This paper investigates two fixpoint approaches for minimal model reasoning with disjunctive logic programs \( \mathcal{P} \). The first one, called \textit{model generation}, is based on an operator \( \mathcal{F}_{\mathcal{P}}^{\text{INT}} \) defined on sets of Herbrand interpretations whose least fixpoint is logically equivalent to the set of minimal Herbrand models of the program. The second approach, called \textit{state generation}, uses a fixpoint operation \( \mathcal{F}_{\mathcal{G}} \) based on \textit{hyperresolution}. It operates on disjunctive Herbrand states, and its least fixpoint is the set of logical consequences of \( \mathcal{P} \), the so-called \textit{minimal model state} of the program. We establish a useful relationship between hyperresolution by \( \mathcal{F}_{\mathcal{G}} \) and model generation by \( \mathcal{F}_{\mathcal{P}}^{\text{INT}} \). Then we investigate the problem of \textit{continuity} of the two operators \( \mathcal{F}_{\mathcal{P}}^{\text{INT}} \) and \( \mathcal{F}_{\mathcal{G}} \). It is known that the operator \( \mathcal{F}_{\mathcal{G}} \) is continuous, and so it reaches its least fixpoint in at most \( \omega \) iterations. On the other hand, the question of whether \( \mathcal{F}_{\mathcal{P}}^{\text{INT}} \) is continuous has been open. We show by a counterexample that \( \mathcal{F}_{\mathcal{P}}^{\text{INT}} \) is not continuous. Nevertheless, we prove that it converges towards its least fixpoint in at most \( \omega \) iterations, too, as follows from the relationship that we show exists between hyperresolution and model generation. We define an iterative version of \( \mathcal{F}_{\mathcal{P}}^{\text{INT}} \) that computes the \textit{perfect model semantics} of stratified disjunctive logic programs. On each stratum of the program, this operator converges in at most \( \omega \) iterations. Model generations for the \textit{stable semantics} and the \textit{partial stable semantics} are respectively achieved by using this iterative operator together with the \textit{evidential transformation} and the \textit{3-S transformation}. © Elsevier Science Inc., 1997
1. INTRODUCTION

The semantics of a disjunctive logic program has been characterized by its set of minimal Herbrand models or, equivalently, by the collection of all positive disjunctions that hold in every minimal Herbrand model of the program (see [15]). This collection is called the minimal model state of the program.

These equivalent semantic definitions gave rise to two alternative ways of computing the meaning of a program. The first one, denoted here by model generation, relies on a fixpoint operator $\mathcal{F}_\varphi^M$ that operates on sets of Herbrand interpretations and whose least fixpoint is the set of minimal Herbrand models of the program. This operator was originally introduced by Fernández and Minker in [5] (see also [3, 6]) for the case of disjunctive logic programs without function symbols.\(^1\)

The second approach, developed by Minker and Rajasekar [16], is based on a fixpoint operator $\mathcal{F}_\varphi^s$ defined on sets of positive disjunctions called states. This operator uses hyperresolution (cf. Robinson [21]) to construct the model state of the program as its least fixpoint. We refer to this approach as state generation.

In this paper, we further investigate the nature of model and state generations and prove some useful relationships between them. In particular, we investigate the problem of continuity of the two operators $\mathcal{F}_\varphi^s$ and $\mathcal{F}_\varphi^M$. It is known that the operator $\mathcal{F}_\varphi^s$ is continuous [16], and so it reaches its least fixpoint in at most $\omega$ iterations. On the other hand, the question of whether $\mathcal{F}_\varphi^M$ is continuous when applied to arbitrary disjunctive logic programs (with function symbols) has been open. We argue that this problem is ill-posed, as the domain of $\mathcal{F}_\varphi^M$ is not closed under least upper bounds. We then give a natural extension of $\mathcal{F}_\varphi^M$ in terms of a new operator $\mathcal{F}_\varphi^{INT}$ defined on a more suitable domain and reformulate the continuity problem for the new operator. We prove, by means of a counterexample, that $\mathcal{F}_\varphi^{INT}$ is not continuous. Nevertheless, from a relationship that we show exists between state generation by $\mathcal{F}_\varphi^s$ and model generation by $\mathcal{F}_\varphi^{INT}$, we prove that $\mathcal{F}_\varphi^{INT}$ reaches its least fixpoint in at most $\omega$ iterations, too.

We define an iterative version of $\mathcal{F}_\varphi^{INT}$ that computes the perfect model semantics of stratified disjunctive logic programs. On each stratum of the program, this operator converges in at most $\omega$ iterations. Due to the characterizations of the stable and the partial stable (and so the well-founded) semantics respectively presented in [4] and [22], this iterative operator can be used to construct the stable models and partial stable models (the well-founded model in particular) of normal disjunctive logic programs.

The paper consists of the following sections. Section 2 presents some basic definitions and notations. Section 3 summarizes the main properties of state generation. Section 4 surveys model generation, defines the new operator $\mathcal{F}_\varphi^{INT}$, and shows that it is not continuous. This section also proves the existence of minimal models of disjunctive logic programs. Section 5 establishes some useful relationships between model generation and state generation. As a consequence of these relationships, it is proven that $\mathcal{F}_\varphi^{INT}$ reaches its least fixpoint in at most $\omega$ iterations. Section 6 provides an iterated version of $\mathcal{F}_\varphi^{INT}$ that constructs the perfect models of stratified disjunctive logic programs, and is used to generate the

\(^1\) Further approaches to model generation can be found in [1, 13, 14, 17].
(partial) stable models of normal disjunctive logic programs. Section 7 concludes the paper.

2. BASIC DEFINITIONS AND NOTATIONS

Given a first-order language, a disjunctive logic program $\mathcal{P}$ consists of logical inference rules of the form

$$A_1 \lor \cdots \lor A_k \leftarrow B_i \land \cdots \land B_m,$$

(2.1)

where $A_i, i \in \langle 1, k \rangle$, and $B_i, i \in \langle 1, m \rangle$, are (positive) atoms in the language and $k, m \in \mathbb{N}_0$. A rule is called a fact if $m = 0$. The set of all ground instances of the rules and facts in $\mathcal{P}$ is denoted by $\text{gnd}(\mathcal{P})$. A disjunctive logic program $\mathcal{P}$ is called a disjunctive deductive database if the program does not contain any function symbols. Two important concepts are associated with a logic program $\mathcal{P}$.

(i) First, a subset $I \subseteq \text{HB}_\mathcal{P}$ of the Herbrand base $\text{HB}_\mathcal{P}$ is called a Herbrand interpretation. The set of all Herbrand interpretations is denoted by $\mathcal{H}_\mathcal{P}$. A Herbrand interpretation $I$ is called a model of $\mathcal{P}$ if for all ground rules of the form (2.1) in $\mathcal{P}$, it holds that $(B_i | i \in \langle 1, m \rangle) \subseteq I$ implies that there exists $i \in \langle 1, k \rangle$ such that $A_i \in I$.

The set of all Herbrand models of $\mathcal{P}$ is denoted by $\mathcal{M}_\mathcal{P}$, and sometimes by $\mathcal{M}_\mathcal{D}(\mathcal{P})$.

(ii) Second, the set $\text{DHB}_\mathcal{P}$ of all positive ground disjunctions $A_1 \lor \cdots \lor A_k$, $k \in \mathbb{N}_0$, that can be formed by atoms $A_i \in \text{HB}_\mathcal{P}$, is called the disjunctive Herbrand base of $\mathcal{P}$. A subset $S \subseteq \text{DHB}_\mathcal{P}$ is called a disjunctive Herbrand state.

For reasoning with disjunctive deductive logic programs, two main approaches have been developed. The first approach generates the set $\mathcal{M}_\mathcal{P}$ of all minimal Herbrand models of $\mathcal{P}$. A model $I$ is called minimal if there is no model $I'$ of $\mathcal{P}$ for which $I' \subseteq I$. The second approach uses hyperresolution for deriving the set $\text{MS}_\mathcal{P}$ of all positive ground disjunctions $C \in \text{DHB}_\mathcal{P}$ that are logical consequences of the logic program, i.e., the disjunctions that hold in all models of the logic program. This set is called the minimal model state of the logic program [16].

Below, we review some important notation and results on partial orderings and lattices and on fixpoint theory on complete lattices, cf., e.g., [11, 2].

Let $\mathcal{S} = \langle S, \leq \rangle$, where $S$ is a set and $\leq$ is a binary relation on $S$. $\mathcal{S}$ is called a partial ordering on $S$ if $\leq$ is reflexive, transitive, and antisymmetric. A partial ordering $\mathcal{S}$ is called a complete lattice if for all subsets $X \subseteq S$, there exists a least upper bound, denoted by $\text{lub}(X)$, and a greatest lower bound, denoted by $\text{glb}(X)$, in $S$. A very common example of a complete lattice that will occur in this paper is $\mathcal{S} = \langle 2^U, \subseteq \rangle$ where $2^U$ is the power set of some set $U$, ordered by set inclusion, and least upper bounds and greatest lower bounds are given by the operations union and intersection, respectively.

---

2 By $\mathbb{N}_+$ we denote the set $\{1, 2, 3, \ldots\}$ of positive natural numbers, whereas $\mathbb{N}_0$ denotes the set $\{0, 1, 2, \ldots\}$ of all natural numbers, $\langle n, m \rangle$ denotes the interval $\{n, n + 1, \ldots, m\}$ of natural numbers.

3 The notations $\text{lub}(X)$ and $\text{glb}(X)$ are justified since for every set $X$, its least upper bound and its greatest lower bound are unique if they exist.
Certain mappings \( \mathcal{F}: L \rightarrow L \) on a complete lattice \( \mathcal{E} = \langle L, \leq \rangle \) are of special interest: monotonic mappings which preserve the partial ordering and continuous mappings which commute with the least upper bound operator of the lattice. \( \mathcal{F} \) is called monotonic if \( x_1 \leq x_2 \) implies \( \mathcal{F}(x_1) \leq \mathcal{F}(x_2) \), for all elements \( x_1, x_2 \in L \). A subset \( X \subseteq L \) is called directed if for every finite subset \( X' \) of \( X \), there exists an upper bound of \( X' \) in \( X \). \( \mathcal{F} \) is called continuous if \( \mathcal{F}(\text{lub}(X)) = \text{lub}(\mathcal{F}(X)) \) for all directed subsets \( X \subseteq L \), where \( \mathcal{F}(X) = \{ \mathcal{F}(x) | x \in X \} \). Every continuous mapping on a complete lattice is also monotonic. The converse, however, it not true.

