Universidade Nova de Lisboa
Faculdade de Ciências e Tecnologia
Departamento de Informática

Technische Universität Dresden
Fakultät Informatik

A Comparative Study of Disjunctive Well-Founded Semantics

Matthias Knorr

Master Thesis in
Computational Logic

Dissertação apresentada na Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa para obtenção do grau de Mestre em Lógica Computacional

supervision:
Prof. Dr. Reinhard Kahle
Prof. Dr. Steffen Hölldobler

Lisboa
2006

Abstract

A lot of different proposals were made claiming to extend the well-founded semantics to logic programs with disjunction. None of these however has been established as 'the' disjunctive well-founded semantics which is mainly caused by the different intuitive ideas on which these semantics are based. In fact, some of them are even incomparable with respect to their derivable knowledge, and systematic comparisons are rather rarely done.

A recently introduced framework for characterizing logic programming semantics was quite successfully used for comparing the major semantics for normal logic programs. We are going to extend this framework to disjunctive logic programs and will present alternative characterizations for three semantics for disjunctive logic programs, namely the strong well-founded semantics (SWFS), the generalized disjunctive well-founded semantics (GDWFS), and the disjunctive well-founded semantics (D-WFS). We will see that this cannot be done in a straightforward way. Whereas the derivation of positive information is more or less uniform, obtained negative information differs and some of the constructions even show the limits of the applied framework. We will also mention the difficulties when characterizing two major two-valued semantics for disjunctive logic programs, minimal models and disjunctive stable models.

Acknowledgment

This work has been supported by the German Academic Exchange Service (DAAD) and by the 'Gesellschaft von Freunden und Förderern der Technischen Universität Dresden e.V.'.

Contents

1 Introduction

During the last thirty years a lot of semantics have been defined in the field of logic programming covering a large variety of different kinds of programs. For normal programs, i.e. programs consisting of Horn clauses extended by default negation in the body, two approaches are nowadays considered to be the most important ones. Stable model semantics ([14]) is the main two-valued approach allowing the truth values true and false whereas the major three-valued semantics allowing an additional third truth value undefined is the well-founded semantics ([34]). Both semantics are closely related as shown in [33]. While default negation assumes everything false which cannot proven to be true, explicit negation only allows to draw this conclusion if there is evidence given in the program. For these generalized logic programs, answer set programming ([19]), the well-founded semantics with explicite negation WFSX ([2]) respectively, is the main two-valued, respectively three-valued, semantics.

Another possible extension of normal logic programs is the occurrence of indefinite information in the programs represented by disjunctions. There is a huge amount of proposals for semantics of disjunctive logic programs caused by the different interpretions of a disjunction and the desired reasoning capabilities. Disjunctions may be interpreted exclusively, meaning that as few as possible disjuncts are true, or inclusively where we always consider the possibility that all disjuncts might be true. It would be far beyond the scope to mention all the introduced semantics and their deduction abilities. Instead we refer to [21] and [24] for an overview. While various proposals were mainly initiated by intuitive considerations, some researchers tried to extend the main semantics for (definite) logic programs to disjunctive ones, and minimal models ([22]) and disjunctive stable models ([26]) are straightforward extensions of the least and the stable model semantics. Unfortunately, up to now there is no semantics established to be the extension of the well-founded semantics although several proposals exist. In [29], Ross introduced the strong well-founded semantics (SWFS) based on a top-down procedure using derivation trees. The generalized disjunctive well-founded semantics (GDWFS) was introduced by Baral, Lobo, and Minker in [3], built on several bottom-up operators and the extended generalized closed world assumption ([37]). Brass and Dix proposed the disjunctive well-founded semantics (D-WFS) in [6] based on two operators iterating over conditional facts. In [35], Wang introduced the well-founded disjunctive semantics (WFDS) based on an abduction framework and quite recently, in [1], the well-founded semantics with disjunction (WFS_d) was proposed by Alcântara, Damásio, and Pereira defined by means of an operator on sets of interpretations. However, since the constructions on which these semantics are based differ so much, it is rather difficult to compare them.

In [10], Dix and Müller presented a framework for comparing the semantics of disjunctive programs. But this framework alone is not sufficient since it only provides certain properties which should be satisfied by a semantics for disjunctive logic programs. Recently, a methodology for uniformly characterizing semantics has been proposed in [17]. It uses level mappings which allow for describing syntactic and semantic dependencies in logic programs. This results in characterizations allowing easy comparisons of the corresponding semantics. With the introduction of the framework, normal logic programs have been studied and compared in [17] and [15], and in [16] the level-mapping approach has been compared with the meta-theory of selector generated semantics ([30]).

We will present level mapping characterizations for three of the previously mentioned approaches which aim at extending the well-founded semantics to disjunctive logic programs, namely SWFS, GDWFS, and D-WFS. We proceed as follows. In Section 2, we present basic notions and recall the stable models semantics and the well-founded semantics including their characterizations by means of level mappings. Then we devote one section to each of the three semantics first recalling the approach itself and then presenting the level mapping characterization including a proof of the equivalence. We start with SWFS in Section 3, continue with GDWFS in Section 4 and end with D-WFS in Section 5. After that, in Section 6 we recall results from older comparisons of these three semantics and we compare the characterizations looking for common conditions which might be further properties for an appropriate well-founded semantics for disjunctive programs. We also mention the difficulties we had with a characterization of minimal models which are used in case of GDWFS. We conclude with Section 7 and point out further work.

2 General Notions and Preliminaries

2.1 Terminology

A *disjunctive logic program* Π consists of finitely many universally quantified *clauses* of the form

$$H_1 \vee \cdots \vee H_l \leftarrow A_1 \wedge \cdots \wedge A_n \wedge \neg B_1 \wedge \cdots \wedge \neg B_m$$

usually written as

$$H_1 \vee \cdots \vee H_l \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$$

where H_k, A_i, and B_j, for $k = 1, \ldots, l$, $i = 1, \ldots, n$, and $j = 1, \ldots, m$, are atoms of a given first order language, consisting of predicate symbols, function symbols, constants and variables, and the symbol $\neg$ is representing default negation. An *atom* is of the form $p(t_1, \ldots, t_n)$, where p is a n-nary predicate symbol and the t_i, for $i = 1, \ldots, n$ and $n \geq 0$, are terms. A *term* is a constant, a variable or a compound term $f(t_1, \ldots, t_m)$ where f is a function symbol of arity m with m terms as arguments and is, for $m = 0$, equivalent to a constant. A clause can be divided into the *head* $H_1 \vee \cdots \vee H_l$ and the *body* $A_1, \ldots, A_n$, $\neg B_1, \cdots, \neg B_m$. If the body is empty, the clause is called a *fact*.
We also identify the elements in the head and in the body with the following sets $\mathcal{H} = \{H_1, \ldots, H_l\}$, $\mathcal{A} = \{A_1, \ldots, A_n\}$, and $\neg\mathcal{B} = \{\neg B_1, \ldots, \neg B_m\}$, where $\mathcal{B} = \{B_1, \ldots, B_m\}$, and allow alternatively to abbreviate clauses by $\mathcal{H} \leftarrow \mathcal{A} \wedge \neg \mathcal{B}$, where $\mathcal{H}$, $\mathcal{A}$, and $\mathcal{B}$ are each sets of pairwise distinct atoms. Likewise, for a given a disjunction D and a conjunction C, we use $\mathcal{D}$ and $\mathcal{C}$ for denoting the set of all atoms occurring in D, in C respectively. This alternative representation also exemplifies that the disjunction in the head and the conjunction in the body are commutative, i.e. the order of the elements in the head and in the body does not matter. In a *normal clause* there is exactly only one atom occurring in $\mathcal{H}$ and we call a program consisting only of normal clauses a *normal (logic) program*. If additionally $\mathcal{B}$ is empty then the corresponding clause is called *definite*. A program just consisting of definite clauses is a *definite (logic) program*, respectively a *definite disjunctive (logic) program* if there is a clause with a head containing more than one atom. We usually denote disjunction-free programs by P to distinguish from disjunctive programs which are usually represented by Π.

Any expression, i.e. term, atom, disjunction, conjunction, clause and program, is called *ground* if it contains no variables. The *Herbrand Universe* U_Π is the set of all ground terms that can be formed from the constants and function symbols occurring in Π. If there is no constant in Π then an arbitrary constant is added. The *Herbrand base* B_Π is the set of all ground atoms that can be formed by using the predicate symbols from Π with terms as arguments from U_Π. A *literal* is either a *positive literal*, respectively an atom, or a *negative literal*, a negated atom, and usually we denote by $A, B, \ldots$ atoms and by $L, M, \ldots$ literals. Moreover, we use the notion disjunctive literal for both, disjunctions and negated disjunctions. The *extended Herbrand base* EB_Π is the set of all disjunctions that can be formed using pairwise distinct atoms from B_Π. The *conjunctive Herbrand base* CB_Π is the set of all conjunctions that can be formed using pairwise distinct atoms from B_Π. Finally, the set of all ground instances of the clauses of a program Π with respect to B_Π is called $\mathsf{ground}(\Pi)$. It is important to note that all these notions may not be finite because of the possible occurrence of non-nullary function symbols, even though the program itself is only a finite set of (non-ground) clauses.

2.2 Mathematical Notions

We are going to recall now some mathematical theory from [31] which we will need in the following sections and we start with binary relations and some properties of them.

Definition 2.1 Let X and Y be given sets. A *binary relation* R is a subset of $X \times Y$. If $R \subseteq X \times X$ then R is a binary relation *over* X.

A function is then just a special binary relation.

Definition 2.2 Let $f \subseteq X \times Y$ be a binary relation such that for each $x \in X$ there is exactly one $f(x) = y \in Y$. Then f is called a (total) *function*, also denoted $f : X \to Y$.

Thus, functions also may satisfy the general properties of binary relations which we recall in the following.

Definition 2.3 Let $R \subseteq X \times Y$, i.e. be a binary relation.

- injective: $\forall x, z \in X$, $\forall y \in Y$: $xRy \wedge zRy \to x = z$
- surjective: $\forall y \in Y$: $\exists x \in X$: xRy
- bijective: R is surjective and injective

Example 2.4 Let $X = \{0, 1, 2, 3\}$, $Y = \{4, 5, 6\}$, $R = \{(0, 4), (1, 5), (2, 6)\}$. Then R is injective and surjective, but not a function since there is no value $R(3)$. The relation $R_1 = R \cup \{(3, 5)\}$ now is a function and surjective but not injective because we have $1R_15$ and $3R_15$.

Now we restrict the interest to binary relations where $X = Y$.

Definition 2.5 Given a binary relation R over X we define the following properties.

- reflexive: $\forall x \in X \colon xRx$
- symmetric: $\forall x, y \in X \colon xRy \to yRx$
- antisymmetric: $\forall x, y \in X \colon xRy \wedge yRx \to x = y$
- transitive: $\forall x, y, z \in X \colon xRy \wedge yRz \to xRz$
- linear: $\forall x, y \in X \colon xRy \vee yRx$

These properties allow us to define equivalence relations.

Definition 2.6 An *equivalence relation* is a binary relation over some set X which is reflexive, symmetric, and transitive.

Equivalence relations can be used to define equivalence classes.

Definition 2.7 Given a set X and an equivalence relation R over X we define the *equivalence class* $[c]$ of an element c to be all elements of X which are equivalent to c, i.e. $[c] = \{x \in X \mid xRc\}$.

Example 2.8 Let $X = \{0, 1, 2\}$ and $R = \{(0, 0), (1, 1), (2, 2), (1, 2), (2, 1)\}$. Then R is an equivalence relation and the equivalence classes are $[0] = \{0\}$ and $[1] = \{1, 2\}$.

The properties from Definition 2.5 can also be used to define partially ordered sets.

Definition 2.9 A *poset* (partially ordered set), (X, R), is a set X equipped with a *partial order* R, i.e. a binary relation over X which is reflexive, antisymmetric and transitive.

We are interested in particular in partially ordered sets which are linearly ordered.

Definition 2.10 A binary relation R over X is *linearly ordered* (or a *chain*) if R is antisymmetric, transitive and linear. We also say that (X, R) is a *linearly ordered set*.

We can see that if a binary relation is linear then it is also reflexive, thus linearly ordered sets are in fact special partially ordered sets. Often, we will also be interested in linearly ordered subsets of posets. In this context, also the supremum is of importance.

Definition 2.11 Let $(X, \leq)$ be a poset and $S \subseteq X$. The *supremum* of S, denoted $\mathsf{sup}(S)$ is the element of X such that $s \leq \mathsf{sup}(S)$ for all $s \in S$ and $\mathsf{sup}(S) \leq x$ for all $x \in X$ which satisfies $s \leq x$ for all $s \in S$. The supremum is also called *least upper bound* of S, denoted $\mathsf{lub}(S)$.

Now we recall the notion of an ω-cpo, a particular partially ordered set.

Definition 2.12 A poset is an *ω-cpo* if it has a least element $\perp$ and every linearly ordered sequence has a supremum (least upper bound).

Example 2.13 Let X be the powerset of $U = \{a, b, c\}$ with $\subseteq$ being the subset inclusion of elements from X. Then $(X, \subseteq)$ is a partially ordered set but not a a linearly ordered set because e.g. neither $\{a, b\} \subseteq \{a, c\}$ nor $\{a, c\} \subseteq \{a, b\}$. Nevertheless, it is an ω-cpo since we have a least element $\emptyset$ and every linearly ordered sequence of elements has a supremum.

Another special case of partial ordered sets are well-orderings.

Definition 2.14 A poset $(X, \leq_x)$ is *well-ordered*, (or a *well-ordering*), if each subset of X has a least element.

There are certain general properties which are of particular interest and we recall them in the following definition.

Definition 2.15 Let $(X, \leq_x)$ and $(Y, \leq_y)$ be posets and $f : X \to Y$ be a function. We say that f is *monotonic* if $a \leq_x b$ implies $f(a) \leq_y f(b)$ for all $a, b \in X$. If f is monotonic and bijective then f is an *order isomorphism* between X and Y, which means that X and Y are *isomorphic*.

The notion of an isomorphism together with well-orderings are used to introduce ordinals.

Definition 2.16 An *ordinal* is an equivalence class of a well-ordering under isomorphism. A *successor ordinal* is an ordinal α such that there is a greatest ordinal β with $\beta < \alpha$. In this case α is the *successor* of β and can be denoted $\beta + 1$. Any other ordinal is called *limit ordinal*.

We will identify any ordinal α with the set of all ordinals β such that $\alpha > \beta$. Thus, any mapping $f : X \to \{\beta \mid \beta < \alpha\}$ will be represented by $f : X \to \alpha$.

Example 2.17 Natural numbers are ordinals where each $n \in \mathbb{N}$ with $n > 0$ represents a successor ordinal whereas 0 is a limit ordinal. The limit ordinal for natural numbers is $\omega = \{\beta \mid \beta < \omega \text{ and } \beta \in \mathbb{N}\}$.

We can also define ordinals as a combination of two ordinals.

Definition 2.18 Let α, β be ordinals. Then the lexicographic order $\alpha \times \beta$ is also an ordinal with

> $(a, b) \geq (a', b')$ if and only if $a > a'$ or $a = a'$ and $b \geq b'$ for all $(a, b), (a', b') \in \alpha \times \beta$.

The order can be split into two components:

> $(a, b) >_1 (a', b')$ if and only if $a > a'$ for all $(a, b), (a', b') \in \alpha \times \beta$.
>
> $(a, b) \geq_2 (a', b')$ if and only if $a = a'$ and $b \geq b'$ for all $(a, b), (a', b') \in \alpha \times \beta$.

For proving properties of ordinals we recall the principle of transfinite induction.

Definition 2.19 The *principle of transfinite induction* is given in the following: Suppose we want to prove that a property Q holds for all members of an ordinal α. Then it suffices to show that the following hold.

(i) $Q(0)$ holds.

(ii) For any ordinal β, if $Q(\alpha)$ holds for all ordinals $\alpha < \beta$ then $Q(\beta)$ holds as well.

Often, we divide part (ii) into successor ordinals and limit ordinals.
We continue with recalling the well-known result of Tarski.

Theorem 2.20 ([32]) Let $(X, \leq)$ denote an ω-cpo, let $f : X \to X$ be monotonic and let $x \in X$ be such that $x \leq f(x)$. Then f has a least fixed point a above x, which is also the least pre-fixed point of f above x, and there exists a least ordinal α such that $a = f^{\alpha}(x)$. In particular, f has a least fixed point a which is also its least pre-fixed point.

A fixed point a of f satisfies $f(a) = a$ and in case of a pre-fixed point a of f we have $f(a) \leq a$. This theorem will be used very often in the following because we will define, respectively recall, a lot of functions, respectively operators, which are monotonic and the theorem above then guarantees the existence of a least fixed point.

2.3 Least and Stable Model Semantics

Having defined the terminology, we recall at first two well known two-valued semantics, one for definite logic programs and one for normal logic programs. A two-valued semantics distinguishes just between two truth values: 'true' and 'false', represented by $\mathbf{t}$, respectively $\mathbf{f}$.

At first, we present the definition of interpretations which are basically assignments of truth values to atoms.

