Model Generation and State Generation for Disjunctive Logic Programs *

Dietmar Seipel[1], Jack Minker[2], Carolina Ruiz[2]

[1] Department of Computer Science.
University of Tübingen.
Sand 13, D – 72076 Tübingen, Germany
seipel@informatik.uni-tuebingen.de

[2] Institute for Advanced Computer Studies and
Department of Computer Science.
University of Maryland. College Park, MD 20742 U.S.A.
{minker, cruizc}@cs.umd.edu

Abstract

This paper investigates two fixpoint approaches for minimal model reasoning with disjunctive logic programs $\mathcal{P}$. The first one, called *model generation* [4], is based on an operator $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ 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 *state generation* [12], uses a fixpoint operator $\mathcal{T}_{\mathcal{P}}^{s}$ based on *hyperresolution*. It operates on disjunctive Herbrand *states* and its least fixpoint is the set of logical consequences of $\mathcal{P}$, the so–called *minimal model state* of the program.

We establish a useful relationship between hyperresolution by $\mathcal{T}_{\mathcal{P}}^{s}$ and model generation by $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$. Then we investigate the problem of *continuity* of the two operators $\mathcal{T}_{\mathcal{P}}^{s}$ and $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$. It is known that the operator $\mathcal{T}_{\mathcal{P}}^{s}$ is continuous [12], and so it reaches its least fixpoint in at most ω steps. On the other hand, the question of whether $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ is continuous has been open. We show by a counterexample that $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ is not continuous. Nevertheless, we prove that it converges towards its least fixpoint in at most ω steps too, as follows from the relationship that we show exists between hyperresolution and model generation.

We define an iterative version of $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ that computes the *perfect model semantics* of stratified disjunctive logic programs. On each stratum of the program, this operator converges in at most ω steps. Model generations for the *stable semantics* and the *partial stable* (and so the *well–founded*) *semantics* are respectively achieved by using this iterative operator together with the *evidential transformation* [3] and the *3-S transformation* [16].

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 [11]). This collection is called the *minimal model state* of the program.

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

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 $\mathcal{T}_{\mathcal{P}}^{M}$ 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 [4] (see also [2, 5]) for the case of disjunctive logic programs without function symbols.

The second approach, developed by Minker and Rajasekar [12], is based on a fixpoint operator $\mathcal{T}_{\mathcal{P}}^{s}$ defined on sets of positive disjunctions called *states*. This operator uses *hyperresolution* 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 $\mathcal{T}_{\mathcal{P}}^{s}$ and $\mathcal{T}_{\mathcal{P}}^{M}$. It is known that the operator $\mathcal{T}_{\mathcal{P}}^{s}$ is continuous [12], and so it reaches its least fixpoint in at most ω steps. On the other hand, the question of whether $\mathcal{T}_{\mathcal{P}}^{M}$ 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 $\mathcal{T}_{\mathcal{P}}^{M}$ is not closed under least upper bounds. We then give a natural extension of $\mathcal{T}_{\mathcal{P}}^{M}$ in terms of a new operator $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ defined on a more suitable domain and reformulate the continuity problem for the new operator. We prove, by means of a counterexample, that $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ is not continuous. Nevertheless, from a relationship that we show exists between state generation by $\mathcal{T}_{\mathcal{P}}^{s}$ and model generation by $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$, we prove that $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ reaches its least fixpoint in at most ω steps, too.

We define an iterative version of $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ that computes the perfect model semantics of stratified disjunctive logic programs. On each stratum of the program, this operator converges in at most ω steps. Due to the characterizations of the stable and the partial stable (and so the well–founded) semantics respectively presented in [3] and [16], 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 notation. Section 3 summarizes the main properties of state generation. Section 4 surveys model generation, defines the new operator $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ and shows that it is not continuos. 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 $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ reaches its least fixpoint in at most ω steps. Section 6 provides an iterated version of the operator $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ that constructs the perfect models of stratified disjunctive logic programs, and is used to generate the (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

$$A_1 \vee \ldots \vee A_k \leftarrow B_1 \wedge \ldots \wedge B_m, \tag{1}$$

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 $k, m \in I\!N_0$.[1] A rule is called a fact if $m = 0$. The set of all *ground instances* of the rules and facts in $\mathcal{P}$ is

[1] By $I\!N_+$ we denote the set $\{1, 2, 3, \ldots\}$ of positive natural numbers, whereas $I\!N_0$ denotes the set $\{0, 1, 2, \ldots\}$ of all natural numbers.

denoted by $gnd(\mathcal{P})$. 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 $I \subseteq \mathit{HB}_{\mathcal{P}}$ of the Herbrand base $\mathit{HB}_{\mathcal{P}}$ is called a *Herbrand interpretation*. The set of all Herbrand interpretations is denoted by $\mathcal{HI}_{\mathcal{P}}$. A Herbrand interpretation I is called a *model* of $\mathcal{P}$ if for all ground rules of the form (1) in $\mathcal{P}$, it holds that $\{ B_i \mid i \in \langle 1, m \rangle \} \subseteq I$ 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 $\mathcal{MOD}_{\mathcal{P}}$, and sometimes by $\mathcal{MOD}(\mathcal{P})$.

(ii) Secondly, the set $\mathit{DHB}_{\mathcal{P}}$ of all ground disjunctions $A_1 \vee \ldots \vee A_k$, $k \in \mathbb{N}_0$, that can be formed by atoms $A_i \in \mathit{HB}_{\mathcal{P}}$ is called the *disjunctive Herbrand base* of $\mathcal{P}$. A subset $S \subseteq \mathit{DHB}_{\mathcal{P}}$ 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 $\mathcal{MM}_{\mathcal{P}}$ of all *minimal Herbrand models* of $\mathcal{P}$. A model I is called minimal if there is no smaller model I' of $\mathcal{P}$ for which $I' \subsetneq I$. The second approach uses hyperresolution for deriving the set $\mathit{MS}_{\mathcal{P}}$ of all ground disjunctions $C \in \mathit{DHB}_{\mathcal{P}}$ 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 [12].

Below, we put together some important notation and results on *partial orderings and lattices* and on *fixpoint theory* on complete lattices, cf. e.g. [9, 1].

Let $\mathcal{O} = \langle S, \leq \rangle$, where S is a set and $\leq$ is a binary relation on S. $\mathcal{O}$ is called a *partial ordering* on S, if $\leq$ is reflexive, transitive and antisymmetric. A partial ordering $\mathcal{O}$ is called a *complete lattice*, if for all subsets $X \subseteq S$ there exists a least upper bound, denoted by $lub(X)$, and a greatest lower bound, denoted by $glb(X)$, in S.[2] A very common example of a complete lattice that will occur in this paper is $\mathcal{O} = \langle 2^U, \subseteq \rangle$ 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.

Certain mappings $\mathcal{T} : L \to L$ on a complete lattice $\mathcal{O} = \langle L, \leq \rangle$ 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. $\mathcal{T}$ is called *monotonic*, if $x_1 \leq x_2$ implies $\mathcal{T}(x_1) \leq \mathcal{T}(x_2)$, 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. $\mathcal{T}$ is called *continuous*, if $\mathcal{T}(lub(X)) = lub(\mathcal{T}(X))$ for all directed subsets $X \subseteq L$, where $\mathcal{T}(X) = \{ \mathcal{T}(x) \mid x \in X \}$. Every continuous mapping on a complete lattice is also monotonic. The converse, however, is 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 *fixpoint* of $\mathcal{T}$, if $\mathcal{T}(x) = x$. A fixpoint $x \in L$ of $\mathcal{T}$ is called the *least fixpoint* of $\mathcal{T}$, denoted by $lfp(\mathcal{T})$, if $x \leq x'$ for all fixpoints x' of $\mathcal{T}$.[3]

Given a mapping $\mathcal{T} : L \to L$ on a complete lattice $\mathcal{O} = \langle L, \leq \rangle$ 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 α in conjunction with the least upper bound operation on the lattice. The *ordinal powers* $\mathcal{T}^{\uparrow\alpha} : L \to L$ of $\mathcal{T}$ are defined as follows:

$$\mathcal{T}^{\uparrow 0}(x) \;=\; x,$$

[2] The notations $lub(X)$ and $glb(X)$ are justified since for every set X its least upper bound and its greatest lower bound are unique if they exist.

