A New Fixpoint Semantics for General Logic Programs Compared with the Well-Founded and the Stable Model Semantics

François Fages
LCR Thomson-CSF
and
LIENS, URA 1327 du CNRS,
45 rue d'Ulm
75005 Paris
fages@dmi.ens.fr

Abstract We study a new fixpoint semantics for logic programs with negation. Our construction is intermediate between Van Gelder's well-founded model and Gelfond and Lifschitz's stable model semantics. We show first that the stable models of a logic program P are exactly the well-supported models of P, i.e. the supported models with loop-free finite justifications. Then we associate to any logic program P a non-monotonic operator over the semilattice of justified interpretations, and we define in an abstract form its ordinal powers. We show that the fixpoints of this operator are the stable models of P, and that its ordinal powers after some ordinal α are extensions of the well-founded partial model of P. In particular if P has a well-founded model then that canonical model is also an ordinal power and the unique fixpoint of our operator. We show with examples of logic programs which have a unique stable model but no well-founded model that the converse is false. We relate also our work to Doyle´ s truth maintenance system and some implementations of rule-based expert systems.

1 Introduction

Recent results in foundations of logic programming and deductive databases have greatly clarified the relationship between the procedural semantics of logic programs and their declarative semantics in mathematical logic. The theories of "declarative knowledge" which have been developed in this framework [37] [5] [2] [23] [1] [38] [30] [18] [39] appear to be closely related to other theories of non-monotonic reasoning developed in AI [25] [32] [7] [26] [29] [9]. The reason of this convergence is that logic programs do not rely on logical negation, but use instead a non-monotonic operator called negation by failure or negation by default.

The meaning of a logic program, its *declarative semantics*, can be defined in the framework of first-order classical logic in two different ways. On the one hand the program can be viewed as a notation for a first-order formula obtained by some transformations. The semantics of the program is then identified with the class of its models. This approach is examplified by the Clark's completion semantics in which the rules are seen as definitions of predicates using the equivalence symbol instead of the implication. On the other hand one can see each rule of the program as a universally quantified implication, but identify the logical semantics of the program with some *canonical model* instead of all models. In this paper we follow the second approach. We study a new semantics for logic programs which is intermediate between the well-founded semantics of [39] and the stable model semantics of [18], and show its relationship with truth maintenance systems. We recall here the basic definitions.

Definition. The Herbrand base B_H denotes the set of ground atoms. A logic program is a set of rules of the form

$$L_1, ..., L_n \rightarrow A$$

where A is an atom, called the *conclusion* and denoted by $concl(R)$, and the L_i's are literals (i.e. atoms or negated atoms), called the *premises* and denoted by $prem(R)$. We denote the subset of positive atoms in $prem(R)$ by $pos(R)$, and the set of atoms under a negation by $neg(R)$. The set of ground instances of a logic program P is denoted by $Ground(P)$.

The most general notion of canonical model for a logic program or its Clark's completion, is the one of stable model [18].

Definition. A Herbrand interpretation I of a logic program P is a *stable model* of P iff $I = M_{H(P,I)}$ where $M_{H(P,I)}$ denotes the least Herbrand model [37] of the pure Horn program $H(P,I)$ defined by the stability transformation:

$$H(P,I) = \{pos(R) \rightarrow concl(R) \mid R \in Ground(P) \ \wedge \ neg(R) \cap I = \emptyset\}$$

This transformation eliminates the rules of P which have inconsistent negative premises w.r.t. I, and eliminates the negative premises in the remaining rules. A model is stable if it derives itself by this transformation. A program P is said to be *well-behaved* if it has a unique stable model.

Any stable model of a logic program P is a minimal Herbrand model of P [18]. The stable models of P coincide with the default models of P in Reiter's default theory [27]. For well-behaved programs however the unique stable model lacks a constructive definition. Significantly more constructive, but still not effective, is the well-founded model semantics introduced in [39].

Definition. A *partial interpretation* is a couple of disjoint sets of atoms (T, F), atoms in T are interpreted as *true*, atoms in F are interpreted as $false$, and the other atoms in $B_H \backslash (T \cup F)$ are left uninterpreted.

The *well-founded partial model* of a logic program P is defined as the least fixpoint [24] of the monotonic operator V_P over the complete semilattice of partial interpretations defined by:

$$V_P(T, F) = (T_P(T, F), F_P(T, F))$$

where $T_P(T, F) = \{concl(R) \mid R \in Ground(P) \wedge pos(R) \subseteq T \wedge neg(R) \subseteq F\}$, and where $F_P(T, F)$ denotes the *greatest unfounded set* of P w.r.t. (T, F).

A set of ground atoms $S \subseteq B_H$ is said to be an *unfounded set* of P w.r.t. a partial interpretation (T, F) iff for any atom $A \in S$, for any rule instance $R \in Ground(P)$ with $concl(R) = A$, $prem(R)$ is inconsistent with the partial interpretation (T, F) or with S, that is $T \cap neg(R) \neq \emptyset$ or $F \cap pos(R) \neq \emptyset$ or $S \cap pos(R) \neq \emptyset$. Intuitively the atoms of an unfounded set can be interpreted by false. The union of two unfounded sets is again an unfounded set, the greatest unfounded set is thus the union of all unfounded sets of P w.r.t. (T, F). An alternative constructive definition for F_P is given in [29].

We denote by $(T_{P,\alpha}, F_{P,\alpha}) = V_P \uparrow \alpha$ the partial interpretation obtained as the ordinal power α of V_P. The well-founded partial model of P is equal to the closure ordinal power of V_P. When the well-founded partial model $(T_{P,\alpha}, F_{P,\alpha})$ of P is a model, i.e. $B_H = T_{P,\alpha} \cup F_{P,\alpha}$, this model is called the *well-founded model* of P. The well-founded model semantics for logic programs generalizes the iterated least model semantics for *stratified* logic programs [1] [38] [30].

1.1. Theorem [39]. If P is (locally) stratifiable then P has a well-founded model, which is identical to its iterated least model (perfect model).

1.2. Theorem [39]. If a logic program P has a well-founded model M then M is also the unique stable model of P.

The well-founded model semantics is thus compatible with the stable model semantics, but weaker in the sense that a well-behaved program may have no well-founded model (only a partial model). For example

$$\begin{aligned} P_1 = \{&\neg p \to q, \\ &\neg r \to p, \\ &\neg p, \neg q \to r\} \end{aligned}$$

has a unique stable model $\{p\}$ but its well-founded partial model is $(\emptyset, \emptyset)$. Note that $\{p\}$ is also the unique model of the Clark's completion of P_1:

$$comp(P_1) = \{q \iff \neg p,\ p \iff \neg r,\ r \iff \neg p \wedge \neg q\}$$

In this paper we define a fixpoint semantics for logic programs based on a non-monotonic and non-deterministic operator J_P. We show that the fixpoints of J_P are the stable models of P (theorem 3.3), that each fixpoint can be obtained as an ordinal power of J_P^ρ for some strategy ρ (theorem 3.4), and that for any fair strategy ρ the ordinal powers of J_P^ρ after some ordinal α are extensions of the well-founded partial model of P (theorem 4.2). We say a

Herbrand interpretation M is the rational model of a logic program P iff for any fair strategy ρ, M is a fixpoint and an ordinal power of J_P^ρ. Therefore if P has a well-founded model M then M is also a rational model of P. Conversely we show that P_1 is an example of a well-behaved program which has a rational model but no well-founded model.

