Stable Models Are Generated by a Stable Chain

Heinrich Herre * Gerd Wagner †

Submitted to *JLP*.

Abstract

We propose a general preference criterion selecting the 'intended' models of generalized logic programs which a) is a conservative extension of the stable model semantics for normal logic programs of [GL88], b) is very close to the answer set semantics of [GL91] for disjunctive logic programs, and c) allows for arbitrary formulas in the body and in the head of a rule, i.e. does not depend on the presence of any specific connective, such as negation(-as-failure), nor on any specific syntax of rules. Our preference criterion defines a model of a program as intended if it is generated by a stable chain. We show that stable models and answer sets are generated by a stable chain of length $\leq \omega$.

1 Introduction

A logic program consists of *facts* and *deduction rules*. Facts correspond to sentences of a suitably restricted language, and deduction rules correspond to non-schematic (Gentzen) sequents. While facts express extensional knowledge, rules express intensional knowledge. A set of facts can be viewed as a database whose semantics is determined by its minimal models.

In the case of logic programs, minimal models are not adequate because they are not able to capture the directedness of rules, i.e. they do not satisfy the *groundedness* requirement. Therefore, *stable* models in the form of certain fixpoints have been proposed by Gelfond and Lifschitz [GL88] as the intended models of normal logic programs. We generalize this notion by presenting a definition which is neither fixpoint-based nor dependent on any specific rule syntax.

We call our preferred models *stable generated* because they are generated by a *stable chain*, i.e. a stratified sequence of rule applications where all applied rules

*herre@informatik.uni-leipzig.de, Institut für Informatik, Univ. Leipzig, Augustusplatz 10-11, 04109 Leipzig, Germany

†gw@inf.fu-berlin.de, Institut für Informatik, Univ. Leipzig, Augustusplatz 10-11, 04109 Leipzig, Germany

remain applicable throughout the model computation, and not because there is a close relationship to the concept of stable expansions in autoepistemic logic which was the original motivation of the stable model definition of [GL88]. We believe that in the semantics of rules, the notion of a stable chain is more fundamental than that of a stable expansion. The former is a genuine rule-related concept whereas the latter is just an application of a concept from autoepistemic logic. Interestingly, and making a good point for both approaches, in the base case of normal logic programs, there is a one-to-one correspondence between stable expansions and models generated by a stable chain, i.e. 'GL-stable' models and stable generated models agree.

While Gelfond and Lifschitz had to modify their definition of stable models, resp. answer sets, for each extension of normal logic programs they have considered (see [GL90, GL91, Gel91]), our definition is uniformly applicable without changes in all these extensions.[1] Our definition also accommodates negation(-as-failure) in the head of rules which is not possible in answer set semantics, and would again require a modification of it.[2]

We show that all answer sets of a normal disjunctive programs are generated by a stable chain. It turns out, however, that in such cases like head-cyclic disjunctive programs there are models generated by a stable chain which are not answer sets. We present a simple example of a head-cyclic program whose unique minimal model is generated by a stable chain but is not an answer set. This disagreement may suggest two things: 1) that our stable models are essentially different from answer sets, or 2) that the difference is not really essential but points to a weakness in the definition of answer sets for disjunctive programs. We tend to think that the latter is the case, but only further investigations will be able to decide this question.

While Gelfond and Lifschitz (and many other authors) treat negation-as-failure in a special way and unlike other connectives, and do not make any attempt to relate their semantics of logic programs to a standard model-theoretic account of rules, we emphasize the possibility to analyze logic programs using standard logical means such as the notions of (Gentzen) sequents, (possibly partial) interpretations, an information ordering between interpretations, and preferential entailment. As a consequence, we are able to allow for arbitrary formulas in the head and the body of rules, and to characterize negation-as-failure as standard negation under the preferential semantics of stable models.

The paper has the following structure. After introducing some basic notation in section 2, we recall some facts about Herbrand model theory in section 3, and then introduce the notion of a sequent in section 4. In section 5, we define the general concept of a stable model, and then, in sections 6 and 7, investigate the relationship of our general concept to the original fixpoint-based definitions for normal and normal disjunctive programs.

[1]This is shown for extended programs in [HJW95], for extended disjunctive programs in [HW96], and is briefly sketched for epistemic, i.e. S5 logic programs in the appendix.

[2]Such as proposed, e.g., in [LW92].

2 Preliminaries

A *signature* $\sigma = \langle Rel, Const, Fun\rangle$ consists of a set of relation symbols, a set of constant symbols, and a set of function symbols. U_σ denotes the set of all ground terms of σ. For a tuple $t_1, \dots, t_n$ we will also write $\vec{t}$ when its length is of no relevance. The logical functors are $\neg, \wedge, \vee, |, \supset, \forall, \exists$; where $|$ and $\supset$ are called *exclusive* disjunction, and *material implication*, respectively. $L(\sigma)$ is the smallest set containing the atomic formulas of σ, and being closed with respect to the following conditions: if $F, G \in L(\sigma)$, then $\{\neg F,\ F \wedge G,\ F \vee G,\ F|G,\ F \supset G,\ \exists x F,\ \forall x F\} \subseteq L(\sigma)$.

$L^0(\sigma)$ denotes the corresponding set of sentences (closed formulas). For sublanguages of $L(\sigma)$ formed by means of a subset $\mathcal{F}$ of the logical functors, we write $L(\sigma; \mathcal{F})$. With respect to a signature σ we define the following sublanguages: $\mathrm{At}(\sigma) = L(\sigma; \emptyset)$, the set of all atomic formulas (also called *atoms*); $\mathrm{Lit}(\sigma) = L(\sigma; \neg)$, the set of all *literals*. We introduce the following conventions. When $L \subseteq L(\sigma)$ is some sublanguage, L^0 denotes the corresponding set of sentences. If the signature σ does not matter, we omit it and write, e.g., L instead of $L(\sigma)$.

If Y is a partially ordered set, then $\mathrm{Min}(Y)$ denotes the set of all minimal elements of Y, i.e. $\mathrm{Min}(Y) = \{X \in Y \mid \neg\exists X' \in Y : X' < X\}$.

3 Model Theory

A *Herbrand interpretation* of the language $L(\sigma)$ is one for which the universe equals U_σ, and the function symbols are interpreted canonically. We identify Herbrand interpretations over σ with subsets of $\mathrm{At}^0(\sigma)$.