[3] The notation $lfp(\mathcal{T})$ is justified since for every mapping $\mathcal{T}$ its least fixpoint is unique if it exists.

$$\mathcal{T}^{\uparrow\alpha}(x) = \mathcal{T}(\mathcal{T}^{\uparrow\alpha-1}(x)), \text{ for a successor ordinal } \alpha,$$
$$\mathcal{T}^{\uparrow\alpha}(x) = lub(\{\, \mathcal{T}^{\uparrow\beta}(x) \mid \beta < \alpha \,\}), \text{ for a limit ordinal } \alpha.$$

For the important special case of the bottom element $\bot = glb(L)$ of $\mathcal{O}$, the *ordinal powers* $\mathcal{T} \uparrow \alpha$ of $\mathcal{T}$ are lattice elements given by $\mathcal{T} \uparrow \alpha = \mathcal{T}^{\uparrow\alpha}(\bot)$. The well-known theorem of *Knaster and Tarski*, cf. [8] and [20], relates the fixpoints of a monotonic mapping on a complete lattice to the ordinal powers: If $\mathcal{T}$ is monotonic, then the collection of fixpoints of $\mathcal{T}$ forms a complete lattice and so $\mathcal{T}$ has a unique least fixpoint $lfp(\mathcal{T})$. For any ordinal α, it holds $\mathcal{T} \uparrow \alpha \leq lfp(\mathcal{T})$, and there exists an ordinal α, such that $\mathcal{T} \uparrow \alpha' = lfp(\mathcal{T})$ for all $\alpha' \geq \alpha$. The smallest such α is called the *closure ordinal* of $\mathcal{T}$. If $\mathcal{T}$ is continuous, then $\mathcal{T} \uparrow \omega = lfp(\mathcal{T})$ (see [7]).

3 State Generation

The fixpoint semantics of a disjunctive logic program is based on a disjunctive consequence operator $\mathcal{T}^s_\mathcal{P}$ given in [12] (see also [10]).

Definition 3.1 (Consequence Operator $\mathcal{T}^s_\mathcal{P}$)
Let $\mathcal{P}$ be a disjunctive logic program and let $S \subseteq DHB_\mathcal{P}$ be a disjunctive Herbrand state.
The disjunctive *consequence operator*

$$\mathcal{T}^s_\mathcal{P} : 2^{DHB_\mathcal{P}} \rightarrow 2^{DHB_\mathcal{P}}$$

of $\mathcal{P}$ is defined as

$$\begin{aligned} \mathcal{T}^s_\mathcal{P}(S) = S \cup \{\, & C \vee C_1 \vee \ldots \vee C_m \mid C, C_1, \ldots, C_m \in DHB_\mathcal{P} \text{ and} \\ & \text{there is a rule } C \leftarrow B_1 \wedge \ldots \wedge B_m \in gnd(\mathcal{P}) : \\ & \forall\, i \in \langle 1, m \rangle : B_i \vee C_i \in S \,\}. \end{aligned}$$

Note that according to the above definition the result of applying the operator $\mathcal{T}^s_\mathcal{P}$ 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.

Minker and Rajasekar show in [12] that the disjunctive consequence operator $\mathcal{T}^s_\mathcal{P}$ is *continuous* with respect to the complete lattice $\mathcal{O} = \langle 2^{DHB_\mathcal{P}}, \subseteq \rangle$ on disjunctive Herbrand states, and hence also *monotonic*. The bottom element of $\mathcal{O}$ is $\bot = \emptyset$. Following the general setting given in Section 2, the *ordinal powers* of the disjunctive consequence operator are defined with respect to the lattice $\mathcal{O}$ as follows.

Definition 3.2 (Ordinal Powers for $\mathcal{T}^s_\mathcal{P}$ on $\mathcal{O} = \langle 2^{DHB_\mathcal{P}}, \subseteq \rangle$)
Let $\mathcal{P}$ be a disjunctive logic program.

(i) For $S \subseteq DHB_\mathcal{P}$, the *ordinal powers* $\mathcal{T}^{s\uparrow\alpha}_\mathcal{P}(S)$ are defined by

$$\begin{aligned} \mathcal{T}^{s\uparrow 0}_\mathcal{P}(S) &= S, \\ \mathcal{T}^{s\uparrow\alpha}_\mathcal{P}(S) &= \mathcal{T}^s_\mathcal{P}(\mathcal{T}^{s\uparrow\alpha-1}_\mathcal{P}(S)), \text{ for a successor ordinal } \alpha, \\ \mathcal{T}^{s\uparrow\alpha}_\mathcal{P}(S) &= \bigcup_{\beta<\alpha} \mathcal{T}^{s\uparrow\beta}_\mathcal{P}(S), \text{ for a limit ordinal } \alpha. \end{aligned}$$

(ii) The *ordinal powers* $\mathcal{T}^s_\mathcal{P} \uparrow \alpha$ are defined by $\mathcal{T}^s_\mathcal{P} \uparrow \alpha = \mathcal{T}^{s\uparrow\alpha}_\mathcal{P}(\emptyset)$.

Example 3.3 (Disjunctive Transitive Closure)
Consider the disjunctive logic program

$$\begin{aligned}\mathcal{P} \;=\; \{\; & path(X,Y) \leftarrow arc(X,Z) \wedge path(Z,Y),\\ & path(X,Y) \leftarrow arc(X,Y),\\ & arc(a,b) \vee arc(a,c),\;\; arc(b,d),\;\; arc(c,d)\;\},\end{aligned}$$

that consists of the classical transitive closure rules and some disjunctive facts for the *arc*–relation, cf. Figure 1.

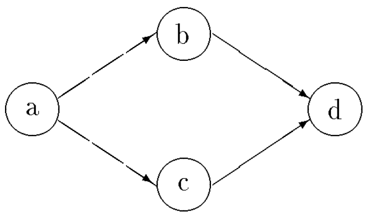

Figure 1: Graph with disjunctive arcs

The ordinal powers $S_n = \mathcal{T}^s_{\mathcal{P}} \uparrow n = \mathcal{T}^{s\uparrow n}_{\mathcal{P}}(\emptyset)$, $n \geq 1$, are given by

$$\begin{aligned}S_1 &= \{\; arc(a,b) \vee arc(a,c),\;\; arc(b,d),\;\; arc(c,d)\;\},\\ S_2 &= S_1 \cup \{\; arc(a,b) \vee path(a,c),\;\; path(a,b) \vee arc(a,c),\\ &\qquad path(b,d),\;\; path(c,d)\;\},\\ S_3 &= S_2 \cup \{\; path(a,b) \vee path(a,c),\;\; path(a,d) \vee arc(a,c),\\ &\qquad path(a,d) \vee arc(a,b),\\ &\qquad path(a,d) \vee path(a,c),\;\; path(a,d) \vee path(a,b)\;\},\\ S_4 &= S_3 \cup \{\; path(a,d)\;\},\\ S_n &= S_4, \quad \text{for all} \quad n \geq 4.\end{aligned}$$

Thus, $\mathcal{T}^s_{\mathcal{P}} \uparrow \omega = \mathcal{T}^s_{\mathcal{P}} \uparrow 4$. This example shows that $\mathcal{T}^s_{\mathcal{P}}$ can also derive definite facts from disjunctive facts.

A disjunction C' is called a *sub-disjunction* of another disjunction C if every atom appearing in C' also appears in C. C' is called a *proper* sub-disjunction of C if $C' \neq C$ and C' is a sub-disjunction of C. For a disjunctive Herbrand state S, let

$$\begin{aligned}can(S) &= \{\, C \in S \mid \nexists C' \in S : C' \text{ is a proper sub-disjunction of } C\,\},\\ exp(S) &= \{\, C \in D_{HB_{\mathcal{P}}} \mid \exists C' \in S : C' \text{ is a sub-disjunction of } C\,\}.\end{aligned}$$

$can(S)$ 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 $S_1 \equiv_{\triangleright} S_2$.

The minimal model state $MS_{\mathcal{P}}$ is equivalent to the *least fixpoint* of the disjunctive consequence operator $\mathcal{T}^s_{\mathcal{P}}$, and it can be derived as $MS_{\mathcal{P}} = exp(\mathcal{T}^s_{\mathcal{P}} \uparrow \omega)$, as was proven by Minker and Rajasekar in [12].

Theorem 3.4 (Characterization of $\mathit{MS}_{\mathcal{P}}$ [12])
Let $\mathcal{P}$ be a disjunctive logic program. Then

$$\mathit{MS}_{\mathcal{P}} \equiv_{\triangleright} \mathit{lfp}(\mathcal{T}^s_{\mathcal{P}}) = \mathcal{T}^s_{\mathcal{P}} \uparrow \alpha,$$

where α is the closure ordinal of $\mathcal{T}^s_{\mathcal{P}}$ on $\mathcal{O} = \langle 2^{\mathit{DHB}_{\mathcal{P}}}, \subseteq \rangle$.