Definition 2.21 A *(total) two-valued interpretation* I of a disjunctive logic program Π is a mapping from the Herbrand base B_Π to the set of truth values $\{\mathbf{t}, \mathbf{f}\}$, represented by the elements of B_Π which are mapped to $\mathbf{t}$. The set of all interpretations is denoted by I_Π.

Note, that this definition and some of the following are stated for the more general case of disjunctive logic programs although the here discussed semantics are disjunction-free. But all of them all perfectly applicable to the special case of normal logic programs.

Given an interpretation I, we can define the truth of a rule occurring in $\mathsf{ground}(\Pi)$.

Definition 2.22 A ground rule $H_1 \vee \cdots \vee H_l \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ is satisfied by an interpretation I if and only if there is at least one $H_k \in I$ or an $A_i \notin I$ or a $B_j \in I$, where $1 \leq k \leq l$, $1 \leq i \leq n$, $1 \leq j \leq m$.

In other words, a clause is satisfied by I, respectively true in I, if and only if the head is true or at least one literal in the body is false in I.
Among the set of interpretations are some of particular interest, called models.

Definition 2.23 A two-valued interpretation I is a *(two-valued) model* of the disjunctive logic program Π if and only if all the rules occurring in $\mathsf{ground}(\Pi)$ are satisfied by I.

A lot of semantics are declarative and their model(s) can be computed by means of an operator and in case of definite logic programs this is done using the immediate consequence operator T_P.

Definition 2.24 Given a normal logic program P, the operator $T_P : I_P \rightarrow I_P$ is defined by setting $T_P(I) = \{H \mid H \leftarrow \mathsf{body} \in \mathsf{ground}(P)$ and body true in $I\}$.

Restricted to definite logic programs the body is just true in I if all atoms in the body are true in I.

For definite logic programs, T_P is ω-continuous [20] and allows thus to obtain the least model of a definite logic program by re-iterating the operator. We define $T_P \uparrow 0 = \emptyset$, $T_P \uparrow (n+1) = T_P(T_P \uparrow n)$ and $T_P \uparrow \omega = \bigcup_{i \geq 0} T_P \uparrow i$, and the least model is obtained as the least fixed point of this operator.

Theorem 2.25 Given a definite logic program P, the *least model* is obtained as the least fixed point of T_P.

The proof can be found for example in [20].
In [17], Hitzler and Wendt presented a uniform framework using level mappings for characterizing logic programming semantics in an alternative way. We recall the basic definition from [17].

Definition 2.26 A *level mapping* for a program P is a mapping $l : B_P \to \alpha$, where α is some ordinal.

The equivalent characterization for the least model semantics of definite logic programs is the following.

Theorem 2.27 ([17]) The least model $T_P \uparrow \omega$ for a definite program P is the unique model M for P satisfying the following condition: there exists a level mapping $l : B_P \to \alpha$ for some ordinal α, such that for each $A \in M$ there is a clause $A \leftarrow \texttt{body}$ in $\texttt{ground}(P)$ with $\texttt{body}$ true in M and $l(B) < l(A)$ for each $B \in \texttt{body}$.

The general idea of this characterization is that the mapping represents the dependencies between atoms and thus the structure of the program. A small example shall demonstrate that.

Example 2.28 Let P be the given program.

$$
\begin{array}{lcl}
a & \leftarrow & \\
b & \leftarrow & a \\
c & \leftarrow & \\
d & \leftarrow & b, c \\
e & \leftarrow & e
\end{array}
$$

Then we calculate the least model in the following way using the operator

T_P.

$$\begin{aligned} T_P \uparrow 0 &= \emptyset. \\ T_P \uparrow 1 &= T_P(\emptyset) = \{a, c\}. \\ T_P \uparrow 2 &= T_P(\{a, c\}) = \{a, b, c\}. \\ T_P \uparrow 3 &= T_P(\{a, b, c\}) = \{a, b, c, d\}. \\ T_P \uparrow 4 &= T_P(\{a, b, c, d\}) = \{a, b, c, d\} = T_P \uparrow 3 = T_P \uparrow \omega. \end{aligned}$$

Following the proof of Theorem 2.27 in [17] the levels are then obtained by setting $l(A) = \mathsf{min}\{n \mid A \in T_P \uparrow (n+1)\}$ for all $A \in M$, i.e. $l(a) = l(c) = 0$, $l(b) = 1$, and $l(d) = 2$. For all $A \notin M$ the level is set to 0, thus $l(e) = 0$ although any other value would be appropriate as well. It is also easy to see that there is no clause with head e which could possibly satisfy the condition given in Theorem 2.27 so e cannot occur in the least model.

Unfortunately, for normal logic programs T_P is in general not ω-continuous. Nevertheless, the least model semantics for definite programs is used to define the standard two-valued semantics for normal logic programs - the stable model semantics.
In [14], Gelfond and Lifschitz, introduced this stable model semantics using a specific program transformation.

Definition 2.29 Let P be a normal logic program and let I be in I_P. The *Gelfond-Lifschitz transform* P/I of P is the set of all clauses $H \leftarrow A_1, \ldots, A_n$ for which there exists a clause $H \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(P)$ with $B_1, \ldots, B_m \notin I$.

The idea is that given a normal logic program P and an interpretation I all clauses which contain a default literal which is false in I are removed from the program and all the default literals occurring in the remaining clauses as well. Since the resulting program P/I is negation-free by construction, i.e. definite, we can apply the least model semantics to P/I.

Definition 2.30 Given a normal logic program P and an interpretation I, the *Gelfond-Lifschitz operator* Γ_P is defined as follows:

$$\Gamma_P(I) = T_{P/I} \uparrow \omega$$

Stable models are then defined as the fixpoints of the Gelfond-Lifschitz operator Γ_P.

Definition 2.31 Let P be a normal logic program. A two-valued interpretation I is a *stable model* of P if and only if $\Gamma_P(I) = I$. The stable model semantics of P is the intersection of all stable models of P.

There is also a straightforward level mapping characterization of the stable model semantics which was introduced by Fages.

Theorem 2.32 ([12]) Let P be a normal program. Then a model M of P is a stable model of P if and only if there exists a total level mapping $l : B_P \to \alpha$ such that for each $A \in M$ there exists a clause $A \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(P)$ with $A_i \in M$, $B_j \notin M$ and $l(A) > l(A_i)$ for all $i = 1, \ldots, n$ and all $j = 1, \ldots, m$.

Example 2.33 Consider the following program and the models $M_1 = \{b, c\}$ and $M_2 = \{a\}$.

$$\begin{array}{rcl} a & \leftarrow & c, \neg b \\ b & \leftarrow & c \\ c & \leftarrow & \neg a \end{array}$$

P/M_1 is obtained by canceling the complete first clause, because $\neg b$ is false in M_1 and by removing $\neg a$ from the third clause.

$$\begin{array}{rcl} b & \leftarrow & c \\ c & \leftarrow & \end{array}$$

Then $T_{P/M_1} \uparrow \omega = \{b, c\}$ and M_1 is a stable model and $l(c) = 0$, $l(b) = 1$, and $l(a) = 0$ are obtained in the same way as in Example 2.28. P/M_2 is

$$\begin{array}{rcl} a & \leftarrow & c \\ b & \leftarrow & c \end{array}$$

and $T_{P/M_2} \uparrow \omega = \emptyset$, thus M_2 is not a stable model. Considering Theorem 2.32 this is no surprise since the only clause with a in the head contains c which is not contained in M_2.

2.4 Well-founded Semantics

The well-founded semantics was introduced by van Gelder, Ross, and Schlipf [34] around the same time as stable models but based on three-valued interpretations for which the set of truth values is extended by introducing a third truth value **u** meaning 'undefined'.

Definition 2.34 A *(partial) three-valued interpretation* I of a normal logic program P is a mapping from the Herbrand base B_P to the set of truth values $\{\mathbf{t}, \mathbf{u}, \mathbf{f}\}$, represented, for $A, B \subseteq B_P$ and $A \cap B = \emptyset$, by the set (A, B), where elements in A are mapped to $\mathbf{t}$, elements in B are mapped to $\mathbf{f}$, and the remaining to $\mathbf{u}$. The set of all three-valued interpretations is denoted by $I_{P,3}$ and each element $(A, B) \in I_{P,3}$ can be represented using the *signed set* $A \cup \neg B$.

Note that here the definition is restricted to normal logic programs because the approaches presented in the following sections are using different concepts for representing derivable knowledge which is also valid for the following definitions. That is why they are also restricted to normal logic programs.
Having extended interpretations we also extend truth of clauses to the three-valued case and we do that starting with the body.

Definition 2.35 Given a three-valued interpretation I, the body of some ground clause $H \leftarrow L_1, \ldots, L_n$ is true in I if and only if all literals L_i, $1 \leq i \leq n$, are true in I, or false in I if and only if at least one of that literals is false in I. Otherwise the body is undefined.

Definition 2.36 Given a three-valued interpretation I, the ground clause $H \leftarrow$ `body` is true in I if and only if (at least) one of the following conditions holds:

- the head H is true in I,
- `body` is false in I,
- `body` is undefined and H is not false in I.

The definition of the a three-valued model is straightforward.

Definition 2.37 Given a normal logic program P, a three-valued interpretation I is a *three-valued model* for P if and only if all clauses in $\mathsf{ground}(P)$ are true in I.

Note the difference in applied truth values: clauses are only true or false although some literals or even the entire head or body is undefined.
To distinguish between several three-valued models, the knowledge ordering [13] for three-valued partial interpretations is recalled in the following.

Definition 2.38 Given two three-valued interpretations (A, B) and (C, D), the *knowledge ordering* $\leq_k$ is defined as $(A, B) \leq_k (C, D)$ if and only if $A \subseteq C$ and $B \subseteq D$.

Obviously, this just coincides with the subset inclusion on signed sets.

Example 2.39 Given the partial interpretations $I_1 = (\{a,b\},\{c\})$, $I_2 = (\{a\},\{c,d\})$ and $I_3 = (\{a,b\},\{c,d\})$ then $I_1 \leq_k I_3$ and $I_2 \leq_k I_3$ but neither $I_1 \leq_k I_2$ nor $I_2 \leq_k I_1$. This can easily be seen by means of the equivalent signed set notion, because $\{a, b, \neg c\}$ and $\{a, \neg c, \neg d\}$ are both subsets of $\{a, b, \neg c, \neg d\}$ but none of them is a subset of the other.

For defining the well-founded semantics the notion of an unfounded set needs to be recalled [34].

Definition 2.40 Given a normal logic program P and $I \in I_{P,3}$, we say that $U \subseteq B_P$ is an *unfounded set* (of P) *with respect to* I if each atom $A \in U$ satisfies the following condition: for each clause $A \leftarrow \texttt{body}$ in $\mathsf{ground}(P)$ at least one of the following holds:

(Ui) Some (positive or negative) literal in `body` is false in I.

(Uii) Some (non-negated) atom in `body` occurs in U.

Given a logic program P and an interpretation $I \in I_{P,3}$, there exists a greatest unfounded set of P with respect to I which is obtained as the union of all unfounded sets of P with respect to I.
We continue by extending the immediate consequence operator T_P to three-valued interpretations.

Definition 2.41 Given a normal logic program P, the operator $T'_P : I_{P,3} \rightarrow I_{P,3}$ is defined by setting $T'_P(I) = \{H \mid H \leftarrow \texttt{body} \in \mathsf{ground}(P)$ and `body` true in the three-valued interpretation $I\}$.

Finally, we recall the definition of the operator W_P.

Definition 2.42 Given a normal logic program, let $U_P(I)$ be the greatest unfounded set (of P) with respect to I. Then for all $I \in I_{P,3}$:

$$W_P(I) = T'_P(I) \cup \neg U_P(I).$$

It was shown in [34] that W_P is a monotonic operator and thus, by Theorem 2.20, has a least fixed point.

Definition 2.43 The *well-founded model* M of a normal logic program is obtained as the least fixed point of the operator W_P.

It should be mentioned that, in general, the operator W_P is defined for four-valued interpretations, including a forth truth value 'both' which is assigned to an atom which occurs in T'_P and U_P. But since it has been shown [34] that the iteration $W_P \uparrow 0 = \emptyset$, $W_P \uparrow (n+1) = W_P(W_P \uparrow n)$, and $W_P \uparrow \alpha = \bigcup_{i<\alpha} W_P \uparrow i$ for limit ordinals α, always yields three-valued interpretations, we simplified the definitions regarding this aspect.

Example 2.44 Let P be the following program.

$$\begin{array}{rcl} a & \leftarrow & \\ b & \leftarrow & \neg a, c \\ c & \leftarrow & \neg b \\ d & \leftarrow & a, \neg d \\ e & \leftarrow & \neg c, f \\ f & \leftarrow & e \end{array}$$

Now we can calculate the least fixed point of W_P.

$$\begin{array}{rcl} W_P \uparrow 0 & = & \emptyset \\ W_P \uparrow 1 & = & \{a, \neg e, \neg f\} \\ W_P \uparrow 2 & = & \{a, \neg b, \neg e, \neg f\} \\ W_P \uparrow 3 & = & \{a, c, \neg b, \neg e, \neg f\} \\ W_P \uparrow 4 & = & \{a, c, \neg b, \neg e, \neg f\} \end{array}$$

Thus $M = \{a, c, \neg b, \neg e, \neg f\}$ is the well-founded model where d remains undefined. Note that e and f occur in the greatest unfounded set by (Uii) whereas b is contained in U_P because of (Ui).

We will also show the characterization of the well-founded semantics using level mappings, taken from [17] but since we are now dealing with three-valued interpretations we have to extend the notion of level mappings appropriately beforehand.

Definition 2.45 For a program P and a three-valued interpretation $I \in I_{P,3}$ an *I-partial level mapping* for P is a partial mapping $l : B_P \to \alpha$ with domain $\mathsf{dom}(l) = \{A \mid A \in I \text{ or } \neg A \in I\}$, where α is some (countable) ordinal. Every such mapping is extended to literals by setting $l(\neg A) = l(A)$ for all $A \in \mathsf{dom}(l)$.

The alternative characterization is the following.

Definition 2.46 Let P be a normal logic program, let I be a model for P, and let l be an I-partial level mapping for P. We say that P *satisfies* (WF) *with respect to* I *and* l if each $A \in \mathsf{dom}(l)$ satisfies one of the following conditions.

(WFi) $A \in I$ and there is a clause $A \leftarrow L_1, \dots, L_n$ in $\mathsf{ground}(P)$ such that $L_i \in I$ and $l(A) > l(L_i)$ for all i.

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

(WFiia) There exists i with $\neg A_i \in I$ and $l(A) \geq l(A_i)$.

(WFiib) There exists j with $B_j \in I$ and $l(A) > l(B_j)$.

If $A \in \mathsf{dom}(l)$ satisfies (WFi), then we say that A *satisfies* (WFi) *with respect to* I *and* l, and similarly if $A \in \mathsf{dom}(l)$ satisfies (WFii).
The next theorem states the correspondence between the well-founded model and this definition.

Theorem 2.47 ([17]) Let P be a normal logic program with well-founded model M. Then, in the knowledge ordering, M is the greatest model amongst all models I for which there exists an I-partial level mapping l for P such that P satisfies (WF) with respect to I and l.

Example 2.48 Reconsider the program given in Example 2.44:

$$\begin{array}{rcl} a & \leftarrow & \\ b & \leftarrow & \neg a, c \\ c & \leftarrow & \neg b \\ d & \leftarrow & a, \neg d \\ e & \leftarrow & \neg c, f \\ f & \leftarrow & e \end{array}$$

We know already, that $M = \{a, c, \neg b, \neg e, \neg f\}$ is the well-founded model of this program. The level mapping is $l(a) = 0$, $l(b) = 1$, $l(c) = 2$, and $l(e) = l(f) = 0$ which can just be obtained by setting $l(A) = \mathsf{min}\{n \mid$ either A or $\neg A$ occur in $W_P \uparrow (n+1)\}$. Note that d can neither be true, because it also occurs as default literal in the body of the only clause with head d, thus violating (WFi), nor false, since (WFiib) cannot be satisfied by the same reason and (WFiia) cannot be satisfied because a is true. We also see that e.g. $\{a, c, \neg b\}$ is a model of this program which satisfies the conditions from Definition 2.46, however it is not the greatest one and thus not the well-founded model according to Theorem 2.47.

3 Strong Well-founded Semantics

We start with the strong well-founded semantics which was introduced by Ross [29], one of the developers of the well-founded semantics. At first we extend some notions from three-valued semantics of normal logic programs to disjunctive logic programs and therefore we recall the notions of closure and consistency from [29].

Definition 3.1 Let I be a set of disjunctive literals. The *closure* of I, written $\mathsf{cl}(I)$, is the least set I' with $I \subseteq I'$ which additionally satisfies the following two conditions.

1. If $D \in I'$ then every disjunction containing D is also contained in I'.
2. For all disjunctions D_1 and D_2, $\neg D_1 \in I'$ and $\neg D_2 \in I'$ if and only if $\neg(D_1 \vee D_2) \in I'$.

I is *consistent* if there is no $D \in \mathsf{cl}(I)$ with $\neg D \in \mathsf{cl}(I)$ as well.

The general idea behind these conditions is that if we e.g. have a consistent set I containing p and q then $\neg(p \vee q)$ cannot occur in I. However, I is also consistent if $(p \vee q)$ is not contained in I, i.e. undefined. Thus, a consistent set does not contain contradictory information but does also not guarantee to be closed either in opposite to what was mentioned in [29].