Although it is not effective our fixpoint semantics has also a strong operational connotation. In the last section of this paper we show that the bottom-up procedure suggested by our construction can be considered as an alternative to top-down SLDNF-resolution procedures augmented with loop-checks [40] [28] [1] [38], when one has to implement the canonical model semantics instead of the weaker semantics of program's completion or of standard SLDNF-resolution [5] [23] [6] [21]. In the last section we relate also our work to the truth maintenance system of [9] and some implementations of rule-based expert systems.

2 Well-Supported Interpretations

Definition. We say a Herbrand interpretation I is *well-supported* iff there exists a strict well-founded partial ordering $\prec$ on I such that for any atom $A \in I$ there exists a rule $R \in Ground(P)$ with $concl(R) = A$, $I \models prem(R)$ and for any $B \in pos(R)$, $B \prec A$.

2.1. Theorem. For a general logic program P, the well-supported models of P are exactly the stable models of P.

Proof. If M is a stable model of P, let $\prec$ be the strict well-founded partial ordering on M defined by $A \prec B$ iff for some integer i, $A \in T_{H(P,M)} \uparrow i$ and $B \in T_{H(P,M)} \uparrow i+1 \backslash T_{H(P,M)} \uparrow i$, where $T_{H(P,M)}$ is the (monotonic) immediate consequence operator associated with the program obtained by the stability transformation. We show that $\prec$ establishes the well-supportedness of M. For any $A \in M$ there is an integer $i > 0$ with $A \in T_{H(P,M)} \uparrow i$, let i be the least such integer. By definition of $T_{H(P,M)}$ there exists a rule $R \in Ground(P)$ with $concl(R) = A$, $neg(R) \cap M = \emptyset$ and $pos(R) \subseteq T_{H(P,M)} \uparrow i-1$. Therefore $M \models prem(R)$ and as $A \notin T_{H(P,M)} \uparrow i-1$ we have $\forall B \in pos(R)\ B \prec A$, that is M is a well-supported model of P.

Conversely we shall use the notion of a *rank* of an element e w.r.t. to a strict well-founded partial ordering $\prec$, which is defined as the sup of the ordinals α such that there exists a chain $e_0 \prec e_1 \prec ... \prec e_\alpha = e$. If M is a well-supported model of P, let $\prec$ be a strict well-founded partial ordering on M that establishes the well-supportedness of M. We show by transfinite induction on δ that for any atom $A \in M$ at rank δ w.r.t. $\prec$, there exists an integer $i > 0$ where $A \in T_{H(P,M)} \uparrow i$. In this way we get $M \subseteq T_{H(P,M)} \uparrow \omega$. This suffices to prove that M is a stable model of P. Indeed since M is a model of P, M is also a model of $H(P,M)$, and since $T_{H(P,M)} \uparrow \omega$ is the least model of $H(P,M)$ we have $T_{H(P,M)} \uparrow \omega \subseteq M$.

1) In the base case, an atom A at rank 0 in M is supported by a rule $R \in \mathit{Ground}(P)$ with $\mathit{pos}(R) = \emptyset$ and $M \models \mathit{neg}(R)$, hence $A \in H(P,M)$ and $A \in T_{H(P,M)} \uparrow 1$.

2) An atom A at a successor or limit ordinal rank δ in M w.r.t. $\prec$ is supported by a rule R such that $A = \mathit{concl}(R)$, $M \models \mathit{prem}(R)$ and $\forall B \in \mathit{pos}(R)\ B \prec A$. We have $(\mathit{pos}(R) \rightarrow A) \in H(P,M)$. By transfinite induction for any $B \in \mathit{pos}(R)$ there exists an integer i_B such that $B \in T_{H(P,M)} \uparrow i_B$. Let i be the sup of the i_B's, as $T_{H(P,M)}$ is monotonic we get $\forall B \in \mathit{pos}(R)\ B \in T_{H(P,M)} \uparrow i$. Hence $A \in T_{H(P,M)} \uparrow i+1$. QED

This result has been shown independently by [12] in the case where P is a propositional program. A propositional program like for example $P_2 = \{p \rightarrow p,\ \ \neg p \rightarrow q\}$ has two supported minimal models, $\{p\}$ and $\{q\}$, which are both models of $\mathit{comp}(P_2)$, but only one well-supported model $\{q\}$ (called a grounded model in [12]) which is also the unique stable model and the iterated least model in the stratified semantics of [1] [38].

In [1] it is proved that the models of $\mathit{comp}(P)$ are exactly the supported models of P, so our characterization of the stable models of P as the well-supported models of P clarifies the difference between the program's completion semantics and the stable model semantics. In particular if all supports for positive atoms are necessarily finite and loop-free, as it is the case in hierarchical programs or more generally in positive-order-consistent programs [13], both semantics coincide: the stable models of P are exactly the Herbrand models of $\mathit{comp}(P)$. We define now the set of justified interpretations on which our fixpoint semantics is based, and show the equivalence with well-supportedness.

Definition. A *justification* is a *finite* set of ground literals. A *justified atom* is a pair A/Γ where A is a ground atom and Γ is a justification *not containing* A.

The set of justified atoms is countable. Any subset S defines an interpretation denoted by $\overline{S}$ which is obtained by forgetting justifications in S. In this paper we shall consider only sets of *uniquely* justified atoms, that is sets S such that any atom in $\overline{S}$ has a unique justification in S.

Definition. Let P be a general logic program. A *justified interpretation* J of P is a set of uniquely justified atoms such that for any $A/\Gamma \in J$ there exists a rule instance $R \in \mathit{Ground}(P)$ such that $\mathit{concl}(R) = A$, $\overline{J} \models \mathit{prem}(R)$ and $\Gamma = \mathit{prem}(R) \cup \bigcup_{i=1}^{m} \Gamma_i$ where $\mathit{pos}(R) = \{A_1, ..., A_m\}$, $A_i/\Gamma_i \in J$ for $1 \leq i \leq m$, and $A \notin \Gamma$. We say that an interpretation I is *justifiable* iff there exists a justified interpretation J such that $\overline{J} = I$.