4 Model Generation

The model generation approach constructs the minimal Herbrand models of a given logic program. For a definite logic program $\mathcal{P}$ (without disjunctions) and a given Herbrand interpretation I the classical consequence operator $\mathcal{T}_{\mathcal{P}}$ of van Emden and Kowalski [21] computes the Herbrand interpretation J that consists of the head atoms of all rules in $\mathit{gnd}\,(\mathcal{P})$, 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* (*c*ollection *o*f *in*terpretations). We use the following two operations *min* and *exp* for a coin $\mathcal{I}$:

$$\begin{aligned} \mathit{min}(\mathcal{I}) &= \{\, I \in \mathcal{I} \mid \not\exists J \in \mathcal{I} : J \subsetneq I \,\}, \\ \mathit{exp}\,(\mathcal{I}) &= \{\, I \in 2^{\mathcal{HI}_{\mathcal{P}}} \mid \exists J \in \mathcal{I} : J \subseteq I \,\}. \end{aligned}$$

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 refering. A coin $\mathcal{I}$ is called *canonical* if it does not contain two different Herbrand interpretations I, J such that $I \subsetneq J$, i.e. if $\mathcal{I} = \mathit{min}(\mathcal{I})$. A coin $\mathcal{I}$ is called *expanded* if for each Herbrand interpretation $I \in \mathcal{I}$ it also contains all Herbrand interpretations that are supersets of I, i.e. if $\mathcal{I} = \mathit{exp}\,(\mathcal{I})$. E.g. for the Herbrand base $\mathit{HB}_{\mathcal{P}} = \{\, a, b, c \,\}$ and the coin $\mathcal{I} = \{\, \{a\}, \{a, b\}, \{b, c\} \,\}$, we get $\mathit{min}(\mathcal{I}) = \{\, \{a\}, \{b, c\} \,\}$ and $\mathit{exp}\,(\mathcal{I}) = \mathcal{I} \cup \{\, \{a, c\}, \{a, b, c\} \,\}$.

The following consequence operator $\mathcal{T}^{\mathrm{INT}}_{\mathcal{P}}$ generalizes the operator $\mathcal{T}_{\mathcal{P}}$ of van Emden and Kowalski to the case of disjunctive logic programs $\mathcal{P}$. $\mathcal{T}^{\mathrm{INT}}_{\mathcal{P}}$ maps coins to coins.

Definition 4.1 (Consequence Operators $\mathcal{T}^{\mathrm{INT}}_{\mathcal{P}}$ and $\mathcal{T}^{M}_{\mathcal{P}}$)
Let $\mathcal{P}$ be a disjunctive logic program.

(i) The *consequence operator*

$$\mathcal{T}^{\mathrm{INT}}_{\mathcal{P}} : 2^{\mathcal{HI}_{\mathcal{P}}} \to 2^{\mathcal{HI}_{\mathcal{P}}}$$

operates on sets $\mathcal{I} \in 2^{\mathcal{HI}_{\mathcal{P}}}$ of Herbrand interpretations:

$$\mathcal{T}^{\mathrm{INT}}_{\mathcal{P}}(\mathcal{I}) = \bigcup_{I \in \mathcal{I}} \mathcal{MOD}(\mathcal{T}^s_{\mathcal{P}}(I)).$$

(ii) The *consequence operator*

$$\mathcal{T}^{M}_{\mathcal{P}} : 2^{\mathcal{HI}_{\mathcal{P}}} \to 2^{\mathcal{HI}_{\mathcal{P}}}$$

operates on sets $\mathcal{I} \in 2^{\mathcal{HI}_{\mathcal{P}}}$ of Herbrand interpretations:

$$\mathcal{T}^{M}_{\mathcal{P}}(\mathcal{I}) = \mathit{min}(\mathcal{T}^{\mathrm{INT}}_{\mathcal{P}}(\mathcal{I})).$$

For each $I \in \mathcal{I}$, $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}(\mathcal{I})$ 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 $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}(\mathcal{I})$ is expanded.

The operator $\mathcal{T}_{\mathcal{P}}^{M}$ was originally introduced by Fernández and Minker [4] (see also [2, 5]) 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 $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}(\mathcal{I}) = exp\,(\mathcal{T}_{\mathcal{P}}^{M}(\mathcal{I}))$. [4]

Example 4.2 (Consequence Operator $\mathcal{T}_{\mathcal{P}}^{M}$)
Consider the disjunctive deductive database $\mathcal{P}$ of Example 3.3.

(i) For the Herbrand interpretation $I = \emptyset$ the rules whose bodies are satisfied by I are precisely the facts of $\mathcal{P}$. Thus, $\mathcal{T}_{\mathcal{P}}^{M}(\{\emptyset\}) = \mathcal{I}_1 = \{I_1, J_1\}$, is the set of minimal Herbrand interpretations of the facts of $\mathcal{P}$, where

$$\begin{aligned} I_1 &= \{\, arc(a,b), arc(b,d), arc(c,d)\,\}, \\ J_1 &= \{\, arc(a,c), arc(b,d), arc(c,d)\,\}. \end{aligned}$$

(ii) For the Herbrand interpretations in $\mathcal{I}_1$ the bodies of some ground instances of the second rule are satisfied. Thus, $\mathcal{T}_{\mathcal{P}}^{M}$ extends I_1, J_1. We get $\mathcal{T}_{\mathcal{P}}^{M}(\mathcal{I}_1) = \mathcal{I}_2 = \{I_2, J_2\}$, where

$$\begin{aligned} I_2 &= I_1 \cup \{\, path(a,b), path(b,d), path(c,d)\,\}, \\ J_2 &= J_1 \cup \{\, path(a,c), path(b,d), path(c,d)\,\}. \end{aligned}$$

Fernández and Minker (cf. [5]) investigated some of the properties of the consequence operator $\mathcal{T}_{\mathcal{P}}^{M}$ based on the following subsumption relation $\sqsubseteq$ defined for coins $\mathcal{I}, \mathcal{J} \in 2^{\mathcal{HI}_{\mathcal{P}}}$:

$$\mathcal{I} \sqsubseteq \mathcal{J} \;\text{ iff }\; \forall\, J \in \mathcal{J} : \exists\, I \in \mathcal{I} : I \subseteq J.$$

E.g. for $\mathcal{I} = \{\,\{a\}, \{a,b\}, \{b,c\}\,\}$ and $\mathcal{J} = \{\,\{a\}, \{c\}\,\}$ we get $\mathcal{J} \sqsubseteq \mathcal{I}$.

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

$$\mathcal{I} \equiv_{\sqsubseteq} \mathcal{J} \;\text{ iff }\; \mathcal{I} \sqsubseteq \mathcal{J} \text{ and } \mathcal{J} \sqsubseteq \mathcal{I}.$$

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

[4] 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 exist some $I \in \mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}(\mathcal{I})$ for which there is no minimal interpretation in $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}(\mathcal{I})$ contained in I.

Theorem 4.3 (Characterization of $\mathcal{MM}_\mathcal{P}$ [5])
Let $\mathcal{P}$ be a disjunctive deductive database. Then

$$\mathcal{MM}_\mathcal{P} = lfp(\mathcal{T}_\mathcal{P}^M) = \mathcal{T}_\mathcal{P}^M \uparrow \alpha,$$

where α is the closure ordinal of $\mathcal{T}_\mathcal{P}^M$ on $\mathcal{O}_{min} = \langle\, 2^{\mathcal{HI}_\mathcal{P}}_{min}, \sqsubseteq \rangle$.

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

Example 4.4 (Disjunctive Transitive Closure)
For the disjunctive logic program $\mathcal{P}$ of Example 3.3 all ordinal powers $\mathcal{T}_\mathcal{P}^M \uparrow n = \{\, I_n, J_n \,\}$, $n \geq 1$, consist of two Herbrand interpretations, where I_1, J_1 and I_2, J_2 have been given in Example 4.2 and

$$\begin{array}{rcl} I_3 & = & I_2 \cup \{\, path(a,d) \,\}, \\ J_3 & = & J_2 \cup \{\, path(a,d) \,\}. \end{array}$$

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

In principle, one can apply the operator $\mathcal{T}_\mathcal{P}^M$ 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 $\mathcal{O}_{min} = \langle\, 2^{\mathcal{HI}_\mathcal{P}}_{min}, \sqsubseteq \rangle$ 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 $\mathcal{O}_{exp} = \langle\, 2^{\mathcal{HI}_\mathcal{P}}_{exp}, \sqsubseteq \rangle$, where $2^{\mathcal{HI}_\mathcal{P}}_{exp}$ consists of all *expanded* coins. It is easy to show that in this subdomain of expanded coins the relation $\sqsubseteq$ reduces to superset inclusion, since for all $\mathcal{I}, \mathcal{J} \in 2^{\mathcal{HI}_\mathcal{P}}$ it holds

$$\mathcal{I} \sqsubseteq \mathcal{J} \ \text{ iff } \ exp(\mathcal{I}) \supseteq exp(\mathcal{J}).$$

This shows that $\mathcal{O}_{exp} = \langle\, 2^{\mathcal{HI}_\mathcal{P}}_{exp}, \sqsubseteq \rangle$ is a *complete* lattice, and that the least upper bound and the greatest lower bound of a set $X \subseteq 2^{\mathcal{HI}_\mathcal{P}}_{exp}$ of expanded coins are given by

$$lub_{\,exp}(X) = \bigcap_{\mathcal{I} \in X} \mathcal{I}, \quad glb_{\,exp}(X) = \bigcup_{\mathcal{I} \in X} \mathcal{I}.$$

The bottom element of $\mathcal{O}_{exp}$ is the coin $\bot_{exp} = \mathcal{HI}_\mathcal{P}$ consisting of all Herbrand interpretations. Note that both the intersection and the union of a set X of expanded coins is an expanded coin.[5] We remark that the construction of $\mathcal{O}_{exp}$ is similar to the concept of *free ω-completion* of partial orderings, cf. [23].