The elements of a lattice which are invariant under a mapping on the lattice are called fixpoints of the mapping. That is, an element \( x \in L \) is called a fixed point of \( \mathcal{F} \) if \( \mathcal{F}(x) = x \). A fixpoint \( x \in L \) of \( \mathcal{F} \) is called the least fixpoint of \( \mathcal{F} \), denoted by \( \text{lfp}(\mathcal{F}) \), if \( x \leq x' \) for all fixpoints \( x' \) of \( \mathcal{F} \).

Given a mapping \( \mathcal{F}: L \rightarrow L \) on a complete lattice \( \mathcal{E} = \langle L, \leq \rangle \), we can define its ordinal powers—corresponding to repeated applications of the mapping—as new mappings on the same lattice by using transfinite recursion on \( \alpha \) in conjunction with the least upper bound operation on the lattice. The ordinal powers \( \mathcal{F}^\alpha \) of \( \mathcal{F} \) are defined as follows:

\[
\mathcal{F}^0(x) = x, \\
\mathcal{F}^\alpha(x) = \mathcal{F}(\mathcal{F}^{\alpha-1}(x)), \quad \text{for a successor ordinal } \alpha, \\
\mathcal{F}^\alpha(x) = \text{lub}\{\mathcal{F}^\beta(x) | \beta < \alpha\}, \quad \text{for a limit ordinal } \alpha.
\]

For the important special case of the bottom element \( \bot = \text{glb}(L) \) of \( \mathcal{E} \), the ordinal powers \( \mathcal{F}^\alpha \) of \( \mathcal{F} \) are lattice elements given by \( \mathcal{F}^\alpha = \mathcal{F}^\alpha(\bot) \). The well-known theorem of Knaster and Tarski, as contained in Lloyd [11] (based on [10, 26]), relates the fixpoints of a monotonic mapping on a complete lattice to the ordinal powers: If \( \mathcal{F} \) is monotonic, then the collection of fixpoints of \( \mathcal{F} \) forms a complete lattice and so \( \mathcal{F} \) has a unique least fixpoint \( \text{lfp}(\mathcal{F}) \). For any ordinal \( \alpha \), it holds that \( \mathcal{F}^\alpha \leq \text{lfp}(\mathcal{F}) \), and there exists an ordinal \( \alpha \), such that \( \mathcal{F}^\alpha = \text{lfp}(\mathcal{F}) \) for all \( \alpha' \geq \alpha \). The smallest such \( \alpha \) is called the closure ordinal of \( \mathcal{F} \). If \( \mathcal{F} \) is continuous, then \( \mathcal{F}^\omega = \text{lfp}(\mathcal{F}) \) (see [9]).

3. STATE GENERATION

The fixpoint semantics of a disjunctive logic program is based on a disjunctive consequence operator \( \mathcal{F}_D \) given in [16] (see also [12]). The definition of \( \mathcal{F}_D \) uses the concept of hyperresolution, which was first introduced by Robinson, cf. [21].

**Definition 3.1** (Consequence operator \( \mathcal{F}_D \)). Let \( \mathcal{P} \) be a disjunctive logic program and let \( S \subseteq \text{DHB}_{\mathcal{P}} \) be a disjunctive Herbrand state. The disjunctive consequence operator \( \mathcal{F}_D: \text{DHB}_{\mathcal{P}} \rightarrow \text{DHB}_{\mathcal{P}} \) is defined as

\[
\mathcal{F}_D(S) = S \cup \{ C \vee C_1 \vee \cdots \vee C_m \mid C, C_1, \ldots, C_m \in \text{DHB}_{\mathcal{P}} \text{ and there is a rule } C \leftarrow B_1 \land \cdots \land B_m \in \text{gnd}(\mathcal{P}) : \forall i \in \langle 1, m \rangle : B_i \lor C_i \in S \}.
\]

The notation \( \text{lfp}(\mathcal{F}) \) is justified since for every mapping \( \mathcal{F} \), its least fixpoint is unique if it exists.
The above definition differs from that in [16] as follows. First, the result of applying the operator ~ to a disjunctive Herbrand state S contains the state S as well as all disjunctions that can be derived from S and the rules in \( \mathcal{P} \) by one step of hyperresolution. Second, since we consider only ground disjunctions, and throughout the paper we implicitly consider these disjunctions as sets of atoms, we do not need to use factorization, i.e., we do not take the “smallest factor” of the resolvents \( C \lor C_1 \lor \cdots \lor C_m \).

Minker and Rajasekar show in [16] that the disjunctive consequence operator \( \mathcal{T}_p \) is continuous with respect to the complete lattice \( \mathcal{O} = \langle 2^\mathbb{DHB}_p, \subseteq \rangle \) on disjunctive Herbrand states, and hence also monotonic.

Following the general setting given in Section 2, the ordinal powers of the disjunctive consequence operator are defined with respect to the lattice \( \mathcal{O} \), whose bottom element is \( \bot = \emptyset \), as follows.

**Definition 3.2** (Ordinal powers for \( \mathcal{T}_p \) on \( \mathcal{O} = \langle 2^\mathbb{DHB}_p, \subseteq \rangle \)). Let \( \mathcal{P} \) be a disjunctive logic program:

(i) For \( S \subseteq \mathbb{DHB}_p \), the ordinal powers \( \mathcal{T}_p^{\alpha} (S) \) are defined by

\[
\mathcal{T}_p^{\alpha \uparrow 0} (S) = S,
\mathcal{T}_p^{\alpha \uparrow \alpha} (S) = \mathcal{T}_p (\mathcal{T}_p^{\alpha - 1} (S)), \text{ for a successor ordinal } \alpha,
\mathcal{T}_p^{\alpha \uparrow \alpha} (S) = \bigcup_{\beta < \alpha} \mathcal{T}_p^{\beta \uparrow \alpha} (S), \text{ for a limit ordinal } \alpha.
\]

(ii) The ordinal powers \( \mathcal{T}_p^{\alpha} \uparrow \alpha \) are defined by \( \mathcal{T}_p^{\alpha} \uparrow \alpha = \mathcal{T}_p^{\alpha \uparrow \alpha} (\emptyset) \).

**Example 3.1** (Disjunctive transitive closure). Consider the disjunctive logic program

\[
\mathcal{P} = \{ \text{path}(X,Y) \leftarrow \text{arc}(X,Z) \land \text{path}(Z,Y), \text{path}(X,Y) \leftarrow \text{arc}(X,Y), \text{arc}(a,b) \lor \text{arc}(a,c), \text{arc}(b,d), \text{arc}(c,d) \},
\]

that consists of the classical transitive closure rules and some disjunctive facts for the \( \text{arc} \)-relation, cf. Figure 1, where definite arcs are represented by solid arrows and indefinite arcs are represented by dotted arrows.

The original powers \( S_n = \mathcal{T}_p^{\alpha \uparrow \alpha} (\emptyset), n \geq 1 \), are given by

\[
S_1 = \{ \text{arc}(a,b) \lor \text{arc}(a,c), \text{arc}(b,d), \text{arc}(c,d) \},
S_2 = S_1 \cup \{ \text{arc}(a,b) \lor \text{path}(a,c), \text{path}(a,b) \lor \text{arc}(a,c), \text{path}(b,d), \text{path}(c,d) \},
S_3 = S_2 \cup \{ \text{path}(a,b) \lor \text{path}(a,c), \text{path}(a,d) \lor \text{arc}(a,c), \text{path}(a,d) \lor \text{arc}(a,b), \text{path}(a,d) \lor \text{path}(a,c), \text{path}(a,d) \lor \text{path}(a,b) \},
S_4 = S_3 \cup \{ \text{path}(a,d) \},
S_n = S_4, \text{ for all } n \geq 4.
\]
Thus, \( F_\varphi \uparrow \omega = F_\varphi \uparrow 4 \). This example shows that \( F_\varphi \) can also derive definite facts from disjunctive facts.

A positive disjunction \( C' \) is called a subdisjunction of another positive disjunction \( C \) if every atom appearing in \( C' \) also appears in \( C \). \( C' \) is called a proper subdisjunction of \( C \) if \( C' \neq C \) and \( C' \) is a subdisjunction of \( C \). For a disjunctive Herbrand state \( S \), let

\[
\text{can}(S) = \{ C \in S | \exists C' \in S: C' \text{ is a proper subdisjunction of } C \},
\]

\[
\text{exp}(S) = \{ C \in DHB,\varphi | \exists C' \in S: C' \text{ is a subdisjunction of } C \}.
\]

can(\( S \)) and exp(\( S \)) are respectively the canonization and the expansion of \( S \), and it holds that can(\( S \)) \( \subseteq \) \( S \) \( \subseteq \) exp(\( S \)). Two disjunctive Herbrand states \( S_1 \) and \( S_2 \) are called equivalent if exp(\( S_1 \)) = exp(\( S_2 \)). This is denoted by \( S_1 \equiv_\varphi S_2 \).

The minimal model state \( MS_\varphi \) is equivalent to the least fixpoint of the disjunctive consequence operator \( F_\varphi \), and it can be derived as \( MS_\varphi = \text{exp}(F_\varphi \uparrow \omega) \), as was proven by Minker and Rajasekar in [16].

**Theorem 3.1 (Characterization of \( MS_\varphi \) [16]).** Let \( \varphi \) be a disjunctive logic program. Then

\[
MS_\varphi \equiv_\varphi \text{lfp}(F_\varphi) = F_\varphi \uparrow \alpha,
\]

where \( \alpha \) is the closure ordinal of \( F_\varphi \) on \( \varnothing = (2^{DHB,\varphi}, \subseteq) \).

4. MODEL GENERATION

The model generation approach constructs the minimal Herbrand models of a given logic program.