Justified interpretations are similar to quasi-interpretations defined independently in [10]. Here in a justified interpretation the justification attached to an atom contains exactly the premises of some supporting rule together with the justifications of the positive premises. The restriction to finite and loop-free justifications is motivated by the following lemma:

2.2. Lemma. An interpretation is well-supported if and only if it is justifiable.

Proof. If J is a justified interpretation, it is easy to check that the relation defined on $\overline{J}$ by $A \prec B$ iff $A/\Gamma \in J$, $B/\Delta \in J$ and $A \in \Delta$, proves that $\overline{J}$ is well-supported. By definition of a justified interpretation, for any $A \in \overline{J}$, let $A/\Gamma \in J$, there exists a rule $R \in Ground(P)$ with $concl(R) = A$, $\overline{J} \models prem(R)$, and for any $B \in pos(R)$, $B \in \Gamma$, hence $B \prec A$. Furthermore $\prec$ is transitive by the definition of a justified interpretation, and $\prec$ admits no infinite decreasing chain $A_1 \succ A_2 \succ ...$, since otherwise, let $A_1/\Gamma_1 \in J, A_2/\Gamma_2 \in J, ...$, we would have an infinite chain of set inclusions, $\Gamma_i \supseteq \Gamma_{i+1} \cup \{A_{i+1}\}$, contradicting either the finiteness of the justifications, or the absence of loop $A_i \notin \Gamma_i$.

Conversely let I be a well-supported interpretation and $\prec$ be a strict well-founded partial ordering on I that establishes the well-supportedness of I. Every atom $A \in I$ is supported by a rule $R_A \in Ground(P)$ such that $concl(R_A) = A$, $I \models prem(R_A)$ and for any $B \in pos(R_A)$, $B \prec A$.

Let J be the set of justified atoms defined by

$$J_0 = \{A/neg(R_A) \mid A \text{ is a } \prec\text{-minimal element}\}$$

$$J_{i+1} = J_i \cup \{A/\Gamma \mid pos(R_A) = \{A_1, ..., A_n\},\ A_k/\Gamma_k \in J_i$$
$$\Gamma = prem(R_A) \cup \{\Gamma_k \mid 1 \leq k \leq n\}\ \}$$

$$J = \bigcup_{i \geq 0} J_i$$

It is straightforward to verify that J is a justified interpretation of P and $\overline{J} = I$. QED

2.3. Corollary. A Herbrand model M of a logic program P is a stable model of P if and only if there exists a justified interpretation J of P such that $M = \overline{J}$.

The set of justified interpretations of P, denoted by JI_P, together with set inclusion forms a *complete semilattice*, i.e. a partially ordered set endowed with a least element, such that every non-empty subset has a greatest lower bound *glb*, and every chain (i.e. every totally ordered subset) has a least upper bound *lub*. In the following we study fixpoint equations in this domain.

3 The Non-Monotonic Justification Maintenance Operator J_P

Definition. For a logic program P and an interpretation I we define the *set of satisfied rules* $S_P(I)$ as

$$S_P(I) = \{R \in Ground(P) \mid I \models prem(R)\}$$

and the *conflict set* $C_P(I)$ as

$$C_P(I) = \{R \in S_P(I) \mid concl(R) \notin I\}$$

Definition. Let P be a logic program, $R \in Ground(P)$, $pos(R) = \{A_1, ..., A_m\}$ and $concl(R) = \{A\}$. The application of the *justification maintenance operator associated with* R, denoted by J_R, to a justified interpretation J is defined by:
$J_R(J) = J$ if $R \notin C_P(\overline{J})$, otherwise
$J_R(J) = (J \cup \{A/(prem(R) \cup \bigcup_{i=1}^m \Gamma_i)\}) \backslash \{B/\Gamma \mid \neg A \in \Gamma\}$
where $A_i/\Gamma_i \in J$ for $1 \leq i \leq m$.
It is straightforward to verify that J_R is an operator on JI_P.

Definition. We say a logic program P is *well-formed* iff there is no rule instance $R \in ground(P)$ with $concl(R) \in neg(R)$.

The technical reason to reject rules of the form $...\neg A... \rightarrow A$ lies in the following proposition, and the subsequent identification of the fixpoints of J_P with the stable models of P. If P is not well-formed J_P can have pathological fixpoints which are not stable models of P. This condition is not a real restriction, a rule for which the conclusion is unifiable with one of its negative conditions can be rewritten in two well-formed rules by introducing a new constant or a new function symbol that duplicates the conclusion. For instance a rule like $p(x,y), \neg q(x), r(y) \rightarrow q(y)$ can be rewritten in $p(x,y), \neg q(x), r(y) \rightarrow f1(y)$ and $f1(y) \rightarrow q(y)$. Furthermore our results on the relationship between the ordinal powers of J_P^ρ and the well-founded (partial) model semantics of logic programs are not restricted to well-formed programs.

3.1. Proposition. If P is a well-formed logic program then for any $R \in C_P(\overline{J})$ we have $R \notin C_P(\overline{J_R(J)})$.

Proof. Let $R \in C_P(\overline{J})$. If $concl(R) \in \overline{J_R(J)}$ then $R \notin C_P(\overline{J_R(J)})$. Otherwise by the definition of J_R we have $\neg concl(R) \in prem(R) \cup \bigcup_{i=1}^m \Gamma_i$ where $pos(R) = \{A_1, ..., A_m\}$ and $A_i/\Gamma_i \in J$ for $1 \leq i \leq m$. As P is a well-formed program $\neg concl(R) \notin prem(R)$, so $\neg concl(R) \in \bigcup_{i=1}^m \Gamma_i$. For some i $\neg concl(R) \in \Gamma_i$ so $A_i \notin \overline{J_R(J)}$. Hence we get $R \notin S_P(\overline{J_R(J)})$ thus $R \notin C_P(\overline{J_R(J)})$. QED

Definition. For a logic program P we denote by $J_P : JI_P \rightarrow 2^{JI_P}$ the non-deterministic operator on JI_P defined by:
$J_P(J) = \{J\}$ if $C_P(\overline{J}) = \emptyset$
$J_P(J) = \{J_R(J) \mid R \in C_P(\overline{J})\}$

We say that a justified interpretation J is a fixpoint of J_P if $J_P(J) = \{J\}$. The next proposition shows that the fixpoints of J_P are exactly those interpretations for which the conflict set is empty.

3.2. Proposition. Let J be a justified interpretation of a well-formed logic program P. J is a fixpoint of J_P iff $C_P(\overline{J}) = \emptyset$.