Definition 3.2 A *disjunctive three-valued interpretation* I of a disjunctive logic program Π is a consistent mapping from the extended Herbrand base EB_Π to the set of truth values $\{\mathbf{t}, \mathbf{u}, \mathbf{f}\}$, represented, for $A, B \subseteq EB_\Pi$ and $A \cap B = \emptyset$, by the set (A, B), where elements in A are mapped to $\mathbf{t}$, elements in B are mapped to $\mathbf{f}$, and the remaining to $\mathbf{u}$. The set of all disjunctive three-valued interpretations is denoted by $I_{\Pi,3}$ and each element $(A, B) \in I_{\Pi,3}$ can be represented using the *disjunctive signed set* $A \cup \neg B$.

We continue by extending the notion of truth to disjunctive clauses starting with the body.

Definition 3.3 Given a disjunctive three-valued interpretation I, the body of some ground clause $\mathcal{H} \leftarrow \mathcal{A}, \neg\mathcal{B}$ is true in I if and only if all literals in the body are true in I, or false in I if and only if there is a D such that either $\mathcal{D} \subseteq \mathcal{A}$ and $\neg D \in I$ or $\mathcal{D} \subseteq \mathcal{B}$ with $D \in I$. Otherwise the body is undefined.

The difference to non-disjunctive clauses is caused by the possibly indefinite knowledge contained in disjunctive interpretations. Given a disjunction, even though we do not know the truth value of any single disjunct, the disjunction may be true. But then at least one disjunct must be true, although we do not know which one and the conjunction containing all these disjuncts as negated literals must be false and so is the whole body.

Definition 3.4 Let I be a disjunctive three-valued interpretation. Then the ground clause $\mathcal{H} \leftarrow \texttt{body}$ is true in I if and only if (at least) one of the following conditions holds:

- there is $\mathcal{D} \subseteq \mathcal{H}$ with D true in I,
- $\texttt{body}$ is false in I,
- $\texttt{body}$ is undefined and $\mathcal{H}$ is not false in I.

Apart from the treatment of the disjunctive head, this is rather similar to Definition 2.36 and the notion of a model is extended straightforwardly.

Definition 3.5 Given a disjunctive logic program Π, a disjunctive three-valued interpretation I is a *(disjunctive) three-valued model* for Π if and only if all clauses in $\mathsf{ground}(\Pi)$ are true in I.

Accordingly, we also extend the knowledge ordering [13] for three-valued partial interpretations to disjunctive interpretations.

Definition 3.6 Given two disjunctive three-valued interpretations (A, B) and (C, D), the *disjunctive knowledge ordering* $\preceq_k$ is defined as $(A, B) \preceq_k (C, D)$ if and only if $A \subseteq C$ and $B \subseteq D$.

Obviously, this also just coincides with the subset inclusion on disjunctive signed sets.
Following the previous section we would now continue with the definition of an extension of the unfounded set but Ross showed in [29] with the following example that there is no simple generalization to disjunctive programs.

Example 3.7

$$\begin{aligned} p \leftarrow & \quad t \wedge v \\ t \leftarrow & \quad \neg q \\ v \leftarrow & \quad \neg r \\ q \vee r \leftarrow & \end{aligned}$$

His argument regarding the given program above is the following: Both q and r should be undefined and therefore t and v as well. Nevertheless, one of q and r has to be true, thus either t or v are definitely false although we do not know which. Then p must be false since it requires both, t and v, to be true. If we now try to lift the concept of the unfounded set to disjunctive programs, then for p being contained in the unfounded set, we must include t or v as well, i.e. one of them being false, which is against our intentions. Alternatively, we could extend the unfounded set to include conjunctions. Instead, Ross defined a procedural semantics based on a top-down approach for the well-founded semantics developed by himself [28] and the derivation rules for the strong well-founded semantics are given in the following definition.

Definition 3.8 Let I be a set of disjunctive literals and Π a disjunctive program. The *derivate* I' is *strongly derived* from I (written $I \Leftarrow I'$) if I contains a disjunction D and there is some clause in $\mathsf{ground}(\Pi)$ given by

$$H \leftarrow A_1, \ldots, A_n, \neg B$$

such that either

(S1) $\mathcal{H} \subseteq \mathcal{D}$ and $I' = (I \setminus \{D\}) \cup \{A_1 \vee D, \ldots, A_n \vee D\} \cup \neg\mathcal{B}$

or

(S2) $\mathcal{H} \not\subseteq \mathcal{D}$, $\mathcal{C} = \mathcal{H} \setminus \mathcal{D} \neq \emptyset$, and $I' = (I \setminus \{D\}) \cup \mathcal{A} \cup \neg\mathcal{B} \cup \neg\mathcal{C}$.

$\mathcal{C}$ is the set of atoms which occur in $\mathcal{H}$ but not in the head $\mathcal{D}$. Then $\neg\mathcal{C}$ is just the set of these negated atoms. This slightly differs from the way it is defined in [29]: there, just one negated disjunction containing all that atoms is added to I'. But this will not make a difference as we will see later on.

Example 3.9 Consider the following program Π:

$$\begin{aligned} p \leftarrow &\ q \\ p \leftarrow &\ r \\ q \vee r \leftarrow & \end{aligned}$$

Let I be $\{p\}$. Then $\{p\} \Leftarrow \{p \vee q\}$ by the first clause and (S1), $\{p \vee q\} \Leftarrow \{p \vee q \vee r\}$ by the second clause and (S1), and $\{\}$ is strongly derived from $\{p \vee q \vee r\}$ by the third clause and (S1). Consider alternatively $I_1 = \{q\}$. Then, by rule (S2) we obtain $\{q\} \Leftarrow \{\neg r\}$.

Obviously, there is no rule applicable to $\{\neg r\}$, and we will formalize how to proceed in that case starting with the notion of a strong derivation sequence.

Definition 3.10 Let D be a ground disjunction and let $I_0 = \{D\}$. Suppose I_1 is strongly derived from I_0, I_2 is strongly derived from I_1 and so on. We call the sequence $I_0, I_1, \ldots$ a *strong derivation sequence* for D.

A strong derivation sequence may be finite or infinite, and there may be more than one derivation sequence for a particular disjunction.

Definition 3.11 An *active* strong derivation sequence for D is a finite strong derivation sequence for D whose last element is either empty or contains only negative literals. The last element of an active strong derivation sequence is called *basis* of D.

Thus we obtained two active strong derivations in Example 3.9, one with the basis $\{\}$ and the other with $\{\neg r\}$. Consider the very simple program just consisting of one single clause:

$$p \leftarrow \quad p$$

Then obviously $\{p\} \Leftarrow \{p\} \Leftarrow \ldots$ is not an active strong derivation sequence.

We can use derivation sequences to construct derivation trees starting with a single disjunction where the derivation sequences are branches and the bases are leaves. These leaves are used to connect different trees in the following way.

Definition 3.12 If $I = \{\neg l_1, \ldots, \neg l_n\}$ is an arbitrary set of ground negated disjunctions then $\bar{I} = l_1 \vee \cdots \vee l_n$. If I is empty, denoting $\mathbf{t}$, then $\bar{I}$ denotes $\mathbf{f}$.

Thus the basis is just negated and the obtained disjunction is the root of a new derivation tree, including the possibility of having no tree at all if the basis is empty. We can see now that the mentioned difference of Definition 3.8 to the original one from [29] does not matter. There, a negated disjunction $\neg(D_1 \vee \ldots \vee D_n)$ is added to the derivation. This negated disjunction occurs unchanged in the basis I. Then $(D_1 \vee \ldots \vee D_n)$ is a subdisjunction of $\bar{I}$ and thus all D_i, for $1 \leq i \leq n$, occur in $\bar{I}$. The very same thing also happens if we directly add all $\neg D_i$, for $1 \leq i \leq n$, to the particular derivation. So only negative literals may occur in an element of a derivation sequence which also explains our slight simplification of Definition 3.11 compared to [29].

Definition 3.13 The *strong global tree* Γ_D^S for a given disjunction $D \in EB_\Pi$ is defined as follows:

- The root of Γ_D^S is D.
- If the disjunction D' is any node of Γ_D^S then its children are all disjunctions of the form $\bar{I}$, where I ranges over all bases for D'.

Truth assignments of ground disjunctions in Γ_D^S are defined as follows:

- If every child of D' is true, then D' is false. (In particular, D' is false if it has no children.)
- If some child of D' is false, then D' is true.
- Any node which is not true or false according to the above rules is undefined.

These assignments may also include nodes which are not the root of the tree. However, there is a strong global tree for each disjunction and the strong well-founded semantics is defined by means of these trees.

Definition 3.14 Let Π be a disjunctive logic program and $M^S_{WF}(\Pi)$ denote the set of disjunctive literals such that for all disjunctions D

- If D is true in Γ^S_D then $D \in M^S_{WF}(\Pi)$.
- If D is false in Γ^S_D then $\neg D \in M^S_{WF}(\Pi)$.
- If D is undefined in Γ^S_D then neither $D \in M^S_{WF}(\Pi)$ nor $\neg D \in M^S_{WF}(\Pi)$.

We call $M^S_{WF}(\Pi)$ the *strong well-founded model* of Π.

Example 3.15 Recall the program from Example 3.7.

$$
\begin{aligned}
p \leftarrow &\quad t \wedge v \\
t \leftarrow &\quad \neg q \\
v \leftarrow &\quad \neg r \\
q \vee r \leftarrow &
\end{aligned}
$$

Let us start with the strong global tree Γ^S_p: $\{p\} \Leftarrow \{p \vee t, p \vee v\}$ by $(S1)$ and the first clause. Then $\{p \vee t, p \vee v\} \Leftarrow \{\neg q, p \vee v\} \Leftarrow \{\neg q, \neg r\}$ follows according to the second and third clause and (S1). (One could switch the order of applying the clauses but the result is obviously the same basis.) So $I_3 = \{\neg q, \neg r\}$ is the only basis in this tree (although occurring twice) and $\bar{I}_3 = \{q \vee r\}$ is the only child of $\{p\}$. But due to the forth clause there is only one child $\{\}$ of $\{q \vee r\}$ which denotes false, so $\{q \vee r\}$ is true and since the only child in Γ^S_p is true, p is false.

As already mentioned $q \vee r$ is contained in $M^S_{WF}(\Pi)$ but this should not be the case for q respectively r. Consider e.g. $\{q\}$ then $\{\neg r\}$ is strongly derived from $\{q\}$ by (S2) and $\{r\}$ is the only child of Γ^S_q. But $\{r\} \Leftarrow \{\neg q\}$ and $\{q\}$ is the only child of Γ^S_r and this alternates forever. So there is neither a false child nor all children are true, hence r and q are both undefined in the strong well-founded model.

Nevertheless, $q \vee r$ is true, meaning one of them is true although we do not know which one. Both, t and v, depend negatively on one of them, so we should be able to derive the truth of the disjunction $t \vee v$ as well, without knowing which specifically is true. Unfortunately this is not the case: We have $\{t \vee v\} \Leftarrow \{\neg q\}$ by (S1) and the second rule, so q is one child, and by using the third rule equivalently we obtain r as a child as well. Both children are undefined so $t \vee v$ remains undefined. This would not be the case if the two clauses would be positively depending on r, respectively q. Then we would derive $\{t \vee v\} \Leftarrow \{t \vee v \vee q\} \Leftarrow \{t \vee v \vee q \vee r\} \Leftarrow \{\}$ by applying (S1) to the second, third, and forth clause and $t \vee v$ would be true.

We continue with recalling two results presented by Ross.

Lemma 3.16 ([29]) Given a disjunctive logic program Π, $M^S_{WF}(\Pi)$ is a consistent interpretation.

Thus $M^S_{WF}(\Pi)$ is in fact a disjunctive three-valued interpretation which allows us to conclude that the strong well-founded model does not contain contradictory information.

Theorem 3.17 ([29]) For normal programs, the strong well-founded semantics coincides with the standard well-founded semantics.

Example 3.18 We demonstrate this theorem with the following example.

$$\begin{aligned} p &\leftarrow \quad p \\ q &\leftarrow \quad \neg p \\ r &\leftarrow \quad \neg q \\ r &\leftarrow \quad \neg s \\ s &\leftarrow \quad \neg r \end{aligned}$$

The well-founded model of that program contains $\neg p$ by means of the unfounded set and thus q by the second clause. Note that r and s must be undefined since the forth and fifth clause do not satisfy one of the conditions of the unfounded set. We have that p is also false in M^S_{WF} because the tree Γ^S_p has no children at all. Because of that, q is true since the only child of Γ^S_q is p. Γ^S_r has a true child, q, but also a child s and Γ^S_s only has one child r, hence s and r are both also undefined in the strong well-founded semantics.

There is, however, an important note. Disjunctive three-valued interpretations may also contain disjunctions and not just atoms, i.e. in the example above, $(p \vee q)$ is in the strong well-founded model whereas it is not contained in the well-founded model. Precisely spoken, the coincidence only holds if we restrict the disjunctive model to (non-disjunctive) literals.

The notion of a level, respectively a stage, of a disjunction was already used for this semantics by Ross in [29], although the definition itself differs. We restate that definition and use only the notion stage to avoid ambiguities.

Definition 3.19 Let Π be a disjunctive logic program and the *stage* s be a partial function $s : EB_\Pi \to \alpha$ for some ordinal α. If $D \in M_{WF}^S$ then D has a stage one more than the minimum stage of all its false children. If $\neg D \in M_{WF}^S$ then D has a stage one more than the least upper bound of the level of its true children.

Unfortunately, this definition is a bit imprecise as the following example will show.

Example 3.20 Let Π be consisting of the following clauses.

$$\begin{aligned} p(0) &\leftarrow \\ p(X) &\leftarrow \\ p(X) &\leftarrow \quad \neg q(X) \\ q(X) &\leftarrow \quad \neg p(s(X)) \end{aligned}$$

Clearly, all $p(s^n(0))$, for $n \geq 0$, are true in the strong well-founded model since each tree $\Gamma^S_{p(s^n(0))}$ contains a basis $\{\}$ because of the second clause and (S1) and thus a false child. Furthermore, since all $\Gamma^S_{q(s^n(0))}$, for $n \geq 0$, contain only one basis $\{\neg p(s^{n+1}(0))\}$ by the forth clause and (S1), i.e. only one child which is true, all $q(s^n(0))$ are false. Now we try to determine the stage of e.g. $p(0)$. The tree $\Gamma^S_{p(0)}$ has three children: twice the empty child and $q(0)$ and all children are false. By definition, the stage of $p(0)$ should be one more than the minimum stage of its false children. Since the empty children do no have any children themselves, we set their stage to 0. There clearly is no smaller value, i.e. the minimum stage is 0. However, being precise, we at first need the stage of $q(0)$ before we can decide the stage of $p(0)$. The stage of a disjunction that is false has stage one more than the least upper bound of the stage of its true children. Since $q(0)$ has only one true child, namely $p(s(0))$ we need to know at first the stage of $p(s(0))$ which brings us back to the initial problem since $\Gamma^S_{p(s(0))}$ also has a false child $q(s(0))$ which again has a true child $p(s(s(0)))$ and so on. We see that we run into a loop if we apply the definition of the stage strictly recursive.

Fortunately, we do not have to apply the definition strictly recursive. Every passed child in this calculation necessarily increases the stage and we use the minimal number of children to be passed in depth to establish the truth value of a disjunction to show that the stage is defined for all disjunctions occurring in the strong well-founded model.

Definition 3.21 Given a disjunctive logic program Π we define the partial function of the minimal number of children to be passed, $MCh(D) : EB_\Pi \rightarrow \alpha$ for some ordinal α.

- $D \in M^S_{WF}$: $MCh(D) = \mathsf{min}\{MCh(C) \mid C$ is a false child of $\Gamma^S_D\} + 1$.
- $\neg D \in M^S_{WF}$:
  - $MCh(D) = 0$ if Γ^S_D does not contain any children.
  - $MCh(D) = \mathsf{lub}\{MCh(C) \mid C$ is child of $\Gamma^S_D\} + 1$.

The general idea is to find the minimal number of children to be passed to evaluate a disjunction to true, respectively false. This means in particular, that in case of a false disjunction we have to take the least upper bound of the values of the children since a disjunction D is only known to be false if we know that all children in Γ^S_D are true.

Example 3.22 Reconsider Example 3.20. Then $MCh(p(s^n(0))) = 2n + 1$ and $MCh(q((s^n(0))) = 2n + 2$. If we add a further clause $r \leftarrow \neg p(X)$ then we obtain $MCh(r) = \omega + 1$ since the least upper bound of all values $MCh(p(s^n(0)))$ of the children in Γ^S_r is ω.

It is easy to see that in fact this definition coincides with the stage but represents in a better way that a disjunction which is true or false depends on one or more sequences of children which end with the empty child, respectively a child without any children itself.

Lemma 3.23 Let Π be a disjunctive logic program and $D \in EB_\Pi$. If D is true or false in M^S_{WF} then $D \in \mathsf{dom}(s)$.

Proof: We show at first, for all $D \in EB_\Pi$ which are true or false in M^S_{WF} with $D \in \mathsf{dom}(MCh)$, by transfinite induction on $MCh(D)$ that $s(D) \leq MCh(D)$.

