from which we conclude that

$$M \models \text{SM}(Π, PQ) \land \lnot \exists P'Q'[\text{SM}(Π, PQ)[PQ/P'Q'] \land (\text{Deg}(Π)^P < \text{Deg}(Π)^P$$

$$\land \text{Deg}(Π)^Q < \text{Deg}(Π)^Q)],$$

$$M \models \text{SM}(Π, PQ) \land \lnot \exists P'Q'[\text{SM}(Π, PQ)[PQ/P'Q'] \land \text{Deg}(Π)^P < \text{Deg}(Π)^P Q',$$

and

$$M \models \text{PSM}(Π, PQ).$$

This completes the proof of Theorem 10. □

9. Conclusions

Preference plays an important role in commonsense reasoning, while developing an effective yet expressive mechanism of handling preference in Answer Set Programming is technically challenging, e.g., (Brewka, Truszczyński, & Woltran, 2010; Delgrande & Tompits, 2004). In this paper we have developed a formulation of first-order LPODs which may be viewed as a natural generalization of propositional LPODs.

The proposed both the second-order stable model semantics and the progression semantics capture two important aspects of LPODs: their relationship to classical logic as well as the underlying reasoning feature involving ordered disjunction on the first-order level. The translations from
LPODs to first-order sentences/theories extend the previous work on ordered completion and loop formula for normal logic programs, and we argue that these translations will be useful in developing an effective solver for first-order LPODs. The complexity and expressiveness results confirm that LPODs remain in NP and the hardness holds even if for positive LPODs, while almost positive LPODs capture the full class of normal logic programs. Our logic characterization of preferred stable model semantics reveals that we can eventually use a classical second-order sentence to precisely represent the preference relation among stable models so that such first-order preference semantics may be formalized in a unified way as LPOD first-order stable model semantics. As for the case of general stable model theories, under certain conditions, LPODs also satisfy the modular property via splitting even if the preference on stable models is taken into account.

For future work, we are planning to develop a first-order LPOD solver based on the results presented in this paper. Another interesting work is to extend our semantics for first-order LPODs by allowing both ordered and unordered disjunction in the heads of rules, as discussed in (Cabalar, 2011) for the propositional case, and explore their logical and computational properties and possible applications in practical domains.

References