Proof. If $C_P(\overline{J}) = \emptyset$ then by definition $J_P(J) = \{J\}$. Conversely the proof is by contradiction. Let J be a fixpoint of J_P, let us suppose that $C_P(\overline{J}) \neq \emptyset$, let $R \in C_P(\overline{J})$. By 3.1, $R \notin C_P(\overline{J_R(J)})$, that is either $concl(R) \in \overline{J_R(J)}$ or $\overline{J_R(J)} \not\models prem(R)$. In any case $\overline{J} \neq \overline{J_R(J)}$ thus we get the contradiction $J_P(J) \neq \{J\}$. QED

3.3. Theorem. For any well-formed logic program P the fixpoints of J_P coincide with the stable models of P.

Proof. J is a fixpoint of J_P
iff J is a justified interpretation and $C_P(\overline{J}) = \emptyset$ (by 3.2), i.e. J is a justified model of P,
iff $\overline{J}$ is a well-supported model of P (by 2.3),
iff $\overline{J}$ is a stable model of P (by 2.1). QED

Although J_P is not monotonic we shall use the idea of constructing solutions to fixpoint equations by iterating the operator from the least element. For example on $P_1 = \{\neg p \rightarrow q, \quad \neg r \rightarrow p, \quad \neg p, \neg q \rightarrow r\}$ all rule selection strategies lead to the unique stable model of P_1 that is $\{p\}$:

$\emptyset \triangleright \{q/\neg p\} \triangleright \{p/\neg r\}$
$\emptyset \triangleright \{p/\neg r\}$
$\emptyset \triangleright \{r/\neg p, \neg q\} \triangleright \{q/\neg p\} \triangleright \{p/\neg r\}$

In the following we show that each stable model of P can be obtained as an ordinal power of J_P^ρ for some particular strategy ρ, and that if P has a well-founded model M then M is an ordinal power of J_P^ρ for any fair strategy ρ. To this end we define now in an abstract form the ordinal powers of a non-monotonic operator.

Definition. Let T be an arbitrary (non-monotonic) operator over a complete semilattice L. We define the ordinal powers of T as:

$T \uparrow 0 = \bot$
$T \uparrow \alpha + 1 = T(T \uparrow \alpha)$
$T \uparrow \alpha = lub\{E_\beta | \beta < \alpha\}$ if α is a limit ordinal and $(E_\beta)_{\beta<\alpha}$ is the greatest increasing chain contained in $(T \uparrow \beta)_{\beta<\alpha}$, i.e. constructively $E_\beta = glb\{T \uparrow \gamma \mid \beta \leq \gamma < \alpha\}$.

The intuitive idea of this definition is that we retain at a limit ordinal α only the information which was persistent in the preceding non-monotonic iterations. We remark that if T is monotonic then in the definition of a limit ordinal power we get $E_\beta = T \uparrow \beta$ which makes our definition equivalent to the classical definition for monotonic operators. If T is not monotonic we still have that if $A \in T \uparrow \alpha + i$ for any integer i then $A \in T \uparrow \beta$ for β the next limit ordinal after α. In particular if $T \uparrow \alpha$ is a fixpoint of T then for any ordinal $\beta \geq \alpha$, $T \uparrow \beta = T \uparrow \alpha$.

This notion of non-monotonic induction has been introduced independently for logic programs by [3] who proved that the non-monotonic operator T_P of a strongly determined program converges to the unique stable model of the program. Strongly determined programs however form a restrictive class of programs that does not include stratified programs. We show here that any stable model of a general logic program P can be approximated by a sequence of ordinal powers of the justification maintenance operator J_P, yet the price to pay is non-determinism: we have to consider all rule selection strategies. To overcome this difficulty we shall study in the next section the case where the unique stable model of the program is approximated by *any* fair strategy.

We denote by $(J_P^\rho \uparrow \alpha)_{\alpha \leq \beta}$ the transfinite sequence of ordinal powers of J_P under a rule selection strategy ρ, which gives the choice of the rule in the conflict set used to obtain the successor element in the sequence.

3.4. Theorem. Let P be a well-formed logic program. M is a stable model of P if and only if for some strategy ρ $M = \overline{J_P^\rho} \uparrow \omega$ and $J_P^\rho \uparrow \omega$ is a fixpoint of J_P.

Proof. If $J_P^\rho \uparrow \omega$ is a fixpoint of J_P then $\overline{J_P^\rho \uparrow \omega}$ is a stable model of P by 3.3. Conversely if M is a stable model of P then by 2.3 there exists a justified interpretation J such that $\overline{J} = M$. Let $\bigcup_{i \geq 0} J_i$ be an enumeration of J. Notice that by the definition of a justified interpretation, for any couple of elements, $J_i = A/\Gamma$, $J_j = B/\Delta$, such that $B \in \Gamma$ and $i < j$, we have $\Delta \subseteq \Gamma$ so $A \notin \Delta$. Let $\bigcup_{i \geq 0} J'_i$ be the enumeration obtained from $\bigcup_{i \geq 0} J_i$ by repeatedly exchanging such couples. $\bigcup_{i \geq 0} J'_i$ is an enumeration of J in which any justified atom is preceded by its positive justifications. For any $i \geq 0$ there exists a rule $R_i \in \mathit{Ground}(P)$ such that $J'_i = \mathit{concl}(R_i)/\mathit{prem}(R_i) \cup \Gamma$ with $\overline{J} \models \mathit{prem}(R_i)$ and $\Gamma = \{\Delta \mid B \in \mathit{pos}(R_i),\ B/\Delta \in J_j,\ j < i\}$. Let us consider the strategy ρ which selects the rule R_i at step i. Clearly we have $J = J_P^\rho \uparrow \omega$ so $M = \overline{J_P^\rho \uparrow \omega}$, and by 3.3, $J_P^\rho \uparrow \omega$ is a fixpoint of J_P. QED

4 The rational model semantics for general logic programs

In this section we define the notion of transfinite fairness and show how the unique stable model of a logic program can be approximated by iterating the justification maintenance operator under an arbitrary fair strategy. This provides a simple method of constructing the iterated least model of a stratified program, the unique perfect model of a locally stratified program and the two-valued well-founded model of a program when it exists.

Definition. We say a rule $R \in Ground(P)$ is *treated at ordinal* β if $R \notin S_P(\overline{J_P^\rho \uparrow \beta})$ or R is selected at ordinal β or for any $\gamma \geq \beta$ $concl(R) \in \overline{J_P^\rho \uparrow \gamma}$.

We say a rule selection strategy ρ is *fair* iff whenever $R \in S_P(\overline{J_P^\rho \uparrow \alpha})$ for some ordinal α, there exists an ordinal $\beta \geq \alpha$ where R is treated.

4.1. Proposition. For any logic program P the transfinite "first-in-first-out" ordering of S_P constitutes a fair strategy.

Proof. The transfinite FIFO ordering of $S_P(\overline{J_P^\rho \uparrow \alpha})$ for any ordinal α respects the partial ordering in which rules were satisfied, that is for all rules $R_1, R_2 \in S_P(\overline{J_P^\rho \uparrow \alpha})$, if there exists an ordinal $\beta \leq \alpha$ where $R_2 \notin S_P(\overline{J_P^\rho \uparrow \beta})$ and for any γ, $\beta \leq \gamma \leq \alpha$, we have $R_1 \in S_P(\overline{J_P^\rho \uparrow \gamma})$, then $R_1 \prec R_2$.