In the domain $\mathcal{O}_{exp}$ we can work with the operator $\mathcal{T}_\mathcal{P}^{\mathrm{INT}}$ that produces expanded coins. For every coin $\mathcal{I}$ it holds $\mathcal{I} \sqsubseteq \mathcal{T}_\mathcal{P}^{\mathrm{INT}}(\mathcal{I})$, since every Herbrand interpretation $J \in \mathcal{T}_\mathcal{P}^{\mathrm{INT}}(\mathcal{I})$ is derived by extending an Herbrand interpretation $I \in \mathcal{I}$, where $I \subseteq J$. Moreover, the operator $\mathcal{T}_\mathcal{P}^{\mathrm{INT}}$ is *monotonic* on $\mathcal{O} = \langle\, 2^{\mathcal{HI}_\mathcal{P}}, \sqsubseteq \rangle$ (and hence in $\mathcal{O}_{exp}$):

$$\mathcal{I} \sqsubseteq \mathcal{J} \ \text{ implies } \ \mathcal{T}_\mathcal{P}^{\mathrm{INT}}(\mathcal{I}) \sqsubseteq \mathcal{T}_\mathcal{P}^{\mathrm{INT}}(\mathcal{J}).$$

Following the general setting given in Section 2, the *ordinal powers* of $\mathcal{T}_\mathcal{P}^{\mathrm{INT}}$ are defined with respect to the complete lattice $\mathcal{O}_{exp}$ as follows.

[5] This does not hold for canonical coins.

Definition 4.5 (Ordinal Powers for $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ on $\mathcal{O}_{exp} = \langle 2^{\mathcal{H}\mathcal{I}_{\mathcal{P}}}_{exp}, \sqsubseteq \rangle$)
Let $\mathcal{P}$ be a disjunctive logic program.

(i) For $\mathcal{I} \in 2^{\mathcal{H}\mathcal{I}_{\mathcal{P}}}_{exp}$, the *ordinal powers* $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}\uparrow\alpha}(\mathcal{I})$ are defined by

$$\begin{aligned}
\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}\uparrow 0}(\mathcal{I}) &= \mathcal{I}, \\
\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}\uparrow\alpha}(\mathcal{I}) &= \mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}(\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}\uparrow\alpha-1}(\mathcal{I})), \text{ for a successor ordinal } \alpha, \\
\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}\uparrow\alpha}(\mathcal{I}) &= \bigcap_{\beta<\alpha} \mathcal{T}_{\mathcal{P}}^{\mathrm{INT}\uparrow\beta}(\mathcal{I}), \text{ for a limit ordinal } \alpha.
\end{aligned}$$

(ii) The *ordinal powers* $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}} \uparrow \alpha$ are defined by $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}} \uparrow \alpha = \mathcal{T}_{\mathcal{P}}^{\mathrm{INT}\uparrow\alpha}(\bot_{exp})$.

For disjunctive deductive databases it can be shown that the respective ordinal powers of the operators $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ and $\mathcal{T}_{\mathcal{P}}^{M}$ are equivalent, i.e. $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}} \uparrow \alpha \equiv_{\sqsubseteq} \mathcal{T}_{\mathcal{P}}^{M} \uparrow \alpha$. It holds $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}} \uparrow \alpha = exp(\mathcal{T}_{\mathcal{P}}^{M} \uparrow \alpha)$ and $\mathcal{T}_{\mathcal{P}}^{M} \uparrow \alpha = min(\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}} \uparrow \alpha)$.

We show now by means of a counterexample that the operator $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ is *not continuous*.

Example 4.6 (Non–continuity of Model Generation through $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$)
Consider the disjunctive logic program

$$\begin{aligned}
\mathcal{P} = \{\, & a_0 \leftarrow a(X) \wedge a(Y) \wedge diff(X,Y), \\
& a(X) \leftarrow a(Y), \\
& d(f(c)) \leftarrow d(c) \,\}
\end{aligned}$$

and the complete lattice $\mathcal{O}_{exp} = \langle 2^{\mathcal{H}\mathcal{I}_{\mathcal{P}}}_{exp}, \supseteq \rangle$. Consider the following Herbrand interpretation for the predicate symbol *diff*:

$$I_{diff} = \{\, diff(f^m(c), f^n(c)) \mid m, n \in \mathbb{N}_0, m \neq n \,\}.$$

This Herbrand interpretation I_{diff} says that for $diff(X,Y)$ to be true, it must hold that X and Y denote different terms. Let a_n denote the atom $a(f^n(c))$, for all $n \in \mathbb{N}_+$, and let $I_a = \{a_n | n \in \mathbb{N}_+\}$. We are interested in the following expanded coins $\mathcal{I}_n$, where all $I \in \mathcal{I}_n$ contain the Herbrand interpretation I_{diff}:

$$\mathcal{I}_n = exp(\{\, \{a_m\} \cup I_{diff} \mid m \geq n \,\}), \text{ for } n \in \mathbb{N}_+.$$

The set $X = \{\, \mathcal{I}_n \mid n \in \mathbb{N}_+ \,\}$ of coins forms a *chain*, since it holds $\mathcal{I}_n \subseteq \mathcal{I}_k$, for all $n \geq k$. Thus, X is a *directed* set of coins. The least upper bound of X is given by

$$lub_{exp}(X) = \bigcap_{n \in \mathbb{N}_+} \mathcal{I}_n = \{\, I \subseteq HB_{\mathcal{P}} \mid I \cap I_a \text{ is infinite and } I_{diff} \subseteq I \,\},$$

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

$$\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}(lub_{exp}(X)) = exp(\{\, \{a_0\} \cup I_a \cup I_{diff} \,\}).$$

On the other hand it holds

$$\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}(\mathcal{I}_n) = exp\,(\{\, I_a \cup I_{diff} \,\}),\ \text{for all}\ n \in I\!N_+,$$

i.e. all coins $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}(\mathcal{I}_n)$, $n \in I\!N_+$, are identical. Thus,

$$lub_{\,exp}(\{\, \mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}(\mathcal{I}_n) \mid n \in I\!N_+ \,\}) = \bigcap_{n \in I\!N_+} \mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}(\mathcal{I}_n) = exp\,(\{\, I_a \cup I_{diff} \,\}).$$

This shows that the operator $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ does not commute with $lub_{\,exp}$, i.e.

$$\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}(lub_{\,exp}(X)) \neq lub_{\,exp}(\{\, \mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}(\mathcal{I}_n) \mid n \in I\!N_+ \,\}).$$

This implies also that $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ is not continuous with respect to $\mathcal{O}_{exp} = \langle\, 2^{\mathcal{H}\mathcal{I}_{\mathcal{P}}}_{exp}, \sqsubseteq \,\rangle$ nor with respect to $\mathcal{O} = \langle\, 2^{\mathcal{H}\mathcal{I}_{\mathcal{P}}}, \sqsubseteq \,\rangle$.

$\square$

In the next section we show that even though $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ is not continuous it reaches its least fixpoint in at most ω steps. Furthermore, based on the operator $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$, 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 $\mathcal{M}\mathcal{M}_{\mathcal{P}} \equiv_{\sqsubseteq} lfp(\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}) = \mathcal{T}_{\mathcal{P}}^{\mathrm{INT}} \uparrow \omega$ (see Theorem 5.5).

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 $M_1 \supsetneq M_2 \supsetneq M_3 \supsetneq \ldots$ of models of $\mathcal{P}$.