For a definite logic program \( \varphi \) (without disjunctions) and a given Herbrand interpretation \( I \), the classical consequence operator \( F_\varphi \) of van Emden and Kowalski [27] computes the Herbrand interpretation \( J \) that consists of the head atoms of all rules in \( \text{gnd}(\varphi) \), such that the bodies of the rules are satisfied by \( I \). The unique minimal Herbrand model of the program is precisely the least fixpoint of this operator.

For disjunctive logic programs, model generation deals with sets of Herbrand interpretations. For conciseness, we abbreviate the set of Herbrand interpretations as coin (collection of interpretations). We use the following two operations min and exp for a coin \( I \):

\[
\text{min}(I) = \{ I \in I | \exists J \in I: J \subseteq I \},
\]

\[
\text{exp}(I) = \{ I \in 2^{I} | \exists J \in I: J \subseteq I \}.
\]
Note that we use the operator \( \exp \) for states as well as for coins, but it will be clear from the context to which case we are referring.

A coin \( \mathcal{J} \) is called **canonical** if it does not contain two different Herbrand interpretations \( I, J \) such that \( I \subseteq J \), i.e., if \( \mathcal{J} = \min(\mathcal{J}) \). A coin \( \mathcal{J} \) is called **expanded** if for each Herbrand interpretation \( I \in \mathcal{J} \), it also contains all Herbrand interpretations that are supersets of \( I \), i.e., if \( \mathcal{J} = \exp(\mathcal{J}) \).

For example, for the Herbrand base \( HB = \{ a, b, c \} \) and the coin \( \mathcal{J} = \{\{a\}, \{a, b\}, \{b, c\}\} \), we get \( \min(\mathcal{J}) = \{\{a\}, \{b, c\}\} \) and \( \exp(\mathcal{J}) = \{\{a\}, \{a, b, c\}\} \).

The following consequence operator \( \mathcal{F}^{\text{INT}} \) generalizes the operator \( \mathcal{F} \) of van Emden and Kowalski to the case of disjunctive logic programs \( \mathcal{P} \). \( \mathcal{F}^{\text{INT}} \) maps coins to coins.

**Definition 4.1 (Consequence operators \( \mathcal{F}^{\text{INT}} \) and \( \mathcal{F}^{\text{M}} \)).** Let \( \mathcal{P} \) be a disjunctive logic program:

(i) The consequence operator

\[
\mathcal{F}^{\text{INT}} : 2^{\mathcal{P}^+} \rightarrow 2^{\mathcal{P}^+}
\]

operates on sets \( \mathcal{J} \in \mathcal{S}^{\mathcal{P}^+} \) of Herbrand interpretations:

\[
\mathcal{F}^{\text{INT}}(\mathcal{J}) = \bigcup_{I \in \mathcal{J}} \mathcal{M}_{\mathcal{P}}(\mathcal{F}_p(I)).
\]

(ii) The consequence operator

\[
\mathcal{F}^{\text{M}} : 2^{\mathcal{P}^+} \rightarrow 2^{\mathcal{P}^+}
\]

operates on sets \( \mathcal{J} \in 2^{\mathcal{P}^+} \) of Herbrand interpretations:

\[
\mathcal{F}^{\text{M}}(\mathcal{J}) = \min(\mathcal{F}^{\text{INT}}(\mathcal{J})).
\]

For each \( I \in \mathcal{I} \), \( \mathcal{F}^{\text{INT}}(\mathcal{J}) \) contains all Herbrand interpretations \( J \) which extend \( I \) and at the same time satisfy all ground rules of \( \mathcal{P} \) whose bodies are satisfied by \( I \). Thus, from each interpretation, several interpretations may be derived. Furthermore, the result \( \mathcal{F}^{\text{INT}}(\mathcal{J}) \) is expanded.

The operator \( \mathcal{F}^{\text{M}} \) was originally introduced by Fernández and Minker [5] (see also [3, 6]) to compute the minimal Herbrand models of a disjunctive deductive database (i.e., a disjunctive logic program with no function symbols). In this case, due to the fact that the Herbrand base of a database is finite, coins are finite sets of finite interpretations and so have the property that \( \mathcal{F}^{\text{INT}}(\mathcal{J}) = \exp(\mathcal{F}^{\text{M}}(\mathcal{J})) \).

**Example 4.1 (Consequence operator \( \mathcal{F}^{\text{M}} \)).** Consider the disjunctive deductive database \( \mathcal{P} \) of Example 3.1:

(i) For the Herbrand interpretation \( I = \emptyset \), the rules whose bodies are satisfied by \( I \) are precisely the facts of \( \mathcal{P} \). Thus, \( \mathcal{F}^{\text{M}}(\emptyset) = \mathcal{J} = \{I_1, J_1\} \) is the set of minimal Herbrand interpretations of the facts of \( \mathcal{P} \), where

\[
I_1 = \{\text{arc}(a, b), \text{arc}(b, d), \text{arc}(c, d)\}, \\
J_1 = \{\text{arc}(a, c), \text{arc}(b, d), \text{arc}(c, d)\}.
\]

---

5 For an arbitrary disjunctive logic program \( \mathcal{P} \), a coin can be an infinite set and it can contain infinite interpretations. Thus, this property may not hold for \( \mathcal{P} \), since there may exists some \( I \in \mathcal{F}^{\text{INT}}(\mathcal{J}) \) for which there is no minimal interpretation in \( \mathcal{F}^{\text{INT}}(\mathcal{J}) \) contained in \( I \).
(ii) For the Herbrand interpretations in $\mathcal{I}$, the bodies of some ground instances of the second rule are satisfied. Thus, $\mathcal{F}_\varphi^M$ extends $I_1$, $J_1$. We get $\mathcal{F}_\varphi^M(\mathcal{I}) = \mathcal{I}_2 = \{I_2, J_2\}$, where

$$
I_2 = I_1 \cup \{\text{path}(a, b), \text{path}(b, d), \text{path}(c, d)\},
$$

$$
J_2 = J_1 \cup \{\text{path}(a, c), \text{path}(b, d), \text{path}(c, d)\}.
$$

Fernández and Minker (cf. [16]) investigated some of the properties of the consequence operator $\mathcal{F}_\varphi^M$ based on the following subsumption relation $\subseteq$ defined for coins $\mathcal{I}, \mathcal{J} \in 2^x_{\varphi^+}$:

$$
\mathcal{I} \subseteq \mathcal{J} \iff \forall \mathcal{J} \in \mathcal{I}: \exists \mathcal{I} \in \mathcal{J}: \mathcal{I} \subseteq \mathcal{J},
$$

e.g., for $\mathcal{I} = \{(a), (a, b), (b, c)\}$ and $\mathcal{J} = \{(a), (c)\}$, we get $\mathcal{I} \subseteq \mathcal{J}$.

The pair $\Theta = \langle 2^x_{\varphi^+}, \subseteq \rangle$, however, is only a quasi-ordering, since on $2^x_{\varphi^+}$ the relation $\subseteq$ is reflexive and transitive, but not antisymmetric. To overcome this, one can work with equivalence classes of coins as follows. Two coins $\mathcal{I}, \mathcal{J} \in 2^x_{\varphi^+}$ are called equivalent with respect to the quasi-ordering $\subseteq$, denoted by $\mathcal{I} \equiv \mathcal{J}$, if they subsume each other, i.e.,

$$
\mathcal{I} \equiv \mathcal{J} \iff \mathcal{I} \subseteq \mathcal{J} \text{ and } \mathcal{J} \subseteq \mathcal{I}.
$$

Fernández and Minker then restricted the domain of $\mathcal{F}_\varphi^M$ to $\Theta_{\min} = \langle 2^x_{\varphi^+}, \subseteq \rangle$, where $2^x_{\varphi^+}$ consists of all canonical coins. This subdomain does form a partial ordering. For disjunctive databases, this partial ordering is also complete since the Herbrand base of such a database is finite. Based on this, they proved the monotonicity of $\mathcal{F}_\varphi^M$ on $\Theta_{\min}$ and the following characterization of the set $\mathcal{M}_\varphi^M$ of minimal Herbrand models of $\varphi$ in terms of the consequence operator $\mathcal{F}_\varphi^M$ and its ordinal powers $\mathcal{F}_\varphi^M \uparrow \alpha = \mathcal{F}_\varphi^M \uparrow \alpha(\{\varphi\})$ with respect to $\Theta_{\min}$.

**Theorem 4.1 (Characterization of $\mathcal{M}_\varphi^M$ [6]).** Let $\varphi$ be a disjunctive deductive database. Then

$$
\mathcal{M}_\varphi^M = \text{lfp}(\mathcal{F}_\varphi^M) = \mathcal{F}_\varphi^M \uparrow \alpha,
$$

where $\alpha$ is the closure ordinal of $\mathcal{F}_\varphi^M$ on $\Theta_{\min} = \langle 2^x_{\varphi^+}, \subseteq \rangle$.

Since the Herbrand base of a disjunctive deductive database is finite, the operator $\mathcal{F}_\varphi^M$ reaches its least fixpoint in a finite number of iterations.

**Example 4.2 (Disjunctive transitive closure).** For the disjunctive logic program $\varphi$ of Example 3.1, all ordinal powers $\mathcal{F}_\varphi^M \uparrow n = \{I_n, J_n\}$, $n \geq 1$, consist of two Herbrand interpretations, where $I_1, J_1$ and $I_2, J_2$ have been given in Example 4.1 and

$$
I_3 = I_2 \cup \{\text{path}(a, d)\},
$$

$$
J_3 = J_2 \cup \{\text{path}(a, d)\}.
$$

It holds that $I_n = I_3$ and $J_n = J_3$, for all $n \geq 3$. Thus, the set of minimal Herbrand models of $\varphi$ is given by $\mathcal{M}_\varphi^M = \{I_3, J_3\}$.

In principle, one can apply the operator $\mathcal{F}_\varphi^M$ to disjunctive logic programs containing function symbols. The question of whether or not the operator is
continuous in this extended context has been open. Notice, however, that in the extended context, the partial ordering \( \mathcal{C}_{\text{min}} = \langle 2^{\mathcal{C}_{\text{min}}}, \subseteq \rangle \) is not a complete lattice (the least upper bound of a collection \( X \) of canonical coins may not exist). Hence the question of continuity in this subdomain is ill-posed.