For any satisfied rule $R \in S_P(\overline{J_P^\rho \uparrow \alpha})$ at rank δ w.r.t. $\prec$, we show by transfinite induction on δ that there exists an ordinal $\beta \geq \alpha$ where R is treated.

1) In the base case $\delta = 0$. If for some ordinal $\beta \geq \alpha$, $concl(R) \notin \overline{J_P^\rho \uparrow \beta}$, let β be the least such ordinal, then either $R \notin S_P(\overline{J_P^\rho \uparrow \beta})$ or R is selected at ordinal $\beta + 1$. Otherwise $\forall \gamma \geq \alpha$ $concl(R) \in \overline{J_P^\rho \uparrow \gamma}$. In any case R is treated at some ordinal greater or equal than α.

2) If δ is a successor ordinal, the proof is similar. If for some ordinal $\beta \geq \alpha$ $concl(R) \in \overline{J_P^\rho \uparrow \beta}$, let β be the least such ordinal, then either $R \notin S_P(\overline{J_P^\rho \uparrow \beta})$ or R is selected at ordinal $\beta+1$, or another rule of lesser rank than R is selected at ordinal $\beta+1$, so in this case the rank of R in $S_P(\overline{J_P^\rho \uparrow \beta + 1})$ decreases, thus by transfinite induction we get that R is treated. Otherwise $\forall \gamma \geq \alpha$ $concl(R) \in \overline{J_P^\rho \uparrow \gamma}$. In all cases R is treated at some ordinal greater than or equal to α.

3) If δ is a limit ordinal, by transfinite induction all rules of rank $\delta' < \delta$ are treated at some ordinal greater than or equal to α. Let β be the least ordinal where all rules of rank $\delta' < \delta$ have been treated. Then either R has been treated at an ordinal γ, $\alpha \leq \gamma \leq \beta$, or by the definition of ρ R has rank 0 in $S_P(\overline{J_P^\rho \uparrow \beta})$, in which case we get by induction that R is treated at an ordinal greater than or equal to β. QED

On example

$$P_3 = \{\neg b \rightarrow a, \quad \neg a \rightarrow b, \quad a, b \rightarrow c, \quad \neg c \rightarrow a\}$$

from [39] as on example P_1, any fair strategy leads to the unique stable model $\{a\}$ which is also the unique model of $comp(P_3)$, while the well-founded partial model is empty,

$\emptyset \triangleright \{a/\neg b\}$

$\emptyset \triangleright \{b/\neg a\} \triangleright \{a/\neg c\}$

$\emptyset \triangleright \{a/\neg c\}$

However there are still examples like

$$P_4 = \{\neg b \to a,\\ \neg a \to b,\\ b \to a\}$$

which have a unique stable model, here $\{a\}$, while some fair strategies do not reach a fixpoint:

$\emptyset \triangleright \{a/\neg b\}$

$\emptyset \triangleright \{b/\neg a\} \triangleright \emptyset \triangleright \{a/\neg b\}$

$\emptyset \triangleright \{b/\neg a\} \triangleright \emptyset \triangleright \{b/\neg a\} \triangleright \emptyset \triangleright ...$

The well-founded partial model of P_4 is empty. The same situation arises with example $P_5 = \{\neg b \to a, \quad \neg a \to b, \quad \neg p \to p, \quad \neg b \to p\}$ from [39].

Definition. We say that an interpretation M is the *rational model* of a general logic program P if for any fair strategy ρ there exists an ordinal α such that $J^\rho_P \uparrow \alpha$ is a fixpoint of J_P and $M = \overline{J^\rho_P \uparrow \alpha}$.

4.2. Main theorem. Let P be a general logic program and ρ be a fair strategy. After some ordinal β the ordinal powers of J^ρ_P are extensions of the well-founded partial model of P. More precisely

$$\forall \alpha\ \exists \beta\ \forall \gamma \geq \beta\ \ T_{P,\alpha} \subseteq \overline{J^\rho_P \uparrow \gamma}\ \wedge\ F_{P,\alpha} \cap \overline{J^\rho_P \uparrow \gamma} = \emptyset$$

Proof. By transfinite induction on α.

1) The base case is trivial as $V_P \uparrow 0 = (\emptyset, \emptyset)$.

2) In the case of a successor ordinal, let $V_P \uparrow \alpha + 1 = (T_{P,\alpha+1}, F_{P,\alpha+1})$. By transfinite induction $\exists \beta\ \forall \gamma \geq \beta\ T_{P,\alpha} \subseteq \overline{J^\rho_P \uparrow \gamma}\ \wedge\ F_{P,\alpha} \cap \overline{J^\rho_P \uparrow \gamma} = \emptyset$. We show first that $\exists \beta' \geq \beta\ \forall \gamma \geq \beta'\ T^{\alpha+1}_P \subseteq \overline{J^\rho_P \uparrow \gamma}$.

Let $C = \{R \in \mathit{Ground}(P) \mid \mathit{pos}(R) \subseteq T^\alpha_P\ \wedge\ \mathit{neg}(R) \subseteq F^\alpha_P\}$, by definition $T^{\alpha+1}_P = \{\mathit{concl}(R) \mid R \in C\}$. By induction $\forall R \in C\ \forall \gamma \geq \beta\ \overline{J^\rho_P \uparrow \gamma} \models \mathit{prem}(R)$. So for any $R \in C$, by the fairness of ρ there exists $\beta' \geq \beta$ where R is treated, that is in this case either $\forall \gamma \geq \beta'\ \mathit{concl}(R) \in \overline{J^\rho_P \uparrow \gamma}$, or R is selected at ordinal β' so again $\forall \gamma \geq \beta'\ \mathit{concl}(R) \in \overline{J^\rho_P \uparrow \gamma}$.

We show now that $\forall \gamma \geq \beta\ F_{P,\alpha+1} \cap \overline{J^\rho_P \uparrow \gamma} = \emptyset$. By definition $F_{P,\alpha+1} = F_P(T_{P,\alpha}, F_{P,\alpha})$, so for any $A \in F_{P,\alpha+1}$, for any $R \in \mathit{Ground}(P)$ with $\mathit{concl}(R) = A$, we have

$T^\alpha_P \cap \mathit{neg}(R) \neq \emptyset$ and by induction $\overline{J^\rho_P \uparrow \gamma} \cap \mathit{neg}(R) \neq \emptyset$,

or $F^\alpha_P \cap \mathit{pos}(R) \neq \emptyset$ and by induction $\mathit{pos}(R) \not\subseteq \overline{J^\rho_P \uparrow \gamma}$,

or $F^{\alpha+1}_P \cap \mathit{pos}(R) \neq \emptyset$.