Definition 1 (Interpretation) *Let $\sigma = \langle Rel, Const, Fun\rangle$ be a signature. A Herbrand σ-interpretation can be represented as a set of atoms $I \subseteq \mathrm{At}^0(\sigma)$. Its* universe *is equal to the set of all ground terms U_σ; its canonical interpretation of ground terms is the identity mapping. A Herbrand interpretation I assigns a relation*

$$r^I = \{\vec{t} \in U_\sigma^{a(r)} : r(\vec{t}) \in I\}$$

to every relation symbol $r \in Rel$, where $a(r)$ denotes the arity of r.

The class of all Herbrand σ-interpretations is denoted by $\boldsymbol{I}_H(\sigma) = 2^{\mathrm{At}^0(\sigma)}$. In the sequel we shall also simply say 'interpretation' instead of 'Herbrand interpretation'. A *valuation* over an interpretation I is a function ν from the set of all variables Var into the Herbrand universe U_σ, which can be naturally extended to arbitrary terms by $\nu(f(t_1, \dots, t_n)) = f(\nu(t_1), \dots, \nu(t_n))$. Analogously, a valuation ν can be canonically extended to arbitrary formulas F, where we write $F\nu$ instead of $\nu(F)$. Note that for a constant c, being a 0-ary function, we have $\nu(c) = c$. The model relation $\models \subseteq \boldsymbol{I}_H(\sigma) \times L^0(\sigma)$ between an interpretation and a sentence is defined inductively as follows.

Definition 2 (Model Relation) *1. $I \models r(\vec{t})$ iff $r(\vec{t}) \in I$.*

2. $I \models F \wedge G$ iff $I \models F$ and $I \models G$.

3. $I \models F \vee G$ iff $I \models F$ or $I \models G$.

4. $I \models \exists x F(x)$ iff $I \models F(t)$ for some $t \in U_\sigma$.

5. $I \models \forall x F(x)$ iff $I \models F(t)$ for all $t \in U_\sigma$.

6. $I \models \neg F$ iff $I \not\models F$.

In addition, we have the following definitions for exclusive disjunction,

$$F|G \longrightarrow (F \wedge \neg G) \vee (G \wedge \neg F)$$

and material implication,

$$F \supset G \longrightarrow \neg F \vee G$$

in the sense that for every rewrite rule $LHS \longrightarrow RHS$, we define

$$I \models LHS \quad \text{iff} \quad I \models RHS$$

The model relation between an interpretation $I \in \boldsymbol{I}_H(\sigma)$ and a formula $F \in L(\sigma)$ is defined by

$$I \models F \text{ iff } I \models F\nu \text{ for every valuation } \nu : Var \rightarrow U_\sigma$$

$\mathrm{Mod}_H(X) = \{I \in \boldsymbol{I}_H : I \models X\}$ denotes the Herbrand model operator, and $\models_H$ denotes the corresponding consequence relation, i.e. $X \models_H F$ iff $\mathrm{Mod}_H(X) \subseteq \mathrm{Mod}_H(F)$.

Definition 3 (Extension of an Interpretation) *Let $I, I' \in \boldsymbol{I}_H$ be two interpretations. We say that I'* extends *I, resp. I' is* informationally greater *or equal than I, symbolically $I \leq I'$, if $I \subseteq I'$.*

Definition 4 (Minimal Models) *For $F \in L(\sigma) \supseteq X$, we define $\mathrm{Mod}_m(X) = \mathrm{Min}(\mathrm{Mod}_H(X))$, and $X \models_m F$ iff $\mathrm{Mod}_m(X) \subseteq \mathrm{Mod}_H(F)$.*

Definition 5 (Persistent Formula) *A formula $F \in L(\sigma)$ is called* persistent *if for arbitrary Herbrand interpretations $I, J \in \boldsymbol{I}_H(\sigma)$ satisfying $I \leq J$, and every valuation $\nu : Var \rightarrow U_\sigma$ the condition $I \models F\nu$ implies $J \models F\nu$.*

Observation 1 *Every formula $F \in L(\sigma; \wedge, \vee, \exists, \forall)$ is persistent.*

4 Sequents

Traditionally, Gentzen sequents are used in a schematic way in sequent calculi in order to express valid transitions from one argument schema to another. In other words, a sequent in a sequential inference rule stands for a whole class of propositional substitution instances. Here, we propose to use sequents in a non-schematic way for the purpose of representing rule knowledge. A sequent, then, is not a schematic but a concrete expression representing some piece of knowledge.

Definition 6 (Satisfaction Set) *Let $I \in \boldsymbol{I}_H(\sigma)$, and $F \in L(\sigma)$. Then*

$$\mathrm{Sat}_I(F) = \{\nu \in {U_I}^{Var} : I \models F\nu\}$$

Definition 7 (Sequent) *A* sequent *s is an expression of the form*

$$F_1, \ldots, F_m \Rightarrow G_1, \ldots, G_n$$

where $F_i, G_j \in L(\sigma)$ for $i = 1, \ldots, m$ and $j = 1, \ldots, n$. The body *of s, denoted by Bs, is given by $\{F_1, \ldots, F_m\}$, and the* head *of s, denoted by Hs, is given by $\{G_1, \ldots, G_n\}$.* $\mathrm{Seq}(\sigma)$ *denotes the class of all sequents s such that $Hs, Bs \subseteq L(\sigma)$, and for a given set $S \subseteq \mathrm{Seq}(\sigma)$, $[S]$ denotes the set of all ground instances of sequences from S.*

Definition 8 (Model of a Sequent) *Let $I \in \boldsymbol{I}_H$. Then,*

$$I \models F_1, \ldots, F_m \Rightarrow G_1, \ldots, G_n \quad \textit{iff} \quad \bigcap_{i \leq m} Sat_I(F_i) \subseteq \bigcup_{j \leq n} Sat_I(G_j)$$

Observation 2 (Sequents and Material Implication) *Let $B \Rightarrow H$ be any sequent. Then, for any $I \in \boldsymbol{I}_H$,*

$$I \models B \Rightarrow H \quad \textit{iff} \quad I \models \bigwedge B \supset \bigvee H$$

We define the following classes of sequents corresponding to positive, normal, and generalized (disjunctive) logic programs, respectively.

1. $\mathrm{PLP}(\sigma) = \{s \in \mathrm{Seq}(\sigma) : Hs \in \mathrm{At}(\sigma),\ Bs \subseteq \mathrm{At}(\sigma)\}$.

2. $\mathrm{PDLP}(\sigma) = \{s \in \mathrm{Seq}(\sigma) : Bs, Hs \subseteq \mathrm{At}(\sigma)\}$.