Let $MCh(D)$ be 0. If $D \in M^S_{WF}$ then, by Definition 3.21, $MCh(D) = 1 + \mathsf{min}\{MCh(C) \mid C$ is false child of $\Gamma^S_D\}$. Since there is no ordinal smaller than 0, there can be no child in Γ^S_D and D cannot be true. So let $\neg D \in M^S_{WF}$ then Γ^S_D does not contain any children. The stage of D is 0 and thus $s(D) \leq MCh(D)$.

Suppose that for all $C \in \mathsf{dom}(MCh)$ with $MCh(C) < \alpha$ we know that $s(C) \leq MCh(C)$.

Let $MCh(D) = \alpha$ and assume first that $\neg D \in M^S_{WF}$. Then, by Definition 3.21, $MCh(D) = 1 + \mathsf{lub}\{MCh(C) \mid C \text{ is child of } \Gamma^S_D\}$, i.e. all $MCh(C) < \alpha$. Then, by induction hypothesis, for all children C in Γ^S_D we have that $s(C) \leq MCh(C)$. By Definition 3.19, we know that the stage of a false disjunction is one more than the least upper bound of its true children, so $s(D) \leq 1 + \mathsf{lub}\{MCh(C) \mid C \text{ is child of } \Gamma^S_D\}$ and thus $s(D) \leq MCh(D)$. Alternatively, assume that $D \in M^S_{WF}$, i.e. $MCh(D) = 1 + \mathsf{min}\{MCh(C) \mid C$ is false child of $\Gamma^S_D\}$ by Definition 3.21. Then $\mathsf{min}\{MCh(C) \mid C$ is false child of $\Gamma^S_D\} < \alpha$. By induction hypothesis, for these children C with minimal value $MCh(C)$ we have $s(C) \leq MCh(C)$. The stage of a true disjunction is defined to be one more than the minimal stage of its false children. The minimal stage can only be smaller or equal to $s(C) \leq MCh(C)$ for children C with minimal $MCh(C)$ and thus $s(D) \leq MCh(D)$.

Hence, whenever $MCh(D)$ is defined then the stage of D is defined as well with a smaller or exactly the same value. So finally assume, that $MCh(D)$ is undefined even though D, respectively $\neg D$, occurs in the strong well founded-model. If $\neg D \in M^S_{WF}$ then at least one of the true children cannot be defined with respect to MCh as well. If $D \in M^S_{WF}$ then all of its false children have no defined value MCh otherwise the defined value $MCh(D')$ of a false child D' represents a limit: in every sequence of passed children each passed child in depth necessarily increases the value MCh and if every sequence contains more elements than $MCh(D')$ then $MCh(D')+1$ is $MCh(D)$ and the value is defined for D which is a contradiction. Thus all false children have no defined value. We can apply this argument to the complete sequence and thus all false children are undefined with respect to MCh and each false child contains at least one true child with undefined value MCh. But then, there is not even a transfinite sequence of children which establishes the truth, respectively the falsity, of the disjunction D hence by Definition 3.13, the truth value of D is not determinable, which contradicts our initial assumption. ■

Even though the stage is defined for all disjunctions which are true or false in the strong well-founded model, it is not the desired result for the alternative characterization.

Example 3.24

$$\begin{aligned} p &\leftarrow \\ q &\leftarrow \quad p \\ r &\leftarrow \quad r \\ s &\leftarrow \quad \neg r \\ t &\leftarrow \quad \neg s \end{aligned}$$

Γ_p^S has only one child, the empty child which is true, i.e. $s(p) = 1$. Obviously, Γ_q^S also only has that child, thus we obtain for q the same stage. Γ_r^S has no children and we set the stage of r to 0. Then, Γ_s^S has only one false child, r, i.e. is of stage 1 and similarly Γ_t^S has only one true child, namely s, thus $s(t) = 2$.
Since this program is normal, we may also apply the level mapping characterization of the well-founded semantics. According to Theorem 2.47, we have that r is false in the well-founed model according to (WFiia) of Definition 2.46 and set the level to 0. Then s is true in the well-founded model by (WFi) of Definition 2.46 and we set $l(s) = 1$. We also have $l(t) = 2$ by (WFiib) of the very same definition and we obtain the same dependencies we obtained when using the stage. But $l(p) = 0$ and $l(q) = 1$ by Theorem 2.47 and (WFi) of Definition 2.46 and we obtain the dependency between p and q given by the second clause which is lost in the stage assignment.

By Theorem 3.17, we know that for normal programs like the one given above the strong well-founded semantics coincides with the well-founded semantics, thus we prefer to have a characterization which for normal programs also coincides with the characterization of the well-founded semantics.
Nevertheless, the definition of the stage will be of importance because we use it to prove certain properties of the strong well-founded model. We show at first the following lemma.

Lemma 3.25 Let Π be a disjunctive logic program and $D, D' \in EB_\Pi$ with $\mathcal{D}' \subseteq \mathcal{D}$ and $\mathcal{D}'' = \mathcal{D} \setminus \mathcal{D}'$. If $\Gamma_{D'}^S$ contains an active strong derivation sequence with child D_1' then Γ_D^S also contains an active strong derivation sequence with child D_1 such that $\mathcal{D}_1' \setminus (\mathcal{D}'' \setminus \mathcal{D}_1) = \mathcal{D}_1$.

Proof: Consider the active strong derivation sequence $\{D'\}, I_1', \ldots, I_r'$ where $\mathcal{H} \leftarrow A_1, \ldots A_n, \neg B_1, \ldots, \neg B_m$ is the clause c in $\mathsf{ground}(\Pi)$ which is used for the first derivation $\{D'\} \Leftarrow I_1'$ of that sequence. We consider two cases.

We apply (S1). Then $\mathcal{H} \subseteq \mathcal{D}'$ and $I_1' = \{A_1 \vee D', \ldots, A_n \vee D', \neg B_1, \ldots, \neg B_m\}$ by Definition 3.8. But then we also have a derivation $\{D\} \Leftarrow I_1$ with $I_1 = \{A_1 \vee D, \ldots, A_n \vee D, \neg B_1, \ldots, \neg B_m\}$ by (S1) since $\mathcal{H} \subseteq \mathcal{D}' \subseteq \mathcal{D}$. If there is no atom A_i, $1 \leq i \leq n$, occurring in c then we already obtained the basis and in both cases the child is $B_1 \vee \ldots \vee B_m$, i.e. $\mathcal{D}_1 = \mathcal{D}'_1$, $\mathcal{D}'' \setminus \mathcal{D}_1$ only contains atoms not occurring in $\mathcal{D}_1$, thus also not in $\mathcal{D}'_1$, and $\mathcal{D}'_1 \setminus (\mathcal{D}'' \setminus \mathcal{D}_1) = \mathcal{D}_1$. Otherwise, there is at least one atom A_i in c such that we have $A_i \vee D'$ in I'_1 and $A_i \vee D$ in I_1 with $(\{A_i\} \cup \mathcal{D}') \subseteq (\{A_i\} \cup \mathcal{D})$. Nevertheless, the sets of negative atoms obtained in the first derivation step are identical, i.e. $\mathcal{B}_1 = \mathcal{B}'_1$ and the sets of additional negated atoms $\mathcal{C}_1$ and $\mathcal{C}'_1$ are empty since we applied (S1) where the index, in this case 1, represents the derivation step.

Alternatively, we apply (S2). Then $\mathcal{H} \not\subseteq \mathcal{D}'$, $\mathcal{C}' = \mathcal{H} \setminus \mathcal{D}' \neq \emptyset$ and $I'_1 = \{A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m, \neg \mathcal{C}'\}$ by Definition 3.8. Since $\mathcal{D}' \subseteq \mathcal{D}$ we also have $\mathcal{H} \setminus \mathcal{D} \neq \emptyset$.
If $\mathcal{H} \not\subseteq \mathcal{D}$ then by (S2) we also have a derivation $\{D\} \Leftarrow I_1$ with $\mathcal{C} = \mathcal{H} \setminus \mathcal{D}$ and $I_1 = \{A_1, \ldots, A_n,\ \neg B_1, \ldots, \neg B_m, \neg \mathcal{C}\}$. If $\mathcal{D}'' \cap \mathcal{H} = \emptyset$ then $\mathcal{C}' = \mathcal{C}$. If $\mathcal{D}'' \cap \mathcal{H} \neq \emptyset$ then $\mathcal{C}' = \mathcal{C} \cup (\mathcal{D}'' \cap \mathcal{H})$.
Otherwise, $\mathcal{H} \subseteq \mathcal{D}$ and we can apply (S1) instead, yielding $\{D\} \Leftarrow I_1$ with $I_1 = \{A_1 \vee D, \ldots, A_n \vee D, \neg B_1, \ldots, \neg B_m\}$. Then $\mathcal{C}' \subseteq \mathcal{D}''$ since $\mathcal{H} \subseteq \mathcal{D}$, $\mathcal{C}' = \mathcal{H} \setminus \mathcal{D}'$ and $\mathcal{D}'' = \mathcal{D} \setminus \mathcal{D}'$.
Again, if c does not contain any A_i, $1 \leq i \leq n$, then I'_1 is a basis and $\mathcal{B} \cup \mathcal{C}'$ is a child in $\Gamma^S_{D'}$, and thus $\mathcal{B} \cup \mathcal{C}$, respectively $\mathcal{B}$ in case of (S1), is a child in Γ^S_D. We either have $\mathcal{C}' = \mathcal{C}$ or $\mathcal{C}' = \mathcal{C} \cup (\mathcal{D}'' \cap \mathcal{H})$ and in both cases, since $\mathcal{B}' = \mathcal{B}$, we have $(\mathcal{B}'_1 \cup \mathcal{C}'_1) \setminus (\mathcal{D}'' \setminus (\mathcal{B}_1 \cup \mathcal{C}_1)) = \mathcal{B}_1 \cup \mathcal{C}_1$, i.e. $\mathcal{D}'_1 \setminus (\mathcal{D}'' \setminus \mathcal{D}_1) = \mathcal{D}_1$. Note that the now introduced additional indices refer to the derivation step. Otherwise, there is at least one A_i in c such that we have A_i in I'_1 and A_i in I_1, respectively $A_i \vee D$ in I_1 (depending on whether we applied (S1) or (S2)), with $\{A_i\} \subseteq \{A_i\}$, respectively $\{A_i\} \subseteq (\{A_i\} \cup \mathcal{D})$, where $(\mathcal{B}'_1 \cup \mathcal{C}'_1) \setminus (\mathcal{D}'' \setminus (\mathcal{B}_1 \cup \mathcal{C}_1)) = \mathcal{B}_1 \cup \mathcal{C}_1$ as well.
As we have seen, no matter whether we apply (S1) or (S2), for each resulting positive disjunction in I'_1 there is also a positive disjunction in I_1 which subsumes the one from I'_1 which allows us to apply the same argument also to the following derivation steps of the active strong derivation sequence $I'_1, \ldots, I'_r$. We obtain the active strong derivation sequence $\{D\}, I_1, \ldots, I_r$ in Γ^S_D and that $(\mathcal{B}'_q \cup \mathcal{C}'_q) \setminus (\mathcal{D}'' \setminus (\mathcal{B}_q \cup \mathcal{C}_q)) = \mathcal{B}_q \cup \mathcal{C}_q$ holds for each $1 \leq q \leq r$ and thus $\mathcal{D}'_q \setminus (\mathcal{D}'' \setminus \mathcal{D}_q) = \mathcal{D}_q$. But then $\bigcup_{q=1\ldots r} (\mathcal{D}'_q \setminus (\mathcal{D}'' \setminus \mathcal{D}_q)) = \bigcup_{q=1\ldots r} \mathcal{D}_q$ where $\bigcup_{q=1\ldots r} \mathcal{D}'_q$ is the child in $\Gamma^S_{D'}$ and $\bigcup_{q=1\ldots r} \mathcal{D}_q$ is the child in Γ^S_D, so the claimed property holds. ■

We use this rather technical lemma to show that whenever we know that a disjunction D is true with a certain stage then all disjunctions containing D are also true with at most the same stage, respectively whenever we know that a disjunction D is false with a stage $s(D)$ then all subdisjunctions of D are also false with a stage smaller or equal to $s(D)$.

Lemma 3.26 Let $D \in EB_{\Pi}$ with $s(D) = \alpha$.

1. If $D \in M^S_{WF}$ then $D' \in M^S_{WF}$ and $s(D') \leq \alpha$ for all disjunctions D' with $\mathcal{D} \subseteq \mathcal{D}'$.

2. If $\neg D \in M^S_{WF}$ then $\neg D' \in M^S_{WF}$ and $s(D') \leq \alpha$ for all disjunctions D' with $\mathcal{D}' \subseteq \mathcal{D}$.

Proof: We are going to prove the two statements by one transfinite induction on the stage of D.

Let $s(D) = 0$. If $D \in M^S_{WF}$ then, by Definition 3.14, D is true in Γ^S_D, i.e., by Definition 3.13, there is at least one child in Γ^S_D which is false. But by Definition 3.19, the stage of D is one more than the minimum of its false children, which is not possible since there is no ordinal smaller than 0, thus D cannot be true. So $\neg D \in M^S_{WF}$ and, by Definition 3.14, D is false in Γ^S_D, i.e., by Definition 3.13, all children in Γ^S_D are true. By Definition 3.19, the stage of D is one more than the least upper bound of its true children in Γ^S_D, thus there are no children at all. Assume that there is an active strong derivation sequence in $\Gamma^S_{D'}$ with child D'_1. By Lemma 3.25 and $\mathcal{D}' \subseteq \mathcal{D}$, we then have that Γ^S_D also contains an active strong derivation sequence with child D_1 which contradicts the assumption. Hence, $\Gamma^S_{D'}$ has no children, $\neg D' \in M^S_{WF}$, and $s(D') = 0$.

Suppose that the lemma holds for all disjunctions C with $s(C) \leq \beta$, i.e. if $C \in M^S_{WF}$ then $C' \in M^S_{WF}$ and $s(C') \leq \beta$ for all disjunctions C' with $\mathcal{C} \subseteq \mathcal{C}'$, and if $\neg C \in M^S_{WF}$ then $\neg C' \in M^S_{WF}$ and $s(C') \leq \beta$ for all disjunctions C' with $\mathcal{C}' \subseteq \mathcal{C}$. Let $s(D) = \alpha = \beta + 1$ be a successor ordinal. We have to consider two cases.

If $D \in M^S_{WF}$ then, by Definition 3.14, D is true in Γ^S_D, i.e., by Definition 3.13, there is at least one child in Γ^S_D which is false. By Definition 3.19, the stage of D is one more than the minimal stage of its false children. Consider the corresponding active strong derivation sequence of a false child C with minimal stage β. By Lemma 3.25, for any D' with $\mathcal{D} \subseteq \mathcal{D}'$ we know that there also is an active strong derivation sequence with child C' such that $\mathcal{C} \setminus (\mathcal{D}'' \setminus \mathcal{C}') = \mathcal{C}'$ where $\mathcal{D}'' = \mathcal{D} \setminus \mathcal{D}'$ and thus $\mathcal{C}' \subseteq \mathcal{C}$. By induction hypothesis we have $\neg C' \in M^S_{WF}$ and $s(C') \leq \beta$. Hence, $s(D') \leq \beta + 1 = \alpha$ by Definition 3.19.

If $\neg D \in M^S_{WF}$ then, by Definition 3.14, D is false in Γ^S_D, i.e., by Definition 3.13, all children in Γ^S_D are true. Consider any disjunction D' with $\mathcal{D}' \subseteq \mathcal{D}$. If $\Gamma^S_{D'}$ does not contain any active derivation sequences then $s(D') = 0 \leq \beta + 1$ and $\neg D_1 \in M^S_{WF}$. Thus consider alternatively any active strong derivation sequence with child C'. Then, by Lemma 3.25, Γ^S_D also contains an active strong derivation sequence with child C such that $\mathcal{C}' \setminus (\mathcal{D}'' \setminus \mathcal{C}) = \mathcal{C}$ where

$\mathcal{D}'' = \mathcal{D} \setminus \mathcal{D}'$ and thus $\mathcal{C} \subseteq \mathcal{C}'$. By Definition 3.19, the stage of D is one more than the least upper bound of its true children in Γ^S_D, i.e. $s(C) \leq \beta$ and $C \in M^S_{WF}$ since all children of D are true. Thus, by induction hypothesis and $\mathcal{C} \subseteq \mathcal{C}'$, $C' \in M^S_{WF}$ and $s(C') \leq \beta$. Since we have shown for any arbitrary child C' that it is true with stage less than or equal to β, we know that the least upper bound of the stages of all true children in $\Gamma^S_{D'}$ is smaller or equal to β, hence, by Definition 3.19, $s(D') \leq \beta + 1 = \alpha$.