We reformulate the continuity problem in a more appropriate domain \( \mathcal{C}_{\text{exp}} = \langle 2^{\mathcal{C}_{\text{exp}}}, \subseteq \rangle \), where \( 2^{\mathcal{C}_{\text{exp}}} \) consists of all expanded coins. It is easy to show that in this subdomain of expanded coins, the relation \( \subseteq \) reduces to superset inclusion, since for all \( \mathcal{I}, \mathcal{J} \in 2^{\mathcal{C}_{\text{exp}}} \), it holds that

\[
\mathcal{I} \subseteq \mathcal{J} \iff \exp(\mathcal{I}) \supseteq \exp(\mathcal{J}).
\]

This shows that \( \mathcal{C}_{\text{exp}} = \langle 2^{\mathcal{C}_{\text{exp}}}, \subseteq \rangle \) is a complete lattice, and that the least upper bound and the greatest lower bound of a set \( X \subseteq 2^{\mathcal{C}_{\text{exp}}} \) of expanded coins are given by

\[
lub_{\text{exp}}(X) = \bigcap_{\mathcal{I} \in X} \mathcal{I}, \quad \text{glb}_{\text{exp}}(X) = \bigcup_{\mathcal{I} \in X} \mathcal{I}.
\]

The bottom element of \( \mathcal{C}_{\text{exp}} \) is the coin \( \bot_{\text{exp}} = \mathcal{C}_{\text{exp}} \) consisting of all Herbrand interpretations. Note that both the intersection and the union of a set \( X \) of expanded coins are expanded coins.\(^6\)

In the domain \( \mathcal{C}_{\text{exp}} \), we can work with the operator \( \mathcal{I}_{\text{NT}} \) that produces expanded coins. For every coin \( \mathcal{I} \), it holds that \( \mathcal{I} \subseteq \mathcal{I}_{\text{NT}}(\mathcal{I}) \), since every Herbrand interpretation \( I \in \mathcal{I}_{\text{NT}}(\mathcal{I}) \) is derived by extending a Herbrand interpretation \( l \in \mathcal{I} \), where \( l \in I \). Moreover, the operator \( \mathcal{I}_{\text{NT}} \) is monotonic on \( \mathcal{C} = \langle 2^{\mathcal{C}_{\text{exp}}}, \subseteq \rangle \) (and hence in \( \mathcal{C}_{\text{exp}} \)):

\[
\mathcal{I} \subseteq \mathcal{J} \text{ implies } \mathcal{I}_{\text{NT}}(\mathcal{I}) \subseteq \mathcal{I}_{\text{NT}}(\mathcal{J}).
\]

Following the general setting given in Section 2, the ordinal powers of \( \mathcal{I}_{\text{NT}} \) are defined with respect to the complete lattice \( \mathcal{C}_{\text{exp}} \) as follows.

**Definition 4.2 (Ordinal powers for \( \mathcal{I}_{\text{NT}} \) on \( \mathcal{C}_{\text{exp}} = \langle 2^{\mathcal{C}_{\text{exp}}}, \subseteq \rangle \)).** Let \( \mathcal{P} \) be a disjunctive logic program:

(i) For \( \mathcal{I} \in 2^{\mathcal{C}_{\text{exp}}} \), the ordinal powers \( \mathcal{I}_{\text{NT}} \uparrow^0(\mathcal{I}) \) are defined by

\[
\mathcal{I}_{\text{NT}} \uparrow^0(\mathcal{I}) = \mathcal{I},
\]

\[
\mathcal{I}_{\text{NT}} \uparrow^a(\mathcal{I}) = \mathcal{I}_{\text{NT}}(\mathcal{I}_{\text{NT}} \uparrow^{a-1}(\mathcal{I})), \text{ for a successor ordinal } \alpha,
\]

\[
\mathcal{I}_{\text{NT}} \uparrow^a(\mathcal{I}) = \bigcap_{\beta < \alpha} \mathcal{I}_{\text{NT}} \uparrow^\beta(\mathcal{I}), \text{ for a limit ordinal } \alpha.
\]

(ii) The ordinal powers \( \mathcal{I}_{\text{NT}} \uparrow^a \) are defined by \( \mathcal{I}_{\text{NT}} \uparrow^a = \mathcal{I}_{\text{NT}} \uparrow^a(\bot_{\text{exp}}) \).

For disjunctive deductive databases, it can be shown that the respective ordinal powers of the operators \( \mathcal{I}_{\text{NT}} \) and \( \mathcal{I}_{\text{M}} \) are equivalent, i.e., \( \mathcal{I}_{\text{NT}} \uparrow^a = \mathcal{I}_{\text{M}} \uparrow^a \).

Due to the fact that for single applications of the operators it holds that \( \mathcal{I}_{\text{NT}}(\mathcal{I}) = \exp(\mathcal{I}_{\text{M}}(\mathcal{I})) \), it can be concluded that \( \mathcal{I}_{\text{NT}} \uparrow^a = \exp(\mathcal{I}_{\text{M}} \uparrow^a) \) and \( \mathcal{I}_{\text{M}} \uparrow^a = \exp(\mathcal{I}_{\text{NT}} \uparrow^a) \).

\(^6\) This does not hold for canonical coins.
\[ \min(\mathcal{F}_\mathcal{P}^{INT} \uparrow \alpha). \] From this, the equivalence of the ordinal powers of the operators \( \mathcal{F}_\mathcal{P}^{INT} \) and \( \mathcal{F}_\mathcal{M}^{\alpha} \) follows.

We show now by means of a counterexample that the operator \( \mathcal{F}_\mathcal{P}^{INT} \) is not continuous.

**Example 4.3** (Noncontinuity of model generation by \( \mathcal{F}_\mathcal{P}^{INT} \)). Consider the disjunctive logic program

\[
\mathcal{P} = \{ a_0 \leftarrow a(X) \land a(Y) \land \text{diff}(X,Y), \\
a(X) \leftarrow a(Y) \land \text{term}(X), \\
d(f(c)) \leftarrow d(c) \}
\]

and the complete lattice \( \mathcal{E}_{exp} = (2^{\mathcal{E}_{exp}}, \supseteq) \). Consider the following Herbrand interpretation for the predicate symbol \text{diff}:

\[
\mathcal{I}_{\text{diff}} = \{ \text{diff}(f^m(c), f^n(c)) \mid m, n \in \mathbb{N}_0, m \neq n \},
\]

and the following Herbrand interpretation for the predicate symbol \text{term}:

\[
\mathcal{I}_{\text{term}} = \{ \text{term}(f^n(c)) \mid n \in \mathbb{N}_0 \},
\]

and let \( \mathcal{I}_* = \mathcal{I}_{\text{diff}} \cup \mathcal{I}_{\text{term}} \). The Herbrand interpretation \( \mathcal{I}_{\text{diff}} \) says that for \( \text{diff}(X,Y) \) to be true, it must hold that \( X \) and \( Y \) denote different terms. The Herbrand interpretation \( \mathcal{I}_{\text{term}} \) says that for \( \text{term}(X) \) to be true, it must hold that \( X \) is a term of the form \( f^n(c) \). Let \( a_n \) denote the atom \( a(f^n(c)) \), for all \( n \in \mathbb{N}_+ \), and let \( \mathcal{I}_a = \{ a_n \mid n \in \mathbb{N}_+ \} \).

We are interested in the following expanded coins \( \mathcal{J}_n \), where all \( \mathcal{I} \in \mathcal{J}_n \) contain the Herbrand interpretation \( \mathcal{I}_* \):

\[
\mathcal{J}_n = \exp(\{ \{ a_m \} \cup \mathcal{I}_a \mid m \geq n \}), \quad \text{for} \ n \in \mathbb{N}_+.
\]

For the set \( X = \{ \mathcal{J}_n \mid n \in \mathbb{N}_+ \} \) of coins, it holds that \( \mathcal{J}_k \supseteq \mathcal{J}_n \), for all \( k \leq n \), i.e., \( X \) forms a decreasing chain \( \mathcal{J}_n \supseteq \mathcal{J}_k \supseteq \cdots \supseteq \mathcal{J}_0 \supseteq \mathcal{J}_- \). This especially implies that \( X \) also is a directed set of coins. The least upper bound of \( X \) is given by

\[
lub_{\exp}(X) = \bigcap_{n \in \mathbb{N}_+} \mathcal{J}_n = \{ \mathcal{I} \subseteq \text{HB}_{\mathcal{P}} \mid \mathcal{I} \cap \mathcal{I}_a \text{ is infinite and } \mathcal{I}_* \subseteq \mathcal{I} \},
\]

i.e., it consists exactly of those Herbrand interpretations \( \mathcal{I} \) which contain infinitely many of the atoms \( a_n \) and also contain the whole Herbrand interpretation \( \mathcal{I}_* \). Thus, each of these Herbrand interpretations \( \mathcal{I} \) contains at least two different atoms \( a_n = a(f^n(c)) \) and \( a_k = a(f^k(c)) \) and the corresponding atom \( \text{diff}(f^n(c), f^k(c)) \). This implies that the first rule of the logic program will extend \( \mathcal{I} \) with the atom \( a_0 \), and that the second rule of the logic program will extend \( \mathcal{I} \) with all other atoms \( a_m \in \mathcal{I}_a \). Thus, it holds that

\[
\mathcal{F}_\mathcal{P}^{INT}(\lub_{\exp}(X)) = \text{ex}(\{ \{ a_0 \} \cup \mathcal{I}_a \cup \mathcal{I}_* \})).
\]

\[ \text{The disjunctive logic program } \mathcal{P} \text{ is range-restricted, i.e., every variable that occurs in a rule also occurs in the positive body of the rule. The atom term}(X) \text{ in the second rule is only used for making } \mathcal{P} \text{ range-restricted.} \]
On the other hand, it holds that

\[ \mathcal{F}^{\text{INT}}(\mathcal{I}_n) = \exp(\{I_a \cup I_b\}), \quad \text{for all } n \in \mathbb{N}_+, \]

i.e., all coins \( \mathcal{F}^{\text{INT}}(\mathcal{I}_n), n \in \mathbb{N}_+, \) are identical. Thus,

\[ \lub_{\exp}(\{\mathcal{F}^{\text{INT}}(\mathcal{I}_n) | n \in \mathbb{N} \pm \}) = \bigcap_{n \in \mathbb{N}_+} \mathcal{F}^{\text{INT}}(\mathcal{I}_n) = \exp(\{I_a \cup I_b\}). \]

This shows that the operator \( \mathcal{F}^{\text{INT}} \) does not commute with \( \lub_{\exp} \), i.e.,

\[ \mathcal{F}^{\text{INT}}(\lub_{\exp}(X)) \neq \lub_{\exp}(\{\mathcal{F}^{\text{INT}}(\mathcal{I}_n) | n \in \mathbb{N}_+ \}). \]

This implies also that \( \mathcal{F}^{\text{INT}} \) is not continuous with respect to the lattice \( \mathbb{C}_{\exp} = (2^{\mathbb{E}_{\exp}}, \subseteq) \) nor with respect to \( \mathcal{O} = (2^\mathcal{F}_{\exp}, \subseteq) \).