3. $\mathrm{NLP}(\sigma) = \{s \in \mathrm{Seq}(\sigma) : Hs \in \mathrm{At}(\sigma),\ Bs \subseteq \mathrm{Lit}(\sigma)\}$.

4. $\mathrm{NDLP}(\sigma) = \{s \in \mathrm{Seq}(\sigma) : Hs \subseteq \mathrm{At}(\sigma),\ Bs \subseteq \mathrm{Lit}(\sigma), Hs \neq \emptyset\}$.

5. $\mathrm{GLP}(\sigma) = \{s \in \mathrm{Seq}(\sigma) : Hs, Bs \subseteq L(\sigma; \neg, \wedge, \vee, |, \supset)\}$.

For $S \subseteq \mathrm{GLP}(\sigma)$, we define the model operators

$$\begin{array}{rcl}\mathrm{Mod}_H(S) & = & \{I \in \boldsymbol{I}_H(\sigma) : I \models s, \text{ for all } s \in S\} \\ \mathrm{Mod}_m(S) & = & \mathrm{Min}(\mathrm{Mod}_H(S))\end{array}$$

The associated entailment relations are defined as follows:

$$S \models_* F \quad \text{iff} \quad \mathrm{Mod}_*(S) \subseteq \mathrm{Mod}(F)$$

where $* = H, m$, and $F \in L(\sigma)$.

Observation 3 *Let $S \subseteq \mathrm{GLP}$ be a generalized logic program. Then,*

$$\mathrm{Mod}_H(S) = \mathrm{Mod}_H([S])$$

where $[S]$ is the Herbrand instantiation of S.

With respect to a class of interpretations $\boldsymbol{K}$, we write $\boldsymbol{K} \models F$ iff $I \models F$ for all $I \in \boldsymbol{K}$. We denote the set of all sequents from a sequent set S which are applicable in $\boldsymbol{K}$ by

$$S_{\mathbf{K}} = \{s \in [S] : \boldsymbol{K} \models Bs\}$$

If $\boldsymbol{K}$ is a singleton, we omit brackets. A preferential semantics for sequents is given by a preferred model operator $\Phi : 2^{\mathrm{Seq}} \to 2^{\mathbf{I}_H}$, satisfying the condition $\Phi(S) \subseteq \mathrm{Mod}_H(S)$, and defining the preferential entailment relation

$$S \models_\Phi F \;\text{ iff }\; \Phi(S) \subseteq \mathrm{Mod}_H(F)$$

The question is: which preferred model operator captures the intended semantics for (generalized) logic programs ?

If we assume Mod_H as the model operator for the semantics of sequents then there is no difference between sequents and material implications, since for $F, G \in L$, it holds that

$$\mathrm{Mod}_H(F \Rightarrow G) = \mathrm{Mod}_H(F \supset G).$$

Which models of a set of sequents should be considered *intended* ? A sequent may have several meanings. The traditional meaning of a sequent s, based on classical model theory, is given by the formula $\bigwedge Bs \supset \bigvee Hs$ and the usual truth relation. Our intuitive understanding of rules suggests another meaning which interprets a sequent as a rule for generating information. We may consider a model I of a set S of sequents as intended if I can be generated bottom-up starting from zero information by an iterated application of the sequents $s \in [S]$. In other words, the intended models of S should be *grounded* in S. Furthermore, we want to have the property that every element $a \in I$ is supported by a sequent $s \in [S]$ satisfying $I \models Bs$. Below, we will make these ideas precise.

If a knowledge base contains sequents with non-persistent body formulas, it may have minimal models which are not grounded in the sense sketched above. Consequently, minimal model semantics does, in general, not guarantee groundedness. This is illustrated by the following example.

Example 1 *Let* $S_1 = \{\Rightarrow r(c);\ \Rightarrow p(a), p(b);\ \neg p(x) \Rightarrow q(x)\}$. *Every intended model of* S_1 *should contain* $q(c)$. *Assume* $I \models S_1$, *but* $q(c) \notin I$. *Then* $p(c) \in I$. *But* $p(c)$ *cannot be generated from below starting with* $\emptyset$ *by applying the sequents from* $[S_1]$ *because* $p(c)$ *does not appear in the head of any rule* $s \in [S_1]$. *But* $M_1 = \{r(c), p(a), p(b), p(c)\}$ *is a minimal model of* S_1. *Hence,* $S_1 \not\models_m q(c)$.

5 Stable Models

Definition 9 (Interpretation Interval) *Let* $I_1, I_2 \in \boldsymbol{I}_H$. *Then,* $[I_1, I_2] = \{I \in \boldsymbol{I}_H : I_1 \leq I \leq I_2\}$.

The following definition of a *stable generated model* is inspired by the definition of a *stable closure* of a set of rules in [Wag94]. In the sequel we assume the result that every consistent set of quantifier free formulas has a minimal Herbrand model, see [HJW95].

Definition 10 (Stable Generated Model) *Let* $S \subseteq \mathrm{GLP}(\sigma)$. *A model* M *of* S *is called* stable generated, *symbolically* $M \in \mathrm{Mod}_s(S)$, *if there is a chain of Herbrand interpretations* $I_0 \leq \ldots \leq I_\kappa$ *such that* $M = I_\kappa$, *and*

1. $I_0 = \emptyset$.
2. *For successor ordinals* α *with* $0 < \alpha \leq \kappa$, I_α *is a minimal extension of* $I_{\alpha-1}$ *satisfying the heads of all sequents whose bodies hold in* $[I_{\alpha-1}, M]$, *i.e.*
$$I_\alpha \in \mathrm{Min}\{I \in \boldsymbol{I}_H(\sigma) : I \geq I_{\alpha-1},\ \textit{and}\ I \models \bigvee Hs,\ \textit{f.a.}\ s \in S_{[I_{\alpha-1}, M]}\}$$
3. *For limit ordinals* $\lambda \leq \kappa$, $I_\lambda = \sup_{\alpha < \lambda} I_\alpha$.

We also say that M *is* generated *by the* S-stable chain $I_0 \leq \ldots \leq I_\kappa$.

