

Wright State University

CORE Scholar

---

Computer Science and Engineering Faculty  
Publications

Computer Science & Engineering

---

6-2005

## Towards a Systematic Account of Different Semantics for Logic Programs

Pascal Hitzler  
pascal.hitzler@wright.edu

Follow this and additional works at: <https://corescholar.libraries.wright.edu/cse>



Part of the [Computer Sciences Commons](#), and the [Engineering Commons](#)

---

### Repository Citation

Hitzler, P. (2005). Towards a Systematic Account of Different Semantics for Logic Programs. *Journal of Logic and Computation*, 15 (3), 391-404.

<https://corescholar.libraries.wright.edu/cse/166>

This Article is brought to you for free and open access by Wright State University's CORE Scholar. It has been accepted for inclusion in Computer Science and Engineering Faculty Publications by an authorized administrator of CORE Scholar. For more information, please contact [library-corescholar@wright.edu](mailto:library-corescholar@wright.edu).

# Towards a Systematic Account of Different Semantics for Logic Programs

Pascal Hitzler\*\*

AIFB, Universität Karlsruhe, Germany  
<http://www.aifb.uni-karlsruhe.de/WBS/phi/hitzler@aifb.uni-karlsruhe.de>

**Abstract.** In [14, 15], a new methodology has been proposed which allows to derive uniform characterizations of different declarative semantics for logic programs with negation. One result from this work is that the well-founded semantics can formally be understood as a stratified version of the Fitting (or Kripke-Kleene) semantics. The constructions leading to this result, however, show a certain asymmetry which is not readily understood. We will study this situation here with the result that we will obtain a coherent picture of relations between different semantics for normal logic programs.

## 1 Introduction

Within the past twenty years, many different declarative semantics for logic programs with negation have been developed. Different perspectives on the question what properties a semantics should foremost satisfy, have led to a variety of diverse proposals. From a knowledge representation and reasoning point of view it appears to be important that a semantics captures established non-monotonic reasoning frameworks, e.g. Reiter's default logic [24], and that they allow intuitively appealing, i.e. "common sense", encodings of AI problems. The semantics which, due to common opinion by researchers in the field, satisfy these requirements best, are the least model semantics for definite programs [18], and for normal programs the stable [11] and the well-founded semantics [26]. Of lesser importance, albeit still acknowledged in particular for their relation to resolution-based logic programming, are the Fitting semantics [8] and approaches based on stratification [2, 22].

The semantics just mentioned are closely connected by a number of well- (and some lesser-) known relationships, and many authors have contributed to this understanding. Fitting [10] provides a framework using

---

\*\* The author acknowledges support by the German Federal Ministry for Education and Research (BMBF) under the SmartWeb project, and by the European Union under the KnowledgeWeb Network of Excellence.

Belnap’s four-valued logic which encompasses supported, stable, Fitting, and well-founded semantics. His work was recently extended by Denecker, Marek, and Truszczyński [4]. Przymusiński [23] gives a version in three-valued logic of the stable semantics, and shows that it coincides with the well-founded one. Van Gelder [25] constructs the well-founded semantics using the Gelfond-Lifschitz operator originally associated with the stable semantics. Dung and Kanchanasut [6] define the notion of fixpoint completion of a program which provides connections between the supported and the stable semantics, as well as between the Fitting and the well-founded semantics, studied by Fages [7] and Wendt [28]. Hitzler and Wendt [14, 15] have recently provided a unifying framework using level mappings, and results which amongst other things give further support to the point of view that the stable semantics is a formal and natural extension to normal programs of the least model semantics for definite programs. Furthermore, it was shown that the well-founded semantics can be understood, formally, as a stratified version of the Fitting semantics.

This latter result, however, exposes a certain asymmetry in the construction leading to it, and it is natural to ask the question as to what exactly is underlying it. This is what we will study in the sequel. In a nutshell, we will see that formally this asymmetry is due to the well-known preference of falsehood in logic programming semantics. More importantly, we will also see that a “dual” theory, obtained from preferring truth, can be established which is in perfect analogy to the close and well-known relationships between the different semantics mentioned above.

We want to make it explicit from the start that we do not intend to provide new semantics for practical purposes<sup>1</sup>. We rather want to focus on the deepening of the theoretical insights into the relations between different semantics, by painting a coherent and complete picture of the dependencies and interconnections. We find the richness of the theory very appealing, and strongly supportive of the opinion that the major semantics studied in the field are founded on a sound theoretical base. Indeed, from a mathematical perspective one expects major notions in a field to be strongly interconnected, and historic developments show that such foundational underpinnings are supportive of a wide and lasting impact of a field. The results in this paper aim at establishing these foundations in a clean and formally satisfying manner.

The plan of the paper is as follows. In Section 2 we will introduce notation and terminology needed for proving the results in the main body of the paper. We will also review in detail those results from [14, 15]

---

<sup>1</sup> Although there may be some virtue to this perspective, see [12].

which triggered and motivated our investigations. In Section 3 we will provide a variant of the stable semantics which prefers truth, and in Section 4 we will do likewise for the well-founded semantics. Throughout, our definitions will be accompanied by results which complete the picture of relationships between different semantics.

This paper is a revised version of the conference contribution [13].

*Acknowledgements.* I am grateful for comments by anonymous referees which helped to improve the presentation, and in particular for bringing my attention to the related and independent work reported in [3, 5].

## 2 Preliminaries and Notation

A (*normal*) *logic program* is a finite set of (universally quantified) *clauses* of the form  $\forall(A \leftarrow A_1 \wedge \dots \wedge A_n \wedge \neg B_1 \wedge \dots \wedge \neg B_m)$ , commonly written as  $A \leftarrow A_1, \dots, A_n, \neg B_1, \dots, \neg B_m$ , where  $A$ ,  $A_i$ , and  $B_j$ , for  $i = 1, \dots, n$  and  $j = 1, \dots, m$ , are atoms over some given first order language.  $A$  is called the *head* of the clause, while the remaining atoms make up the *body* of the clause, and depending on context, a body of a clause will be a set of literals (i.e. atoms or negated atoms) or the conjunction of these literals. Care will be taken that this identification does not cause confusion. We allow a body, i.e. a conjunction, to be empty, in which case it always evaluates to true. A clause with empty body is called a *unit clause* or a *fact*. A clause is called *definite*, if it contains no negation symbol. A program is called *definite* if it consists only of definite clauses. We will usually denote atoms with  $A$  or  $B$ , and literals, which may be atoms or negated atoms, by  $L$  or  $K$ .

Given a logic program  $P$ , we can extract from it the components of a first order language, and we always make the mild assumption that this language contains at least one constant symbol. The corresponding set of ground atoms, i.e. the *Herbrand base* of the program, will be denoted by  $B_P$ . For a subset  $I \subseteq B_P$ , we set  $\neg I = \{\neg A \mid A \in B_P\}$ . The set of all ground instances of  $P$  with respect to  $B_P$  will be denoted by  $\text{ground}(P)$ . For  $I \subseteq B_P \cup \neg B_P$ , we say that  $A$  is *true with respect to* (or *in*)  $I$  if  $A \in I$ , we say that  $A$  is *false with respect to* (or *in*)  $I$  if  $\neg A \in I$ , and if neither is the case, we say that  $A$  is *undefined with respect to* (or *in*)  $I$ . A (*three-valued* or *partial*) *interpretation*  $I$  for  $P$  is a subset of  $B_P \cup \neg B_P$  which is *consistent*, i.e. whenever  $A \in I$  then  $\neg A \notin I$ . A body, i.e. a conjunction of literals, is true in an interpretation  $I$  if every literal in the body is true in  $I$ , it is false in  $I$  if one of its literals is false in  $I$ , and otherwise it is

undefined in  $I$ . For a negated literal  $L = \neg A$  we will find it convenient to write  $\neg L \in I$  if  $A \in I$ . By  $I_P$  we denote the set of all (three-valued) interpretations of  $P$ . Both  $I_P$  and  $B_P \cup \neg B_P$  are complete partial orders (cpo) via set-inclusion, i.e. they contain the empty set as least element, and every ascending chain has a supremum, namely its union. A *model* of  $P$  is an interpretation  $I \in I_P$  such that for each clause  $A \leftarrow \mathbf{body}$  we have that  $\mathbf{body} \subseteq I$  implies  $A \in I$ . A *total* interpretation is an interpretation  $I$  such that no  $A \in B_P$  is undefined in  $I$ .