Lemma 4.7 (Existence of Minimal Models)
Let $\mathcal{P}$ be a disjunctive logic program. Then

$$\mathcal{M}\mathcal{M}_{\mathcal{P}} \equiv_{\sqsubseteq} \mathcal{M}\mathcal{O}\mathcal{D}_{\mathcal{P}}.$$

Proof:

For each $M \in \mathcal{M}\mathcal{O}\mathcal{D}_{\mathcal{P}}$ we want to show that there exists some $I^* \in \mathcal{M}\mathcal{M}_{\mathcal{P}}$ such that $I^* \subseteq M$. Let $\mathcal{I}_M = \{\, I \in \mathcal{M}\mathcal{O}\mathcal{D}_{\mathcal{P}} \mid I \subseteq M \,\}$. We prove below that each chain $\mathcal{K} \subseteq \mathcal{I}_M$ [6] has a lower bound I^* in $\mathcal{I}_M$. Using Zorn's lemma, this implies that the coin $\mathcal{I}_M$ contains a minimal element. Hence, $\mathcal{M}\mathcal{M}_{\mathcal{P}} \sqsubseteq \mathcal{M}\mathcal{O}\mathcal{D}_{\mathcal{P}}$. The statement $\mathcal{M}\mathcal{O}\mathcal{D}_{\mathcal{P}} \sqsubseteq \mathcal{M}\mathcal{M}_{\mathcal{P}}$ is trivial, since $\mathcal{M}\mathcal{M}_{\mathcal{P}} \subseteq \mathcal{M}\mathcal{O}\mathcal{D}_{\mathcal{P}}$.

Now let $\mathcal{K} \subseteq \mathcal{I}_M$ be a chain of Herbrand models. Obviously, the Herbrand interpretation

$$I^* = \bigcap_{I \in \mathcal{K}} I$$

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

$$A_1 \vee \ldots \vee A_k \leftarrow B_1 \wedge \ldots \wedge B_m \in gnd\,(\mathcal{P})$$

such that $I^* \models B_1 \wedge \ldots \wedge B_m$. Since $\mathcal{K}$ is a chain, we get that

$$\mathcal{J} = \{\, I \cap \{\, A_1, \ldots, A_k \,\} \mid I \in \mathcal{I} \,\}$$

[6] i.e. for each $I, J \in \mathcal{K}$ it either holds $I \subseteq J$ or $J \subseteq I$.

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

$$\forall J \in \mathcal{J} : J' \subseteq J.$$

Consider the Herbrand interpretation $I' \in \mathcal{K}$, such that $J' = I' \cap \{ A_1, \ldots, A_k \}$. Since $I^* \models B_1 \wedge \ldots \wedge B_m$ and $I^* \subseteq I'$, it holds $I' \models B_1 \wedge \ldots \wedge B_m$. From $I' \in \mathcal{MOD}_{\mathcal{P}}$ we get $I' \models A_1 \vee \ldots \vee A_k$, i.e. $J' = I' \cap \{ A_1, \ldots, A_k \} \neq \emptyset$. Since all elements of $\mathcal{J}$ are supersets of J', we get

$$\forall I \in \mathcal{K} : J' \subset I.$$

This shows that $J' \subseteq I^*$, i.e. $I^* \models A_1 \vee \ldots \vee A_k$. Thus, I^* is a Herbrand model of $\mathcal{P}$ contained in $\mathcal{I}_M$.

□

5 Relationships between State Generation and Model Generation

For disjunctive logic programs $\mathcal{P}$ the minimal model state $MS_{\mathcal{P}}$ and the set $\mathcal{MM}_{\mathcal{P}}$ of minimal Herbrand models of $\mathcal{P}$ are *dual* concepts, i.e., they can be derived from each other. According to [10], it holds

$$\begin{aligned} \mathcal{MM}_{\mathcal{P}} &= \mathcal{MM}(MS_{\mathcal{P}}), \\ MS_{\mathcal{P}} &= MS(\mathcal{MM}_{\mathcal{P}}), \end{aligned}$$

where the dualization operations are defined by

$$\begin{aligned} MS(\mathcal{I}) &= \{C \in D_{HB_{\mathcal{P}}} \mid \forall I \in \mathcal{I} : I \models C \}, \\ \mathcal{MM}(S) &= min(\{I \in \mathcal{HI}_{\mathcal{P}} \mid \forall C \in S : I \models C \}). \end{aligned}$$

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 $\mathcal{MM}_{\mathcal{P}}$ represents the disjunction of the conjunctions formed by its models.

This duality relates the least fixpoints $MS_{\mathcal{P}}$ and $\mathcal{MM}_{\mathcal{P}}$ ($\equiv_{\sqsubseteq} \mathcal{MOD}_{\mathcal{P}}$) of the consequence operators $\mathcal{T}^s_{\mathcal{P}}$ and $\mathcal{T}^{\text{INT}}_{\mathcal{P}}$, respectively (see Figure 2). In the following we also compare the *intermediate results* of the respective fixpoint iterations. Corollary 5.3 establishes the relationship between the ordinal powers $\mathcal{T}^s_{\mathcal{P}} \uparrow n$ and $\mathcal{T}^{\text{INT}}_{\mathcal{P}} \uparrow n$.

Lemma 5.1 (Connection between $\mathcal{T}^s_{\mathcal{P}}$ and $\mathcal{T}^{\text{INT}}_{\mathcal{P}}$ [17])
Let $\mathcal{P}$ be a disjunctive logic program, let $\mathcal{I}$ be a coin, and let $S \subseteq D_{HB_{\mathcal{P}}}$ be a disjunctive Herbrand state, such that $\mathcal{I} \subseteq \mathcal{MOD}(S)$. Then

$$\begin{aligned} \mathcal{T}^{\text{INT}}_{\mathcal{P}}(\mathcal{I}) &\subseteq \mathcal{MOD}(\mathcal{T}^s_{\mathcal{P}}(S)), \\ \mathcal{T}^s_{\mathcal{P}}(S) &\subseteq MS(\mathcal{T}^{\text{INT}}_{\mathcal{P}}(\mathcal{I})). \end{aligned}$$

$$
\begin{tikzcd}
\mathcal{T}^s_\mathcal{P} \uparrow 0 = \emptyset \arrow[r, shift left, "\mathcal{M}od"] \arrow[d, "\mathcal{T}^s_\mathcal{P}"] & \mathcal{T}^{\mathrm{INT}}_\mathcal{P} \uparrow 0 = \bot_{exp} \arrow[l, shift left, "MS"] \arrow[d, "\mathcal{T}^{\mathrm{INT}}_\mathcal{P}"] \\
\mathcal{T}^s_\mathcal{P} \uparrow n \arrow[d, "\mathcal{T}^s_\mathcal{P}"] & \mathcal{T}^{\mathrm{INT}}_\mathcal{P} \uparrow n \arrow[d, "\mathcal{T}^{\mathrm{INT}}_\mathcal{P}"] \\
\mathcal{T}^s_\mathcal{P} \uparrow \omega \equiv_{\triangleright} MS_\mathcal{P} \arrow[r, shift left, "\mathcal{M}od"] & \mathcal{T}^{\mathrm{INT}}_\mathcal{P} \uparrow \omega \equiv_{\sqsubseteq} \mathcal{MM}_\mathcal{P} \arrow[l, shift left, "MS"]
\end{tikzcd}
$$

Figure 2: Comparing Fixpoint Computations

Proof:

(i) Assume M is a Herbrand interpretation, such that

$$M \in \mathcal{T}^{\mathrm{INT}}_\mathcal{P}(\mathcal{I}).$$

We will show that $M \in \mathcal{M}od(\mathcal{T}^s_\mathcal{P}(S))$.

There is some $I \in \mathcal{I}$, such that $M \in \mathcal{M}od(\mathcal{T}^s_\mathcal{P}(I))$. Thus, $I \subseteq M$. Since $\mathcal{I} \subseteq \mathcal{M}od(S)$, we get $I \in \mathcal{M}od(S)$. Since I is a model of S, we get that M is a model of S. For each $C \in \mathcal{T}^s_\mathcal{P}(S) \setminus S$, there is a rule

$$C' \leftarrow B_1 \wedge \ldots \wedge B_m \in gnd(\mathcal{P})$$

and there are facts $B_i \vee C_i \in S$, such that $C = C' \vee C_1 \vee \ldots \vee C_m$. If $I \models C_1 \vee \ldots \vee C_m$, then $M \models C$. Otherwise, since $I \models B_i \vee C_i$, $1 \le i \le m$, we get $I \models B_1 \wedge \ldots \wedge B_m$. Thus, $C' \in \mathcal{T}^s_\mathcal{P}(I)$. Since $M \in \mathcal{M}od(\mathcal{T}^s_\mathcal{P}(I))$, we get that $M \models C'$. Thus, again $M \models C$. Summarizing, M is a model of $\mathcal{T}^s_\mathcal{P}(S) \setminus S$.