In the next section, we show that even though \( \mathcal{F}^{\text{INT}} \) is not continuous, it reaches its least fixpoint in at most \( \omega \) iterations. Furthermore, based on the operator \( \mathcal{F}^{\text{INT}} \), we are able to generalize the characterization of the minimal Herbrand model semantics that Fernández and Minker obtained for disjunctive deductive databases to arbitrary disjunctive logic programs, namely, that \( \mathcal{M}_{\mathcal{F}} \subseteq \text{lfp}(\mathcal{F}^{\text{INT}}) = \mathcal{F}^{\text{INT}} \uparrow \omega \) (see Theorem 5.2).

The following lemma is used in the next section. It shows that the set of minimal Herbrand models of a disjunctive logic program is equivalent to the set of all Herbrand models of the logic program with respect to the quasi-ordering \( \sqsubseteq \). In other words, this means that each Herbrand model of the logic program contains a minimal Herbrand model of the program. This result is not trivial since, for an arbitrary disjunctive logic program \( \mathcal{P} \), it could have been the case that there were an infinitely decreasing chain \( M_1 \supseteq M_2 \supseteq M_3 \supseteq \cdots \) of models of \( \mathcal{P} \).

**Lemma 4.1 (Existence of minimal models).** Let \( \mathcal{P} \) be a disjunctive logic program. Then

\[ \mathcal{M}_{\mathcal{P}} \equiv_{\subseteq} \mathcal{M}_{\text{int}}. \]

**Proof.** For each \( M \in \mathcal{M}_{\text{int}} \), we want to show that there exists some \( I^* \in \mathcal{M}_{\mathcal{P}} \) such that \( I^* \subseteq M \). Let \( \mathcal{M} = \{I \in \mathcal{M}_{\text{int}} | I \subseteq M\} \). We prove below that each chain \( \mathcal{I} \subseteq \mathcal{M} \) has a lower bound \( I^* \) in \( \mathcal{M} \). Using Zorn's lemma, this implies that the coin \( \mathcal{M} \) contains a minimal element. Hence, \( \mathcal{M}_{\mathcal{P}} \subseteq \mathcal{M}_{\text{int}} \). The statement \( \mathcal{M}_{\text{int}} \subseteq \mathcal{M}_{\mathcal{P}} \) is trivial, since \( \mathcal{M}_{\mathcal{P}} \subseteq \mathcal{M}_{\text{int}} \).

Now let \( \mathcal{I} \subseteq \mathcal{M} \) be a chain of Herbrand models. Obviously, the Herbrand interpretation

\[ I^* = \bigcap_{I \in \mathcal{I}} I \]

is a lower bound of \( \mathcal{I} \) and \( I^* \subseteq M \). We want to show that \( I^* \) is also a model of \( \mathcal{P} \), i.e., \( I^* \in \mathcal{M}_{\mathcal{P}} \). Consider a rule

\[ A_1 \lor \cdots \lor A_k \leftarrow B_1 \land \cdots \land B_m \in \text{gnd}(\mathcal{P}) \]

\[ \text{i.e., for each } I, J \in \mathcal{I}, \text{ it either holds that } I \subseteq J \text{ or } J \subseteq I. \]
such that \( I^* = B_1 \land \cdots \land B_m \). Since \( \mathcal{H} \) is a chain, we get that
\[
\mathcal{J} = \{ I \cap \{ A_1, \ldots, A_k \} | I \in \mathcal{J} \}
\]
is a chain, too. Since all elements of \( \mathcal{J} \) have a finite cardinality between 0 and \( k \), there exists an element \( J' \in \mathcal{J} \) with minimal cardinality. Since \( \mathcal{J} \) is a chain, it holds that
\[
\forall J \in \mathcal{J}: J' \subseteq J.
\]
Consider the Herbrand interpretation \( I' \in \mathcal{H} \), such that \( J' = I' \cap \{ A_1, \ldots, A_k \} \).
Since \( I^* \models B_1 \land \cdots \land B_m \) and \( I^* \subseteq I' \), it holds that \( I' \models B_1 \land \cdots \land B_m \). From \( I' \in \mathcal{M}_{\mathcal{H}, \mathcal{P}} \), we get \( J' = I' \cap \{ A_1, \ldots, A_k \} \), i.e., \( J' = I' \cap \{ A_1, \ldots, A_k \} \neq \emptyset \). Since all elements of \( \mathcal{J} \) are supersets of \( J' \), we get
\[
\forall J \in \mathcal{J}: J \subseteq J'.
\]
This shows that \( J' \subseteq I^* \), i.e., \( I^* \models A_1 \lor \cdots \lor A_k \). Thus, \( I^* \) is a Herbrand model of \( \mathcal{P} \) contained in \( \mathcal{J}_M \). □

5. RELATIONSHIPS BETWEEN STATE GENERATION AND MODEL GENERATION

For disjunctive logic programs \( \mathcal{P} \), the minimal model state \( MS_{\mathcal{P}} \) and the set \( \mathcal{M}_{\mathcal{P}} \) of minimal Herbrand models of \( \mathcal{P} \) are dual concepts, i.e., they can be derived from each other. According to [12], it holds that
\[
\mathcal{M}_{\mathcal{P}} = \mathcal{M}(MS_{\mathcal{P}}),
\]
\[
MS_{\mathcal{P}} = MS(\mathcal{M}_{\mathcal{P}}),
\]
where the dualization operations are defined by
\[
MS(\mathcal{J}) = \{ C \in DHB_{\mathcal{P}} | \forall I \in \mathcal{J}: I \models C \},
\]
\[
\mathcal{M}(S) = \min(\{ I \in \mathcal{H}_{\mathcal{P}} | \forall C \in S: I \models C \}).
\]
This resembles the duality between the conjunctive and the disjunctive normal form of boolean formulas, since \( MS_{\mathcal{P}} \) represents the conjunction of its disjunctions, whereas \( \mathcal{M}_{\mathcal{P}} \) represents the disjunction of the conjunctions formed by its models.

This duality relates the least fixpoints \( MS_{\mathcal{P}} \) and \( \mathcal{M}_{\mathcal{P}} \) of the consequence operators \( \mathcal{F}_{\mathcal{P}} \) and \( \mathcal{F}_{\mathcal{P}}^{INT} \), respectively (see Figure 2). In the following, we also compare the intermediate results of the respective fixpoint iterations. Corollary 5.1 establishes the relationship between the ordinal powers \( \mathcal{F}_{\mathcal{P}}^{\uparrow n} \) and \( \mathcal{F}_{\mathcal{P}}^{\uparrow INT} \uparrow n \).

Lemma 5.1 (Connection between \( \mathcal{F}_{\mathcal{P}}^{\uparrow} \) and \( \mathcal{F}_{\mathcal{P}}^{\uparrow INT} \) [23]). Let \( \mathcal{P} \) be a disjunctive logic program, let \( \mathcal{J} \) be a coin, and let \( S \subseteq DHB_{\mathcal{P}} \) be a disjunctive Herbrand state, such that \( \mathcal{J} \subseteq \mathcal{M}(S) \). Then
\[
\mathcal{F}_{\mathcal{P}}^{INT}(\mathcal{J}) \subseteq \mathcal{M}(\mathcal{F}_{\mathcal{P}}^{\uparrow}(S)),
\]
\[
\mathcal{F}_{\mathcal{P}}^{\uparrow}(S) \subseteq MS(\mathcal{F}_{\mathcal{P}}^{INT}(\mathcal{J})).
\]
\[ T^0_p \uparrow \emptyset \xrightarrow{\text{Mod}} T^0_{p^\text{INT}} \uparrow \bot_{\text{esp}} \]

\[ T^0_p \uparrow n \xrightarrow{\text{Mod}} T^0_{p^\text{INT}} \uparrow n \]

\[ T^0_p \uparrow \omega \equiv \omega \xrightarrow{\text{Mod}} T^0_{p^\text{INT}} \uparrow \omega \equiv \omega \]

**FIGURE 2.** Comparing fixpoint computations.

**PROOF.** (i) Assume \( M \) is a Herbrand interpretation, such that

\[ M \in \mathcal{F}^\text{INT}_p (\mathcal{I}) \]

We will show that \( M \in \mathcal{M}_\mathcal{D}(\mathcal{F}^\text{INT}_p (S)) \).

There is some \( I \in \mathcal{I} \), such that \( M \in \mathcal{M}_\mathcal{D}(\mathcal{F}^\text{INT}_p (I)) \). Thus, \( I \subseteq M \). Since \( \mathcal{I} \subseteq \mathcal{M}_\mathcal{D}(S) \), we get \( I \in \mathcal{M}_\mathcal{D}(S) \). Since \( I \) is a model of \( S \), we get that \( M \) is a model of \( S \). For each \( C \in \mathcal{F}^\text{INT}_p (S) \setminus S \), there is a rule

\[ C' \leftarrow B_1 \wedge \cdots \wedge B_m \in \text{gnd}(\mathcal{P}) \]

and there are facts \( B_i \vee C_i \in S \), such that \( C = C' \vee C_1 \vee \cdots \vee C_m \). If \( I \models C_1 \vee \cdots \vee C_m \), then \( M \models C \). Otherwise, since \( I \models B_i \vee C_i \), \( 1 \leq i \leq m \), we get \( I \models B_1 \wedge \cdots \wedge B_m \). Thus, \( C' \in \mathcal{F}^\text{INT}_p (I) \). Since \( M \in \mathcal{M}_\mathcal{D}(\mathcal{F}^\text{INT}_p (I)) \), we get that \( M \models C' \). Thus, again \( M \models C \).