Finally, let $s(D) = \alpha$ be a limit ordinal, i.e there is no greatest ordinal β with $\beta < \alpha$. If $D \in M^S_{WF}$ then, by Definition 3.19, the stage of D is one more than the minimal stage of its false children. Thus we have a greatest ordinal β with $\beta < \alpha$, the minimal stage of its false children, and $D \in M^S_{WF}$ is not possible. If $\neg D \in M^S_{WF}$ then, by Definition 3.19, the stage of D is one more than the least upper bound of its true children. But then the least upper bound of its true children is the greatest ordinal with $\beta < \alpha$ and $\neg D \in M^S_{WF}$ does not hold either and we conclude that $s(D) = \alpha$ cannot be a limit ordinal which finishes the induction step. ■

Example 3.27 We demonstrate this with the given program Π.

$$\begin{aligned} p &\leftarrow p \\ q &\leftarrow \neg s, \neg t \\ s \vee t &\leftarrow \end{aligned}$$

We know that $p \vee q$ only has one child $s \vee t$ which is true with stage 1. Thus $p \vee q$ is false with $s(p \vee q) = 2$. For the same reason, q is false with $s(q) = 2$. The stage of a subdisjunction does not have to be equal: Γ^S_p does not have any children and thus p is false but with stage 0.

Lemma 3.26 shows that the strong well-founded model satisfies the first and one direction of the second condition of the closure (cf. Definition 3.1). However, for showing that it is in fact closed we also need to show the other direction of the second condition of this definition, which appears to be rather difficult.
Table 1 shows our current knowledge about the assignment of truth values in the strong well-founded semantics for a disjunction $(p \vee q)$ given the values of its two disjuncts p and q, respectively the truth value of a disjunct s given the truth value of the disjunction $(r \vee s)$ and the other disjunct r. The three entries 'n.a.' stand for 'not allowed' because if we e.g. already know that r is true then $(r \vee s)$ cannot be undefined by Lemma 3.26. The truth values in parentheses with question mark are not intended but we did not prove yet that they cannot occur.

Table 1: Truth values in the strong well-founded semantics

p	q	$(p \vee q)$		r	$(r \vee s)$	s
f	**f**	**f** $(/\mathbf{u}?^a)$		**f**	**f**	**f**
f	**u**	**u** $(/\mathbf{t}?^d)$		**f**	**u**	**u** $(/\mathbf{f}?^c)$
f	**t**	**t**		**f**	**t**	**t** $(/\mathbf{u}?^b)$
u	**f**	**u** $(/\mathbf{t}?^d)$		**u**	**f**	n.a.
u	**u**	**u/t**		**u**	**u**	**f/u**
u	**t**	**t**		**u**	**t**	**u/t** $(/\mathbf{f}?^e)$
t	**f**	**t**		**t**	**f**	n.a.
t	**u**	**t**		**t**	**u**	n.a.
t	**t**	**t**		**t**	**t**	**f/u/t**

The first thing we see is that the assignment of the truth value to a disjunction is not functional. Given two undefined disjuncts the disjunction may be true or undefined. As an example consider the program just consisting of $p \vee q \leftarrow$. Since p has q as the only child and q has only one child p and this alternates forever, both, p and q, are undefined. Nevertheless, $(p \vee q)$ has one empty child and is thus true. Alternatively, consider the program with the two clauses $p \leftarrow \neg p$ and $q \leftarrow \neg q$. Then p and q are both undefined again but now $(p \vee q)$ is also undefined since it has now only the two children p and q. This example also shows that q may be undefined if we know that p and $(p \vee q)$ are undefined, and if we drop the second clause then q may also be false in this case.

Let us now consider the cases where not intended results remain to be removed. If p and q are false then $(p \vee q)$ cannot be true by consistency of the strong well-founded model but it may be undefined (a). If we can show that $(p \vee q)$ has to be false then knowing that r is false and $(r \vee s)$ is undefined s cannot be false (c). If r is false and $(r \vee s)$ is true then s cannot be false by consistency however it may be undefined (b). But if we can show that it has to be true in general then $(p \vee q)$ cannot be true given a false and an undefined disjunct (d) and likewise given a true disjunction $(r \vee s)$ and r undefined then s cannot be false (e).
Showing one of these properties is most likely done by an transfinite induction over the stage similar to Lemma 3.26. Since undefined disjunctions cannot have a value for the stage it is most reasonable trying to prove that the cases (a) and (b) cannot occur because all assumptions in both cases have a defined stage. However, the following example will show that even that is rather complicated.

Example 3.28 Let Π be a disjunctive logic program and let the following clauses be all clauses in $\mathsf{ground}(\Pi)$ which contain p, q, a, or b in the head.

$$\begin{aligned} p \vee q \leftarrow & \quad a, \neg s \\ p \vee q \leftarrow & \quad b, \neg t \\ a \vee b \leftarrow & \quad \neg r \end{aligned}$$

The disjunction $(p \vee q)$ has three different children. We may apply (S1) with the first clause and (S2) with the third clause and obtain $\{p \vee q\} \Leftarrow \{p \vee q \vee a, \neg s\} \Leftarrow \{\neg b, \neg r, \neg s\}$. Or we apply (S1) with the second clause and (S2) with the third clause and obtain $\{p \vee q\} \Leftarrow \{p \vee q \vee b, \neg t\} \Leftarrow \{\neg a, \neg r, \neg t\}$. Alternatively, we can also apply (S1) with the first and the second clause (in arbitrary order) and then (S1) with the third clause and obtain e.g. $\{p \vee q\} \Leftarrow \{p \vee q \vee a, \neg t\} \Leftarrow \{p \vee q \vee a \vee b, \neg t, \neg s\} \Leftarrow \{\neg r, \neg s \neg t\}$. Thus the three children are $(b \vee r \vee s)$, $(a \vee r \vee t)$, and $(r \vee s \vee t)$.
For p, and symmetrically for q, we only have two children. We may apply (S2) with the first clause and (S2) with the third clause and obtain $\{p\} \Leftarrow \{a, \neg s, \neg q\} \Leftarrow \{\neg b, \neg r, \neg s, \neg q\}$. Or we apply (S2) with the second clause and (S2) with the third clause and obtain $\{p\} \Leftarrow \{b, \neg t, \neg q\} \Leftarrow \{\neg a, \neg r, \neg t, \neg q\}$. So we have two children $(b \vee r \vee s \vee q)$ and $(a \vee r \vee t \vee q)$ for p and two children $(b \vee r \vee s \vee p)$ and $(a \vee r \vee t \vee p)$ for q.

Assume we want to show in general that if p and q are false then also $(p \vee q)$ is false using an inductive argument over the stage and assume that we have shown that if $p' \vee q'$ is true and p' is false then q' has to be true. Since p and q are false, all children of them are true. Since q is false and $(b \vee r \vee s \vee q)$ is a true child of p we know that $(b \vee r \vee s)$ has to be true by our additional assumption. This allows us to derive that two of the children of $(p \vee q)$ have to be true. Unfortunately, this argument cannot be applied to the third child $(r \vee s \vee t)$ and it is not clear at all how it could be proven that $(r \vee s \vee t)$ is true. One attempt could be to show that a or b is false as well. Then $(r \vee s)$ or $(r \vee t)$ had to be true by our assumption and thus also $(r \vee s \vee t)$ by Lemma 3.26. But there is no evident argument stating that a or b should be false, both may also be undefined.

If we want to prove that q is true given that $(p \vee q)$ is true and p is false the problem is the very same. One of the children of $(p \vee q)$ has to be false and we assume that if p' and q' are false that $(p' \vee q')$ is false as well. If $(b \vee r \vee s)$ or $(a \vee r \vee t)$ is the false child then by our assumption $(b \vee r \vee s \vee p)$ or $(a \vee r \vee t \vee p)$ is also false since p is false. Otherwise $(r \vee s \vee t)$ is the false child and thus all disjuncts are false by Lemma 3.26. Then in this case we need to show that a or b is false for showing that there is a false child for q but it also is not clear at all how this could be done.

Obviously, in case of the example we already could not find an appropriate argument to show that the intended properties hold and a general proof of these intended properties from Table 1 seems to be even more difficult. But no counterexample was found either, so we leave this as an open question and will only make remarks in the following where this influences the results.

Even though we could not prove that all desired properties hold for the strong well-founded semantics, we may use the results from Lemma 3.26 to represent the strong well-founded model by a minimal set: whenever a disjunction is contained in the strong well-founded model then any superset of that disjunction is contained implicitly as well and, likewise, whenever a negated disjunction occurs then implicitly any subset of the disjunction occurs negated as well.

All the semantics for normal logic programs which were up to now characterized by means of level mappings [17] are declarative and based on operators which is of main importance for the approach itself and the proof of any characterization. Unfortunately, the strong well-founded semantics is not based on an operator and we introduce a construction which is based on derivation trees.

Definition 3.29 Let Π be a disjunctive logic program. Then Γ^S_Π is the set of all strong global trees with respect to Π and $\boldsymbol{\Gamma^S_\Pi}$ is the power set of Γ^S_Π, i.e. the set of all subsets of Γ^S_Π.

Note, that there is only one strong global tree for each disjunction occurring in Π. Thus the set of the roots of all trees occurring in Γ^S_Π is exactly EB_Π. In other words, a set of trees, i.e. an element of $\boldsymbol{\Gamma^S_\Pi}$, corresponds to a set of disjunctive literals. We use that idea to define an ω-cpo.

Proposition 3.30 Let Π be a disjunctive logic program and $\subseteq$ be the subset inclusion. Then $(\boldsymbol{\Gamma^S_\Pi}, \subseteq)$ is an ω-cpo.

Proof: At first we show that $(\boldsymbol{\Gamma^S_\Pi}, \subseteq)$ is a partially ordered set which is rather simple.

- reflexivity: We obviously have that $\forall \Gamma_1 \in \boldsymbol{\Gamma^S_\Pi}$ the subset inclusion of itself holds, i.e. $\Gamma_1 \subseteq \Gamma_1$.
- antisymmetry: By means of $\subseteq$, antisymmetry also holds for all $\Gamma_1, \Gamma_2 \in \boldsymbol{\Gamma^S_\Pi}$. If $\Gamma_1 \subseteq \Gamma_2$ and $\Gamma_2 \subseteq \Gamma_1$ then $\Gamma_1 = \Gamma_2$.
- transitivity: By the same reason the proof of transitivity is trivial. For all $\Gamma_1, \Gamma_2, \Gamma_3 \in \boldsymbol{\Gamma^S_\Pi}$, if $\Gamma_1 \subseteq \Gamma_2$ and $\Gamma_2 \subseteq \Gamma_3$ then also $\Gamma_1 \subseteq \Gamma_3$.

The least element of $\mathbf{\Gamma_\Pi^S}$ is $\emptyset$. The only thing remaining for the proof is to show that every linearly ordered sequence has a supremum. Take a linearly ordered sequence $\Gamma_1 \subseteq \Gamma_2 \subseteq \Gamma_3 \subseteq \ldots$ where $\Gamma_j \in \mathbf{\Gamma_\Pi^S}$ for all $j > 0$. We claim that $\bigcup_{j>0} \Gamma_j$ is the supremum. Since all Γ_j, $j > 0$, are sets of trees, the union of these sets is again a set of trees and thus contained in $\mathbf{\Gamma_\Pi^S}$. Furthermore, it is an upper bound because all Γ_j are contained in $\bigcup_{j>0} \Gamma_j$. Take any upper bound U for this sequence. Then $\Gamma_j \in U$ for all $j > 0$. Thus $\bigcup_{j>0} \Gamma_j \subseteq U$ for any upper bound and $\bigcup_{j>0} \Gamma_j$ is the least upper bound for that sequence. ■

We are now going to define operators which given the well-founded model simulate a declarative definition.

Definition 3.31 Let Π be a disjunctive logic program, $M_{WF}^S(\Pi)$ the strong well-founded model, and $\Gamma \in \mathbf{\Gamma_\Pi^S}$. We define:

- $T_\Pi^S(\Gamma) = \{\Gamma_D^S \in \Gamma_\Pi^S \mid \Gamma_D^S$ contains an active strong derivation sequence $\{D\}, I_1, \ldots, I_r$ with child $C = \bar{I}_r$ and $I_1 = \{D_1, \ldots, D_n, \neg D_{n+1}, \ldots, \neg D_m\}$ where $\neg C \in M_{WF}^S(\Pi)$, $\Gamma_C^S \in \Gamma$ if $C \neq \{\}$, $\Gamma_{D_i}^S \in \Gamma$, $D_i \in M_{WF}^S$, $\Gamma_{D_j}^S \in \Gamma$, $\neg D_j \in M_{WF}^S$ for all $i = 1, \ldots, n$ and $j = n+1, \ldots, m\}$
- $U_\Pi^S(\Gamma) = \{\Gamma_D^S \in \Gamma_\Pi^S \mid$ for all active strong derivation sequences in Γ_D^S the corresponding child C is true in $M_{WF}^S(\Pi)$ and $\Gamma_C^S \in \Gamma\}$
- $W_\Pi^S(\Gamma) = T_\Pi^S(\Gamma) \cup U_\Pi^S(\Gamma)$
- $W_\Pi^S \uparrow 0 = \emptyset$, $W_\Pi^S \uparrow n+1 = W_\Pi^S(W_\Pi^S \uparrow n)$ and $W_\Pi^S \uparrow \alpha = \bigcup_{\beta<\alpha} W_\Pi^S \uparrow \beta$ for limit ordinal α

The operator T_Π^S collects certain trees which contain a false child, U_Π^S accepts trees where all children are true, and W_Π^S just combines the results like W_P in case of the well-founded semantics (cf. Definition 2.42). Note that even though $\neg D \in M_{WF}^S$ we just take Γ_D^S and neither $\Gamma_{\neg D}^S$ nor $\neg\Gamma_D^S$.
If we could have shown that all the not intended properties given in Table 1 do not hold then it would not be necessary to mention the child C in case of T_Π^S and the additional conditions related to it. It would be possible to show that all the true elements in the first derivate already imply that.

Example 3.32 Reconsider the program Π from Example 3.24.

$$
\begin{aligned}
p &\leftarrow \\
q &\leftarrow \quad p \\
r &\leftarrow \quad r \\
s &\leftarrow \quad \neg r \\
t &\leftarrow \quad \neg s
\end{aligned}
$$

Using the minimal set representation, the strong well-founded model M_{WF}^S is $\{p, q, s, \neg(r \vee t)\}$. Recalling definition 3.31 we have $W_\Pi^S \uparrow 0 = \emptyset$ and then $W_\Pi^S \uparrow 1 = W_\Pi^S(\emptyset) = T_\Pi^S(\emptyset) \cup U_\Pi^S(\emptyset)$ and $T_\Pi^S(\emptyset) = \{\Gamma_p^S\}$ because Γ_p^S has only one empty child and the first derivate in the corresponding sequence is empty. We also use here the minimal representation because any tree $\Gamma_{p \vee D}^S$ for some disjunction D contains the very same active strong derivation sequence. $U_\Pi^S(\emptyset) = \{\Gamma_r^S\}$ because there is no active strong derivation sequence in Γ_r^S and $W_\Pi^S \uparrow 1 = \{\Gamma_p^S, \Gamma_r^S\}$. Then we have $W_\Pi^S \uparrow 2 = \{\Gamma_p^S, \Gamma_r^S, \Gamma_q^S, \Gamma_s^S\}$ and finally $W_\Pi^S \uparrow 3 = \{\Gamma_p^S, \Gamma_r^S, \Gamma_q^S, \Gamma_s^S, \Gamma_t^S, \Gamma_{(r \vee t)}^S\}$ which also is the least fixed point for this example. We can see that this also includes implicitly all trees whose roots are disjunctive combinations of one of these elements and any arbitrary element of EB_Π, i.e. in case of that program Π the least fixed point is in fact Γ_Π^S. Moreover, we now keep the dependency between p and q since Γ_q^S only appears in $W_\Pi^S \uparrow 2$ after we know that Γ_p^S, respectively $\Gamma_{p \vee q}^S$, is contained in $W_\Pi^S \uparrow 1$.

We will now show that this operator is monotonic.

Proposition 3.33 Given a logic program Π and the strong well-founded model $M_{WF}^S(\Pi)$, the operator W_Π^S is monotonic.

Proof: Let $\Gamma_1, \Gamma_2 \in \boldsymbol{\Gamma_\Pi^S}$, $\Gamma_1 \subseteq \Gamma_2$, and $\Gamma_D^S \in W_\Pi^S(\Gamma_1)$. We have to show that $\Gamma_D^S \in W_\Pi^S(\Gamma_2)$ as well.

Consider at first that $\Gamma_D^S \in T_\Pi^S(\Gamma_1)$. Then Γ_D^S contains an active strong derivation sequence $\{D\}, I_1, \ldots, I_r$ with child $C = \bar{I}_r$ and $I_1 = \{D_1, \ldots, D_n, \neg D_1, \ldots, \neg D_m\}$ where $\neg C \in M_{WF}^S(\Pi)$, $\Gamma_C^S \in \Gamma$ if $C \neq \{\}$, $\Gamma_{D_i}^S \in \Gamma$, $D_i \in M_{WF}^S$, $\Gamma_{D_j}^S \in \Gamma$, $\neg D_j \in M_{WF}^S$, for all $i = 1, \ldots, n$ and $j = 1, \ldots, m\}$. Since $\Gamma_1 \subseteq \Gamma_2$ we know that also $\Gamma_C^S \in \Gamma_2$ if $C \neq \{\}$, $\Gamma_{D_i}^S \in \Gamma_2$, and $\Gamma_{D_j}^S \in \Gamma_2$ for all $i = 1, \ldots, n$ and $j = 1, \ldots, m$. Hence $\Gamma_D^S \in T_\Pi^S(\Gamma_2)$ and thus $\Gamma_D^S \in W_\Pi^S(\Gamma_2)$.