Hence for any $A \in F_P^{\alpha+1} \cap \overline{J_P^\rho \uparrow \gamma}$, let $A/\Gamma \in J_P^\rho \uparrow \gamma$, there must exist $R \in Ground(P)$ with $concl(R) = A$ and $prem(R) \subseteq \Gamma$ such that $F_P^{\alpha+1} \cap pos(R) \neq \emptyset$. Hence for any $A \in F_P^{\alpha+1} \cap \overline{J_P^\rho \uparrow \gamma}$ there exists $B \in F_P^{\alpha+1} \cap \overline{J_P^\rho \uparrow \gamma}$ with $B \prec A$, where $\prec$ is a strict well-founded partial ordering that establishes the well-supportedness of $J_P^\rho \uparrow \gamma$. Consequently $F_{P,\alpha+1} \cap \overline{J_P^\rho \uparrow \gamma} = \emptyset$, otherwise we would have an infinite chain w.r.t. $\prec$.

3) In the case of a limit ordinal we have $T_P^\alpha = \bigcup_{\delta<\alpha} T_{P,\delta}$ and $F_P^\alpha = \bigcup_{\delta<\alpha} F_{P,\delta}$. By transfinite induction $\forall \delta < \alpha\ \exists \beta_\delta\ \forall \gamma \geq \beta_\delta\ T_{P,\delta} \subseteq \overline{J_P^\rho \uparrow \gamma}$ and $F_{P,\delta} \cap \overline{J_P^\rho \uparrow \gamma} = \emptyset$. Let β be the sup of the β_δ's for $\delta < \alpha$, we get for any $\gamma \geq \beta$, $T_{P,\alpha} \subseteq \overline{J_P^\rho \uparrow \gamma}$ and $F_{P,\alpha} \cap \overline{J_P^\rho \uparrow \gamma} = \emptyset$. QED

An immediate corollary is the following theorem of [39].

4.3. Corollary. Any stable model of P is an extension of the well-founded partial model of P.

Proof. By 3.3 any stable model of a well-formed logic program P is a fixpoint of J_P (only the converse of this proposition needs the well-formedness assumption), therefore by 4.2 we get that any stable of P is an extension of the well-founded partial model of P. QED

4.4. Corollary. If a logic program has a two-valued well-founded model, then that model is the rational model of the program.

Proof. Let P be a general logic program having a two valued well-founded model. Let α be the closure ordinal of V_P. By theorem 4.2, for any fair strategy ρ we have $\exists \beta\ \forall \gamma \geq \beta\ T_{P,\alpha} \subseteq \overline{J_P^\rho \uparrow \gamma}$ and $F_{P,\alpha} \cap \overline{J_P^\rho \uparrow \gamma} = \emptyset$. Since (T_P^α, F_P^α) is a model, $T = B_H \backslash F_{P,\alpha}$ thus $T_{P,\alpha} = \overline{J_P^\rho \uparrow \gamma}$ for any $\gamma \geq \beta$. Therefore for any fair strategy ρ there exists β such that $J_P^\rho \uparrow \beta$ is a fixpoint of J_P and $T_{P,\alpha} = \overline{J_P^\rho \uparrow \beta}$, i.e. $T_{P,\alpha}$ is the rational model of P. QED

Examples P_1 and P_3 show that the converse of this corollary is false. In this sense the rational model semantics generalizes the well-founded model semantics for logic programs. However some well-behaved logic programs such as P_4 and P_5 have no rational model. A fundamental question underlying this difficulty is the logical complexity of the unique stable model of a well-behaved logic program [33] [4] [22].

5 Operational Semantics and Relation to Truth Maintenance Systems

General logic programs when interpreted by SLDNF-resolution as in standard Prolog implementations lack a fully equivalent declarative semantics. Under strong restrictions on the use of variables and negation, SLDNF-resolution is sound and complete w.r.t. $comp(P)$ [23] [6] [21]. In the general case it is well known that SLDNF-resolution is sound but not complete w.r.t. $comp(P)$, for example with $P_6 = \{p \rightarrow p, \ \ p \rightarrow q, \ \ \neg p \rightarrow q\}$, q is a logical consequence of $comp(P_6)$ but it cannot be obtained by SLDNF-resolution. Following [15], [22] proposed another declarative semantics for which SLDNF-resolution is sound but still not complete. For that semantics a complete interpreter does exist but is not practical. In [34] an extension of the negation as failure rule is proved sound and complete for a weak form of $comp(P)$.

For stratified logic programs [1] [38] the iterated least model semantics is a very appealing declarative semantics but for which no complete interpreter can exist. SLDNF-resolution is not sound w.r.t. that semantics. To achieve soundness, [1] and [38] augment SLDNF-resolution with loop checks and ground instantiation of subgoals. Completeness is obtained in the finite case, that is when B_H is finite (there are no function symbols only constants) or when P satisfies the bounded term size property [38]. This interpreter is practical however only if ground instantiation of subgoals does not occur. This is the case for instance for logic programs that satisfy in addition the ground derivation property, then starting with a ground goal SLDNF-resolution (with loop checks) always generates ground goals.

We describe here an alternative interpreter which is sound w.r.t. the rational model semantics, and complete if B_H is finite. This interpreter differs from the bottom-up evaluator proposed in [35] by the absence of backtracking. When B_H is finite it is more convenient to redefine fairness in order to force satisfied rules to be treated in a finite number of steps. We say that a strategy ρ is fair if whenever $R \in S_P(\overline{J_P^\rho \uparrow \alpha})$ for some ordinal α, there exists an integer i such that R is treated at ordinal $\alpha + i$.

5.1. Lemma. If P has a rational model M and B_H is finite, then J_P^ρ finitely converges to M for any fair strategy ρ.

Proof. As B_H and P are finite, JI_P are finite. It is not true with our definition of ordinal powers that a non-monotonic operator T over a finite semilattice either converges finitely or none of its ordinal powers is a fixpoint. However one can conclude by the definition of the limit ordinal powers of T that $T \uparrow \omega \subseteq T \uparrow j$ for all j greater than some integer i. Therefore there exists an integer i such that $J_P^\rho \uparrow \omega \subseteq J_P^\rho \uparrow i$. As $J_P^\rho \uparrow i$ is a justified interpretation, there exists a strategy ρ' which coincides with ρ on the finite powers, $\forall k \ J_P^\rho \uparrow k = J_P^{\rho'} \uparrow k$, and for some integer j we have $J_P^{\rho'} \uparrow (\omega + j) = J_P^{\rho'} \uparrow i$. In this way for any limit ordinal α we have $J_P^{\rho'} \uparrow \alpha = J_P^\rho \uparrow \omega$. As ρ is fair so is ρ', hence by hypothesis we have that $J_P^{\rho'} \uparrow \alpha$ is a fixpoint of J_P for some ordinal α and $M = \overline{J_P^{\rho'} \uparrow \alpha}$. Henceforth $M = \overline{J_P^{\rho} \uparrow \omega}$. Now $J_P^{\rho} \uparrow \omega \subseteq J_P^{\rho} \uparrow i$ and $C_P(\overline{J_P^{\rho} \uparrow \omega}) = \emptyset$ by 3.2, so we get $J_P^{\rho} \uparrow \omega = J_P^{\rho} \uparrow i$, that is J_P^{ρ} finitely converges to M. QED