For an interpretation  $I$  and a program  $P$ , an  *$I$ -partial level mapping* for  $P$  is a partial mapping  $l : B_P \rightarrow \alpha$  with domain  $\mathbf{dom}(l) = \{A \mid A \in I \text{ or } \neg A \in I\}$ , where  $\alpha$  is some (countable) ordinal. We extend every level mapping to literals by setting  $l(\neg A) = l(A)$  for all  $A \in \mathbf{dom}(l)$ . A (*total*) *level mapping* is a total mapping  $l : B_P \rightarrow \alpha$  for some (countable) ordinal  $\alpha$ .

Given a normal logic program  $P$  and some  $I \subseteq B_P \cup \neg B_P$ , we say that  $U \subseteq B_P$  is an *unfounded set (of  $P$ ) with respect to  $I$*  if each atom  $A \in U$  satisfies the following condition: For each clause  $A \leftarrow \mathbf{body}$  in  $\mathbf{ground}(P)$  (at least) one of the following holds.

- (Ui) Some (positive or negative) literal in  $\mathbf{body}$  is false in  $I$ .
- (Uii) Some (non-negated) atom in  $\mathbf{body}$  occurs in  $U$ .

Given a normal logic program  $P$ , we define the following operators on  $B_P \cup \neg B_P$ .  $T_P(I)$  is the set of all  $A \in B_P$  such that there exists a clause  $A \leftarrow \mathbf{body}$  in  $\mathbf{ground}(P)$  such that  $\mathbf{body}$  is true in  $I$ .  $F_P(I)$  is the set of all  $A \in B_P$  such that for all clauses  $A \leftarrow \mathbf{body}$  in  $\mathbf{ground}(P)$  we have that  $\mathbf{body}$  is false in  $I$ . Both  $T_P$  and  $F_P$  map elements of  $I_P$  to elements of  $I_P$ . Now define the operator  $\Phi_P : I_P \rightarrow I_P$  by

$$\Phi_P(I) = T_P(I) \cup \neg F_P(I).$$

This operator is due to [8] and is well-defined and monotonic on the cpo  $I_P$ , hence has a least fixed point by the Knaster-Tarski<sup>2</sup> fixed-point theorem, and we can obtain this fixed point by defining, for each monotonic

---

<sup>2</sup> We follow the terminology from [16]. The Knaster-Tarski theorem is sometimes called Tarski theorem and states that every monotonic function on a cpo has a least fixed point, which can be obtained by transfinitely iterating the bottom element of the cpo. The Tarski-Kantorovitch theorem is sometimes referred to as the Kleene theorem or the Scott theorem (or even as “the” fixed-point theorem) and states that if the function is additionally Scott (or order-) continuous, then the least fixed point can be obtained by an iteration which is not transfinite, i.e. closes off at  $\omega$ , the least infinite ordinal. In both cases, the least fixed point is also the least pre-fixed point of the function.

operator  $F$ , that  $F \uparrow 0 = \emptyset$ ,  $F \uparrow (\alpha + 1) = F(F \uparrow \alpha)$  for any ordinal  $\alpha$ , and  $F \uparrow \beta = \bigcup_{\gamma < \beta} F \uparrow \gamma$  for any limit ordinal  $\beta$ , and the least fixed point of  $F$  is obtained as  $F \uparrow \alpha$  for some ordinal  $\alpha$ . The least fixed point of  $\Phi_P$  is called the *Kripke-Kleene model* or *Fitting model* of  $P$ , determining the *Fitting semantics* of  $P$ .

Now, for  $I \subseteq B_P \cup \neg B_P$ , let  $U_P(I)$  be the greatest unfounded set (of  $P$ ) with respect to  $I$ , which always exists due to [26]. Finally, define

$$W_P(I) = T_P(I) \cup \neg U_P(I)$$

for all  $I \subseteq B_P \cup \neg B_P$ . The operator  $W_P$ , which operates on the cpo  $B_P \cup \neg B_P$ , is due to [26] and is monotonic, hence has a least fixed point by the Knaster-Tarski<sup>2</sup> fixed-point theorem, as above for  $\Phi_P$ . It turns out that  $W_P \uparrow \alpha$  is in  $I_P$  for each ordinal  $\alpha$ , and so the least fixed point of  $W_P$  is also in  $I_P$  and is called the *well-founded model* of  $P$ , giving the *well-founded semantics* of  $P$ .

In order to avoid confusion, we will use the following terminology: the notion of *interpretation*, and  $I_P$  will be the set of all those, will by default denote consistent subsets of  $B_P \cup \neg B_P$ , i.e. interpretations in three-valued logic. We will sometimes emphasize this point by using the notion *partial interpretation*. By *two-valued interpretations* we mean subsets of  $B_P$ . Both interpretations and two-valued interpretations are ordered by subset inclusion. Each two-valued interpretation  $I$  can be identified with the partial interpretation  $I' = I \cup \neg(B_P \setminus I)$ . Note however, that in this case  $I'$  is always a maximal element in the ordering for partial interpretations, while  $I$  is in general not maximal as a two-valued interpretation<sup>3</sup>. Given a partial interpretation  $I$ , we set  $I^+ = I \cap B_P$  and  $I^- = \{A \in B_P \mid \neg A \in I\}$ .

Given a program  $P$ , we define the operator  $T_P^+$  on subsets of  $B_P$  by  $T_P^+(I) = T_P(I \cup \neg(B_P \setminus I))$ . The pre-fixed points of  $T_P^+$ , i.e. the two-valued interpretations  $I \subseteq B_P$  with  $T_P^+(I) \subseteq I$ , are exactly the models, in the sense of classical logic, of  $P$ . Post-fixed points of  $T_P^+$ , i.e.  $I \subseteq B_P$  with  $I \subseteq T_P^+(I)$  are called *supported interpretations* of  $P$ , and a supported model of  $P$  is a model  $P$  which is a supported interpretation. The supported models of  $P$  thus coincide with the fixed points of  $T_P^+$ . It is well-known that for definite programs  $P$  the operator  $T_P^+$  is monotonic on the set of all subsets of  $B_P$ , with respect to subset inclusion. Indeed it is Scott-continuous [18, 1] and, via the Tarski-Kantorovich<sup>2</sup> fixed-point theorem, achieves its least pre-fixed point  $M$ , which is also a fixed point, as the supremum of the iterates  $T_P^+ \uparrow n$  for  $n \in \mathbb{N}$ . So  $M = \text{lfp}(T_P^+) = T_P^+ \uparrow \omega$

<sup>3</sup> These two orderings in fact correspond to the knowledge and truth orderings as discussed in [9].

is the least two-valued model of  $P$ . Likewise, since the set of all subsets of  $B_P$  is a complete lattice, and therefore has greatest element  $B_P$ , we can also define  $T_P^+ \downarrow 0 = B_P$  and inductively  $T_P^+ \downarrow (\alpha + 1) = T_P^+(T_P^+ \downarrow \alpha)$  for each ordinal  $\alpha$  and  $T_P^+ \downarrow \beta = \bigcap_{\gamma < \beta} T_P^+ \downarrow \gamma$  for each limit ordinal  $\beta$ . Again by the Knaster-Tarski fixed-point theorem, applied to the superset inclusion ordering (i.e. reverse subset inclusion) on subsets of  $B_P$ , it turns out that  $T_P^+$  has a greatest fixed point,  $\mathbf{gfp}(T_P^+)$ .