Alternatively, suppose that $\Gamma_D^S \in U_\Pi^S(\Gamma_1)$. Then for all active strong derivation sequences in Γ_D^S the corresponding child C is true in $M_{WF}^S(\Pi)$ and $\Gamma_C^S \in \Gamma_1$. Again, since $\Gamma_1 \subseteq \Gamma_2$, for all children C, we have $\Gamma_C^S \in \Gamma_2$. Hence $\Gamma_D^S \in U_\Pi^S(\Gamma_2)$ and thus $\Gamma_D^S \in W_\Pi^S(\Gamma_2)$. ■

Thus we can apply Theorem 2.20 which yields that the operator W_Π^S always has a least fixed point. This least fixed point coincides with M_{WF}^S in so far that whenever a tree Γ_D^S is contained in the least fixed point then D or $\neg D$ is contained in the strong well-founded model and vice versa.

Proposition 3.34 Let Π be a disjunctive logic program. If $\Gamma_D^S \in \mathsf{lfp}(W_\Pi^S)$ then $D \in M_{WF}^S$ or $\neg D \in M_{WF}^S$.

Proof: Let $\Gamma_D^S \in \mathsf{lfp}(W_\Pi^S)$. By Theorem 2.20, we know that there is a least ordinal α such that $\mathsf{lfp}(W_\Pi^S) = W_\Pi^S \uparrow \alpha$. Then $W_\Pi^S \uparrow (\alpha+1) = W_\Pi^S \uparrow \alpha$, $W_\Pi^S \uparrow (\alpha+1) = W_\Pi^S(W_\Pi^S \uparrow \alpha)$ and $W_\Pi^S(W_\Pi^S \uparrow \alpha) = T_\Pi^S(W_\Pi^S \uparrow \alpha) \cup U_\Pi^S(W_\Pi^S \uparrow \alpha)$.
Let $\Gamma_D^S \in T_\Pi^S(W_\Pi^S \uparrow \alpha)$. By Definition 3.31, we know that there is an active strong derivation sequence with child C such that $\neg C \in M_{WF}^S$. By Definition 3.13, we know that D is true in Γ_D^S and thus, by Definition 3.14, that $D \in M_{WF}^S$.
Let $\Gamma_D^S \in U_\Pi^S(W_\Pi^S \uparrow \alpha)$. By Definition 3.31, we know that for all active strong derivation sequences with child C we have $\neg C \in M_{WF}^S$. By Definition 3.13, we know that D is false in Γ_D^S and thus by Definition 3.14 that $\neg D \in M_{WF}^S$. ■

Before we show the other direction we present a property necessary for that.

Lemma 3.35 Let Π be a disjunctive logic program and $\{D\}, I_1, \ldots, I_r$ be an active strong derivation sequence in Γ_D^S with child $C = \bar{I}_r$ such that $\neg C \in M_{WF}^S$. For all q, $1 \leq q \leq r$, if $D' \in I_q$ then $D' \in M_{WF}^S$ and if $\neg D' \in I_q$ then $\neg D' \in M_{WF}^S$.

Proof: We are going to prove that lemma for each derivate starting with the basis I_r. The basis does not contain any positive disjunctions so we only consider $\neg D' \in I_r$. Then $D' \in C$. We know that $\neg C \in M_{WF}^S$ and thus, by Lemma 3.26, that $\neg D' \in M_{WF}^S$.
Assume we have shown for I_q, $q \leq r$, that if $D' \in I_q$ then $D' \in M_{WF}^S$ and if $\neg D' \in I_q$ then $\neg D' \in M_{WF}^S$. We show that the claim holds for the derivate I_{q-1}. Let $\neg D'$ be in I_{q-1} where D' is an arbitrary atom. No rule is applicable to $\neg D'$ and thus it occurs unchanged in I_q. Then $\neg D' \in M_{WF}^S$ by assumption. Alternatively, let D' be in I_{q-1} where D' is an arbitrary disjunction. If D' is not the disjunction used in the derivation step $I_{q-1} \Leftarrow I_q$ then it occurs unchanged in I_q which by assumption means $D' \in M_{WF}^S$. So let D' be the disjunction used for the derivation step $I_{q-1} \Leftarrow I_q$. But then we can construct an active strong derivation sequence in $\Gamma_{D'}^S$ with a false child by choosing the very same clause and application rule like in $I_{q-1} \Leftarrow I_q$ for the first derivation step $\{D'\} \Leftarrow I'_1$. All elements of I'_1 occur also in I_q and we thus construct the remaining sequence by applying to each positive disjunction in I'_1 the same rule and clause as in the sequence $\{D\}, I_1, \ldots, I_r$, and likewise for each thereby obtained positive disjunction in any $I_{q'}$, $q' > 1$. We obtain the sequence $\{D'\}, I'_1, \ldots, I'_{r'}$ and $I'_{r'} \subseteq I_r$ since any negative literal obtained in any derivation step also occurs in I_r by construction. Then $\neg \bar{I}'_{r'} \in M_{WF}^S$, $\Gamma_{D'}^S$ contains a false child and thus $D' \in M_{WF}^S$, by Definition 3.13 and 3.14. ■

Unfortunately, there is no similar property for active strong derivation sequences with true children. Recall Example 3.15. We have shown that p is false because there an active strong derivation sequence with true child $(q \vee r)$. But e.g. the basis contains $\neg q$ and $\neg r$ and both are undefined in the strong well-founded model.

We now show that whenever a disjunction is true or false in the strong well-founded model then its tree is also contained in the least fixed point of W_{Π}^{S}.

Proposition 3.36 Let Π be a disjunctive logic program. If $D \in M_{WF}^{S}$ or $\neg D \in M_{WF}^{S}$ then Γ_{D}^{S} is contained in $\mathsf{lfp}(W_{\Pi}^{S})$.

Proof: Suppose that $D \in M_{WF}^{S}$ or $\neg D \in M_{WF}^{S}$. By Lemma 3.23, we know that the stage $s(D)$ is defined, thus we are going to prove by transfinite induction on $s(D)$ that Γ_{D}^{S} is contained in $\mathsf{lfp}(W_{\Pi}^{S})$.

Let $s(D)$ be 0. If D is true in M_{WF}^{S} then, by Definition 3.19, D has a stage one more than the minimum of its false children. Since there are no children with a stage smaller than 0, we conclude that D has no children and cannot be true by Definition 3.13. So D is false. By Definition 3.19, D has a stage one more than the least upper bound of all its true children, thus again Γ_{D}^{S} has no children at all, i.e. no active strong derivation sequence at all. Then $\Gamma_{D}^{S} \in U_{\Pi}^{S}(\emptyset)$, $\Gamma_{D}^{S} \in W_{\Pi}^{S}(\emptyset)$ and $\Gamma_{D}^{S} \in \mathsf{lfp}(W_{\Pi}^{S})$.

Suppose for all C with $s(C) < \alpha$ that $\Gamma_{C}^{S} \in \mathsf{lfp}(W_{\Pi}^{S})$. Let $s(D) = \alpha$. We have to consider two cases.

Let $\neg D \in M_{WF}^{S}$. By Definition 3.14 we know that D is false in Γ_{D}^{S}, i.e. for each active strong derivation sequence in Γ_{D}^{S} the corresponding child C is true. By Definition 3.19, the stage of D has to be one more than the least upper bound of all its true children, thus $s(C) < \alpha$. Then, by induction hypothesis, $\Gamma_{C}^{S} \in \mathsf{lfp}(W_{\Pi}^{S})$ for all children C, and hence, by Definition of U_{Π}^{S}, $\Gamma_{D}^{S} \in \mathsf{lfp}(W_{\Pi}^{S})$.

Let $D \in M_{WF}^{S}$. By Definition 3.14 we know that D is true in Γ_{D}^{S}, i.e. there is at least one active strong derivation sequence where the corresponding child C is false. By Definition 3.19, $s(D)$ is one more than the minimum of the stages of its false children, thus $\mathsf{min}\{s(C)\} = \beta < \alpha$. Then, by induction hypothesis, $\Gamma_{C}^{S} \in \mathsf{lfp}(W_{\Pi}^{S})$ for all false children C with minimal stage.
Let $\{D\}, I_1, \ldots, I_r$ be an active strong strong derivation sequence with child C of minimal stage where I_r is the basis. We are going to prove for all $D' \in I_q$, respectively $\neg D' \in I_q$, $0 \leq q \leq r$ where $I_0 = \{D\}$, that $\Gamma_{D'}^{S} \in \mathsf{lfp}(W_{\Pi}^{S})$. This will finish the proof, because D is the only disjunction occurring in I_0.

Let $q = r$. Since I_r is the basis there is no positive disjunction in I_r we only have to consider $\neg D' \in I_r$. Since $D' \in C$ and $\neg C \in M_{WF}^{S}$, by Lemma

3.26, we know that $\neg D' \in M^S_{WF}$ and $s(D') \leq \beta$. Then $\Gamma^S_{D'} \in \mathsf{lfp}(W^S_\Pi)$ by induction hypothesis.

Suppose we have shown the claim for q, i.e. for all $D' \in I_q$, respectively $\neg D' \in I_q$, $q \leq r$, that $\Gamma^S_{D'} \in \mathsf{lfp}(W^S_\Pi)$ and consider I_{q-1}.
Let $\neg D' \in I_{q-1}$. No rule is applicable to $\neg D'$ and it occurs unchanged in I_q, thus $\Gamma^S_{D'} \in \mathsf{lfp}(W^S_\Pi)$.
Let $D' \in I_{q-1}$. If D' is not used for the derivation $I_{q-1} \Leftarrow I_q$ then it occurs unchanged in I_q, thus $\Gamma^S_{D'} \in \mathsf{lfp}(W^S_\Pi)$ as well. So let D' be the disjunction which is used for the derivation $I_{q-1} \Leftarrow I_q$. Analogous to the proof of Lemma 3.35, we can construct an active strong derivation sequence $\{D'\}, I'_1, \ldots, I'_{r'}$ where $C' = \bar{I}'_{r'}$ is a subdisjunction of C and thus, by Lemma 3.26, $\neg C' \in M^S_{WF}$ and $s(C') \leq \beta$. Then by induction hypothesis $\Gamma^S_{C'} \in \mathsf{lfp}(W^S_\Pi)$. All elements of I'_1 also occur in I_q, so by assumption for all $D'' \in I'_1$, respectively $\neg D'' \in I'_1$, $\Gamma^S_{D''} \in \mathsf{lfp}(W^S_\Pi)$. Additionally, by Lemma 3.35, if $D'' \in I'_1$ then $D'' \in M^S_{WF}$ and if $\neg D'' \in I'_1$ then $\neg D'' \in M^S_{WF}$. But then, by definition of T^S_Π, we have that $\Gamma^S_{D'} \in \mathsf{lfp}(W^S_\Pi)$. ■

Since for every disjunction D occurring (possibly negated) in the strong well-founded model the tree Γ^S_D is contained in W^S_Π and the stage is defined, for each such disjunction we can lift Lemma 3.26 to the operator W^S_Π.

Lemma 3.37 Let Π be a disjunctive logic program and $\Gamma^S_D \in (W^S_\Pi \uparrow \alpha)$.

1. If $D \in M^S_{WF}$ then $\Gamma^S_{D'} \in (W^S_\Pi \uparrow \alpha)$ for all $\mathcal{D} \subseteq \mathcal{D}'$.

2. If $\neg D \in M^S_{WF}$ then $\Gamma^S_{D'} \in (W^S_\Pi \uparrow \alpha)$ for all $\mathcal{D}' \subseteq \mathcal{D}$.

Proof: We proof the statements by transfinite induction on α.

Let $\alpha = 0$. $W^S_\Pi \uparrow 0 = \emptyset$, by Definition 3.31, and the claim holds automatically.

Suppose the two properties hold for all ordinals β with $\beta < \alpha$ and that $\Gamma^S_D \in (W^S_\Pi \uparrow \alpha)$.

Let α be a successor ordinal, i.e. $\alpha = \beta + 1$. We then have that $\Gamma^S_D \in W^S_\Pi(W^S_\Pi \uparrow \beta)$ and $\Gamma^S_D \in (T^S_\Pi(W^S_\Pi \uparrow \beta) \cup U^S_\Pi(W^S_\Pi \uparrow \beta))$ by Definition 3.31.
If $D \in M^S_{WF}$ then Γ^S_D has a false child and thus $\Gamma^S_D \in T^S_\Pi(W^S_\Pi \uparrow \beta)$. Then Γ^S_D contains an active strong derivation sequence $\{D\}, I_1, \ldots, I_r$ with child $C = \bar{I}_r$, $I_1 = \{D_1, \ldots, D_n, \neg D_1, \ldots, \neg D_m\}$ where $\neg C \in M^S_{WF}(\Pi)$, $\Gamma^S_C \in (W^S_\Pi \uparrow \beta)$ if $C \neq \{\}$, $\Gamma^S_{D_i} \in (W^S_\Pi \uparrow \beta)$, $D_i \in M^S_{WF}$, $\Gamma^S_{D_j} \in (W^S_\Pi \uparrow \beta)$, $\neg D_j \in M^S_{WF}$, for all $i = 1, \ldots, n$ and $j = 1, \ldots, m$. Consider any disjunction D' with $\mathcal{D} \subseteq \mathcal{D}'$. By Lemma 3.25, $\Gamma^S_{D'}$ also contains an active strong derivation sequence with child C' such that $\mathcal{C} \setminus (\mathcal{D}'' \setminus \mathcal{C}') = \mathcal{C}'$ where $\mathcal{D}'' = \mathcal{D} \setminus \mathcal{D}'$ and thus $\mathcal{C}' \subseteq \mathcal{C}$. Since $C \in M^S_{WF}$, by Lemma 3.26, $\neg C' \in M^S_{WF}$. Furthermore, by induction hypothesis,

we have that $\Gamma_{C'} \in (W_\Pi^S \uparrow \beta)$. Let $\{D'\} \Leftarrow I'_1$ be the first derivation of that active strong derivation sequence where $I'_1 = \{D'_1, \ldots, D'_{n'}, \neg D'_1, \ldots, \neg D'_{m'}\}$. We reconsider the way we obtained the corresponding child C' in the proof of Lemma 3.25 and have to consider two cases:
If (S1) is applied for $\{D\} \Leftarrow I_1$ then $\mathcal{H} \subseteq \mathcal{D}$. Since $\mathcal{D} \subseteq \mathcal{D}'$, we also have $\mathcal{H} \subseteq \mathcal{D}'$ and (S1) is applied for $\{D'\} \Leftarrow I'_1$. Then, for each $D'_{i'} \in I'_1$, there is a $D_i \in I_1$ such that $\mathcal{D}_i \subseteq \mathcal{D}'_{i'}$ where $1 \leq i \leq n$ and $1 \leq i' \leq n'$. Since all D_i are true in M_{WF}^S, we know by Lemma 3.26 that each $D'_{i'} \in M_{WF}^S$. Moreover, by induction hypothesis, we have that $\Gamma_{D'_{i'}} \in (W_\Pi^S \uparrow \beta)$. The sets of negated atoms obtained in I_1 and I'_1 are identical, i.e. $\neg D'_{j'} \in M_{WF}^S$ and $\Gamma_{D'_{j'}}^S \in (W_\Pi^S \uparrow \beta)$ for all $j' = 1, \ldots, m'$. But then $\Gamma_{D'}^S \in T_\Pi^S(W_\Pi^S \uparrow \beta)$.
If (S2) is applied for $\{D\} \Leftarrow I_1$ then $\mathcal{H} \not\subseteq \mathcal{D}$ and $\mathcal{H} \setminus \mathcal{D} \neq \emptyset$. Since $\mathcal{D} \subseteq \mathcal{D}'$, we have $\mathcal{H} \cap \mathcal{D}' \neq \emptyset$ as well and either (S1) or (S2) is applied in the derivation step $\{D'\} \Leftarrow I'_1$. In both cases, for each $D'_{i'} \in I'_1$, there is a $D_i \in I_1$ such that $\mathcal{D}_i \subseteq \mathcal{D}'_{i'}$ where $1 \leq i \leq n$ and $1 \leq i' \leq n'$. Again, since all D_i are true in M_{WF}^S, we know by Lemma 3.26 that each $D'_{i'} \in M_{WF}^S$ and, by induction hypothesis, that $\Gamma_{D'_{i'}} \in (W_\Pi^S \uparrow \beta)$. If we applied (S1) for $\{D'\} \Leftarrow I'_1$ then for each $\neg D'_{j'} \in I'_1$ there is a $\neg D_j \in I_1$ such that $D_j = D'_{j'}$ where $1 \leq j \leq m$ and $1 \leq j' \leq m'$. In case of (S2) we have $\mathcal{H} \setminus \mathcal{D}' \subseteq \mathcal{H} \setminus \mathcal{D}$, so for each $\neg D'_{j'} \in I'_1$ there is a $\neg D_j \in I_1$ with $D_j = D'_{j'}$, $1 \leq j \leq m$ and $1 \leq j' \leq m'$. Thus in both cases, for all $j' = 1, \ldots, m'$, $\neg D'_{j'} \in M_{WF}^S$ and $\Gamma_{D'_{j'}}^S \in (W_\Pi^S \uparrow \beta)$ and thus $\Gamma_{D'}^S \in T_\Pi^S(W_\Pi^S \uparrow \beta)$ as well. Altogether, $\Gamma_{D'}^S \in W_\Pi^S(W_\Pi^S \uparrow \beta)$, no matter whether (S1) or (S2) is applied in the first derivation step and thus $\Gamma_{D'}^S \in W_\Pi^S \uparrow \alpha$.
If $\neg D \in M_{WF}^S$ then all children C in Γ_D^S are true and $\Gamma_D^S \in U_\Pi^S(W_\Pi^S \uparrow \beta)$. Then for each active strong derivation sequence in Γ_D^S with true child C we have $\Gamma_C^S \in (W_\Pi^S \uparrow \beta)$. Consider any D' with $\mathcal{D}' \subseteq \mathcal{D}$. If $\Gamma_{D'}^S$ does not contain any active derivation sequences then $\Gamma_{D'}^S \in (W_\Pi^S \uparrow 1)$ and the claim holds by monotonicity. Thus consider alternatively any active strong derivation sequence with child C'. Then, by Lemma 3.25, Γ_D^S also contains an active strong derivation sequence with child C such that $\mathcal{C}' \setminus (\mathcal{D}'' \setminus \mathcal{C}) = \mathcal{C}$ where $\mathcal{D}'' = \mathcal{D} \setminus \mathcal{D}'$ and thus $\mathcal{C} \subseteq \mathcal{C}'$. Since C is true in M_{WF}^S, by Lemma 3.26, $C' \in M_{WF}^S$ as well. Furthermore, by induction hypothesis, we have that $\Gamma_{C'} \in (W_\Pi^S \uparrow \beta)$. Since this holds for each child in $\Gamma_{D'}^S$ we know by definition of U_Π^S that $\Gamma_{D'}^S \in U_\Pi^S(W_\Pi^S \uparrow \beta)$ and thus $\Gamma_{D'}^S \in W_\Pi^S(W_\Pi^S \uparrow \beta)$, hence $\Gamma_{D'}^S \in W_\Pi^S \uparrow \alpha$.