This shows that $M \in \mathcal{M}od(\mathcal{T}^s_\mathcal{P}(S))$, i.e.

$$\mathcal{T}^{\mathrm{INT}}_\mathcal{P}(\mathcal{I}) \subseteq \mathcal{M}od(\mathcal{T}^s_\mathcal{P}(S)).$$

(ii) Application of MS to (i) yields

$$MS(\mathcal{M}od(\mathcal{T}^s_\mathcal{P}(S))) \subseteq MS(\mathcal{T}^{\mathrm{INT}}_\mathcal{P}(\mathcal{I})).$$

Since $\mathcal{T}^s_\mathcal{P}(S) \subseteq MS(\mathcal{M}od(\mathcal{T}^s_\mathcal{P}(S)))$, we get

$$\mathcal{T}^s_\mathcal{P}(S) \subseteq MS(\mathcal{T}^{\mathrm{INT}}_\mathcal{P}(\mathcal{I})).$$

$\square$

From this connection we can derive some relationships between *state generation* and *model generation*, i.e., between the ordinal powers $\mathcal{T}^s_\mathcal{P} \uparrow \alpha$ of the operator $\mathcal{T}^s_\mathcal{P}$ on disjunctive Herbrand states and the ordinal powers $\mathcal{T}^{\mathrm{INT}}_\mathcal{P} \uparrow \alpha$ of the operator $\mathcal{T}^{\mathrm{INT}}_\mathcal{P}$ on sets of Herbrand interpretations.

Theorem 5.2 (State Generation vs. Model Generation)
Let $\mathcal{P}$ be a disjunctive logic program, let $S \subseteq \mathit{DHB}_{\mathcal{P}}$ be a disjunctive Herbrand state of $\mathcal{P}$ and let $\mathcal{I}$ be a coin, such that $\mathcal{I} \subseteq \mathcal{MOD}(S)$ and $S \subseteq \mathit{MS}(\mathcal{I})$. For all ordinals α it holds

$$\begin{aligned} \mathcal{T}_{\mathcal{P}}^{\mathrm{INT}\uparrow\alpha}(\mathcal{I}) &\subseteq \mathcal{MOD}(\mathcal{T}_{\mathcal{P}}^{s\uparrow\alpha}(S)), \\ \mathcal{T}_{\mathcal{P}}^{s\uparrow\alpha}(S) &\subseteq \mathit{MS}(\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}\uparrow\alpha}(\mathcal{I})). \end{aligned}$$

Proof:

(i) The first set inclusion is shown by *induction* on α.

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

$$\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}\uparrow 0}(\mathcal{I}) = \mathcal{I} \subseteq \mathcal{MOD}(S) = \mathcal{MOD}(\mathcal{T}_{\mathcal{P}}^{s\uparrow 0}(S)).$$

$\alpha \to \alpha+1$: Using the induction assumption for α and the monotonicity of $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ we get

$$\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}\uparrow\alpha+1}(\mathcal{I}) = \mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}(\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}\uparrow\alpha}(\mathcal{I})) \subseteq \mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}(\mathcal{MOD}(\mathcal{T}_{\mathcal{P}}^{s\uparrow\alpha}(S))).$$

Let $\mathcal{I}' = \mathcal{MOD}(\mathcal{T}_{\mathcal{P}}^{s\uparrow\alpha}(S))$ and $S' = \mathcal{T}_{\mathcal{P}}^{s\uparrow\alpha}(S)$. Then $\mathcal{I}' = \mathcal{MOD}(S')$. Applying the first set inclusion of Lemma 5.1 to S' and $\mathcal{I}'$ yields $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}(\mathcal{I}') \subseteq \mathcal{MOD}(\mathcal{T}_{\mathcal{P}}^{s}(S'))$, i.e.,

$$\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}(\mathcal{MOD}(\mathcal{T}_{\mathcal{P}}^{s\uparrow\alpha}(S))) \subseteq \mathcal{MOD}(\mathcal{T}_{\mathcal{P}}^{s}(\mathcal{T}_{\mathcal{P}}^{s\uparrow\alpha}(S))) = \mathcal{MOD}(\mathcal{T}_{\mathcal{P}}^{s\uparrow\alpha+1}(S)).$$

By transitivity we get:

$$\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}\uparrow\alpha+1}(\mathcal{I}) \subseteq \mathcal{MOD}(\mathcal{T}_{\mathcal{P}}^{s\uparrow\alpha+1}(S)).$$

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

$$\bigcap_{\beta<\alpha} \mathcal{T}_{\mathcal{P}}^{\mathrm{INT}\uparrow\beta}(\mathcal{I}) \subseteq \bigcap_{\beta<\alpha} \mathcal{MOD}(\mathcal{T}_{\mathcal{P}}^{s\uparrow\beta}(S)).$$

Since $\bigcap_{\beta<\alpha} \mathcal{T}_{\mathcal{P}}^{\mathrm{INT}\uparrow\beta}(\mathcal{I}) = \mathcal{T}_{\mathcal{P}}^{\mathrm{INT}\uparrow\alpha}(\mathcal{I})$, and

$$\bigcap_{\beta<\alpha} \mathcal{MOD}(\mathcal{T}_{\mathcal{P}}^{s\uparrow\beta}(S)) = \mathcal{MOD}(\bigcup_{\beta<\alpha} \mathcal{T}_{\mathcal{P}}^{s\uparrow\beta}(S)) = \mathcal{MOD}(\mathcal{T}_{\mathcal{P}}^{s\uparrow\alpha}(S)),$$

we get that

$$\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}\uparrow\alpha}(\mathcal{I}) \subseteq \mathcal{MOD}(\mathcal{T}_{\mathcal{P}}^{s\uparrow\alpha}(S)).$$

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

$$\mathit{MS}(\mathcal{MOD}(\mathcal{T}_{\mathcal{P}}^{s\uparrow\alpha}(S))) \subseteq \mathit{MS}(\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}\uparrow\alpha}(\mathcal{I})).$$

By chaining with $\mathcal{T}_{\mathcal{P}}^{s\uparrow\alpha}(S) \subseteq \mathit{MS}(\mathcal{MOD}(\mathcal{T}_{\mathcal{P}}^{s\uparrow\alpha}(S)))$ we get the second set inclusion.

$\square$

Corollary 5.3 (State Generation vs. Model Generation)
Let $\mathcal{P}$ be a disjunctive logic program. For all ordinals α it holds

$$\begin{aligned} \mathcal{T}_{\mathcal{P}}^{\mathrm{INT}} \uparrow \alpha &\subseteq \mathcal{MOD}(\mathcal{T}_{\mathcal{P}}^{s} \uparrow \alpha), \\ \mathcal{T}_{\mathcal{P}}^{s} \uparrow \alpha &\subseteq \mathit{MS}(\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}} \uparrow \alpha). \end{aligned}$$

Sometimes, the set inclusions in Corollary 5.3 are strict, as the following example shows.

Example 5.4 (State Generation vs. Model Generation)
Given some $n \in I\!N_+$, consider the disjunctive logic program

$$\mathcal{P} = \{ a \leftarrow b_i \mid 1 \le i \le n \} \cup \{ b_1 \vee \ldots \vee b_n \}$$

with n rules and one fact. Let $S_m = \mathcal{T}^s_\mathcal{P} \uparrow m$ and $\mathcal{I}_m = \mathcal{T}^M_\mathcal{P} \uparrow m$. For $n = 2$ we get

$$\begin{array}{rcl} S_1 & = & \{ b_1 \vee b_2 \}, \\ S_2 & = & \{ b_1 \vee b_2,\ a \vee b_1,\ a \vee b_2 \}, \\ \mathcal{I}_1 & = & \{ \{ b_1 \}, \{ b_2 \} \}, \\ \mathcal{I}_2 & = & \{ \{ b_1, a \}, \{ b_2, a \} \}. \end{array}$$

Thus, for the atom a we get $a \in MS(\mathcal{I}_2)$, but $a \notin S_2$. For $n \ge 2$ we get the disjunctive Herbrand states

$$\begin{array}{rcl} S_m & = & \{ b_1 \vee \ldots \vee b_n \} \cup \\ & & \{ a \vee b_{i_1} \vee \ldots \vee b_{i_k} \mid \{i_1, \ldots, i_k\} \subseteq \langle 1, n \rangle,\ n - m + 1 \le k \le n - 1 \}, \\ & & \text{for all } m \in I\!N_+, \end{array}$$

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,

$$\begin{array}{rcl} \mathcal{I}_1 & = & \{ \{b_1\}, \ldots, \{b_n\} \}, \\ \mathcal{I}_m & = & \{ \{ b_i, a \} \mid 1 \le i \le n \}, \text{ for all } m \ge 2. \end{array}$$

The least fixpoint of $\mathcal{T}^M_\mathcal{P}$ is always reached after two steps: $lfp(\mathcal{T}^M_\mathcal{P}) = \mathcal{I}_2$, whereas the least fixpoint of $\mathcal{T}^s_\mathcal{P}$ is reached after $n+1$ steps: $lfp(\mathcal{T}^s_\mathcal{P}) = S_{n+1} = S_n \cup \{a\}$. We can show that

$$\begin{array}{rcl} \mathcal{MOD}(S_n) & = & exp(\mathcal{I}_n \cup \{I_b\}), \\ MS(\mathcal{I}_n) & = & exp(S_n \cup \{a\}), \end{array}$$

where $I_b = \{ b_i \mid 1 \le i \le n \}$. This shows that $a \in MS(\mathcal{I}_n) \setminus S_n$, $I_b \in \mathcal{MOD}(S_n) \setminus \mathcal{I}_n$, i.e., the set inclusions in Corollary 5.3 are strict.