Summarizing, \( M \) is a model of \( \mathcal{F}^\text{INT}_p (S) \setminus S \).

This shows that \( M \in \mathcal{M}_\mathcal{D}(\mathcal{F}^\text{INT}_p (S)) \), i.e.,

\[ \mathcal{F}^\text{INT}_p (\mathcal{I}) \subseteq \mathcal{M}_\mathcal{D}(\mathcal{F}^\text{INT}_p (S)) \]

(ii) Application of \( \text{MS} \) to (i) yields

\[ \text{MS}(\mathcal{M}_\mathcal{D}(\mathcal{F}^\text{INT}_p (S))) \subseteq \text{MS}(\mathcal{F}^\text{INT}_p (\mathcal{I})) \]

Since \( \mathcal{F}^\text{INT}_p (S) \subseteq \text{MS}(\mathcal{M}_\mathcal{D}(\mathcal{F}^\text{INT}_p (S))) \), we get

\[ \mathcal{F}^\text{INT}_p (S) \subseteq \text{MS}(\mathcal{F}^\text{INT}_p (\mathcal{I})) \]

From this connection, we can derive some relationships between state generation and model generation, i.e., between the ordinal powers \( \mathcal{F}^\text{INT}_p \uparrow \alpha \) of the operator \( \mathcal{F}^\text{INT}_p \) on disjunctive Herbrand states and the ordinal powers \( \mathcal{F}^\text{INT}_p \uparrow \alpha \) of the operator \( \mathcal{F}^\text{INT}_p \) on sets of Herbrand interpretations.

**Theorem 5.1** (State generation vs. model generation). Let \( \mathcal{P} \) be a disjunctive logic program, let \( S \subseteq \text{DnB}_p \) be a disjunctive Herbrand state of \( \mathcal{P} \), and let \( \mathcal{I} \) be a coin, such that \( \mathcal{I} \subseteq \mathcal{M}_\mathcal{D}(S) \) and \( S \subseteq \text{MS}(\mathcal{I}) \). For all ordinals \( \alpha \), it holds that

\[ \mathcal{F}^\text{INT}_p \uparrow ^\alpha (\mathcal{I}) \subseteq \mathcal{M}_\mathcal{D}(\mathcal{F}^\text{INT}_p \uparrow ^\alpha (S)) \]

\[ \mathcal{F}^\text{INT}_p \uparrow ^\alpha (S) \subseteq \text{MS}(\mathcal{F}^\text{INT}_p \uparrow ^\alpha (\mathcal{I})) \]
Proof. (i) The first set inclusion is shown by induction on $\alpha$:

$\alpha = 0$: The induction basis is shown by

$$\mathcal{F}^\alpha_{\mu}(\mathcal{J}) = \mathcal{E} \subseteq \mathcal{M}(\mathcal{S}) = \mathcal{M}(\mathcal{F}^\alpha_{\mu}(\mathcal{S})).$$

$\alpha \rightarrow \alpha + 1$: Using the induction assumption for $\alpha$ and the monotonicity of $\mathcal{F}^\alpha_{\mu}$, we get

$$\mathcal{F}^{\alpha + 1}_{\mu}(\mathcal{J}) = \mathcal{F}^\alpha_{\mu}(\mathcal{F}^{\alpha + 1}_{\mu}(\mathcal{J})) \subseteq \mathcal{M}(\mathcal{F}^{\alpha + 1}_{\mu}(\mathcal{S})).$$

Let $\mathcal{J}' = \mathcal{M}(\mathcal{F}^{\alpha + 1}_{\mu}(\mathcal{S}))$ and $\mathcal{S}' = \mathcal{F}^{\alpha + 1}_{\mu}(\mathcal{S})$. Then $\mathcal{J}' = \mathcal{M}(\mathcal{S}')$. Applying the first set inclusion of Lemma 5.1 to $\mathcal{S}'$ and $\mathcal{J}'$ yields $\mathcal{F}^\alpha_{\mu}(\mathcal{S}') \subseteq \mathcal{M}(\mathcal{F}^{\alpha + 1}_{\mu}(\mathcal{S}'))$, i.e.,

$$\mathcal{F}^\alpha_{\mu}(\mathcal{M}(\mathcal{F}^{\alpha + 1}_{\mu}(\mathcal{S}))) \subseteq \mathcal{M}(\mathcal{F}^{\alpha + 1}_{\mu}(\mathcal{S}))) = \mathcal{M}(\mathcal{F}^{\alpha + 1}_{\mu}(\mathcal{S})).$$

By transitivity, we get

$$\mathcal{F}^\alpha_{\mu}(\mathcal{J}) \subseteq \mathcal{M}(\mathcal{F}^{\alpha + 1}_{\mu}(\mathcal{S})).$$

$\alpha$ a limit ordinal: Intersecting the set inclusion for all $\beta < \alpha$ yields

$$\bigcap_{\beta < \alpha} \mathcal{F}^\beta_{\mu}(\mathcal{J}) \subseteq \bigcap_{\beta < \alpha} \mathcal{M}(\mathcal{F}^\beta_{\mu}(\mathcal{S})).$$

Since $\bigcap_{\beta < \alpha} \mathcal{F}^\beta_{\mu}(\mathcal{J}) = \mathcal{F}^\alpha_{\mu}(\mathcal{J})$, and

$$\bigcap_{\beta < \alpha} \mathcal{M}(\mathcal{F}^\beta_{\mu}(\mathcal{S})) = \mathcal{M}\left(\bigcup_{\beta < \alpha} \mathcal{F}^\beta_{\mu}(\mathcal{S})\right) = \mathcal{M}(\mathcal{F}^\alpha_{\mu}(\mathcal{S})),$$

we get that

$$\mathcal{F}^\alpha_{\mu}(\mathcal{J}) \subseteq \mathcal{M}(\mathcal{F}^{\alpha + 1}_{\mu}(\mathcal{S})).$$

(ii) If we apply MS to the first set inclusion, we get

$$\text{MS}(\mathcal{M}(\mathcal{F}^{\alpha + 1}_{\mu}(\mathcal{S}))) \subseteq \text{MS}(\mathcal{F}^\alpha_{\mu}(\mathcal{J})).$$

By chaining with $\mathcal{F}^{\alpha + 1}_{\mu}(\mathcal{S}) \subseteq \text{MS}(\mathcal{M}(\mathcal{F}^{\alpha + 1}_{\mu}(\mathcal{S})))$, we get the second set inclusion. \qed

**Corollary 5.1** (State generation vs. model generation). Let $\mathcal{P}$ be a disjunctive logic program. For all ordinals $\alpha$, it holds that

$$\mathcal{F}^\alpha_{\mu}(\mathcal{J}) \subseteq \mathcal{M}(\mathcal{F}^{\alpha + 1}_{\mu}(\mathcal{J})), \quad \mathcal{F}^\alpha_{\mu}(\mathcal{J}) \subseteq \text{MS}(\mathcal{F}^\alpha_{\mu}(\mathcal{J})).$$

Sometimes, the set inclusions in Corollary 5.1 are strict, as the following example shows.

**Example 5.1** (State generation vs. model generation). Given some $n \in \mathbb{N}_+$, consider the disjunctive logic program

$$\mathcal{P} = \{ a \leftarrow b_i \mid 1 \leq i \leq n \} \cup \{ b_1 \vee \cdots \vee b_n \}$$
with \( n \) rules and one fact. Let \( S_m = \mathcal{F}_p^m \uparrow m \) and \( \mathcal{J}_m = \mathcal{F}_p^M \uparrow m \). For \( n = 2 \), we get
\[
S_1 = \{ b_1 \lor b_2 \}, \\
S_2 = \{ b_1 \lor b_2, a \lor b_1, a \lor b_2 \}, \\
\mathcal{J}_1 = \{ \{ b_1 \}, \{ b_2 \} \}, \\
\mathcal{J}_2 = \{ \{ b_1, a \}, \{ b_2, a \} \}.
\]
Thus, for the atom \( a \), we get \( a \in MS(\mathcal{J}_1) \), but \( a \notin S_2 \). For \( n \geq 2 \), we get the disjunctive Herbrand states
\[
S_m = \{ b_1 \lor \cdots \lor b_n \} \\
\cup \{ a \lor b_{i_1} \lor \cdots \lor b_{i_k} \mid \{ i_1, \ldots, i_k \} \subseteq \{1, n\}, \\
n - m + 1 \leq k \leq n - 1 \}, \quad \text{for all } m \in \mathbb{N}_+,
\]
where \( \langle n, m \rangle \) denotes the interval \( \{n, n+1, \ldots, m\} \) of all natural numbers between \( n \) and \( m \). On the other hand,
\[
\mathcal{J}_1 = \{ \{ b_1 \}, \ldots, \{ b_n \} \}, \\
\mathcal{J}_m = \{ \{ b_i, a \} \mid 1 \leq i \leq n \}, \quad \text{for all } m \geq 2.
\]
The least fixpoint of \( \mathcal{F}_p^M \) is always reached after two iterations: \( lfp(\mathcal{F}_p^M) = \mathcal{J}_2 \), whereas the least fixpoint of \( \mathcal{F}_p^p \) is reached after \( n + 1 \) iterations: \( lfp(\mathcal{F}_p^p) = S_{n+1} = S_n \cup \{ a \} \). We can show that
\[
\mathcal{M}(S_n) = \exp(\mathcal{J}_n \cup \{ I_b \}), \\
MS(\mathcal{J}_n) = \exp(S_n \cup \{ a \}),
\]
where \( I_b = \{ b_i \mid 1 \leq i \leq n \} \). This shows that \( a \in MS(\mathcal{J}_n) \setminus S_n, I_b \in \mathcal{M}(S_n) \setminus \mathcal{J}_n \), i.e., the set inclusions in Corollary 5.1 are strict.