The stable model semantics due to [11] is intimately related to the well-founded semantics. Let  $P$  be a normal program, and let  $M \subseteq B_P$  be a set of atoms. Then we define  $P/M$  to be the (ground) program consisting of all clauses  $A \leftarrow A_1, \dots, A_n$  for which there is a clause  $A \leftarrow A_1, \dots, A_n, \neg B_1, \dots, \neg B_m$  in  $\mathbf{ground}(P)$  with  $B_1, \dots, B_m \notin M$ . Since  $P/M$  does no longer contain negation, it has a least two-valued model  $T_{P/M}^+ \uparrow \omega$ . For any two-valued interpretation  $I$  we can therefore define the operator  $\mathbf{GL}_P(I) = T_{P/I}^+ \uparrow \omega$ , and call  $M$  a *stable model* of the normal program  $P$  if it is a fixed point of the operator  $\mathbf{GL}_P$ , i.e. if  $M = \mathbf{GL}_P(M) = T_{P/M}^+ \uparrow \omega$ . As it turns out, the operator  $\mathbf{GL}_P$  is in general not monotonic for normal programs  $P$ . However it is *antitonic*, i.e. whenever  $I \subseteq J \subseteq B_P$  then  $\mathbf{GL}_P(J) \subseteq \mathbf{GL}_P(I)$ . As a consequence, the operator  $\mathbf{GL}_P^2$ , obtained by applying  $\mathbf{GL}_P$  twice, is monotonic, and hence has a least fixed point  $L_P$  and a greatest fixed point  $G_P$ . In [25] it was shown that  $\mathbf{GL}_P(L_P) = G_P$ ,  $L_P = \mathbf{GL}_P(G_P)$ , and that  $L_P \cup \neg(B_P \setminus G_P)$  coincides with the well-founded model of  $P$ . This is called the *alternating fixed point characterization* of the well-founded semantics.

## Some Results

The following is a straightforward result which has, to the best of our knowledge, first been formally reported in [15], where a proof can be found.

**Theorem 1.** *Let  $P$  be a definite program. Then there is a unique two-valued model  $M$  of  $P$  for which there exists a (total) level mapping  $l : B_P \rightarrow \alpha$  such that for each atom  $A \in M$  there exists a clause  $A \leftarrow A_1, \dots, A_n$  in  $\mathbf{ground}(P)$  with  $A_i \in M$  and  $l(A) > l(A_i)$  for all  $i = 1, \dots, n$ . Furthermore,  $M$  is the least two-valued model of  $P$ .*

The following result is due to [7], and is striking in its similarity to Theorem 1.

**Theorem 2.** *Let  $P$  be normal. Then a two-valued model  $M \subseteq B_P$  of  $P$  is a stable model of  $P$  if and only if there exists a (total) level*

mapping  $l : B_P \rightarrow \alpha$  such that for each  $A \in M$  there exists  $A \leftarrow A_1, \dots, A_n \neg B_1, \dots, \neg B_m$  in  $\text{ground}(P)$  with  $A_i \in M$ ,  $B_j \notin M$ , and  $l(A) > l(A_i)$  for all  $i = 1, \dots, n$  and  $j = 1, \dots, m$ .

We next recall the following alternative characterization of the Fitting model, due to [14, 15].

**Definition 1.** Let  $P$  be a normal logic program,  $I$  be a model of  $P$ , and  $l$  be an  $I$ -partial level mapping for  $P$ . We say that  $P$  satisfies (F) with respect to  $I$  and  $l$ , if each  $A \in \text{dom}(l)$  satisfies one of the following conditions.

- (Fi)  $A \in I$  and there exists a clause  $A \leftarrow L_1, \dots, L_n$  in  $\text{ground}(P)$  such that  $L_i \in I$  and  $l(A) > l(L_i)$  for all  $i$ .
- (Fii)  $\neg A \in I$  and for each clause  $A \leftarrow L_1, \dots, L_n$  in  $\text{ground}(P)$  there exists  $i$  with  $\neg L_i \in I$  and  $l(A) > l(L_i)$ .

**Theorem 3.** Let  $P$  be a normal logic program with Fitting model  $M$ . Then  $M$  is the greatest model among all models  $I$ , for which there exists an  $I$ -partial level mapping  $l$  for  $P$  such that  $P$  satisfies (F) with respect to  $I$  and  $l$ .

Let us recall next the definition of a (locally) stratified program, due to [2, 22]: A normal logic program is called *locally stratified* if there exists a (total) level mapping  $l : B_P \rightarrow \alpha$ , for some ordinal  $\alpha$ , such that for each clause  $A \leftarrow A_1, \dots, A_n, \neg B_1, \dots, \neg B_m$  in  $\text{ground}(P)$  we have that  $l(A) \geq l(A_i)$  and  $l(A) > l(B_j)$  for all  $i = 1, \dots, n$  and  $j = 1, \dots, m$ . The notion of (locally) stratified program was developed with the idea of preventing *recursion through negation*, while allowing recursion through positive dependencies. (Locally) stratified programs have total well-founded models.

There exist locally stratified programs which do not have a total Fitting semantics and vice versa — just consider the programs consisting of the single clauses  $p \leftarrow p$ , respectively,  $p \leftarrow \neg p, q$ . In fact, condition (Fii) requires a strict decrease of level between the head and a literal in the rule, independent of this literal being positive or negative. But, on the other hand, condition (Fii) imposes no further restrictions on the remaining body literals, while the notion of local stratification does. These considerations motivate the substitution of condition (Fii) by the condition (Cii), as done for the following definition.

**Definition 2.** Let  $P$  be a normal logic program,  $I$  be a model of  $P$ , and  $l$  be an  $I$ -partial level mapping for  $P$ . We say that  $P$  satisfies (WF) with



respect to  $I$  and  $l$ , if each  $A \in \text{dom}(l)$  satisfies (Fi) or the following condition.

(Cii)  $\neg A \in I$  and for each clause  $A \leftarrow A_1, \dots, A_n, \neg B_1, \dots, \neg B_m$  contained in  $\text{ground}(P)$  (at least) one of the following conditions holds:

(Ciiia) There exists  $i \in \{1, \dots, n\}$  with  $\neg A_i \in I$  and  $l(A) \geq l(A_i)$ .

(Ciiib) There exists  $j \in \{1, \dots, m\}$  with  $B_j \in I$  and  $l(A) > l(B_j)$ .

So, in the light of Theorem 3, Definition 2 should provide a natural “stratified version” of the Fitting semantics. And indeed it does, and furthermore, the resulting semantics coincides with the well-founded semantics, which is a very satisfactory result from [14, 15].

**Theorem 4.** *Let  $P$  be a normal logic program with well-founded model  $M$ . Then  $M$  is the greatest model among all models  $I$ , for which there exists an  $I$ -partial level mapping  $l$  for  $P$  such that  $P$  satisfies (WF) with respect to  $I$  and  $l$ .*

For completeness, we remark that an alternative characterization of the weakly perfect model semantics [21] can also be found in [14, 15].

The approach which led to the results just mentioned, originally put forward in [14, 15], provides a general methodology for obtaining uniform characterizations of different semantics for logic programs.

### 3 Maximally Circular Stable Semantics

We note that condition (Fi) has been reused in Definition 2. Thus, Definition 1 has been “stratified” only with respect to condition (Fii), yielding (Cii), but not with respect to (Fi). Indeed, also replacing (Fi) by a stratified version such as the following seems not satisfactory at first sight.

(Ci)  $A \in I$  and there exists a clause  $A \leftarrow A_1, \dots, A_n, \neg B_1, \dots, \neg B_m$  in  $\text{ground}(P)$  such that  $A_i, \neg B_j \in I$ ,  $l(A) \geq l(A_i)$ , and  $l(A) > l(B_j)$  for all  $i$  and  $j$ .