If B_H is finite and P has a rational model M, then the naive interpreter which is sound and complete w.r.t. M, computes M as a finite power of J_P^{ρ} with a fair strategy ρ and retrieves in M the answers to the query Q. From the point of view of the worst-case time complexity, the size of M is bounded by the size of B_H, that is by $r.c^a$ if there are no function symbols and r denotes the number of relation symbols, a the maximum arity and c the number of constants. An example of a logic program with n rules (without negation) where SLDNF-resolution is forced to enumerate in exponential time all of B_H of size $O(2^n)$ is given in [1]. Here the proof of termination of J_P^{ρ} under the assumption that a rational model exists and B_H is finite, is not informative. In fact the worst-case time complexity of a logic program with negation can be a double exponential in the number of rules, i.e. $O(2^{2^n})$, but is still simply exponential under the best strategy, that is when J_P^{ρ} is increasing.

To show this pathological behaviour we can modify the 3 bits counter program from [1] by making a copy of the rules augmented with a negative literal. Let

$C_3 = \{c_3(0,0,0),$
$\quad c_3(x,y,0) \rightarrow c_3(x,y,1),$
$\quad c_3(x,0,1) \rightarrow c_3(x,1,0),$
$\quad c_3(0,1,1) \rightarrow c_3(1,0,0),$
$\quad \neg c_3(1,1,1),\ c_3(x,y,0) \rightarrow c_3(x,y,1),$
$\quad \neg c_3(1,1,1),\ c_3(x,0,1) \rightarrow c_3(x,1,0),$
$\quad \neg c_3(1,1,1),\ c_3(0,1,1) \rightarrow c_3(1,0,0)\ \}$

By an obvious generalization let us consider the logic program C_n with $2n+1$ rules. Under the worst (fair) strategy the rules with negation are selected first, the enumeration of B_H takes $2 \times \ldots \times 2$, 2^n times, steps, that is $O(2^{2^n})$. With the best strategy the rules with negation are not selected, J_P^{ρ} is increasing, and B_H is enumerated optimally in $O(2^n)$ steps.

As it stands the naive interpreter is not practical and perfectly useless, independently of any worst-case time complexity argument. To make that interpreter "practical" the idea is to direct the search as in a backward chaining procedure by introducing in the rules metalevel conditions that match the current goals.

Definition. For a given logic program P we denote by $\hat{P}$ the logic program obtained by transforming each rule with $n \geq 0$ premises,

$$A_1, \ldots, A_m, \neg A_{m+1}, \ldots, \neg A_n \rightarrow A$$

in the following $n+1$ rules:

$$goal(A) \rightarrow goal(A_1)$$

$$goal(A), A_1 \rightarrow goal(A_2)$$

$$...$$

$$goal(A), A_1, ..., A_m, \neg A_{m+1}, ..., \neg A_{n-1} \rightarrow goal(A_n)$$

$$goal(A), A_1, ..., A_m, \neg A_{m+1}, ..., \neg A_n \rightarrow A$$

This transformation is reminiscent to the magic set method for deductive databases [31]. Here the transformed rules share the same conditions, so they constitute essentially only one rule when they are compiled in a data driven fashion with Rete algorithm [17] [14].

Given a logic program P and a query Q, we call J_P^ρ*-resolution* the procedure that consists of applying the naive interpreter on $\hat{P} \cup \{goal(Q)\}$. We say that an atom A (resp. a negated atom $\neg A$) is *derivable* by J_P^ρ-resolution if $J^\rho_{\hat{P} \cup \{goal(A)\}}$ finitely converges to a stable model M and A is true (resp. false) in M. It is important to realize that in this bottom-up procedure not only the proof of already established facts is factorized, as proposed for top-down procedures in [36], but subgoals also are memorized, that is the *search* for a proof is equally factorized and loop-checks of [40] [1] [38] are built-in. The drawback is in the space complexity necessary to memorize all the current goals. The translation into $\hat{P}$ is however such that only the subgoals which are supported by the preceding conditions are considered. From the point of view of worst-case time complexity we still have a double exponential under the worst strategy and a simple exponential under the best strategy when J_P^ρ is increasing.

5.2. Theorem. If P is a logic program having a rational model M then J_P^ρ-resolution is sound w.r.t. M. Furthermore if B_H is finite, J_P^ρ-resolution is a complete interpreter w.r.t. M for every fair strategy ρ.

We see the transformation $\hat{P}$ as the rationale of forward chaining as used in rule-based expert systems for problem solving [16]. In these systems the forward chaining rules for subgoaling are usually written by hand, and extra control conditions can be added to direct the search even more accurately. Best-first strategies that rely on fuzzy matching or on uncertainty coefficients can be implemented also but they are generally not fair. In concrete implementations of justification maintenance multiple justifications for a fact are allowed (the conflict set is taken to be the set of satisfied rules). A common "error" in existent systems of this kind is that they omit loop checks in justifications for efficiency reasons, so non well-supported models may be computed. The point is that Rete algorithm can be easily extended to implement very efficiently the maintenance of justifications without loop checks.

J_P^ρ-resolution for logic programs can be seen also as an instance of the more general "truth maintenance system" (TMS) of [9]. The TMS deals with logic programs extended principally with a new constant standing for "inconsistency". In a TMS when an inconsistency is encountered the current set of "beliefs" is revised and "intelligent backtracking" is performed by an analysis of the sources of the contradiction. On the other hand the abductive extension of logic programming in [11] can be compared with the "assumption-based truth maintenance system" (ATMS) of [8].

6 Conclusion

There are different semantics for logic programs that have their own merit and that are not subsumed by one or another. The stable model semantics provides a simple and natural definition for canonical models. Their characterization as the well-supported models reinforces that claim and clarifies the difference between the stable model semantics and the Clark's completion semantics. The well-founded semantics is a construction of stable models with many nice properties, but it is revealed to be too weak on some examples. The rational model semantics is intended to solve that problem while remaining a reasonably intuitive and implementable semantics. On the other hand the stratification of the program remains the principal decidable criterion that ensures the existence of a unique stable model.

Our construction can be seen also as a particular instance of the "truth maintenance system" of [9]. Its previous formalization in a non-monotonic logic based on a modal operator standing for non-monotonic provability was criticized in [7]. The connection with logic programming sheds a new light on the relationship between non-monotonic reasoning and logic programming, as well as on some fundamental bottom-up procedures used in rule-based expert systems. In this direction the problem to generalize the declarative semantics of logic programs to allow negation in the conclusion of the rules, as studied in [19], is directly relevant to the semantics of *production rules* used in expert systems [16] and deductive databases.