Alternatively, let α be a limit ordinal. Then $W_\Pi^S \uparrow \alpha = \bigcup_{\beta<\alpha} W_\Pi^S \uparrow \beta$. If $\Gamma_D^S \in \bigcup_{\beta<\alpha}(W_\Pi^S \uparrow \beta)$ then there is a least ordinal β such that $\Gamma_D^S \in (W_\Pi^S \uparrow \beta)$ with $\alpha > \beta$. Assume that β is itself a limit ordinal. Then $\Gamma_D^S \in \bigcup_{\gamma<\beta} W_\Pi^S \uparrow \gamma$

and β is not the least ordinal such that $\Gamma_D^S \in W_\Pi^S \uparrow \beta$. So β has to be a successor ordinal. In this case we can apply the very same argument we used in case that α is a successor ordinal since $\beta < \alpha$. ∎

We now extend the notion of a level mapping to disjunctive literals.

Definition 3.38 For a disjunctive program Π and a disjunctive interpretation I an *disjunctive I-partial level mapping* for Π is a partial mapping $l : EB_\Pi \rightarrow \alpha$ with domain $\mathsf{dom}(l) = \{D \mid D \in I \text{ or } \neg D \in I\}$, where α is some (countable) ordinal. Every such mapping is extended to negated disjunctions by setting $l(\neg D) = l(D)$ for all $D \in EB_\Pi$.

In the following we introduce a level mapping characterization.

Definition 3.39 Let Π be a disjunctive logic program, let I be a model for Π, and let l be a disjunctive partial level mapping for Π. We say that Π satisfies (SWF) with respect to I and l if each $D \in \mathsf{dom}(l)$ satisfies one of the following conditions:

> (SWFi) $D \in I$ and Γ_D^S contains an active strong derivation sequence with child C, $\neg C \in I$ and $l(D) > l(C)$ if $C \neq \{\}$, and there is a clause $\mathcal{H} \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$ which is used for the first derivation of that sequence such that $\neg B_j \in I$ and $l(D) > l(B_j)$, $1 \leq j \leq m$, and one of the following conditions holds:
>
> > (SWFia) $\mathcal{H} \subseteq \mathcal{D}$, $(A_i \vee D) \in I$ and $l(D) > l(A_i \vee D)$, $1 \leq i \leq n$.
> >
> > (SWFib) $\mathcal{H} \not\subseteq \mathcal{D}$, $\{C_1, \ldots, C_l\} = \mathcal{H} \setminus \mathcal{D} \neq \emptyset$, $A_i \in I$ and $l(D) > l(A_i)$, $1 \leq i \leq n$, and $\neg C_k \in I$ and $l(D) > l(C_k)$, $1 \leq k \leq l$.
>
> (SWFii) $\neg D \in I$ and for each active strong derivation sequence in Γ_D^S with child $C \in I$ there is a clause $\mathcal{H} \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$ which is used for the first derivation of that sequence such that (at least) one of the following conditions holds:
>
> > (SWFiia') $\mathcal{H} \subseteq \mathcal{D}$ and there exists i, $1 \leq i \leq n$, with $\neg(A_i \vee D) \in I$, $l(D) \geq l(A_i \vee D)$.
> >
> > (SWFiia") $\mathcal{H} \not\subseteq \mathcal{D}$, $\mathcal{H} \setminus \mathcal{D} \neq \emptyset$, and there exists i with $\neg A_i \in I$, $l(D) \geq l(A_i)$, $1 \leq i \leq n$.
> >
> > (SWFiib') $\mathcal{H} \subseteq \mathcal{D}$ and there exists D' with $\mathcal{D}' \subseteq \mathcal{B}$, $D' \in I$ and $l(D) > l(D')$.
> >
> > (SWFiib") $\mathcal{H} \not\subseteq \mathcal{D}$, $\mathcal{C} = (\mathcal{H} \setminus \mathcal{D}) \neq \emptyset$, and there exists D' with $\mathcal{D}' \subseteq (\mathcal{B} \cup \mathcal{C})$, $D' \in I$ and $l(D) > l(D')$.

(SWFiic) $l(D) > l(C)$.

It should be mentioned that (SWFib) would not be necessary because there is always an active strong derivation sequence only by means of (S1) but we cannot show that without solving the open problems shown in Table 1.

Now we show the following equivalence between the strong well-founded model and Definition 3.39.

Theorem 3.40 Let Π be a disjunctive program with strong well-founded model M. Then, in the disjunctive knowledge ordering, M is the greatest model amongst all models I for which there exists a disjunctive I-partial level mapping l for Π such that Π satisfies (SWF) with respect to I and l.

Proof: Let M be the strong well-founded model of Π. We define the disjunctive M-partial level mapping l in the following way: $l(D) = \alpha$, where α is the least ordinal such that $\Gamma_D^S \in (W_\Pi^S \uparrow (\alpha + 1)) = W_\Pi^S(W_\Pi^S \uparrow \alpha)$. This mapping is well defined since we have shown in Lemma 3.36 that the tree of each disjunctive literal which occurs in M is contained in the least fixed point of W_Π^S. We show that Π satisfies (SWF) with respect to M and l. Let $D \in \mathsf{dom}(l)$ and $l(D) = \alpha$. We have to consider two cases:
If $D \in M$ then there is an active strong derivation sequence in Γ_D^S with false child. Since $\Gamma_D^S \in W_\Pi^S(W_\Pi^S \uparrow \alpha) = T_\Pi^S(W_\Pi^S \uparrow \alpha) \cup U_\Pi^S(W_\Pi^S \uparrow \alpha)$, we know that $\Gamma_D^S \in T_\Pi^S(W_\Pi^S \uparrow \alpha)$ by definition of T_Π^S and U_Π^S. By Definition 3.31, Γ_D^S contains an active strong derivation sequence $\{D\}, I_1, \ldots, I_r$ with child $C = \bar{I}_r$ and $I_1 = \{D_1, \ldots, D_n, \neg D_1, \ldots, \neg D_m\}$ where $\neg C \in M$, $\Gamma_C^S \in (W_\Pi^S \uparrow \alpha)$ if $C \neq \{\}$, $\Gamma_{D_i}^S \in (W_\Pi^S \uparrow \alpha)$, $D_i \in M$, $\Gamma_{D_j}^S \in (W_\Pi^S \uparrow \alpha)$, $\neg D_j \in M$ for all $i = 1, \ldots, n$ and $j = 1, \ldots, m$. Let $\mathcal{H} \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_{m'}$ be the clause used in the first derivation step $\{D\} \Leftarrow I_1$. If we use (S1) for that derivation step then $\mathcal{H} \subseteq \mathcal{D}$ and $I_1 = \{A_1 \vee D, \ldots, A_n \vee D, \neg B_1, \ldots, \neg B_{m'}\}$ by Definition 3.8. We already know that all positive disjunctions and all negative literals occurring in I_1 are contained in M. Moreover, since all these disjunctions, negative literals and the child C occur in $(W_\Pi^S \uparrow \alpha)$ we know that $l(A_i \vee D) < \alpha$, $l(B_j) < \alpha$, and $l(C) < \alpha$ if $C \neq \{\}$. Then D satisfies (SWFia) with respect to M and l. If we use (S2) for the first derivation step $\{D\} \Leftarrow I_1$ then $\mathcal{H} \not\subseteq \mathcal{D}$, $\{C_1, \ldots, C_l\} = \mathcal{H} \setminus \mathcal{D}$, and $I_1 = \{A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_{m'}, \neg C_1, \ldots, \neg C_l\}$. Again, all the positive disjunctions and all negative literals occurring in I_1 are contained in M. Likewise, all these disjunctions, negative literals, and the child C occur in $(W_\Pi^S \uparrow \alpha)$ and $l(A_i) < \alpha$, $l(B_j) < \alpha$, $l(C_k) < \alpha$ and $l(C) < \alpha$. Then D satisfies (SWFib) with respect to M and l.
If $\neg D \in M$ then each child C of an active strong derivation sequence in Γ_D^S is true, thus $\Gamma_D^S \in U_\Pi^S(W_\Pi^S \uparrow \alpha)$ since $\Gamma_D^S \in W_\Pi^S(W_\Pi^S \uparrow \alpha) = T_\Pi^S(W_\Pi^S \uparrow \alpha) \cup U_\Pi^S(W_\Pi^S \uparrow \alpha)$. Then by Definition 3.31, for all active strong derivation

sequences in Γ_D^S the corresponding child C is true in M and $\Gamma_C^S \in (W_\Pi^S \uparrow \alpha)$. Then $l_S(C) < \alpha$ and, for all derivation sequences with child C, D satisfies (SWFiic).

Alternatively, we show that if I is a model of Π and l a disjunctive I-partial level mapping such that Π satisfies (SWF) with respect to I and l then $I \subseteq M_{WF}^S$. We show via transfinite induction on $\alpha = l(D)$, that whenever $D \in I$, respectively $\neg D \in I$, then $\Gamma_D^S \in (W_\Pi^S \uparrow (\alpha+1))$, i.e. $\Gamma_D^S \in W_\Pi^S(W_\Pi^S \uparrow \alpha) = T_\Pi^S(W_\Pi^S \uparrow \alpha) \cup U_\Pi^S(W_\Pi^S \uparrow \alpha)$ and thus $\Gamma_D^S \in \mathsf{lfp}(W_\Pi^S)$. Then D, respectively $\neg D$, occurs in M_{WF}^S by Proposition 3.34.
Let $l(D) = 0$. If $D \in I$ then by (SWFi) Γ_D^S contains an active strong derivation sequence with child C, $\neg C \in I$ and $l(D) > l(C)$ if $C \neq \{\}$, and there is a clause $\mathcal{H} \leftarrow A_1, \dots, A_n, \neg B_1, \dots, \neg B_m$ in $\mathsf{ground}(\Pi)$ which is used for the first derivation of that sequence such that $\neg B_j \in I$ and $l(D) > l(B_j)$, $1 \leq j \leq m$, and either (SWFia) or (SWFib) holds. If (SWFia) holds then $\mathcal{H} \subseteq \mathcal{D}$, $(A_i \vee D) \in I$ and $l(D) > l(A_i \vee D)$, $1 \leq i \leq n$. But there is no ordinal smaller than 0, i.e. the clause is a fact. Thus there is an active strong derivation sequence $\{D\} \Leftarrow \{\}$ with false child and $\Gamma_D^S \in (W_\Pi^S \uparrow 1)$ by definition of T_Π^S. If (SWFib) holds then $\mathcal{H} \not\subseteq \mathcal{D}$, $\{C_1, \dots, C_l\} = \mathcal{H} \setminus \mathcal{D} \neq \emptyset$, $A_i \in I$ and $l(D) > l(A_i)$, $1 \leq i \leq n$, and $\neg C_k \in I$ and $l(D) > l(C_k)$, $1 \leq k \leq l$. There is no ordinal smaller than 0, so there can be no C_k in I_1 and (SWFib) cannot hold.
If $\neg D \in I$ then by (SWFii) for each active strong derivation in Γ_D^S with child $C \in I$ there is a clause $\mathcal{H} \leftarrow A_1, \dots, A_n, \neg B_1, \dots, \neg B_m$ in $\mathsf{ground}(\Pi)$ which is used for the first derivation of that sequence such that at least one of (SWFiia'), (SWFiia"), (SWFiib'), (SWFiib"), and (SWFiic) holds. If (SWFiia') holds then $\mathcal{H} \subseteq \mathcal{D}$ and there exists i, $1 \leq i \leq n$, with $\neg(A_i \vee D) \in I$, $l(D) \geq l(A_i \vee D)$. If (SWFiia") is satisfied then $\mathcal{H} \not\subseteq \mathcal{D}$, $\mathcal{H} \setminus \mathcal{D} \neq \emptyset$, and there exists i with $\neg A_i \in I$, $l(D) \geq l(A_i)$, $1 \leq i \leq n$. If (SWFiib') holds then $\mathcal{H} \subseteq \mathcal{D}$ and there exists D' with $\mathcal{D}' \subseteq \mathcal{B}$, $D' \in I$ and $l(D) > l(D')$. If (SWFiib") is satisfied then $\mathcal{H} \not\subseteq \mathcal{D}$, $\mathcal{C} = (\mathcal{H} \setminus \mathcal{D}) \neq \emptyset$, and there exists D' with $\mathcal{D}' \subseteq (\mathcal{B} \cup \mathcal{C})$, $D' \in I$ and $l(D) > l(D')$ and if (SWFiic) holds then $l(D) > l(C)$. Since there is no ordinal smaller than 0, (SWFiib'), (SWFiib"), and (SWFiic) cannot hold. For the same reason, in case of (SWFiia') and (SWFiia") we can only have $l(D) = l(A_i \vee D)$, $l(D) = l(A_i)$ respectively. Since $(A_i \vee D)$, respectively A_i, is false in I, it has to satisfy (SWF) and the only possibility is again (SWFiia') or (SWFiia") with a positive disjunction of the same level which is false in I. Then this disjunction also has to satisfy (SWF) and the argument can be applied infinitely often. But the considered derivation sequence is active, thus we know that it is finite, so neither (SWFiia') nor (SWFiia") can hold. Hence, Γ_D^S does not have any children and thus $\Gamma_D^S \in (U_\Pi^S \uparrow 1)$ and $\Gamma_D^S \in (W_\Pi^S \uparrow 1)$.

Assume for all $D' \in EB_\Pi$ with $l(D') < \alpha$ that if $D' \in I$, respectively $\neg D' \in I$, then $\Gamma^S_{D'} \in W^S_\Pi \uparrow \alpha$ and let $l(D) = \alpha$. We have to consider two cases again.

If $D \in I$ then, by (SWFi), Γ^S_D contains an active strong derivation sequence with child C, $\neg C \in I$ and $l(D) > l(C)$ if $C \neq \emptyset$, and there is a clause $\mathcal{H} \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$ which is used for the first derivation of that sequence such that $\neg B_j \in I$ and $l(D) > l(B_j)$, $1 \leq j \leq m$, and either (SWFia) or (SWFib) holds. If (SWFia) holds then $\mathcal{H} \subseteq \mathcal{D}$, $(A_i \vee D) \in I$ and $l(D) > l(A_i \vee D)$, $1 \leq i \leq n$. Then $I_1 = \{A_1 \vee D, \ldots, A_n \vee D, \neg B_1, \ldots, \neg B_m\}$ is the derivate of the first derivation of that sequence. By induction hypothesis, the trees of all elements contained in I_1 and Γ^S_C occur in $W^S_\Pi \uparrow \alpha$ and thus in the least fixed point of W^S_Π. Then all elements of I_1 and C are contained in M^S_{WF} by Proposition 3.34. Hence $\Gamma^S_D \in T^S_\Pi \uparrow (\alpha + 1)$, by Definition 3.31, and $\Gamma^S_D \in W^S_\Pi \uparrow (\alpha + 1)$. If (SWFib) holds then $\mathcal{H} \not\subseteq \mathcal{D}$, $\{C_1, \ldots, C_l\} = \mathcal{H} \setminus \mathcal{D} \neq \emptyset$, $A_i \in I$ and $l(D) > l(A_i)$, $1 \leq i \leq n$, and $\neg C_k \in I$ and $l(D) > l(C_k)$, $1 \leq k \leq l$. Then $I_1 = \{A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m, \neg C_1, \ldots, \neg C_l\}$ is the derivate of the first derivation of that sequence. By induction hypothesis, the trees of all elements contained in I_1 and Γ^S_C occur in $W^S_\Pi \uparrow \alpha$ and thus, again, all elements of I_1 and C are contained in M^S_{WF} by Proposition 3.34. Hence $\Gamma^S_D \in T^S_\Pi \uparrow (\alpha + 1)$, by Definition 3.31, and thus $\Gamma^S_D \in W^S_\Pi \uparrow (\alpha + 1)$.