If we replace condition (Fi) by condition (Ci) in Definition 2, then it is not guaranteed that for any given program there is a greatest model satisfying the desired properties, as the following example from [14, 15] shows.

*Example 1.* Consider the program consisting of the two clauses  $p \leftarrow p$  and  $q \leftarrow \neg p$ , and the two (total) models  $M_1 = \{p, \neg q\}$  and  $M_2 = \{\neg p, q\}$ , which are incomparable, and the level mapping  $l$  with  $l(p) = 0$  and  $l(q) = 1$ .

In order to arrive at an understanding of this asymmetry, we consider the setting with conditions (Ci) and (Fii), which is somehow “dual” to the well-founded semantics which is characterized by (Fi) and (Cii).

**Definition 3.** *Let  $P$  be a normal logic program,  $I$  be a model of  $P$ , and  $l$  be an  $I$ -partial level mapping for  $P$ . We say that  $P$  satisfies (CW) with respect to  $I$  and  $l$ , if each  $A \in \text{dom}(l)$  satisfies (Ci) or (Fii).*

By virtue of Definition 3 we will be able to develop a theory which complements the results from Section 2. We will first characterize the greatest model of a definite program analogously to Theorem 1.

**Theorem 5.** *Let  $P$  be a definite program. Then there is a unique two-valued supported interpretation  $M$  of  $P$  for which there exists a (total) level mapping  $l : B_P \rightarrow \alpha$  such that for each atom  $A \notin M$  and for all clauses  $A \leftarrow A_1, \dots, A_n$  in  $\text{ground}(P)$  there is some  $A_i \notin M$  with  $l(A) > l(A_i)$ . Furthermore,  $M$  is the greatest two-valued model of  $P$ .*

*Proof.* Let  $M$  be the greatest two-valued model of  $P$ , and let  $\alpha$  be the least ordinal such that  $M = T_P^+ \downarrow \alpha$ . Define  $l : B_P \rightarrow \alpha$  by setting  $l(A) = \min\{\gamma \mid A \notin T_P^+ \downarrow (\gamma + 1)\}$  for  $A \notin M$ , and by setting  $l(A) = 0$  if  $A \in M$ . The mapping  $l$  is well-defined because  $A \notin M$  with  $A \notin T_P^+ \downarrow \gamma = \bigcap_{\beta < \gamma} T_P^+ \downarrow \beta$  for some limit ordinal  $\gamma$  implies  $A \notin T_P^+ \downarrow \beta$  for some  $\beta < \gamma$ . So the least ordinal  $\beta$  with  $A \notin T_P^+ \downarrow \beta$  is always a successor ordinal. Now assume that there is  $A \notin M$  which does not satisfy the stated condition. We can furthermore assume without loss of generality that  $A$  is chosen with this property such that  $l(A)$  is minimal. Let  $A \leftarrow A_1, \dots, A_n$  be a clause in  $\text{ground}(P)$ . Since  $A \notin T_P^+ (T_P^+ \downarrow l(A))$  we obtain  $A_i \notin T_P^+ \downarrow l(A) \supseteq M$  for some  $i$ . But then  $l(A_i) < l(A)$  which contradicts minimality of  $l(A)$ .

Conversely, let  $M$  be a two-valued model for  $P$  which satisfies the given condition for some mapping  $l : B_P \rightarrow \alpha$ . We show by transfinite induction on  $l(A)$  that  $A \notin M$  implies  $A \notin T_P^+ \downarrow (l(A) + 1)$ , which suffices because it implies that for the greatest two-valued model  $T_P^+ \downarrow \beta$  of  $P$  we have that  $T_P^+ \downarrow \beta \subseteq M$ , and therefore  $T_P^+ \downarrow \beta = M$ . For the inductive proof consider first the case where  $l(A) = 0$ . Then there is no clause in  $\text{ground}(P)$  with head  $A$  and consequently  $A \notin T_P^+ \downarrow 1 = T_P^+(B_P)$ .

Now assume that the statement to be proven holds for all  $B \notin M$  with  $l(B) < \alpha$ , where  $\alpha$  is some ordinal, and let  $A \notin M$  with  $l(A) = \alpha$ . Then each clause in  $\text{ground}(P)$  with head  $A$  contains an atom  $B$  with  $l(B) = \beta < \alpha$  and  $B \notin M$ . Hence  $B \notin T_P^+ \downarrow (\beta + 1)$  and consequently  $A \notin T_P^+ \downarrow (\alpha + 1)$ .

The following definition and theorem are analogous to Theorem 2.

**Definition 4.** Let  $P$  be normal. Then  $M \subseteq B_P$  is called a maximally circular stable model (maxstable model) of  $P$  if it is a two-valued supported interpretation of  $P$  and there exists a (total) level mapping  $l : B_P \rightarrow \alpha$  such that for each atom  $A \notin M$  and for all clauses  $A \leftarrow A_1, \dots, A_n, \neg B_1, \dots, \neg B_m$  in  $\text{ground}(P)$  with  $B_1, \dots, B_m \notin M$  there is some  $A_i \notin M$  with  $l(A) > l(A_i)$ .

**Theorem 6.**  $M \subseteq B_P$  is a maxstable model of  $P$  if and only if  $M = \text{gfp} \left( T_{P/M}^+ \right)$ .

*Proof.* First note that every maxstable model is a supported model. Indeed supportedness follows immediately from the definition. Now assume that  $M$  is maxstable but is not a model, i.e. there is  $A \notin M$  but there is a clause  $A \leftarrow A_1, \dots, A_n$  in  $\text{ground}(P)$  with  $A_i \in M$  for all  $i$ . But by the definition of maxstable model we must have that there is  $A_i \notin M$ , which contradicts  $A_i \in M$ .

Now let  $M$  be a maxstable model of  $P$ . Let  $A \notin M$  and let  $T_{P/M}^+ \downarrow \alpha = \text{gfp} \left( T_{P/M}^+ \right)$ . We show by transfinite induction on  $l(A)$  that  $A \notin T_{P/M}^+ \downarrow (l(A) + 1)$  and hence  $A \notin T_{P/M}^+ \downarrow \alpha$ . For  $l(A) = 0$  there is no clause with head  $A$  in  $P/M$ , so  $A \notin T_{P/M}^+ \downarrow 1$ . Now let  $l(A) = \beta$  for some ordinal  $\beta$ . By assumption we have that for all clauses  $A \leftarrow A_1, \dots, A_n, \neg B_1, \dots, \neg B_m$  with  $B_1, \dots, B_m \notin M$  there exists  $A_i \notin M$  with  $l(A) > l(A_i)$ , say  $l(A_i) = \gamma < \beta$ . Hence  $A_i \notin T_{P/M}^+ \downarrow (\gamma + 1)$ , and consequently  $A \notin T_{P/M}^+ \downarrow (\beta + 1)$ , which shows that  $\text{gfp} \left( T_{P/M}^+ \right) \subseteq M$ .

So let again  $M$  be a maxstable model of  $P$  and let  $A \notin \text{gfp} \left( T_{P/M}^+ \right) = T_{P/M}^+ \downarrow \alpha$  and  $l(A) = \beta$ . Then for each clause  $A \leftarrow A_1, \dots, A_n$  in  $P/M$  there is  $A_i$  with  $A_i \notin T_{P/M}^+ \downarrow \alpha$  and  $l(A) > l(A_i)$ . Now assume  $A \in M$ . Without loss of generality we can furthermore assume that  $A$  is chosen such that  $l(A) = \beta$  is minimal. Hence  $A_i \notin M$ , and we obtain that for each clause in  $P/M$  with head  $A$  one of the corresponding body atoms is

false in  $M$ . By supportedness of  $M$  this yields  $A \notin M$ , which contradicts our assumption. Hence  $A \notin M$  as desired.