In the sequel, we shall simply say 'stable' instead of 'stable generated' model.[3]

Example 2 *Let* $S = \{\Rightarrow e(0);\ \neg e(x) \Rightarrow o(x);\ o(x) \Rightarrow e(s(x))\}$ *represent a theory of even and odd numbers. Identifying successor terms with the corresponding natural number, we get the unique stable model* $M = \{e(0), o(1), e(2), o(3), \ldots\}$ *which is generated by the stable chain* $\emptyset \leq I_1 \leq M$ *with* $I_1 = \{e(0), o(1), o(3), o(5), \ldots\}$ *being a minimal extension of* $\emptyset$ *satisfying all heads of the ground sequent set*

$$S_{[\emptyset, M]} = \{\Rightarrow e(0)\} \cup \left\{ \begin{array}{c} \neg e(1) \Rightarrow o(1) \\ \neg e(3) \Rightarrow o(3) \\ \ldots \end{array} \right\}$$

It is easy to see that M *is a minimal extension of* I_1 *satisfying all heads of*

$$S_{[I_1, M]} = \{\Rightarrow e(0)\} \cup \left\{ \begin{array}{c} \neg e(1) \Rightarrow o(1) \\ \neg e(3) \Rightarrow o(3) \\ \ldots \end{array} \right\} \cup \left\{ \begin{array}{c} o(1) \Rightarrow e(2) \\ o(3) \Rightarrow e(4) \\ \ldots \end{array} \right\}$$

[3] Since Gelfond and Lifschitz defined stable models only for normal programs, and called their counterpart for disjunctive programs *answer sets*, there is no real name conflict.

Notice that our definition of stable models accommodates

1. Negation in the head of a rule, such as in

$$\Rightarrow \neg(\mathit{nationality}(x, \mathit{German}) \wedge \mathit{nationality}(x, \mathit{US})$$

   expressing the integrity constraint that it is not possible to have both the German and the US nationality.

2. Nested negations, such as in $p(x) \wedge \neg(q(x) \wedge \neg r(x)) \Rightarrow s(x)$ which would be the result of folding $p(x) \wedge \neg ab(x) \Rightarrow s(x)$ and $q(x) \wedge \neg r(x) \Rightarrow ab(x)$. The answer set semantics of [GL91] cannot deal with such rules.

The set of stable models of S is denoted by $\mathrm{Mod}_s(S)$, and the set of stable minimal models of S by $\mathrm{Mod}_{sm}(S) = \mathrm{Mod}_s(S) \cap \mathrm{Mod}_m(S)$. The resp. stable entailment relations are defined as follows:

$$S \models_* F \quad \text{iff} \quad \mathrm{Mod}_*(S) \subseteq \mathrm{Mod}(F)$$

where $* = s, sm$, and $F \in L(\sigma)$. With respect to stable models, sequents differ from material implication. For instance,

$$\mathrm{Mod}_s(\neg p \supset q) = \mathrm{Mod}_m(\neg p \supset q) = \{\{p\}, \{q\}\} \neq \mathrm{Mod}_s(\neg p \Rightarrow q) = \{\{q\}\}$$

Example 1 (continued) *Only the following two models of S_1 are stable*:

$$\begin{array}{rcl} M_2 & = & \{r(c), q(c), p(a), q(b)\} \\ M_3 & = & \{r(c), q(c), p(b), q(a)\} \end{array}$$

and hence, $S_1 \models_s q(c)$.

Stable models do not exist in all cases. For instance, $S = \{\neg p \Rightarrow p\}$ has exactly one minimal model, $\mathrm{Mod}_m(S) = \{\{p\}\}$, which is not stable, however. A sequent set, resp. logic program, without stable models will be called *unstable*.

Example 3 $S = \{p \supset q \Rightarrow r;\ r \Rightarrow p\}$ *is unstable.*

It is well-known that stable entailment is not cumulative, i.e. adding lemmas to programs may change their consequence set.

Example 4 *Disjunctive programs like the following, allowing for (possibly negated) conjunctions and disjunctions of literals in the body, are called* superprograms *in [Prz96].*

$$\begin{array}{rl} & \Rightarrow \mathit{VisitEurope} \vee \mathit{VisitAustralia} \\ \mathit{VisitEurope} \vee \mathit{VisitAustralia} & \Rightarrow \mathit{Happy} \\ \mathit{VisitEurope} \wedge \mathit{VisitAustralia} & \Rightarrow \mathit{Bankrupt} \\ \neg\ (\mathit{VisitEurope} \wedge \mathit{VisitAustralia}\ ) & \Rightarrow \mathit{Prudent} \\ \neg\ (\mathit{VisitEurope} \vee \mathit{VisitAustralia}) & \Rightarrow \mathit{Disappointed} \end{array}$$

In our framework this program is a generalized program without negation in heads. It has two stable models: { VisitEurope, Happy, Prudent }, { VisitAustralia, Happy, Prudent }. *Therefore, the following consequences are entailed:* Prudent, Happy, VisitAustralia $\vee$ VisitEurope, $\neg$ Bankrupt, $\neg$ Disappointed. *For this example our stable semantics gives the same results as the* static semantics *of [Prz96]. As opposed to the semantics of [Prz96], however, our stable semantics satisfies the distribution law* $\neg(K \wedge L) \equiv \neg K \vee \neg L$.

We show now that the length of the chain generating a stable model can be restricted by ω.

Claim 1 *Let* $S \subseteq$ GLP, *and let* M *be a stable model of* S *generated by the sequence* $M_0 \leq \ldots \leq M_\kappa$. *Then there is an ordinal* $\beta \leq \omega$ *such that* $M_\beta = M_\kappa$.

Proof: We show that every sequence stabilizes at an ordinal $\beta \leq \omega$. Obviously, if $M_\alpha = M_{\alpha+1}$ then $M_\alpha = M_\gamma$ for all $\alpha < \gamma \leq \kappa$. It is sufficient to prove $M_\omega = M_{\omega+1}$. We proceed in two steps:

(1) First we show that if $s \in [S]$ and $[M_\omega, M] \models Bs$ then there is a number $n < \omega$ such that $[M_n, M] \models Bs$. W.l.o.g. we may assume that Bs is a set of clauses (disjunction of ground literals), i.e. $Bs = \{C_1, \ldots, C_k\}$, $C_i = a^i_1 \vee \ldots \vee a^i_{m_i} \vee \neg b^i_1 \vee \ldots \neg b^i_{n_i}$, $i \in \{1, 2, ..., k\}$. A clause C_i is said to be positive if the set $P(i) := M_\omega \cap \{a^i_1, \ldots, a^i_{m_i}\}$ is nonempty, otherwise it is called negative. Let $\{C_1, \ldots, C_s\}$ be the set of positive and $\{C_{s+1}, \ldots, C_k\}$ the set of negative clauses. Because the set $P := \bigcup_{1 \leq i \leq s} P(i)$ is finite, there is a number $j < \omega$ such that $P \subseteq M_j$. Then, trivially, $[M_j, M] \models C_1, \ldots, C_s$. It remains to show: if $M_j \subseteq J \subseteq M$ then $J \models C_{s+1}, \ldots, C_k$. Assume there is a number j, $s+1 \leq j \leq k$, such that $J \not\models C_j$, $C_j = a^j_1 \vee \ldots \vee a^j_{m_j} \vee \neg b^j_1 \vee \ldots \vee \neg b^j_{n_j}$, then $J \models \neg a^j_1 \wedge \ldots \neg a^j_{m_j} \wedge b^j_1 \ldots \wedge b^j_{n_j}$. We may assume that the elements in the set $\{a^j_1, \ldots, a^j_{m_j}, b^j_1, \ldots, b^j_{n_j}\}$ are pairwise distinct. Now, let $M^\star_\omega = M_\omega \cup \{b^j_1, \ldots, b^j_{n_j}\}$; then $M_\omega \subseteq M^\star_\omega \subseteq M$. Obviously, $M^\star_\omega \not\models C_j$, since $M^\star_\omega \cap \{a^j_1, \ldots, a^j_{m_j}\} = \emptyset$ and $\{b^j_1, \ldots, b^j_{n_j}\} \subseteq M^\star_\omega$. This is a contradiction to $[M_\omega, M] \models C_j$.

(2) Now, we show that $M_\omega = M_{\omega+1}$. It is sufficient to prove: if $s \in S_{[M_\omega, M]}$, then $M_\omega \models \bigvee Hs$. By (1), the condition $s \in S_{[M_\omega, M]}$ implies that $s \in S_{[M_n, M]}$ for a certain number $n < \omega$, and hence for every $j > n : s \in S_{[M_j, M]}$. Hence, $M_j \models \bigvee Hs$ for every j, $n < j < \omega$. Again, we may assume that $\bigvee Hs$ is given as a set of clauses $\{C_1, \ldots, C_n\}$. We have to check that $M_\omega \models C_1, \ldots, C_n$. Assume, there is a j, $1 \leq j \leq n$, such that $M_\omega \not\models C_j$, then $M_\omega \models \neg C_j$, $C_j = a^j_1 \vee \ldots a^j_{m_j} \vee \neg b^j_1 \vee \ldots \neg b^j_{n_j}$, and $M_\omega \models \neg a^j_1 \wedge \ldots \neg a^j_{m_j} \wedge b^j_1 \ldots \wedge b^j_{n_j}$. It is easy to show that there exists a number $m < \omega$ such that $\{b^j_1, \ldots, b^j_{n_j}\} \subseteq M_m$, and from this follows $M_m \not\models C_j$, which is a contradiction. $\Box$.

Corollary 2 *If* M *is a stable model of* $S \subseteq$ GLP, *then there is either a finite* S*-stable chain, or a* S*-stable chain of length* ω, *generating* M.

Definition 11 (Persistent Sequent) *A sequent is called* persistent, *if all body formulas* $F \in Bs$ *are persistent.*

For instance, all sequents from PDLP are persistent.

Claim 3 *Minimal models of positive disjunctive logic programs are stable: if* $S \subseteq \mathrm{PDLP}$, *then* $\mathrm{Mod}_m(S) \subseteq \mathrm{Mod}_s(S)$.

Proof: Let $M \in \mathrm{Mod}_m(S)$. We construct a sequence $I_0, I_1, \ldots$ such that
(1) $I_0 = \emptyset$,
(2) $I_{\alpha+1} \in \mathrm{Min}\{I : I_\alpha \leq I \leq M \quad \text{and} \quad I \models \bigvee Hs \quad \text{for all} \quad s \in S_{I_\alpha}\}$,
(3) $I_\lambda = \bigcup_{\alpha<\lambda} I_\alpha$ for limit ordinals λ.
Since body formulas are persistent we have $s \in S_{I_\alpha}$ iff $s \in S_{[I_\alpha, M]}$; hence $(I_\alpha)_{\alpha=0,1,\ldots}$ satisfies the stable extension condition 2 of the definition of stable models. Obviously, there is an ordinal α such that $I_\alpha = I_{\alpha+1}$. We show that I_α is a model of S. Let $s \in [S]$ and $I_\alpha \models Bs$, then by definition $I_{\alpha+1} \models \bigvee Hs$, and hence $I_\alpha \models \bigvee Hs$. Since M is a minimal model, it follows finally that $I_\alpha = M$. □

Corollary 4 *Minimal models of a set of persistent sequents are stable.*

Proof: the proof of claim 3 depends only on the persistence of body formulas. □

Thus, minimal models provide an adequate semantics for persistent programs. It holds that $\mathrm{Mod}_{sm}(S) = \mathrm{Mod}_m(S)$, whenever S is persistent. The converse of claim 3, however, that stable models of positive disjunctive programs are minimal, ist not true as the following counterexample shows.

Example 5 *Let* $S = \{\Rightarrow a, b;\ a \Rightarrow b\}$. *Then* $M = \{a, b\}$ *is not minimal since* $\{b\}$ *is a model of* S. *But* $\{a, b\}$ *is stable:* $I_0 = \emptyset$, $S_{[\emptyset, \{a,b\}]} = \{\Rightarrow a, b\}$; *and since* $\{a\} \in \mathrm{Min}\{I : \emptyset \leq I \leq M, I \models a \vee b\}$, *we obtain* $S_{[\{a\},\{a,b\}]} = \{\Rightarrow a, b;\ a \Rightarrow b\}$. *Obviously,* $\{a, b\}$ *is a minimal extension of* $\{a\}$ *satisfying* $a \vee b$ *and* b.

This raises the following question.

Question 1 *Is it really clear that the intended models of a logic program have to be minimal ? Don't we rather want to have the stable groundedness of a model and not necessarily its minimality ?*

The answer to this question decides whether we should choose Mod_s or Mod_{sm} as our preferred model operator.

Definition 12 (Least Model Property) *A set* X *of quantifier free formulas has the* least model property *(l.m.p.) if for every set* A *of ground atoms consistent with* X *the set* $X \cup A$ *has a least model. A quantifier free formula* F, *respectively a sequent* $s \in GLP$, *has the l.m.p if* $\{F\}$, *respectively* $\{\bigvee Hs\}$, *has it.*

In particular, all sequents with atomic heads, such as the rules of normal logic programs have the least model property.