Since the operator $\mathcal{T}^s_\mathcal{P}$ is continuous (see [12]), its ordinal powers converge towards its least fixpoint in at most ω steps, i.e., $lfp(\mathcal{T}^s_\mathcal{P}) = \mathcal{T}^s_\mathcal{P} \uparrow \omega \equiv_\triangleright MS_\mathcal{P}$. Corollary 5.3 helps to show that the ordinal powers of the operator $\mathcal{T}^{\mathrm{INT}}_\mathcal{P}$ converge towards its least fixpoint in at most ω steps, too. That is, the closure ordinal of $\mathcal{T}^{\mathrm{INT}}_\mathcal{P}$ on $\mathcal{O}_{exp}$ is ω. Moreover, this least fixpoint is equivalent to the set $\mathcal{MM}_\mathcal{P}$ of the minimal Herbrand models of the program.

Theorem 5.5 (Convergence of Model Generation by $\mathcal{T}^{\mathrm{INT}}_\mathcal{P}$)
Let $\mathcal{P}$ be a disjunctive logic program. Then

$$\mathcal{MM}_\mathcal{P} \equiv_\sqsubseteq lfp(\mathcal{T}^{\mathrm{INT}}_\mathcal{P}) = \mathcal{T}^{\mathrm{INT}}_\mathcal{P} \uparrow \omega.$$

Proof:
We will show that $\mathcal{MM}_\mathcal{P} \sqsubseteq \mathcal{T}^{\mathrm{INT}}_\mathcal{P} \uparrow \omega$ and that $\mathcal{T}^{\mathrm{INT}}_\mathcal{P} \uparrow \omega \sqsubseteq \mathcal{MM}_\mathcal{P}$.

(i) From Corollary 5.3 we know that $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}} \uparrow \omega \subseteq \mathcal{MOD}(\mathcal{T}_{\mathcal{P}}^{s} \uparrow \omega)$. With $\mathcal{T}_{\mathcal{P}}^{s} \uparrow \omega \equiv_{\triangleright} MS_{\mathcal{P}}$, we get

$$\mathcal{MOD}(MS_{\mathcal{P}}) \sqsubseteq \mathcal{T}_{\mathcal{P}}^{\mathrm{INT}} \uparrow \omega.$$

From Lemma 4.7 we know that $\mathcal{MM}(MS_{\mathcal{P}}) \equiv_{\sqsubseteq} \mathcal{MOD}(MS_{\mathcal{P}})$, and from [12] we know that $\mathcal{MM}(MS_{\mathcal{P}}) = \mathcal{MM}_{\mathcal{P}}$. This shows that

$$\mathcal{MM}_{\mathcal{P}} \sqsubseteq \mathcal{T}_{\mathcal{P}}^{\mathrm{INT}} \uparrow \omega.$$

(ii) From

$$\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}} \uparrow 0 = \bot_{exp} \sqsubseteq \mathcal{MM}_{\mathcal{P}}$$

and $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}(\mathcal{MM}_{\mathcal{P}}) = \mathcal{MM}_{\mathcal{P}}$ and the monotonicity of the operator $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ on the partial ordering $\mathcal{O}_{exp} = \langle 2^{\mathcal{HI}_{\mathcal{P}}}_{exp}, \sqsubseteq \rangle$, by induction on $n \in I\!N_0$ we get that

$$\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}} \uparrow n \sqsubseteq \mathcal{MM}_{\mathcal{P}}, \text{ for all } n \in I\!N_0.$$

This shows that

$$\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}} \uparrow \omega = lub_{exp}(\{\, \mathcal{T}_{\mathcal{P}}^{\mathrm{INT}} \uparrow n \mid n \in I\!N_0 \,\}) \sqsubseteq \mathcal{MM}_{\mathcal{P}}.$$

Hence $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}} \uparrow \omega \equiv_{\sqsubseteq} \mathcal{MM}_{\mathcal{P}}$. Since $\mathcal{MM}_{\mathcal{P}}$ is a fixpoint of $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$, i.e., $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}(\mathcal{MM}_{\mathcal{P}}) = \mathcal{MM}_{\mathcal{P}}$, this implies that $lfp(\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}) \equiv_{\sqsubseteq} \mathcal{MM}_{\mathcal{P}}$.

□

Thus, the operator $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ is an example of a monotonic operator that is not continuous, but nevertheless reaches its least fixpoint in at most ω steps.

6 Iterative Model Generation of Perfect and (Partial) Stable Semantics

In [5], 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 $\mathcal{T}_{\mathcal{P}}^{\mathrm{INT}}$ defined on Section 4.

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

$$A_1 \vee \ldots \vee A_k \leftarrow B_1 \wedge \ldots \wedge B_m \wedge not\, C_1 \wedge \ldots \wedge not\, C_n, \tag{2}$$

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; $k, m, n \in I\!N_0$; and *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. $A_1 \vee \ldots \vee A_k$) 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.

A normal disjunctive logic program $\mathcal{P}$ is called *stratified* (see [13]) 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 (2) in $\mathcal{P}$ there exists a constant c, $1 \leq c \leq r$, such that: (1) the definition of each A_i is contained in $\mathcal{P}_c$; (2) the definition of each B_j is contained in $\bigcup_{s \leq c} \mathcal{P}_s$; and (3) the definition of each C_l 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* $\mathcal{P}$.

The intended meaning of a stratified disjunctive logic program $\mathcal{P}$ is given by its collection of *perfect models* as defined in [13]. It is well–known (see e.g. [13]) 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 $\mathcal{PM}_i$ denote the set of perfect models of the first i strata of P, i.e. of the logic program $\bigcup_{s \leq i} \mathcal{P}_s$. 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 $\mathcal{PM}_1 = \mathcal{MM}_{\mathcal{P}_1}$. When $\mathcal{PM}_i$ has been constructed, it is used to evaluate the negated by default literals appearing in the $i+1$ stratum as follows: If $I \in \mathcal{PM}_i$, then $\mathcal{P}^I_{i+1}$ is the disjunctive logic program:

$$\begin{aligned}\mathcal{P}^I_{i+1} = \{ \; & A_1 \vee \ldots \vee A_k \leftarrow B_1 \wedge \ldots \wedge B_m \,| \\ & A_1 \vee \ldots \vee A_k \leftarrow B_1 \wedge \ldots \wedge B_m \wedge not\, C_1 \wedge \ldots \wedge not\, C_n \in \mathcal{P}_{i+1} \text{ and} \\ & \{C_1, \ldots, C_n\} \cap I = \emptyset\}.\end{aligned}$$

$\mathcal{P}^I_{i+1}$ is called the *Gelfond–Lifschitz* transformation of $\mathcal{P}_{i+1}$ with respect to I. The perfect models of the first $i+1$ strata are then taken to be $\bigcup_{I \in \mathcal{PM}_i} \mathcal{MM}_{\mathcal{P}^I_{i+1} \cup I}$. The collection of perfect models of $\mathcal{P}$ is exactly $\mathcal{PM}_r$. It is worth noticing that this collection of models is the same independently 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 $\mathcal{T}^M_{\mathcal{P}}$ to compute this collection of perfect models [5]. We extend now this procedure to arbitrary stratified disjunctive logic programs by defining the iterative version of our operator $\mathcal{T}^{\text{INT}}_{\mathcal{P}}$.

Definition 6.1 (Iterative version of $\mathcal{T}^{\text{INT}}_{\mathcal{P}}$)
Let $\mathcal{P}$ be a stratified disjunctive logic program and $\{\mathcal{P}_1, \ldots, \mathcal{P}_r\}$ be a stratification of $\mathcal{P}$. The iterative version of $\mathcal{T}^{\text{INT}}_{\mathcal{P}}$ is defined as follows:

$$\begin{aligned}\mathcal{T}^I_{\langle \mathcal{P}_1 \rangle} &= min(\mathcal{T}^I_{\mathcal{P}_1} \uparrow \alpha_{\langle 1, \emptyset \rangle}), \\ \mathcal{T}^I_{\langle \mathcal{P}_1, \ldots, \mathcal{P}_{n+1} \rangle} &= \bigcup_{I \in \mathcal{T}^I_{\langle \mathcal{P}_1, \ldots, \mathcal{P}_n \rangle}} min(\mathcal{T}^I_{(\mathcal{P}^I_{n+1} \cup I)} \uparrow \alpha_{\langle n+1, I \rangle})\end{aligned}$$

where $\alpha_{\langle i, I \rangle}$ is the closure ordinal of $\mathcal{T}^I_{(\mathcal{P}^I_i \cup I)}$.