Conversely, let  $M = \mathbf{gfp} \left( T_{P/M}^+ \right)$ . Then as an immediate consequence of Theorem 5 we obtain that  $M$  is maxstable.

## 4 Maximally Circular Well-Founded Semantics

Maxstable models are formally analogous<sup>4</sup> to stable models in that the former are fixed points of the operator  $I \mapsto \mathbf{gfp} \left( T_{P/I}^+ \right)$ , while the latter are fixed points of the operator  $I \mapsto \mathbf{lfp} \left( T_{P/I}^+ \right)$ . Further, in analogy to the alternating fixed point characterization of the well-founded model, we can obtain a corresponding variant of the well-founded semantics, which we will do next. Theorem 6 suggests the definition of the following operator.

**Definition 5.** *Let  $P$  be a normal program and  $I$  be a two-valued interpretation. Then define  $\mathbf{CGL}_P(I) = \mathbf{gfp} \left( T_{P/I}^+ \right)$ .*

Using the operator  $\mathbf{CGL}_P$ , we can define a “maximally circular” version of the alternating fixed-point semantics.

**Proposition 1.** *Let  $P$  be a normal program. Then the following hold.*

- (i)  $\mathbf{CGL}_P$  is antitonic and  $\mathbf{CGL}_P^2$  is monotonic.
- (ii)  $\mathbf{CGL}_P \left( \mathbf{lfp} \left( \mathbf{CGL}_P^2 \right) \right) = \mathbf{gfp} \left( \mathbf{CGL}_P^2 \right)$  and  $\mathbf{CGL}_P \left( \mathbf{gfp} \left( \mathbf{CGL}_P^2 \right) \right) = \mathbf{lfp} \left( \mathbf{CGL}_P^2 \right)$ .

*Proof.* (i) If  $I \subseteq J \in B_P$ , then  $P/J \subseteq P/I$  and consequently  $\mathbf{CGL}_P(J) = \mathbf{gfp} \left( T_{P/J}^+ \right) \subseteq \mathbf{gfp} \left( T_{P/I}^+ \right) = \mathbf{CGL}_P(I)$ . Monotonicity of  $\mathbf{CGL}_P^2$  then follows trivially.

(ii) Let  $L_P = \mathbf{lfp} \left( \mathbf{CGL}_P^2 \right)$  and  $G_P = \mathbf{gfp} \left( \mathbf{CGL}_P^2 \right)$ . Then we can calculate  $\mathbf{CGL}_P^2 \left( \mathbf{CGL}_P(L_P) \right) = \mathbf{CGL}_P \left( \mathbf{CGL}_P^2(L_P) \right) = \mathbf{CGL}_P(L_P)$ , so  $\mathbf{CGL}_P(L_P)$  is a fixed point of  $\mathbf{CGL}_P^2$ , and hence  $L_P \subseteq \mathbf{CGL}_P(L_P) \subseteq G_P$ . Similarly,  $L_P \subseteq \mathbf{CGL}_P(G_P) \subseteq G_P$ . Since  $L_P \subseteq G_P$  we get from the antitonicity of  $\mathbf{CGL}_P$  that  $L_P \subseteq \mathbf{CGL}_P(G_P) \subseteq \mathbf{CGL}_P(L_P) \subseteq G_P$ . Similarly, since  $\mathbf{CGL}_P(L_P) \subseteq G_P$ , we obtain  $\mathbf{CGL}_P(G_P) \subseteq \mathbf{CGL}_P^2(L_P) = L_P \subseteq \mathbf{CGL}_P(G_P)$ , so  $\mathbf{CGL}_P(G_P) = L_P$ , and also  $G_P = \mathbf{CGL}_P^2(G_P) = \mathbf{CGL}_P(L_P)$ .

<sup>4</sup> The term *dual* seems not to be entirely adequate in this situation, although it is intuitively appealing.

We will now define an operator for the maximally circular well-founded semantics. Given a normal logic program  $P$  and some  $I \in I_P$ , we say that  $S \subseteq B_P$  is a *self-founded set (of  $P$ ) with respect to  $I$*  if  $S \cup I \in I_P$  and each atom  $A \in S$  satisfies the following condition: There exists a clause  $A \leftarrow \text{body}$  in  $\text{ground}(P)$  such that one of the following holds.

- (Si)  $\text{body}$  is true in  $I$ .
- (Sii) Some (non-negated) atoms in  $\text{body}$  occur in  $S$  and all other literals in  $\text{body}$  are true in  $I$ .

Self-founded sets are analogous<sup>5</sup> to unfounded sets, and the following proposition holds.

**Proposition 2.** *Let  $P$  be a normal program and let  $I \in I_P$ . Then there exists a greatest self-founded set of  $P$  with respect to  $I$ .*

*Proof.* If  $(S_i)_{i \in \mathcal{I}}$  is a family of sets each of which is a self-founded set of  $P$  with respect to  $I$ , then it is easy to see that  $\bigcup_{i \in \mathcal{I}} S_i$  is also a self-founded set of  $P$  with respect to  $I$ .

Given a normal program  $P$  and  $I \in I_P$ , let  $S_P(I)$  be the greatest self-founded set of  $P$  with respect to  $I$ , and define the operator  $\text{CW}_P$  on  $I_P$  by

$$\text{CW}_P(I) = S_P(I) \cup \neg F_P(I).$$

**Proposition 3.** *The operator  $\text{CW}_P$  is well-defined and monotonic.*

*Proof.* For well-definedness, we have to show that  $S_P(I) \cap F_P(I) = \emptyset$  for all  $I \in I_P$ . So assume there is  $A \in S_P(I) \cap F_P(I)$ . From  $A \in F_P(I)$  we obtain that for each clause with head  $A$  there is a corresponding body literal  $L$  which is false in  $I$ . From  $A \in S_P(I)$ , more precisely from (Sii), we can furthermore conclude that  $L$  is an atom and  $L \in S_P(I)$ . But then  $\neg L \in I$  and  $L \in S_P(I)$  which is impossible by definition of self-founded set which requires that  $S_P(I) \cup I \in I_P$ . So  $S_P(I) \cap F_P(I) = \emptyset$  and  $\text{CW}_P$  is well-defined.

For monotonicity, let  $I \subseteq J \in I_P$  and let  $L \in \text{CW}_P(I)$ . If  $L = \neg A$  is a negated atom, then  $A \in F_P(I)$  and all clauses with head  $A$  contain a body literal which is false in  $I$ , hence in  $J$ , and we obtain  $A \in F_P(J)$ . If  $L = A$  is an atom, then  $A \in S_P(I)$  and there exists a clause  $A \leftarrow \text{body}$  in  $\text{ground}(P)$  such that (at least) one of (Si) or (Sii) holds. If (Si) holds, then  $\text{body}$  is true in  $I$ , hence in  $J$ , and  $A \in S_P(J)$ . If (Sii) holds, then some non-negated atoms in  $\text{body}$  occur in  $S$  and all other literals in  $\text{body}$  are true in  $I$ , hence in  $J$ , and we obtain  $A \in S_P(J)$ .

<sup>5</sup> Again, it is not really a duality.

Since Proposition 3 establishes monotonicity of  $CW_P$ , for normal  $P$ , we conclude that this operator has a least fixed point  $\text{lfp}(CW_P)$ .

**Definition 6.** For a normal program  $P$ , we call  $\text{lfp}(CW_P)$  the maximally circular well-founded model (maxwf model) of  $P$ .

The following theorem relates our observations to Definition 3, in perfect analogy to the correspondence between the stable model semantics, Theorem 1, Fages's characterization from Theorem 2, the well-founded semantics, and the alternating fixed point characterization.

**Theorem 7.** Let  $P$  be a normal program and  $M_P = \text{lfp}(CW_P)$  be its maxwf model. Then the following hold.

- (i)  $M_P$  is the greatest model among all models  $I$  of  $P$  such that there is an  $I$ -partial level mapping  $l$  for  $P$  such that  $P$  satisfies (CW) with respect to  $I$  and  $l$ .
- (ii)  $M_P = \text{lfp}(\text{CGL}_P^2) \cup \neg(B_P \setminus \text{gfp}(\text{CGL}_P^2))$ .