Claim 5 *Let* X *be a set of quantifier free formulas and assume that every* $F \in X$ *has the least model property. Then* X *has the least model property.*

Proof: Let A be a set of ground atoms such that $X \cup A$ is consistent. Then, for every $F \in X$ the set $A \cup \{F\}$ has a least model denoted by $least(A, F)$. We define a sequence $I_0 \leq I_1 \leq \ldots \leq I_n \ldots$ of Herbrand interpretations as follows: $I_0 = \bigcup\{least(A, F) : F \in X\}$, $I_{n+1} = \bigcup\{least(I_n, F) : F \in X\}$, $n < \omega$. Let be $I = \bigcup_{n<\omega} I_n$. We show that I is the least model of $A \cup X$.

I is a model of $A \cup X$. We have to show that every $F \in X$ is true in I. Let $I_0(F) = least(A, \{F\})$, and $I_{n+1}(F) = least(I_n, \{F\})$, $n < \omega$. By construction, $I = \bigcup_{n<\omega} I_n(F)$. Analogously as in the proof of proposition 20 in [HJW95] it can be shown that $\bigcup_{n<\omega} I_n(F) \models F$, hence $I \models F$.

Finally, we show that I is the least model of $A \cup X$. Let $J \subseteq I$ a subset being a model of $A \cup X$. It is easy to show that $I_n(F) \subseteq J$ for every $F \in X$, $n < \omega$. Hence, $J = I$. $\Box$

In the proof of claim 6 we use the following facts. Let X, Y be sets of quantifier free formulas satisfying the l.m.p. and $X \subseteq Y$. Then $least(X) \subseteq least(Y)$. This follows immediately from the following result in [HJW95]:

Let X be a set of quantifier free formulas and p a ground atom. Then $X \models_H p$ if and only if $p \in \bigcap \mathrm{Mod}_H(X)$. Furthermore, if X has a least model then $least(X) = \bigcap \mathrm{Mod}_H(X)$.

Claim 6 *If a generalized logic program contains only sequents satisfying the least model property, then every stable model of it is also a minimal model.*

Proof: Let S be a set of sequents satisfying the l.m.p., and M be a stable model of S generated by $I_0, I_1, \ldots, I_n, \ldots$, $n < \omega$, i.e.: $I_0 = \emptyset$, $I_{n+1} = \min\{I : M \geq I \geq I_n$, and $I \models \mathrm{Heads}(S_{[I_n,M]})\}$, where $\mathrm{Heads}(X) = \{\bigvee Hs : s \in X\}$.

We show: if $J \subseteq M$, and $J \models S$ then $J = M$. We construct a sequence $J_0 \leq J_1 \leq \ldots \leq J_n \ldots$ such that $I_n \subseteq J_n$, and $J_n \subseteq J$, by setting $J_{n+1} = \min\{K : J \geq K \geq J_n$, and $K \models \mathrm{Heads}(S_{[J_n,J]})\}$. Assume $I_n \subseteq J_n$. Then, $S_{[I_n,M]} \subseteq S_{[J_n,J]}$, implying $\mathrm{Heads}(S_{[I_n,M]}) \subseteq \mathrm{Heads}(S_{[J_n,J]})$. By claim 5, I_{n+1} is the least extension of I_n satisfying $\mathrm{Heads}(S_{[I_n,M]})$, and, using the preceding remarks, we get $I_{n+1} \subseteq J_{n+1}$. Consequently, $J = M$. $\Box$

6 Stable Models of Normal Logic Programs

For $B \subseteq \mathrm{Lit}$, let B^- denote the set of atoms which occur negated in B, i.e. $B^- := \{a \in \mathrm{At} : \neg a \in B\}$, and let $B^+ = \{a \in \mathrm{At} : a \in B\}$. It holds that for any $B \subseteq \mathrm{Lit}^0$, and any $I \in \boldsymbol{I}_H$,

$$I \models B \quad \text{iff} \quad B^+ \subseteq I \;\&\; B^- \cap I = \emptyset$$

Definition 13 (Immediate Consequence Operator) *Let* $S \subseteq \mathrm{NLP}$, *and* $I \subseteq \mathrm{At}$. *Then,*

$$T_S(I) = \{a \in \mathrm{At}^0 : \textit{ there exists } (B \Rightarrow a) \in [S], \textit{ s.th. } I \models B\}$$

is called the immediate consequence operator *associated with* S.

Definition 14 (Gelfond-Lifschitz 1988) *Let* $I \subseteq \mathrm{At}$, *and* $S \subseteq \mathrm{NLP}$. *Then the* Gelfond-Lifschitz transformation *of* S *with respect to* I *is defined as*

$$S^I = \{B^+ \Rightarrow a : (B \Rightarrow a) \in [S],\ and\ B^- \cap I = \emptyset\}$$

and the Gelfond-Lifschitz operator Γ_S *is defined as follows:* $\Gamma_S(I)$ *is the unique minimal model of* S^I, *denoted* M_{S^I}, *i.e.* $\Gamma_S(I) = M_{S^I}$. *We also define the fixpoint set* $\mathrm{Fix}(\Gamma_S) = \{I : I = \Gamma_S(I)\}$.

Note that $S^I \subseteq \mathrm{PLP}$, and consequently S^I has a unique minimal model, $\mathrm{Mod}_m(S^I) = \{M_{S^I}\}$.

We shall show below that the stable models of a normal logic program S in our sense agree with the fixpoints of Γ_S, i.e. with stable models as defined in [GL88]. Since the definition of the Gelfond-Lifschitz transformation requires a specific rule syntax (in order to partition the body into a positive and a negative part), the definition of stable models based on it is not very general; as a consequence, Gelfond and Lifschitz are not able to treat negation-as-failure as a standard connective, and to allow for arbitrary formulas in the body of a rule. The interpretation of negation-as-failure according to our stable semantics seems to be the first general standard logical treatment of nonmonotonic logic programs.[4]

In the sequel we use the following simple observation.

Observation 4 *Let* M *be an arbitrary Herbrand interpretation,* $I \subseteq M$, *and* $F = a_1 \wedge \ldots \wedge a_m \wedge \neg b_1 \wedge \ldots \neg b_n$. *Then* $[I, M] \models F$ *iff* $I \models a_1 \wedge \ldots \wedge a_m$ *and* $M \cap \{b_1, \ldots, b_n\} = \emptyset$.