The set of perfect models of $\mathcal{P}$ is given by $\mathcal{T}^I_{\langle \mathcal{P}_1, \ldots, \mathcal{P}_r \rangle}$. Notice that, due to Theorem 5.5, each of the closure ordinals α^I_i in the previous definition is at most ω.

Fernández et al. [3] showed how to transform an arbitrary normal disjunctive logic program $\mathcal{P}$ into a stratified one, denoted by $\mathcal{P}^{\mathcal{E}}$, in such a way that the perfect models of $\mathcal{P}^{\mathcal{E}}$ that satisfy a given set of integrity constraints correspond to the stable models of $\mathcal{P}$ (as defined in [14]). They called this transformation the *evidential transformation.* Using this characterization of stable models, the iterative version of $\mathcal{T}^{\text{INT}}_{\mathcal{P}}$ can be used to construct the stable semantics of $\mathcal{P}$.

Similarly, the operator $\mathcal{T}^{\text{INT}}_{\mathcal{P}}$ can be used to generate the collection of partial (or 3–valued) stable models ([15]) of a normal disjunctive logic program $\mathcal{P}$ by using a characterization of this collection of models given by Ruiz and Minker in [16]. This characterization is based on a transformation called the *3S-transformation* which applied to a normal disjunctive logic program $\mathcal{P}$ produces a constrained 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 ([22]) is a distinct partial stable model of a normal logic program (see [15]), then model generation for the well–founded semantics is also achieved using the operator $\mathcal{T}^{\text{INT}}_{\mathcal{P}}$ together with the 3S-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 $\mathcal{MM}_{\mathcal{P}}$ 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 $\mathcal{T}^s_{\mathcal{P}}$. Using *model generation*, $\mathcal{MM}_{\mathcal{P}}$ is computed as the least fixpoint of the consequence operator $\mathcal{T}^{\mathrm{INT}}_{\mathcal{P}}$.
- By *dualization* of the minimal model state $MS_{\mathcal{P}}$ we get the set of minimal models of $\mathcal{P}$ as $\mathcal{MM}(MS_{\mathcal{P}})$. Similarly, dualization of $\mathcal{MM}_{\mathcal{P}}$ yields $MS_{\mathcal{P}}$ as $MS(\mathcal{MM}_{\mathcal{P}})$.

As shown in this paper, both approaches converge in at most ω steps. For the case of disjunctive deductive databases, these approaches have been implemented within the disjunctive deductive database engine DisLog[7], cf. [19], 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 $n = |can(MS_{\mathcal{P}})|$ of minimal[8] disjunctions in $MS_{\mathcal{P}}$ and the number $m = |\mathcal{MM}_{\mathcal{P}}|$ 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 $\mathcal{MM}_{\mathcal{P}}$ by model generation.
- If n is much bigger than m, then the fastest is to use model generation for deriving $\mathcal{MM}_{\mathcal{P}}$ and to derive $MS_{\mathcal{P}}$ from $\mathcal{MM}_{\mathcal{P}}$ by dualization.
- If n is much smaller than m, then the fastest is to use hyperresolution for deriving $MS_{\mathcal{P}}$ and to derive $\mathcal{MM}_{\mathcal{P}}$ from $MS_{\mathcal{P}}$ by dualization.

Based on the theorems of Section 5 it has been shown in [18] that during a fixpoint iteration with $\mathcal{T}^s_{\mathcal{P}}$ it is possible to switch to some iterations of model generation. Intermediate information about the size of the ordinal powers $\mathcal{T}^s_{\mathcal{P}} \uparrow n$ 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 $\mathcal{T}^{\mathrm{INT}}_{\mathcal{P}}$, 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 $\mathcal{P}^{\mathcal{E}}$ ([3]) and by model generation with the 3S-Transformation $\mathcal{P}^{3S}$ ([16]).

References

[1] *B.A. Davey, H.A. Priestley:* Introduction to Lattices and Order, Cambridge University Press, 1990.

[2] *J.A. Fernández:* Disjunctive Deductive Databases, Ph.D. Thesis, University of Maryland, 1993.

[3] *J.A. Fernández, J. Lobo, J. Minker, V.S. Subrahmanian:* Disjunctive LP + Integrity Constrains = Stable Model Semantics, Annals of Mathematics and Artificial Intelligence, vol. 8 (3-4), 1993, pp. 449-474.

[7] for a demo version of DisLog visit http://www.informatik.uni-tuebingen.de/dislog.html

[8] a disjunction C in $MS_{\mathcal{P}}$ is minimal if there is no sub-disjunction C' of C in $MS_{\mathcal{P}}$.

[4] *J.A. Fernández and J. Minker:* Bottom-up Evaluation of Hierarchical Disjunctive Deductive Databases. Proc. Eighth Intl. Conference on Logic Programming, MIT Press, 1991, pp. 660-675.

[5] *J.A. Fernández and J. Minker:* Bottom-up computation of perfect models for disjunctive theories, Journal of Logic Programming, vol. 25(1), 1995, pp. 33-51.

[6] *K. Inoue, M. Koshimura, R. Hasegawa:* Embedding Negation as Failure into a Model Generation Theorem Prover, Proc. Intl. Conference on Automated Deduction 1992, pp. 400-415.

[7] *S.C. Kleene:* Introduction to MetaMathematics, Van Nostrand, New York, 1962.

[8] *B. Knaster:* Un théorèm sur les fonctions d'ensembles, Annals Soc. Pol. Math., vol. 6, 1928, pp. 133-134.

[9] *J.W. Lloyd:* Foundations of Logic Programming, Springer, second edition, 1987.

[10] *J. Lobo, J. Minker, A. Rajasekar:* Foundations of Disjunctive Logic Programming, MIT Press, 1992.

[11] *J. Minker:* On Indefinite Databases and the Closed World Assumption, Proc. Conference on Automated Deduction 1982, Springer LNCS 138, 1982, pp. 292-308.

[12] *J. Minker, A. Rajasekar:* A Fixpoint Semantics for Disjunctive Logic Programs, Journal of Logic Programming, vol. 9(1), 1990, pp. 45-74.

[13] *T.C. Przymusinski*: On the declarative semantics of Deductive Databases and Logic Programming, in Foundations of Deductive Databases and Logic Programming, J. Minker ed., 1988, pp. 193-216.

[14] *T.C. Przymusinski*: Extended Stable Semantics for Normal and Disjunctive Programs, Proc. of the Seventh Intl. Conference on Logic Programming, 1990, 459-477.

[15] *T.C. Przymusinski*: Stable Semantics for Disjunctive Programs, New Generation Computing, vol. 9, 1991, pp. 401-424.

[16] *C. Ruiz, J. Minker*: Computing Stable and Partial Stable Models of Extended Disjunctive Logic Programs, in Nonmonotonic Extensions of Logic Programming, 1995, J. Dix and L.M. Pereira and T.C. Przymusinski eds., Lecture Notes in Computer Science 927. Springer-Verlag, pp. 205-229.

[17] *D. Seipel*: Efficient Reasoning in Disjunctive Deductive Databases, Habilitation Thesis, University of Tübingen, 1995.

[18] *D. Seipel, U. Güntzer*: Mixed Fixpoint Theory for Disjunctive Deductive Databases, Proc. Intl. Workshop on Logic Programming 1995, pp. 227-236.

[19] *D. Seipel, H. Thöne*: DisLog – A System for Reasoning in Disjunctive Deductive Databases, Proc. Intl. Workshop on the Deductive Approach to Information Systems and Databases 1994, pp. 325-343.

[20] *A. Tarski:* A Lattice-Theoretical Fixpoint Theorem and its Applications, Pacific Journal of Mathematics, vol. 5, 1955, pp. 285-309.

[21] *M.H. van Emden, R.A. Kowalski:* The Semantics of Predicate Logic as a Programming Language, Journal of the ACM, vol. 23(4), 1976, pp. 733-742.

[22] *A. Van Gelder, K.A. Ross, J.S. Schlipf:*, Unfounded Sets and Well-Founded Semantics for General Logic Programs, Proc. of the Seventh ACM Symposium on Principles of Database Systems, 1988, pp. 221-230.

[23] *W. Wechler:* Universal Algebra for Computer Scientists, Springer, 1992.