Since the operator \( \mathcal{F}_p^p \) is continuous (see [16]), its ordinal powers converge towards its least fixpoint in at most \( \omega \) iterations, i.e., \( lfp(\mathcal{F}_p^p) = \mathcal{J}_n \uparrow \omega \equiv_p MS_p \). Corollary 5.1 helps to show that the ordinal powers of the operator \( \mathcal{F}_p^{INT} \) converge towards its least fixpoint in at most \( \omega \) iterations, too. That is, the closure ordinal of \( \mathcal{F}_p^{INT} \) on \( \mathcal{P}_{exp} \) is \( \omega \). Moreover, this least fixpoint is equivalent to the set \( \mathcal{M}(\mathcal{P}_{exp}) \) of the minimal Herbrand models of the program.

**Theorem 5.2** (Convergence of model generation by \( \mathcal{F}_p^{INT} \)). Let \( \mathcal{P} \) be a disjunctive logic program. Then
\[
\mathcal{M}(\mathcal{P}) \equiv \ lfp(\mathcal{F}_p^{INT}) = \mathcal{F}_p^{INT} \uparrow \omega.
\]
**PROOF.** We will show that \( \mathcal{M}(\mathcal{P}) \subseteq \mathcal{F}_p^{INT} \uparrow \omega \) and that \( \mathcal{F}_p^{INT} \uparrow \omega \subseteq \mathcal{M}(\mathcal{P}) \).

(i) From Corollary 5.1, we know that \( \mathcal{F}_p^{INT} \uparrow \omega \subseteq \mathcal{M}(\mathcal{P}) \uparrow \omega \). With \( \mathcal{F}_p \uparrow \omega \equiv_p MS_p \), we get
\[
\mathcal{M}(MS_p) \subseteq \mathcal{F}_p^{INT} \uparrow \omega.
\]
From Lemma 4.1, we know that \( \mathcal{M}(MS_p) \equiv \mathcal{M}(MS_p) \), and from [16], we know that \( \mathcal{M}(MS_p) = \mathcal{M}(\mathcal{P}) \). This shows that
\[
\mathcal{M}(\mathcal{P}) \subseteq \mathcal{F}_p^{INT} \uparrow \omega.
\]
(ii) From
\[ \mathcal{F}^{\text{INT}}_{\varphi} \uparrow 0 = \perp_{\exp} \subseteq \mathbb{M}_{\varphi} \]
and \( \mathcal{F}^{\text{INT}}_{\varphi}(\mathbb{M}_{\varphi}) = \mathbb{M}_{\varphi} \) and the monotonicity of the operator \( \mathcal{F}^{\text{INT}}_{\varphi} \) on the partial ordering \( \mathcal{C}_{\exp} = \langle 2^{\mathcal{C}_{\exp}}, \subseteq \rangle \), by induction on \( n \in \mathbb{N}_0 \) we get that
\[ \mathcal{F}^{\text{INT}}_{\varphi} \uparrow n \subseteq \mathbb{M}_{\varphi}, \quad \text{for all } n \in \mathbb{N}_0. \]

This shows that
\[ \mathcal{F}^{\text{INT}}_{\varphi} \uparrow \omega = \text{lub}_{\exp}(\{ \mathcal{F}^{\text{INT}}_{\varphi} \uparrow n \mid n \in \mathbb{N}_0 \}) \subseteq \mathbb{M}_{\varphi}. \]
Hence, \( \mathcal{F}^{\text{INT}}_{\varphi} \uparrow \omega = \subseteq \mathbb{M}_{\varphi} \). Since \( \mathbb{M}_{\varphi} \) is a fixpoint of \( \mathcal{F}^{\text{INT}}_{\varphi} \), i.e., \( \mathcal{F}^{\text{INT}}_{\varphi}(\mathbb{M}_{\varphi}) = \mathbb{M}_{\varphi} \), this implies that \( \text{lfp}(\mathcal{F}^{\text{INT}}_{\varphi}) = \subseteq \mathbb{M}_{\varphi} \). \( \Box \)

Thus, the operator \( \mathcal{F}^{\text{INT}}_{\varphi} \) is an example of a monotonic operator that is not continuous, but nevertheless reaches its least fixpoint in at most \( \omega \) iterations.

6. ITERATIVE MODEL GENERATION OF PERFECT AND (PARTIAL) STABLE SEMANTICS

In [6], Minker and Fernández defined an iterative method to compute the perfect models of a stratified disjunctive deductive database. In this section, we extend their method to work with arbitrary stratified disjunctive logic programs by using the operator \( \mathcal{F}^{\text{INT}}_{\varphi} \) defined in Section 4.

Given a first-order language, a normal disjunctive logic program \( \mathcal{P} \) consists of logical inference rules of the form
\[ A_1 \lor \cdots \lor A_k \leftarrow B_1 \land \cdots \land B_m \land \text{not} C_1 \land \cdots \land \text{not} C_n, \quad (6.1) \]
where \( A_i, i \in \langle 1, k \rangle, B_i, i \in \langle 1, m \rangle, \) and \( C_i, i \in \langle 1, n \rangle, \) are (positive) atoms in the language, \( k, m, n \in \mathbb{N}_0, \) and \( \text{not} \) is the negation-by-default operator. Given a predicate \( p \) in the language, the definition of \( p \) in \( \mathcal{P} \) is the set of all rules in \( \mathcal{P} \) whose heads (i.e., \( A_1 \lor \cdots \lor A_k \)) contain an atom in which \( p \) appears. The definition of an atom \( A \) is taken to be the definition of the predicate symbol appearing in \( A \).

Definition 6.1 (Stratification [18]). A normal disjunctive logic program \( \mathcal{P} \) is called stratified if it is possible to partition the set of rules of \( \mathcal{P} \) into sets \( \{ \mathcal{P}_1, \ldots, \mathcal{P}_r \} \), called strata, such that for every rule of the form (6.1) in \( \mathcal{P} \), there exists a constant \( c, 1 \leq c \leq r \), such that:

(i) the definition of each \( A_i \) is contained in \( \mathcal{P}_c \);
(ii) the definition of each \( B_i \) is contained in \( \bigcup_{s \leq c} \mathcal{P}_s \); and
(iii) the definition of each \( C_i \) is contained in \( \bigcup_{s < c} \mathcal{P}_s \).

Any partition \( \{ \mathcal{P}_1, \ldots, \mathcal{P}_r \} \) of \( \mathcal{P} \) satisfying the above conditions is called a stratification of \( \mathcal{P} \).
Example 6.1 (Stratification). Consider the following extension of the disjunctive logic program introduced in Example 3.1:

\[ \mathcal{P} = \{ \text{sink}(X) \leftarrow \text{node}(X) \land \neg \text{no}_\text{sink}(X), \quad \mathcal{P}_4 \}
\]

\[ \text{no}_\text{sink}(X) \leftarrow \text{disconnected}(Y, X), \]

\[ \text{disconnected}(X, Y) \leftarrow \text{node}(X) \land \text{node}(Y) \]

\[ \lor \neg \text{equal}(X, Y) \land \neg \text{path}(X, Y), \quad \mathcal{P}_3 \]

\[ \text{equal}(X, X) \land \neg \text{node}(X), \]

\[ \text{node}(a), \text{node}(b), \text{node}(c), \text{node}(d), \quad \mathcal{P}_2 \]

\[ \text{path}(X, Y) \leftarrow \text{arc}(X, Z) \land \text{path}(Z, Y), \]

\[ \text{path}(X, Y) \leftarrow \text{arc}(X, Y), \]

\[ \text{arc}(a, b) \lor \text{arc}(a, c), \text{arc}(b, d), \text{arc}(c, d). \quad \mathcal{P}_1 \]

The partition \( \{ \mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4 \} \) is a stratification of \( \mathcal{P} \).

The intended meaning of a stratified disjunctive logic program \( \mathcal{P} \) is given by its collection of **perfect models** as defined in [18]. It is well known (see, e.g., [18]) that this collection can be constructed by induction on the strata as follows. Given a stratification \( \{ \mathcal{P}_1, \ldots, \mathcal{P}_r \} \) of \( \mathcal{P} \), let \( \mathcal{P}_\mathcal{M}_i \) denote the set of perfect models of the first \( i \) strata of \( \mathcal{P} \), i.e., of the logic program \( \bigcup_{i \leq j} \mathcal{P}_j \). By definition, the lowest stratum \( \mathcal{P}_1 \) (which may be the empty set) is free of negation-by-default and its set of perfect models is given by \( \mathcal{P}_\mathcal{M}_1 = \{ \mathcal{M}_\mathcal{P}_1 \} \). When \( \mathcal{P}_\mathcal{M}_i \) has been constructed, it is used to evaluate the negated-by-default literals appearing in the \( i + 1 \) stratum as follows. If \( I \in \mathcal{P}_\mathcal{M}_i \), then \( \mathcal{P}_{i+1}^I \) is the ground disjunctive logic program:

\[ \mathcal{P}_{i+1}^I = \{ A_1 \lor \cdots \lor A_k \leftarrow B_1 \land \cdots \land B_m \mid 
\]

\[ A_1 \lor \cdots \lor A_k \leftarrow B_1 \land \cdots \land B_m \land \neg C_1 \land \cdots \land \neg C_n \in \text{gnd}(\mathcal{P}_{i+1}) \]

and \( \{ C_1, \ldots, C_n \} \cap I = \emptyset \).

\( \mathcal{P}_{i+1}^I \) is called the **Gelfond–Lifschitz transformation** (cf. [7, 20]) of \( \mathcal{P}_{i+1} \) with respect to \( I \). The perfect models of the first \( i + 1 \) strata are then taken to be \( \mathcal{P}_\mathcal{M}_{i+1} = \bigcup_{I \in \mathcal{P}_\mathcal{M}_i} \{ \mathcal{M}_{\mathcal{P}_{i+1}} \} \). The collection of perfect models of \( \mathcal{P} \) is exactly \( \mathcal{P}_\mathcal{M} \). It is important to note that the collection of perfect models of \( \mathcal{P} \) is the same independent of the particular stratification of \( \mathcal{P} \) used in the induction.