*Proof.* (i) Let  $M_P = \text{lfp}(CW_P)$  and define the  $M_P$ -partial level mapping  $l_P$  as follows:  $l_P(A) = \alpha$ , where  $\alpha$  is the least ordinal such that  $A$  is not undefined in  $CW_P \uparrow (\alpha + 1)$ . The proof will be established by showing the following facts: (1)  $P$  satisfies (CW) with respect to  $M_P$  and  $l_P$ . (2) If  $I$  is a model of  $P$  and  $l$  is an  $I$ -partial level mapping such that  $P$  satisfies (CW) with respect to  $I$  and  $l$ , then  $I \subseteq M_P$ .

(1) Let  $A \in \text{dom}(l_P)$  and  $l_P(A) = \alpha$ . We consider two cases.

(Case i) If  $A \in M_P$ , then  $A \in S_P(CW_P \uparrow \alpha)$ , hence there exists a clause  $A \leftarrow \text{body}$  in  $\text{ground}(P)$  such that (Si) or (Sii) holds with respect to  $CW_P \uparrow \alpha$ . If (Si) holds, then all literals in  $\text{body}$  are true in  $CW_P \uparrow \alpha$ , hence have level less than  $l_P(A)$  and (Ci) is satisfied. If (Sii) holds, then some non-negated atoms from  $\text{body}$  occur in  $S_P(CW_P \uparrow \alpha)$ , hence have level less than or equal to  $l_P(A)$ , and all remaining literals in  $\text{body}$  are true in  $CW_P \uparrow \alpha$ , hence have level less than  $l_P(A)$ . Consequently,  $A$  satisfies (Ci) with respect to  $M_P$  and  $l_P$ .

(Case ii) If  $\neg A \in M_P$ , then  $A \in F_P(CW_P \uparrow \alpha)$ , hence for all clauses  $A \leftarrow \text{body}$  in  $\text{ground}(P)$  there exists  $L \in \text{body}$  with  $\neg L \in CW_P \uparrow \alpha$  and  $l_P(L) < \alpha$ , hence  $\neg L \in M_P$ . Consequently,  $A$  satisfies (Fii) with respect to  $M_P$  and  $l_P$ , and we have established that fact (1) holds.

(2) We show via transfinite induction on  $\alpha = l(A)$ , that whenever  $A \in I$  (respectively,  $\neg A \in I$ ), then  $A \in CW_P \uparrow (\alpha + 1)$  (respectively,  $\neg A \in CW_P \uparrow (\alpha + 1)$ ). For the base case, note that if  $l(A) = 0$ , then  $\neg A \in I$  implies that there is no clause with head  $A$  in  $\text{ground}(P)$ , hence

$\neg A \in \text{CW}_P \uparrow 1$ . If  $A \in I$  then consider the set  $S$  of all atoms  $B$  with  $l(B) = 0$  and  $B \in I$ . We show that  $S$  is a self-founded set of  $P$  with respect to  $\text{CW}_P \uparrow 0 = \emptyset$ , and this suffices since it implies  $A \in \text{CW}_P \uparrow 1$  by the fact that  $A \in S$ . So let  $C \in S$ . Then  $C \in I$  and  $C$  satisfies condition (Ci) with respect to  $I$  and  $l$ , and since  $l(C) = 0$ , we have that there is a definite clause with head  $C$  whose body atoms (if it has any) are all of level 0 and contained in  $I$ . Hence condition (Sii) (or (Si)) is satisfied for this clause and  $S$  is a self-founded set of  $P$  with respect to  $I$ . So assume now that the induction hypothesis holds for all  $B \in B_P$  with  $l(B) < \alpha$ , and let  $A$  be such that  $l(A) = \alpha$ . We consider two cases.

(Case i) If  $A \in I$ , consider the set  $S$  of all atoms  $B$  with  $l(B) = \alpha$  and  $B \in I$ . We show that  $S$  is a self-founded set of  $P$  with respect to  $\text{CW}_P \uparrow \alpha$ , and this suffices since it implies  $A \in \text{CW}_P \uparrow (\alpha + 1)$  by the fact that  $A \in S$ . First note that  $S \subseteq I$ , so  $S \cup I \in I_P$ . Now let  $C \in S$ . Then  $C \in I$  and  $C$  satisfies condition (Ci) with respect to  $I$  and  $l$ , so there is a clause  $A \leftarrow A_1, \dots, A_n, \neg B_1, \dots, \neg B_m$  in  $\text{ground}(P)$  such that  $A_i, \neg B_j \in I$ ,  $l(A) \geq l(A_i)$ , and  $l(A) > l(B_j)$  for all  $i$  and  $j$ . By induction hypothesis we obtain  $\neg B_j \in \text{CW}_P \uparrow \alpha$ . If  $l(A_i) < l(A)$  for some  $A_i$  then we have  $A_i \in \text{CW}_P \uparrow \alpha$ , also by induction hypothesis. If there is no  $A_i$  with  $l(A_i) = l(A)$ , then (Si) holds, while  $l(A_i) = l(A)$  implies  $A_i \in S$ , so (Sii) holds.

(Case ii) If  $\neg A \in I$ , then  $A$  satisfies (Fii) with respect to  $I$  and  $l$ . Hence for all clauses  $A \leftarrow \text{body}$  in  $\text{ground}(P)$  we have that there is  $L \in \text{body}$  with  $\neg L \in I$  and  $l(L) < \alpha$ . Hence for all these  $L$  we have  $\neg L \in \text{CW}_P \uparrow \alpha$  by induction hypothesis, and consequently for all clauses  $A \leftarrow \text{body}$  in  $\text{ground}(P)$  we obtain that  $\text{body}$  is false in  $\text{CW}_P \uparrow \alpha$  which yields  $\neg A \in \text{CW}_P \uparrow (\alpha + 1)$ . This establishes fact (2) and concludes the proof of (i).

(ii) We first introduce some notation. Let

$$\begin{aligned} L_0 &= \emptyset, & G_0 &= B_P, \\ L_{\alpha+1} &= \text{CGL}_P(G_\alpha), & G_{\alpha+1} &= \text{CGL}_P(L_\alpha) \quad \text{for any ordinal } \alpha, \\ L_\alpha &= \bigcup_{\beta < \alpha} L_\beta, & G_\alpha &= \bigcap_{\beta < \alpha} G_\beta \quad \text{for limit ordinal } \alpha, \\ L_P &= \text{lfp}(\text{CGL}_P^2), & G_P &= \text{gfp}(\text{CGL}_P^2). \end{aligned}$$

By transfinite induction, it is easily checked that  $L_\alpha \subseteq L_\beta \subseteq G_\beta \subseteq G_\alpha$  whenever  $\alpha \leq \beta$ . So  $L_P = \bigcup L_\alpha$  and  $G_P = \bigcap G_\alpha$ .

Let  $M = L_P \cup \neg(B_P \setminus G_P)$ . We intend to apply (i) and first define an  $M$ -partial level mapping  $l$ . We will take as image set of  $l$ , pairs  $(\alpha, \gamma)$  of ordinals, with the lexicographic ordering. This can be done without loss

of generality since any set of such pairs, under the lexicographic ordering, is well-ordered, and therefore order-isomorphic to an ordinal. For  $A \in L_P$ , let  $l(A)$  be the pair  $(\alpha, 0)$ , where  $\alpha$  is the least ordinal such that  $A \in L_{\alpha+1}$ . For  $B \notin G_P$ , let  $l(B)$  be the pair  $(\beta, \gamma)$ , where  $\beta$  is the least ordinal such that  $B \notin G_{\beta+1}$ , and  $\gamma$  is least such that  $B \notin T_{P/L_\beta} \downarrow \gamma$ . It is easily shown that  $l$  is well-defined, and we show next by transfinite induction that  $P$  satisfies (CW) with respect to  $M$  and  $l$ .