Claim 7 *Let* $S \subseteq \mathrm{NLP}$. *Every fixpoint of* Γ_S *is a stable model of* S, *i.e.* $\mathrm{Fix}(\Gamma_S) \subseteq \mathrm{Mod}_s(S)$.

Proof: Let M be a fixpoint of Γ_S, i.e. $\mathrm{Mod}_m(S^M) = \{M\}$. Using the immediate consequence operator associated with S^M we construct the sequence $\{I_n\}_{n<\omega}$, such that $I_0 = \emptyset$, $I_{n+1} = T_{S^M}(I_n)$, and $I_\omega = \bigcup_{n<\omega} I_n$. It is well-known that $I_\omega = M$. We show that $I_0 \leq I_1 \leq \ldots \leq I_n \ldots$ is a S-stable chain, i.e. $I_{n+1} \in \mathrm{Min}\{I : I_n \leq I \leq M, \text{ and f.a. } s \in S_{[I_n, M]}, I \models \bigvee Hs\}$. This follows from the fact that for arbitrary normal sequents $B \Rightarrow H$ it holds that $[I_n, M] \models B$ iff $I_n \models B^+$ and $B^- \cap M = \emptyset$. Hence M is stable. □

Claim 8 *Every stable model of* $S \subseteq \mathrm{NLP}$ *is a fixpoint of* Γ_S: $\mathrm{Mod}_s(S) \subseteq \mathrm{Fix}(\Gamma_S)$.

Proof: Let $M \in \mathrm{Mod}_s(S)$. We consider again the sequence generated by the iteration of T_{S^M}: $I_0 = \emptyset, I_{n+1} = T_{S^M}(I_n)$, and $I_\omega = \bigcup_{n<\omega} I_n$. We have to show that $M = I_\omega$.

[4] There have been many meta-logical (notably modal logic) proposals, though.

($I_\omega \subseteq M$): Let $a \in I_\omega$, then there is a sequent $s' = B' \Rightarrow a$ in S^M, such that for some $n < \omega$, $I_n \models B'$, and $a \in I_{n+1}$. By definition, the reexpansion s of s', $s = B \Rightarrow a$, such that $B^+ = B'$ and $B^- \cap M = \emptyset$, is in $[S]$. As an induction hypothesis, we assume that $I_n \subseteq M$, implying that $M \models B'$, and consequently also $M \models B$. Hence, $a \in M$, proving that $I_{n+1} \subseteq M$.

($M \subseteq I_\omega$): Let M be generated by the S-stable chain $(M_i)_{i<\omega}$. We show by induction on n that $M_n \subseteq I_n$. For $n = 0$, this is trivially the case. Let $a \in M_n$. Since by definition, M_n is the least extension of M_{n-1} satisfying $\mathrm{Heads}(S_{[M_{n-1},M]})$, there is a sequent $B \Rightarrow a$ in $[S]$, such that $[M_{n-1}, M] \models B$. This implies that $B^+ \Rightarrow a$ is in S^M. By the induction hypothesis, $M_{n-1} \subseteq I_{n-1}$, implying $I_{n-1} \models B^+$, and consequently $a \in I_n$. □

7 Stable Models of Disjunctive Logic Programs

The fixpoint-based definition of stable models for normal programs of [GL88] was generalized to normal disjunctive programs in [Prz91] and to extended disjunctive programs, which subsume normal disjunctive ones, in [GL91].

Definition 15 *Let $I \subseteq \mathrm{At}$, and $S \subseteq \mathrm{NDLP}$. Then the* Gelfond-Lifschitz transformation *of S with respect to I is defined as*

$$S^I = \{B^+ \Rightarrow H : (B \Rightarrow H) \in [S], \textit{ and } B^- \cap I = \emptyset\}$$

and the generalized Gelfond-Lifschitz operator *Γ_S is defined as follows: $\Gamma_S(I)$ collects all minimal models of S^I, i.e. $\Gamma_S(I) = \mathrm{Mod}_m(S^I)$. We also define the fixpoint set* $\mathrm{Fix}(\Gamma_S) = \{I : I \in \Gamma_S(I)\}$.

Note that $S^I \subseteq \mathrm{PDLP}$, and consequently the minimal models of S^I are the intended ones.

Claim 9 *Let $S \subseteq \mathrm{NDLP}$. Every stable model M of S is also a supported model of S, i.e. for all $a \in M$ exists a sequent $s \in [S]$, such that $a \in Hs$, and $M \models Bs$.*

Proof: Let M be a stable model of S and $(M_i)_{i<\omega}$ a S-stable chain generating M. Let $a \in M$ be given, then there is a number $i < \omega$ such that $a \in M_{i+1} - M_i$. Since M_{i+1} is a minimal extension of M_i satisfying the set $\mathrm{Heads}(S_{[M_i,M]})$ there is a sequent $s \in [S]$ such that $[M_i, M] \models B_s$ and $a \in Hs$. □

We shall now investigate the relationship between the fixpoints of Γ_S, i.e. the answer sets of a normal disjunctive logic program S as defined in [GL91], and the stable models of S.

Claim 10 *Let $S \subseteq \mathrm{NDLP}$. Every fixpoint of Γ_S is a stable minimal model of S, i.e. $\mathrm{Fix}(\Gamma_S) \subseteq \mathrm{Mod}_{sm}(S)$.*

Proof: Let $M \in \mathrm{Mod}_m(S^M)$. By claim 3, M is a stable model of S^M, i.e. there exists a sequence $\emptyset = I_0 \leq I_1 \leq \ldots \leq I_n \leq \ldots$, such that $M = \bigcup_{n<\omega} I_n$, and $I_{n+1} \in \mathrm{Min}\{I' \in \boldsymbol{I}_H \mid I_n \leq I' \leq M$, and f.a $s \in S^M_{I_n} : I' \models \bigvee Hs)\}$. We show that $(I_n)_{n<\omega}$ is a S-stable chain generating M. For this purpose it is sufficient to prove that $I_{n+1} \in \mathrm{Min}\{I' \in \boldsymbol{I}_H \mid I_n \leq I' \leq M$, and f.a. $s \in S_{[I_n,M]} : I' \models \bigvee Hs)\}$. This follows from the fact that $\mathrm{Heads}(S_{[I_n,M]}) = \mathrm{Heads}(S^M_{I_n})$.
We finally show, that M is a minimal model of S. Assume $J \subseteq M$ and $J \models S$. It is sufficient to prove that J is a model of S^M, then, by minimality of M w.r.t. S^M this implies $J = M$. Let be $s' \in S^M$, $s' = B' \Rightarrow H$, and $J \models B'$. By the definition of S^M, there is a sequence $s := B \Rightarrow H \in [S]$ such that $B^+ = B'$, and $B^- \cap M = \emptyset$. Obviously, $J \models B$, and since J is a model of S, we get $J \models \bigvee H$. Hence, $J \models s'$. This shows, that J is a model of S^M. $\Box$.