The fixpoint technique we have used that relies on both non-monotonic and non-deterministic induction, can also be of independent interest, see [3] [20] [1] [24].

Acknowledgements

This work benefited from many fruitful discussions with my colleagues at LIENS and LCR. I wish to thank also Krzysztof Apt and Gilberto File' for their comments.

References

[1] K.R. Apt, H.A. Blair, A. Walker, *Towards a theory of declarative knowledge*, in Foundations of deductive databases and logic programming, Minker, J. (ed.), Morgan Kaufmann, Los Altos (1987).

[2] K.R. Apt, M.H. van Emden, *Contributions to the theory of logic programming*, JACM, 29(3), pp.841-862 (1982).

[3] A. Batarekh, *Fixpoint techniques for non-monotone maps*, Proc. of the 2nf Int. Conf. on Algebraic and Logic Programming, Nancy (1990).

[4] D.R. Brough, A. Walker, *Some practical properties of logic programming interpreters*, Proc. of FGCS'84, Tokyo (1984).

[5] K.L. Clark, *Negation as failure*, In Logic and Databases, H. Gallaire and J. Minker (Eds.), Plenum Press, New York, pp.293-322 (1978).

[6] L. Cavedon, J.W. Lloyd, *A completeness theorem for SLDNF-resolution*, Journal of Logic Programming, 7(3), pp.177-192 (1989).

[7] M. Davis, *The mathematics of non-monotonic reasoning*, Artificial Intelligence, vol. 13, pp.73-80 (1980).

[8] J. de Kleer, *An assumption-based truth maintenance system*, Artificial Intelligence vol. 28-2, pp.127-162 (1986).

[9] J. Doyle, *A truth maintenance system*, Artificial Intelligence, vol. 12, pp.231-272 (1979).

[10] P.M. Dung, K. Kanchanasut, *A fixpoint approach to declarative semantics of logic programs*, NACLP'89. MIT Press (1989).

[11] K. Eshghi, R.A. Kowalski, *Abduction compared with negation by failure*, 6th Intern. Conference on Logic Programming, Lisbonne (1989).

[12] C. Elkan, *A rational reconstruction of nonmonotonic truth maintenance systems*, Journal of Artificial Intelligence, vol. 43, pp.219-234 (1990).

[13] F. Fages, *Consistency of Clark's completion and existence of stable models*, Workshop on non-monotonic reasoning and logic programming, NACLP'90, Austin. Research Report 90-15, LIENS, Ecole Normale Supe'rieure (1990).

[14] F. Fages, *On the Proceduralization of Rules in Expert Systems*, In Programming of Future Generation Computers, Eds M. Nivat and K. Fuchi, North-Holland, (1988).

[15] M.R. Fitting, *A Kripke-Kleene semantics for logic programs*, J. of Logic Programming 2, pp.295-312 (1985).

[16] C.L. Forgy, *OPS5 Users Manual*, Dept. of Cognitive Sciences, Carnegie Mellon Univ., Pittsburgh (1981).

[17] C. Forgy, *Rete: A Fast Algorithm for the Many Pattern-Many Object Pattern Match Problem*, Artificial Intelligence 19, pp. 17-37 (1982).

[18] M. Gelfond, V. Lifschitz, *The stable model semantics for logic programming*, Proc. of the 5th Logic Programming Symposium, pp.1070-1080, MIT press (1988).

[19] M. Gelfond, V. Lifschitz, *Logic programs with classical negation*, Proc. of the 7th Logic Programming Symposium, pp.579-597, MIT press (1990).

[20] I. Guessarian, *Some fixpoint techniques in algebraic structures and applications to computer science*, Fundamentica Informatica X, pp.387-414 (1987).

[21] K. Kunen, *Signed data dependencies in logic programs*, Journal of Logic Programming, 7(3), pp.231-245 (1989).

[22] K. Kunen, *Negation in logic programming*, Journal of logic programming, 4(4), pp.289-308 (1987).

[23] J.W. Lloyd, *Foundations of Logic Programming*, Springer Verlag (1987).

[24] J.L. Lassez, V.L. Nguyen, E.A. Sonenberg, *Fixed point theorems and semantics: a folk tale*, Information Processing Letters 14(3), pp.112-116 (1982).

[25] J. McCarthy, *Circumscription: a form of non-monotonic reasoning*, Artificial Intelligence, 13, pp.27-39 (1980).

[26] R.C. Moore, *Semantic considerations on non-monotonic logic*, Artificial Intelligence vol. 25, pp.75-94 (1985).

[27] W. Marek, M. Truszczynski, *Stable semantics for logic programs and default theories*, NACLP'89. MIT Press (1989).

[28] T. Przymusinski, *On the declarative and procedural semantics of logic programs*, Journal of Automated Reasoning, 5, pp.167-205 (1989).

[29] T. Przymusinski, *Non-monotonic formalisms and logic programming*, Intern. Conference on Logic Programming, Lisbonne (1989).

[30] T. Przymusinski, *On the declarative semantics of stratified deductive databases and logic programming*, in Foundations of deductive databases and logic programming, Minker, J. (ed.), Morgan Kaufmann, Los Altos (1987).

[31] R. Ramakrishnan, *Magic templates: a spellbinding approach to logic programming*, Proc. of the fifth International Conference on Logic Programming, Seattle (1988).

[32] R. Reiter, *A logic for default reasoning*, Artificial Intelligence, 13, pp.81-132 (1980).

[33] J.S. Schlipf, *The expressive powers of the logic programming semantics*, Proc. of PODS'90, pp.196-204, (1990).

[34] J.C. Shepherdson, *A sound and complete semantics for a version of negation as failure*, Journal of Theoretical Computer Science, vol. 65, pp.343-371 (1989).

[35] D. Sacca, C. Zaniolo, *Stable models and non-determinism in logic programs with negation*, PODS'90, pp.205-217, (1990).

[36] H. Tamaki, T. Sato, *OLD-resolution with tabulation*, Proc. of the 3rd Int. Conf. on Logic Programming, pp.84-98, London (1986).

[37] M.H. Van Emden, R.H. Kowalski, *The semantics of predicate logique as a programming language*, JACM, 23(4), pp.733-742 (1976).

[38] A. Van Gelder, *Negation as failure using tight derivations for general logic programs*, in Foundations of deductive databases and logic programming, Minker, J. (ed.), Morgan Kaufmann, Los Altos. Revised version in Journal of Logic Programming 1989 pp.109-133. (1987).

[39] A. Van Gelder, K. Ross, J.S. Schlipf, *Unfounded sets and well-founded semantics for general logic programs*, Proc. of the Symp. on Principles of Databases Systems, ACM-SIGACT-SIGCOM (1988).

[40] D.S. Warren, *The XWAM: a machine that integrates Prolog and deductive database query evaluation*, Technical Report 89/25, Suny at Stony Brook, NY (1989).