Let  $A \in L_1 = \mathbf{gfp} \left( T_{P/B_P}^+ \right)$ . Since  $P/B_P$  contains exactly all clauses from  $\mathbf{ground}(P)$  which contain no negation, we have that  $A$  is contained in the greatest two-valued model of a definite subprogram of  $P$ , namely  $P/B_P$ . So there must be a definite clause in  $\mathbf{ground}(P)$  with head  $A$  whose corresponding body atoms are also true in  $L_1$ , which, by definition of  $l$ , must have the same level as  $A$ , hence (Ci) is satisfied. Now let  $\neg B \in \neg(B_P \setminus G_P)$  such that  $B \in (B_P \setminus G_1) = B_P \setminus \mathbf{gfp} \left( T_{P/\emptyset}^+ \right)$ . Since  $P/\emptyset$  contains all clauses from  $\mathbf{ground}(P)$  with all negative literals removed, we obtain that  $B$  is not contained in the greatest two-valued model of the definite program  $P/\emptyset$ , and (Fii) is satisfied by Theorem 5 using a simple induction argument.

Assume now that, for some ordinal  $\alpha$ , we have shown that  $A$  satisfies (CW) with respect to  $M$  and  $l$  for all  $A \in B_P$  with  $l(A) < (\alpha, 0)$ .

Let  $A \in L_{\alpha+1} \setminus L_\alpha = \mathbf{gfp} \left( T_{P/G_\alpha}^+ \right) \setminus L_\alpha$ . Then  $A \in \left( T_{P/G_\alpha}^+ \downarrow \gamma \right) \setminus L_\alpha$  for some  $\gamma$ ; note that all (negative) literals which were removed by the Gelfond-Lifschitz transformation from clauses with head  $A$  have level less than  $(\alpha, 0)$ . Then  $A$  satisfies (Ci) with respect to  $M$  and  $l$  by definition of  $l$ .

Let  $A \in (B_P \setminus G_{\alpha+1}) \cap G_\alpha$ . Then  $A \notin \mathbf{gfp} \left( T_{P/L_\alpha}^+ \right)$  and we conclude again from Theorem 5, using a simple induction argument, that  $A$  satisfies (CW) with respect to  $M$  and  $l$ .

This finishes the proof that  $P$  satisfies (CW) with respect to  $M$  and  $l$ . It remains to show that  $M$  is greatest with this property.

So assume that  $M_1 \supset M$  is the greatest model such that  $P$  satisfies (CW) with respect to  $M_1$  and some  $M_1$ -partial level mapping  $l_1$ . Assume  $L \in M_1 \setminus M$  and, without loss of generality, let the literal  $L$  be chosen such that  $l_1(L)$  is minimal. We consider two cases.

(Case i) If  $L = \neg A \in M_1 \setminus M$  is a negated atom, then by (Fii) for each clause  $A \leftarrow L_1, \dots, L_n$  in  $\mathbf{ground}(P)$  there exists  $i$  with  $\neg L_i \in M_1$  and  $l_1(A) > l_1(L_i)$ . Hence,  $\neg L_i \in M$  and consequently for each clause  $A \leftarrow \mathbf{body}$  in  $P/L_P$  we have that some atom in  $\mathbf{body}$  is false in  $M =$



$L_P \cup \neg(B_P \setminus G_P)$ . But then  $A \notin \text{CGL}_P(L_P) = G_P$ , hence  $\neg A \in M$ , contradicting  $\neg A \in M_1 \setminus M$ .

(Case ii) If  $L = A \in M_1 \setminus M$  is an atom, then  $A \notin M = L_P \cup \neg(B_P \setminus G_P)$  and in particular  $A \notin L_P = \text{gfp}(T_{P/G_P}^+)$ . Hence  $A \notin T_{P/G_P}^+ \downarrow \gamma$  for some  $\gamma$ , which can be chosen to be least with this property. We show by induction on  $\gamma$  that this leads to a contradiction, to finish the proof.

If  $\gamma = 1$ , then there is no clause with head  $A$  in  $P/G_P$ , i.e. for all clauses  $A \leftarrow \text{body}$  in  $\text{ground}(P)$  we have that  $\text{body}$  is false in  $M$ , hence in  $M_1$ , which contradicts  $A \in M_1$ .

Now assume that there is no  $B \in M_1 \setminus M$  with  $B \notin T_{P/G_P}^+ \downarrow \delta$  for any  $\delta < \gamma$ , and let  $A \in M_1 \setminus M$  with  $A \notin T_{P/G_P}^+ \downarrow \gamma$ , which implies that  $\gamma$  is a successor ordinal. By  $A \in M_1$  and (Ci) there must be a clause  $A \leftarrow A_1, \dots, A_n \neg B_1, \dots, \neg B_m$  in  $\text{ground}(P)$  with  $A_i, \neg B_j \in M_1$  for all  $i$  and  $j$ . However, since  $A \notin T_{P/G_P}^+ \downarrow \gamma$  we obtain that for each  $A \leftarrow A_1, \dots, A_n$  in  $P/G_P$ , hence for each  $A \leftarrow A_1, \dots, A_n, \neg B_1, \dots, \neg B_m$  in  $\text{ground}(P)$  with  $\neg B_1, \dots, \neg B_m \in \neg(B_P \setminus G_P) \subseteq M \subseteq M_1$  there is  $A_i$  with  $A_i \notin T_{P/G_P}^+ \downarrow (\gamma - 1) \subseteq M$ , and by induction hypothesis we obtain  $A_i \notin M_1$ . So  $A_i \in M_1$  and  $A_i \notin M_1$ , which is a contradiction and concludes the proof.

## 5 Related Work

As the purpose of our paper is to present a coherent unified picture of different semantics, it is related to the large body of work on relating semantics and uniform frameworks for semantics of logic programs. For a subjective selection of the probably most prominent approaches we refer to the introduction of this paper and also to the extensive discussions in [15], where the level-mapping approach was introduced and put into perspective. Two very recent developments, however, appear to be very closely related to our approach, and we discuss them shortly. They were both developed independently of our work, and brought to our attention while this paper was being reviewed.

Loyer, Spyrtatos and Stamate, in [19], presented a parametrized approach to different semantics. It allows to substitute the preference for falsehood by preference for truth in the stable and well-founded semantics, but uses entirely different means than presented here. Its purpose is also different — while we focus on the strenghtening of the mathematical foundations for the field, the work in [19] is motivated by the need to deal with open vs. closed world assumption in some application settings. The

exact relationship between their approach and ours remains to be worked out.

Denecker, Bruynooghe, Marek, and Ternovska, in [3, 5], unified different logic programming semantics by identifying them as transfinite inductive definitions. As the latter can also be analysed using fixed-point computations via semantic operators, our level-mapping proof schema as described in [15] should be applicable to this inductive perspective as well. We believe that our approach provides more flexibility and can be more readily extended to other syntactic and semantic features, but further work will be needed to substantiate this. On the other hand, the inductive approach appears to be more intuitively appealing at first sight, and of more general explanatory value.

## 6 Conclusions and Further Work

We have displayed a coherent picture of different semantics for normal logic programs. We have added to well-known results new ones which complete the formerly incomplete picture of relationships. The richness of theory and relationships turns out to be very appealing and satisfactory.

As noted already in the introduction, we did not intend to provide new semantics for practical purposes. We rather wanted to focus on the deepening of the theoretical insights into the relations between different semantics, by painting a coherent and complete picture of their dependencies and interconnections. Nevertheless, our new semantics stands well in the tradition of the original motivation of non-monotonic reasoning research: our semantics is defined by making a selection of the (classical) models of a program, understood as first-order logical formulae. We do not claim that the this line of motivation necessarily carries much further — as repeatedly stated, our purpose is formal, and foundational.