The following example shows that the converse of this claim is not true, i.e. not every stable minimal model of a NDLP S is a fixpoint of Γ_S.

Example 6 *Let $S = \{\Rightarrow a, b; a \Rightarrow b; \neg a \Rightarrow a\}$. We show that $\{a, b\}$ is a stable minimal model of S. Obviously, $\{a, b\}$ is a minimal model of S. $\{a, b\}$ is also stable: $I_0 = \emptyset$, $S_{[\emptyset,\{a,b\}]} = \{\Rightarrow a, b\}$. A minimal extension of $\emptyset$ is either $\{a\}$ or $\{b\}$. Take $I_1 = \{a\}$. Then $S_{[\{a\},\{a,b\}]} = \{a \Rightarrow b, \Rightarrow a, b\}$, and a minimal extension of $\{a\}$ satisfying b gives finally $I_2 = \{a, b\}$. On the other hand, $\{a, b\}$ is not a minimal model of $S^{\{a,b\}} = \{\Rightarrow a, b; a \Rightarrow b\}$, since $\{b\}$ is a model of $S^{\{a,b\}}$. In fact, Γ_S does not have any fixpoint.*

Question 2 *Is it possible to characterize the stable models M of S as particular models of S^M ?*

The following conjecture was suggested to us by Michael Gelfond in personal communication.

Conjecture 1 *If $S \subseteq$ NDLP is head cycle free, then $\mathrm{Mod}_s(S) = \mathrm{Fix}(\Gamma_S)$.*

8 Conclusion

By introducing a new general definition of stable models, we have established the foundation of a *stable model theory* of logic programs which will render it possible to analyze further extensions of normal logic programs, such as quantifiers in bodies and heads of rules.

Acknowledgements: Stimulating discussions with Michael Gelfond and Teodor Przymusinski on the nature of stable models and answer sets have helped us to improve our understanding and the quality of the paper. We are also grateful to the referees who pointed out several weaknesses of the first draft of the paper.

A Stable Models of S5 Logic Programs

In this appendix we show that our definition of stable models is a very general preferential model concept which applies to various logical systems provided that there is a 'natural' information ordering between interpretations.

In order to be able to distinguish between 'definite failure' and 'epistemic failure', Gelfond [Gel91] has proposed to extend the language of logic programs by adding a *belief operator*. This can be motivated by a simple example. Assume that either John or Peter is guilty: $g(J) \vee g(P)$ (1). But since nothing more is known about them, both have to be presumed innocent. The predicate *presumed innocent*, however, cannot be defined by $\neg g(x) \Rightarrow pi(x)$, because then, by the failure of definite failure, neither John nor Peter would be presumed innocent.

Using S5 interpretations $E \subseteq 2^{\mathrm{At}}$, their natural information ordering[5]

$$E_1 \leq E_2 \ :\Leftrightarrow\ \forall I_2 \in E_2 \exists I_1 \in E_1 : I_1 \subseteq I_2$$

and the satisfaction relation

$$E \models F \ :\Leftrightarrow\ \forall I \in E : E, I \models F$$

where $E, I \models F$ is defined as in standard S5 Kripke semantics (i.e. $E, I \models \mathsf{B}F$ iff $E, I' \models F$ for all $I' \in E$), we can define *presumed innocent* as

$$\neg \mathsf{B}\, g(x) \Rightarrow pi(x) \quad (2)$$

The resulting S5 logic program Φ, consisting of (1) and (2), has a unique stable S5-model:

$$E_1 = \{\{g(J), pi(J), pi(P)\}, \{g(P), pi(J), pi(P)\}\}$$

and hence, $\Phi \models_s pi(J) \wedge pi(P)$. It is left to the reader to check that the two other minimal S5 models of Φ, $E_2 = \{\{g(J), pi(P)\}\}$, and $E_3 = \{\{g(P), pi(J)\}\}$, are not generated by a Φ-stable chain.

References

[Gel91] M. Gelfond. Strong introspection. *Proc. of AAAI-91.*

[GL88] M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. In R. A. Kowalski and K. A. Bowen, editors, *Proc. of ICLP*, pages 1070–1080. MIT Press, 1988.

[GL90] M. Gelfond and V. Lifschitz. Logic programs with classical negation. *Proc. of ICLP'90*, pages 579–597. MIT Press, 1990.

[5] Strictly speaking, we have to restrict the admissible interpretations $E \subseteq 2^{\mathrm{At}}$ to *canonical* S5 interpretations for which $E = \mathrm{Min}(E)$. Otherwise the information ordering $\leq$ would only be a preorder but not a partial ordering as required.

[GL91] M. Gelfond and V. Lifschitz. Classical negation in logic programs and disjunctive databases. *New Generation Computing*, 9:365–385, 1991.

[HJW95] H. Herre, J. Jaspars and G. Wagner. Partial logics with two kinds of negation as a foundation for knowledge-based reasoning. Technical report, CWI, Amsterdam, 1995. To appear in D. Gabbay and H. Wansing (Eds.), *What is negation ?*, Oxford University Press.

[HW96] H. Herre and G. Wagner. Stable and grounded models of extended generalized logic programs. Manuscript, 1996, submitted.

[LW92] V. Lifschitz and T.C. Woo. Answer Sets in General Nonmonotonic Reasoning, *Proc. KR'92*, Morgan-Kaufmann, 1992, 603–614.

[Prz91] T.C. Przymusinski. Stable semantics for disjunctive programs. *New Generation Computing*, 9:401–424, 1991.

[Prz96] T.C. Przymusinski. Super Logic Programs and Negation as Belief. In: R. Dyckhoff, H. Herre, P. Schroeder-Heister, editors, *Proc. of the 5th Int. Workshop on Extensions of Logic Programming*, Springer LNAI 1050, 229–236

[Wag94] G. Wagner. Transforming deductive into active databases. In N. Fuchs and G. Gottlob, editors, *Proc. of the 10th Workshop on Logic Programming*. Univ. Zürich, 1994.