For the case of function-free logic programs, Minker and Fernández used an iterative version of their operator \( \mathcal{P}_\mathcal{M}^I \) to compute this collection of perfect models [6]. We extend now this procedure to arbitrary stratified disjunctive logic programs by defining the iterative version of our operator \( \mathcal{P}_{\text{INT}}^I \).

---

The predicate \text{node} has been added to the body of some rules for the sole purpose of making them range-restricted.
Definition 6.2 (Iterative version of $\mathcal{F}_\mathcal{P}^{\text{INT}}$). Let $\mathcal{P}$ be a stratified disjunctive logic program and $(\mathcal{P}_1, \ldots, \mathcal{P}_n)$ be a stratification of $\mathcal{P}$. The iterative version of $\mathcal{F}_\mathcal{P}^{\text{INT}}$ is defined as follows:

$$\mathcal{F}_\mathcal{P}^{\text{INT}} = \min(\mathcal{F}_\mathcal{P}^{\text{INT}} \uparrow \alpha_{\langle 1, \omega \rangle}),$$

$$\mathcal{F}_{\mathcal{P}_1, \ldots, \mathcal{P}_{n+1}}^{\text{INT}} = \bigcup_{I \in \mathcal{F}_{\mathcal{P}_1, \ldots, \mathcal{P}_n}^{\text{INT}}} \min(\mathcal{F}_{\mathcal{P}_{n+1} \cup I}^{\text{INT}} \uparrow \alpha_{\langle n+1, 1 \rangle}),$$

where $\alpha_{\langle n, I \rangle}$ is the closure of $\mathcal{F}_{\mathcal{P}_1, \ldots, \mathcal{P}_n}^{\text{INT}}$.

The set of perfect models of $\mathcal{P}$ is given by $\mathcal{F}_{\mathcal{P}_1}^{\text{INT}}$. Notice that, due to Theorem 5.2, each of the closure ordinals $\alpha_{\langle n, I \rangle}$ in the previous definition is at most $\omega$.

Example 6.2 (Perfect models). Let $\mathcal{P}$ be the stratified disjunctive logic program of Example 6.1. To simplify the notation below, let

$I' = \{ \text{arc}(b, d), \text{arc}(c, d), \text{path}(b, d), \text{path}(c, d), \text{path}(a, d) \},$

$I'' = \{ \text{node}(a), \text{node}(b), \text{node}(c), \text{node}(d), \text{equal}(a, a), \text{equal}(b, b), \text{equal}(c, c), \text{equal}(d, d) \},$

$I''' = \{ \text{disconnected}(b, a), \text{disconnected}(b, c), \text{disconnected}(c, a), \text{disconnected}(c, b), \text{disconnected}(d, a), \text{disconnected}(d, b), \text{disconnected}(d, c), \text{no_sink}(a), \text{no_sink}(b), \text{no_sink}(c) \}.$

The computation of the perfect models of $\mathcal{P}$ using the iterative fixpoint operator introduced in Definition 6.2 is as follows:

$$\mathcal{F}_{\mathcal{P}_1}^{\text{INT}} = \min(\mathcal{F}_{\mathcal{P}_1}^{\text{INT}} \uparrow \omega) = \{ I_1, J_1 \},$$

where $I_1 = I' \cup \{ \text{arc}(a, b), \text{path}(a, b) \}$,

$J_1 = I' \cup \{ \text{arc}(a, c), \text{path}(a, c) \}$ (see Example 3.1),

$$\mathcal{F}_{\mathcal{P}_1, \mathcal{P}_2}^{\text{INT}} = \bigcup_{I \in \mathcal{F}_{\mathcal{P}_1}^{\text{INT}}} \min(\mathcal{F}_{\mathcal{P}_2 \cup I}^{\text{INT}} \uparrow \omega) = \{ I_2, J_2 \},$$

where $I_2 = I_1 \cup I''$, $J_2 = J_1 \cup I''$,

$$\mathcal{F}_{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3}^{\text{INT}} = \bigcup_{I \in \mathcal{F}_{\mathcal{P}_1, \mathcal{P}_2}^{\text{INT}}} \min(\mathcal{F}_{\mathcal{P}_3 \cup I}^{\text{INT}} \uparrow \omega) = \{ I_3, J_3 \},$$

where $I_3 = I_2 \cup I''' \cup \{ \text{disconnected}(a, c) \}$,

$J_3 = J_2 \cup I''' \cup \{ \text{disconnected}(a, b) \}$,

$$\mathcal{F}_{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4}^{\text{INT}} = \bigcup_{I \in \mathcal{F}_{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3}^{\text{INT}}} \min(\mathcal{F}_{\mathcal{P}_4 \cup I}^{\text{INT}} \uparrow \omega) = \{ I_4, J_4 \},$$

where $I_4 = I_3 \{ \text{sink}(d) \}$,

$J_4 = J_3 \cup \{ \text{sink}(d) \}$.

Fernández et al. [4] showed how to transform an arbitrary normal disjunctive logic program $\mathcal{P}$ into a stratified one, denoted by $\mathcal{P}^\#, in such a way that the perfect models of $\mathcal{P}^\#$ that satisfied a given set of integrity constraints correspond to the stable models of $\mathcal{P}$ (as defined in [19]). They called this transformation the
evidential transformation. Using this characterization of stable models, the iterative version of $\mathcal{T}_p^{INT}$ can be used to construct the stable semantics of $\mathcal{P}$.

Similarly, the operator $\mathcal{T}_p^{INT}$ can be used to generate the collection of partial (or 3-valued) stable models ([20]) of a normal disjunctive logic program $\mathcal{P}$ by using a characterization of this collection of models given by Ruiz and Minker in [22]. This characterization is based on a transformation called the 3-S transformation, which, applied to a normal disjunctive logic program $\mathcal{P}$, produces a constraint logic program (free of negation-by-default) $\mathcal{P}^{3S}$, whose minimal (2-valued) consistent models correspond to the partial stable models of the original program $\mathcal{P}$. Since the well-founded model ([28]) is a distinct partial stable model of a normal logic program (see [20]), then model generation for the well-founded semantics is also achieved using the operator $\mathcal{T}_p^{INT}$ together with the 3-S transformation.

7. CONCLUSIONS

Given a disjunctive logic program $\mathcal{P}$, there are two approaches for deriving the minimal model state $MS_{\mathcal{P}}$ and two approaches for deriving the set $\mathcal{M}_{\mathcal{P}}$ of minimal models of the database. Both sets can be derived based on hyperresolution as well as on model generation from $\mathcal{P}$:

- Using hyperresolution, $MS_{\mathcal{P}}$ is computed as the least fixpoint of the disjunctive consequence operator $\mathcal{T}_\mathcal{P}$. Using model generation, $\mathcal{M}_{\mathcal{P}}$ is computed as the least fixpoint of the consequence operator $\mathcal{T}_\mathcal{P}^{INT}$.
- By dualization of the minimal model state $MS_{\mathcal{P}}$, we get the set of minimal models of $\mathcal{P}$ as $\mathcal{M}(MS_{\mathcal{P}})$. Similarly, dualization of $\mathcal{M}_{\mathcal{P}}$ yields $MS_{\mathcal{P}}$ as $MS(\mathcal{M}_{\mathcal{P}})$.

As shown in this paper, both approaches converge in at most $\omega$ iterations. For the case of disjunctive deductive databases, these approaches have been implemented within the disjunctive deductive database engine DisLog,\footnote{For a demo version of DisLog, visit http://www.informatik.uni-tuebingen.de/dislog.html.} cf. [25], developed at the University of Tübingen. Experimenting with DisLog, we have observed that the efficiency of each approach depends on the relation between the number $n = |\text{can}(MS_{\mathcal{P}})|$ of minimal\footnote{A disjunction $C$ in $MS_{\mathcal{P}}$ is minimal if there is no subdisjunction $C'$ of $C$ in $MS_{\mathcal{P}}$.} disjunctions in $MS_{\mathcal{P}}$ and the number $m = |\mathcal{M}_{\mathcal{P}}|$ of minimal models. We conjecture that the following may be the case:

- If $n$ and $m$ are about equal, then for each derivation it is better to use the specialized approach, i.e., to derive $MS_{\mathcal{P}}$ by hyperresolution and to derive $\mathcal{M}_{\mathcal{P}}$ by model generation.
- If $n$ is much bigger than $m$, then it is best to use model generation for deriving $\mathcal{M}_{\mathcal{P}}$ and to derive $MS_{\mathcal{P}}$ from $\mathcal{M}_{\mathcal{P}}$ by dualization.
- If $n$ is much smaller than $m$, then it is best to use hyperresolution for deriving $MS_{\mathcal{P}}$ and to derive $\mathcal{M}_{\mathcal{P}}$ from $MS_{\mathcal{P}}$ by dualization.

Based on the theorems of Section 5, it has been shown in [24] that during a fixpoint iteration with $\mathcal{T}_\mathcal{P}$, it is possible to switch to some iterations of model generation. Intermediate information about the size of the ordinal powers $\mathcal{T}_\mathcal{P}^{INT} \uparrow n$
can be used to decide if this would be advantageous. However, general criteria to decide in advance which method is more efficient when faced with a particular disjunctive logic program are still to be determined.

Using model generation by $\mathcal{F}_{\mathcal{P}}^{\text{INT}}$, perfect, well-founded, stable and partial stable models of normal disjunctive logic programs can also be computed. For a stratified disjunctive logic program $\mathcal{P}$, the perfect models of $\mathcal{P}$ can be computed by an iterative model generation applied to the strata. For a normal disjunctive logic program $\mathcal{P}$, the stable models and the partial stable models (the well-founded model in particular) of $\mathcal{P}$ can be respectively computed by model generation with the evidential transformation $\mathcal{P}_{\mathcal{E}}$ ([4]) and by model generation with the 3-S transformation $\mathcal{P}_{3S}$ ([22]).

The last two authors acknowledge the support of the National Science Foundation under grant number IRI 9300691.

REFERENCES