From a mathematical perspective one expects major notions in a field to be strongly and cleanly interconnected, and it is fair to say that this is the case for declarative semantics for normal logic programs, as our exhibition shows. We would also like to stress that the results presented in this paper are far from straightforward. Intuitively, replacing least fixed points by greatest fixed points appears to be unproblematic, but this is only true on the general conceptual level, and far from obvious, or easy to achieve, formally. The details of the constructions and proofs are indeed involved and not incremental, which is particularly obvious by the proof details for Theorem 7. The fact that a symmetric picture such as the one presented here can be established at all is strongly supportive of

the position that major established notions in logic programming are not only intuitively appealing — this is well-known as intuition was one of the driving forces in the field — but also formally satisfactory.

For normal logic programs, we have obtained a uniform perspective on different semantics. The situation becomes much more difficult when discussing extensions of the logic programming paradigm like disjunctive [27], quantitative [20], or dynamic [17] logic programming. For many of these extensions it is as yet to be determined what the best ways of providing declarative semantics for these frameworks are, and the lack of interconnections between the different proposals in the literature provides an argument for the case that no satisfactory answers have yet been found.

We believe that successful proposals for extensions will have to exhibit similar interrelationships as observed for normal programs. How, and if, this can be achieved, however, is as yet rather uncertain. Formal studies like the one in this paper may help in designing satisfactory semantics, but a discussion of this is outside the scope of our exhibition, and will be pursued elsewhere.

## References

1. Samson Abramsky and Achim Jung. Domain theory. In Samson Abramsky, Dov Gabbay, and Thomas S.E. Maibaum, editors, *Handbook of Logic in Computer Science*, volume 3. Clarendon, Oxford, 1994.
2. Krzysztof R. Apt, Howard A. Blair, and Adrian Walker. Towards a theory of declarative knowledge. In Jack Minker, editor, *Foundations of Deductive Databases and Logic Programming*, pages 89–148. Morgan Kaufmann, Los Altos, CA, 1988.
3. Marc Denecker, Maurice Bruynooghe, and V. Wiktor Marek. Logic programming revisited: logic programs as inductive definitions. *ACM Transactions on Computational Logic*, 2(4):623–654, 2001.
4. Marc Denecker, V. Wiktor Marek, and Mirosław Truszczyński. Approximating operators, stable operators, well-founded fixpoints and applications in non-monotonic reasoning. In Jack Minker, editor, *Logic-based Artificial Intelligence*, chapter 6, pages 127–144. Kluwer Academic Publishers, Boston, 2000.
5. Marc Denecker and Evgenia Ternovska. A logic of non-monotone inductive definitions and its modularity properties. In Vladimir Lifschitz and Ilkka Niemelä, editors, *Logic Programming and Nonmonotonic Reasoning: 7th International Conference*, volume 2923 of *Lecture Notes in Computer Science*, pages 47–60. Springer, 2004.
6. Phan Minh Dung and Kanchana Kanchanasut. A fixpoint approach to declarative semantics of logic programs. In Ewing L. Lusk and Ross A. Overbeek, editors, *Logic Programming, Proceedings of the North American Conference 1989, NACL P'89, Cleveland, Ohio*, pages 604–625. MIT Press, 1989.
7. François Fages. Consistency of Clark's completion and existence of stable models. *Journal of Methods of Logic in Computer Science*, 1:51–60, 1994.
8. Melvin Fitting. A Kripke-Kleene-semantics for general logic programs. *The Journal of Logic Programming*, 2:295–312, 1985.

9. Melvin Fitting. Bilattices and the semantics of logic programming. *The Journal of Logic Programming*, 11:91–116, 1991.
10. Melvin Fitting. Fixpoint semantics for logic programming — A survey. *Theoretical Computer Science*, 278(1–2):25–51, 2002.
11. Michael Gelfond and Vladimir Lifschitz. The stable model semantics for logic programming. In Robert A. Kowalski and Kenneth A. Bowen, editors, *Logic Programming. Proceedings of the 5th International Conference and Symposium on Logic Programming*, pages 1070–1080. MIT Press, 1988.
12. Pascal Hitzler. Circular belief in logic programming semantics. Technical Report WV-02-13, Knowledge Representation and Reasoning Group, Artificial Intelligence Institute, Department of Computer Science, Dresden University of Technology, Dresden, Germany, 2002.
13. Pascal Hitzler. Towards a systematic account of different logic programming semantics. In Andreas Günter, Rudolf Kruse, and Bernd Neumann, editors, *Proceedings of the 26th German Conference on Artificial Intelligence, KI2003, Hamburg, September 2003*, volume 2821 of *Lecture Notes in Artificial Intelligence*, pages 355–369. Springer, Berlin, 2003.
14. Pascal Hitzler and Matthias Wendt. The well-founded semantics is a stratified Fitting semantics. In Matthias Jarke, Jana Koehler, and Gerhard Lakemeyer, editors, *Proceedings of the 25th Annual German Conference on Artificial Intelligence, KI2002, Aachen, Germany, September 2002*, volume 2479 of *Lecture Notes in Artificial Intelligence*, pages 205–221. Springer, Berlin, 2002.
15. Pascal Hitzler and Matthias Wendt. A uniform approach to logic programming semantics. *Theory and Practice of Logic Programming*, 5(1–2):123–159, 2005.
16. Jacek Jachymski. Order-theoretic aspects of metric fixed-point theory. In William A. Kirk and Brailey Sims, editors, *Handbook of Metric Fixed Point Theory*, pages 613–641. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2001.
17. Joao A. Leite. *Evolving Knowledge Bases*, volume 81 of *Frontiers of Artificial Intelligence and Applications*. IOS Press, 2003.
18. John W. Lloyd. *Foundations of Logic Programming*. Springer, Berlin, 1988.
19. Yann Loyer, Nicolas Spyratos, and Daniel Stamate. Parametrized semantics of logic programs — a unifying framework. *Theoretical Computer Science*, 308(1–3):429–447, 2003.
20. Cristinel Mateis. Quantitative disjunctive logic programming: Semantics and computation. *AI communications*, 13(4):225–248, 2000.
21. Halina Przymusinska and Teodor C. Przymusinski. Weakly stratified logic programs. *Fundamenta Informaticae*, 13:51–65, 1990.
22. Teodor C. Przymusinski. On the declarative semantics of deductive databases and logic programs. In Jack Minker, editor, *Foundations of Deductive Databases and Logic Programming*, pages 193–216. Morgan Kaufmann, Los Altos, CA, 1988.
23. Teodor C. Przymusinski. Well-founded semantics coincides with three-valued stable semantics. *Fundamenta Informaticae*, 13(4):445–464, 1989.
24. Raymond Reiter. A logic for default reasoning. *Artificial Intelligence*, 13:81–132, 1980.
25. Allen van Gelder. The alternating fixpoint of logic programs with negation. In *Proceedings of the Eighth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, Philadelphia, Pennsylvania*, pages 1–10. ACM Press, 1989.

26. Allen van Gelder, Kenneth A. Ross, and John S. Schlipf. The well-founded semantics for general logic programs. *Journal of the ACM*, 38(3):620–650, 1991.
27. Kewen Wang. A comparative study of well-founded semantics for disjunctive logic programs. In Thomas Eiter, Wolfgang Faber, and Mirosław Truszczyński, editors, *Logic Programming and Nonmonotonic Reasoning, 6th International Conference, LPNMR 2001, Vienna, Austria, September 17-19, 2001, Proceedings*, volume 2173 of *Lecture Notes in Artificial Intelligence*, pages 133–146. Springer, 2001.
28. Matthias Wendt. Unfolding the well-founded semantics. *Journal of Electrical Engineering, Slovak Academy of Sciences*, 53(12/s):56–59, 2002. (Proceedings of the 4th Slovakian Student Conference in Applied Mathematics, Bratislava, April 2002).