If $\neg D \in I$ then, by (SWFii), for each active strong derivation in Γ^S_D with child $C \in I$ there is a clause $\mathcal{H} \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$ which is used for the first derivation of that sequence such that at least one of (SWFiia'), (SWFiia"), (SWFiib'), (SWFiib"), and (SWFiic) holds. Consider such an arbitrary active strong derivation. We show that $l(D) > l(C)$ holds in all these cases.

- If (SWFiic) holds then $l(D) > l(C)$ is satisfied automatically.
- If (SWFiib') holds then $\mathcal{H} \subseteq \mathcal{D}$ and there exists D' with $\mathcal{D}' \subseteq \mathcal{B}$, $D' \in I$ and $l(D) > l(D')$. By Definition 3.8, all $\neg B_j$, $j = 1 \ldots, m$, occur in I_1, the first derivate of that sequence. No more rule is applicable to any $\neg B_j$ and it also occurs in the basis, thus $B_j \in C$ and so $\mathcal{D}' \subseteq \mathcal{C}$. By induction hypothesis, $\Gamma^S_{D'} \in (W^S_\Pi \uparrow \alpha)$. Then, by Lemma 3.37, $\Gamma_C \in (W^S_\Pi \uparrow \alpha)$ as well and $l(C) < \alpha$ and thus $l(D) > l(C)$.
- If (SWFiib") is satisfied then $\mathcal{H} \not\subseteq \mathcal{D}$, $\mathcal{C}' = (\mathcal{H} \setminus \mathcal{D}) \neq \emptyset$, and there exists D' with $\mathcal{D}' \subseteq (\mathcal{B} \cup \mathcal{C}')$, $D' \in I$ and $l(D) > l(D')$. By Definition 3.8, all elements in $\mathcal{B} \cup \mathcal{C}'$ occur negated in I_1, the first derivate of that sequence. No more rule is applicable to any negated atom and it also occurs in the basis, thus $\mathcal{D}' \subseteq \mathcal{C}$. By induction hypothesis, $\Gamma^S_{D'} \in (W^S_\Pi \uparrow \alpha)$.

Then, by Lemma 3.37, $\Gamma_C \in (W_\Pi^S \uparrow \alpha)$ as well and $l(C) < \alpha$ and thus $l(D) > l(C)$.

- If (SWFiia') holds then $\mathcal{H} \subseteq \mathcal{D}$ and there exists i, $1 \leq i \leq n$, with $\neg(A_i \vee D) \in I$, $l(D) \geq l(A_i \vee D)$ and if (SWFiia") is satisfied then $\mathcal{H} \not\subseteq \mathcal{D}$, $\mathcal{H} \setminus \mathcal{D} \neq \emptyset$, and there exists i with $\neg A_i \in I$, $l(D) \geq l(A_i)$, $1 \leq i \leq n$. We join these two cases since the argument is exactly the same and prove for all positive disjunctions D' occurring in the active strong derivation sequence $\{D\}, I_1, \ldots, I_r$ with $\neg D' \in M_{WF}^S$, $l(D') \leq l(D)$, and D' satisfies (SWFiia') or (SWFiia") that $l(C) < l(D)$. Then the claim also holds for D itself.

  Since I_r is the basis, D' cannot occur in I_r. The same holds for I_{r-1} because I_r cannot contain any $A_i \vee D'$ or A_i which would be necessary for $D' \in I_{r-1}$ satisfying (SWFiia') or (SWFiia"). So let $D' \in I_{r-2}$. Then I_{r-1} contains (at least) one disjunction D'' with $D'' = A_i \vee D'$ or $D'' = A_i$ but in both cases D'' cannot satisfy (SWFiia') or (SWFiia") as already mentioned. Since D'' occurs in I_{r-1}, the clause which is used for $I_{r-1} \Leftarrow I_r$ cannot contain positive literals in the body because I_r is the basis. Thus there is also an active strong derivation sequence in $\Gamma_{D''}^S$ using this clause with child C'. Since $\neg D'' \in I$ we know that D'' satisfies (SWFii) and it cannot satisfy (SWFiia') or (SWFiia") so $l(C') < l(D'')$ as we have already shown for the other cases. We know $l(D) \geq l(D')$ and $l(D') \geq l(D'')$ so we can apply the induction hypothesis and $C' \in (W_\Pi^S \uparrow \alpha)$ and thus $C' \in M_{WF}^S$. By construction, C' is a subset of the child C for the considered sequence in Γ_D^S and, by Lemma 3.37, we know that $C \in (W_\Pi^S \uparrow \alpha)$. Thus $l(C) < l(D)$.

  Finally, consider for all positive disjunctions occurring in I_q with $\neg D' \in M_{WF}^S$, $l(D') \leq l(D)$, and D' satisfies (SWFiia') or (SWFiia") that $l(C) < l(D)$ has been shown. We show that it also holds for all these D' in I_{q-1}. If D' is not used in the derivation step $I_{q-1} \Leftarrow I_q$ then it occurs unchanged in I_q and the claim holds by assumption. So let D' be the disjunction used for that derivation step such that there is either $A_i \vee D'$ or A_i with $l(D') \geq l(A_i \vee D')$, respectively $l(D') \geq l(A_i)$. If $A_i \vee D$, respectively A_i, satisfies (SWFiia') or (SWFiia") then the claim has already been proven. If not, then we can apply the very same argument for I_{q-1} which we used for I_{r-2} above, and the claim holds as well.

Thus, for all derivation sequences with child C, we have $l(D) > l(C)$ and $\Gamma_C^S \in (W_\Pi^S \uparrow \alpha)$. Then $\Gamma_D^S \in U_\Pi^S(W_\Pi^S \uparrow \alpha)$ by definition of U_Π and $\Gamma_D^S \in W_\Pi^S \uparrow (\alpha + 1)$ by Definition 3.31. ■

Once more we mention that without the problems shown in Table 1 the condition (SWFi) would not contain the case (SWFib) and even the reference to the active derivation sequence and the child would not appear. It is only necessary for proving the statement which we cannot do in a better way without solving the mentioned problems.

In case of (SWFii) we only used (SWFiic) for the proof, respectively reduced all the other cases to (SWFiic). Thus the following corollary is straightforward.

Corollary 3.41 Let Π be a disjunctive program with strong well-founded model M. Then, in the disjunctive knowledge ordering, M is the greatest model amongst all models I for which there exists a disjunctive I-partial level mapping l for Π such that Π satisfies (SWF') with respect to I and l where (SWF') is (SWF) substituting (SWFii) by (SWFii').

> (SWFii') $\neg D \in I$ and for each active strong derivation in Γ_D^S with child $C \in I$ we have $l(D) > l(C)$.

This is apparently much shorter then Definition 3.39. We just prefer the more detailed version because it allows to compare the result in a better way to other characterizations.
Moreover, (SWFiic) does not only cover the four other conditions but also combines knowledge derived from several clauses.

Example 3.42 Let Π be the following program.

$$\begin{aligned} p \vee q \leftarrow &\quad r \\ r \leftarrow &\quad \neg s \\ q \vee s \leftarrow & \end{aligned}$$

We have $\{p\} \Leftarrow \{r, \neg q\} \Leftarrow \{\neg q, \neg s\}$ as the only active derivation sequence in Γ_p^S. Since $q \vee s$, the child, is true, p has to be false. However neither $\neg q$, nor r, nor $\neg s$ are false in the strong well-founded model. Thus (SWF) only holds in this case because of (SWFiic).

The following example shows that in case of false disjunctions (SWFii) can be satisfied in different ways even apart from (SWFiic).

Example 3.43

$$\begin{aligned} p \leftarrow &\quad \neg q, \neg r, \neg s \\ r \vee s \leftarrow & \\ q \leftarrow &\quad \neg t \end{aligned}$$

We have that $\{p\} \Leftarrow \{\neg q, \neg r, \neg s\}$ with child $q \vee r \vee s$. We know that q is true since there is only one false child t with $l(t) = 0$ in Γ_q^S. Then $l(q) = 1$ and $l(p) = 2$ by (SWFiib'). But $r \vee s$ is true as well with the empty child even though none of them is true nor false and $l(r \vee s) = 0$ so that $l(p) = 1$ by (SWFiib') and we prefer this case since it yields a smaller level for p.

We wanted to have a level mapping characterization which for normal programs coincides with the characterization of the well-founded semantics. Unfortunately, we are not able to prove that Definition 2.46 and Definition 3.39 coincide for normal programs even restricted to non-disjunctive literals. The problem is that we also have to show that (WF) implies (SWF) and for that we need to introduce derivation sequences and e.g. the false child in case of (WFi). This remains unsolved as we already mentioned for the characterization (SWF) because of the problems shown in Table 1. However, we may give intuitive arguments that the equivalence should hold. Considering normal logic programs all the heads of the clauses just contain one atom. Then rule (S2) is never applicable and thus (SWFib), (SWFiia'), and (SWFiib") cannot be satisfied. Ignoring the reference to the derivation sequence and to the child we see that the conditions in (WFi) and (SWFia) are quite similar. The only difference is that in (SWFia) all $A_i \vee D$ have to be true whereas in (WFi) all A_i have to be true. But since we are dealing with normal programs a disjunction $A_i \vee D$ should only be true if at least one of the disjuncts is true and since we use this derivation to prove that D is true it should be A_i which is true. Likewise (WFiia) should correspond to (SWFiia') with the same additional difference. Ignoring the derivation sequence and the child (WFiib) and (SWFiib') should be equivalent. The only difference, one single negated atom is false vs. a subset of negated atoms is false, should again be resolved by the fact that we are dealing with normal logic programs without indefinite information. The only remaining condition in (SWF) is (SWFiic) which should not be necessary since we are restricted to normal logic programs but, as already mentioned, we cannot prove that in general.

4 Generalized Disjunctive Well-founded Semantics

Baral, Lobo, and Minker introduced the generalized disjunctive well-founded semantics [3] as an extension to the generalized well-founded semantics defined for normal programs [4]. The reason was that the interpretations used in [4] could not reveal the indefinite information contained in disjunctive logic programs. This is why they extended the notion of an interpretation but instead of using a disjunctive three-valued interpretation like in the previous section, they introduced state-pairs consisting of true disjunctions and false conjunctions.

Definition 4.1 A *state-pair* I is a pair (A, B), where A is a subset of EB_Π such that for all D' if $D \in A$ and $\mathcal{D} \subseteq \mathcal{D}'$ then $D' \in A$ and F is a subset of CB_Π such that for all C' if $C \in B$ and $\mathcal{C} \subseteq \mathcal{C}'$ then $C' \in B$. The disjunctions in A are mapped to $\mathbf{t}$, the conjunctions in B are mapped to $\mathbf{f}$, and all the other disjunctions and conjunctions to $\mathbf{u}$. A state-pair can also be represented by the *signed state pair* $A \cup \neg B$.

As in the previous section, there is a notion of consistency for state-pairs.

Definition 4.2 A state-pair (A, B) is consistent if $I = A \cup \neg B$ is consistent, i.e. whenever $D \in A$ then there is at least one $D' \in D$ such that $D' \notin B$ and whenever $C \in B$ then there is at least one $C' \in C$ such that $C' \notin A$.

There are some differences compared to the notion of a three-valued interpretation from the previous section. First of all, false information is represented by conjunctions and not by disjunctions which yields further differences. State-pairs are closed by definition and we allow the occurrence of indefinite negative information: knowing that a conjunction is false does not guarantee that all conjuncts are false a well. Furthermore, a state-pair is not necessarily consistent in opposite to disjunctive interpretations.

Example 4.3 Given the program Π with only one clause $p \vee q$ then $I_1 = \{p, (p \vee q), \neg(p \wedge q)\}$ is a consistent state-pair even though it contains $(p \vee q)$ and $\neg(p \wedge q)$. The indefinite information in I_1 only states that p is true and therefore also $(p \vee q)$ is and that not both p and q can be true. Thus q is either undefined or false. That is why $I_2 = \{p, q, (p \vee q), \neg(p \wedge q)\}$ is not consistent since there is no atom in $\neg(p \wedge q)$ which is not contained in I_2.

Since state-pairs are closed by definition we may represent them by minimal sets only containing those disjunctions and conjunctions which contain the other disjunctions and conjunctions. More formally, by following Definition 4.1, if D occurs in state-pair I then any superset of D is contained implicitly in I and likewise if $\neg C \in I$ then any superset of C is a false conjunction. Thus I_1 from the example above can also be written as $\{p, \neg(p \wedge q)\}$.

A model is defined as in Definition 3.5 only substituting the disjunctive three-valued interpretation by state-pair and the same holds for Definition 3.3 and 3.4 and even for Definition 3.6, the disjunctive knowledge ordering.

Now we recall the operators $\mathcal{T}_S^D$ and $\mathcal{F}_S^D$ for disjunctive logic programs which are generalizations of operators presented in [4].

Definition 4.4 Let S be a state-pair and Π be a disjunctive program. Let $T \subseteq EB_\Pi$ and $F \subseteq CB_\Pi$.

$\mathcal{T}_S^D(T) = \{D \in EB_\Pi \mid D$ is undefined in S and $H \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$ such that for all i, $1 \leq i \leq n$, $(A_i \vee D_i) \in S$ or $(A_i \vee D_i) \in T$, D_i might be empty, $\neg B_j \in S$ for all j, $1 \leq j \leq m$, and $(\mathcal{H} \cup \bigcup_i \mathcal{D}_i) \subseteq \mathcal{D}.\}$

$\mathcal{F}_S^D(F) = \{C \in CB_\Pi \mid C$ is undefined in S, $A \in C$, and for all clauses $H \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$, with $A \in H$, at least one of the following three cases holds:

- $(B_1 \vee \cdots \vee B_m) \in S$.
- $\neg(A_1 \wedge \cdots \wedge A_n) \in S$.
- $\neg(A_1 \wedge \cdots \wedge A_n) \in F$.

The derivable knowledge by means of these operators, T_S^D and F_S^D, is then obtained in the straightforward way:

Definition 4.5

$$\mathcal{T}_S^D \uparrow 0 = \emptyset. \;\; \mathcal{T}_S^D \uparrow (n+1) = \mathcal{T}_S^D(\mathcal{T}_S^D \uparrow n). \;\; \mathcal{T}_S^D = \bigcup_{n<\omega} \mathcal{T}_S^D \uparrow n.$$

$$\mathcal{F}_S^D \downarrow 0 = CB_\Pi. \;\; \mathcal{F}_S^D \downarrow (n+1) = \mathcal{F}_S^D(\mathcal{F}_S^D \downarrow n). \;\; \mathcal{F}_S^D = \bigcap_{n<\omega} \mathcal{F}_S^D \downarrow n.$$

We restrict the iteration to ordinals below ω and iterate $\mathcal{F}_S^D$ from the top: beginning with the conjunctive Herbrand base we remove conjunctions instead of adding them until the fixed point is reached.

Example 4.6 We extend the program from Example 3.9 to this program Π to demonstrate a calculation of the operators defined above.

$$\begin{aligned} p &\leftarrow q \\ p &\leftarrow r \\ q \vee r &\leftarrow \\ s &\leftarrow s \end{aligned}$$

Let S be $\emptyset$. $\mathcal{T}_S^D \uparrow 0 = \emptyset$ as defined. Then $\mathcal{T}_S^D \uparrow 1 = \mathcal{T}_S^D(\emptyset) = \{q \vee r\}$ since the third clause is the only fact and S is empty. Note that the subset inclusion in the last condition of the definition of $\mathcal{T}_S^D$ guarantees that any superset of $\{q, r\}$ is also a true disjunction. We just show the minimal set, all supersets are contained implicitly. Then we calculate $\mathcal{T}_S^D \uparrow 2 = \mathcal{T}_S^D(\{q \vee r\})$. We have $(q \vee r)$ and in the first clause q in the body, thus we can derive $(p \vee r)$ and likewise $(p \vee q)$ by the second clause, i.e. $\mathcal{T}_S^D \uparrow 2 = \{q \vee r, p \vee q, p \vee r\}$. We have $(p \vee q)$ and again q in the body of the first clause, i.e. we derive additionally p and $\mathcal{T}_S^D \uparrow 3 = \{q \vee r, p\}$ which is also the least fixed point and so $\mathcal{T}_S^D = \bigcup_{n<\omega} \mathcal{T}_S^D \uparrow n = \{q \vee r, p\}$. Note that $(p \vee q)$ and $(p \vee r)$ are still true because of p.
Now $\mathcal{F}_S^D \downarrow 0 = CB_\Pi = \{p, q, r, s\}$ in case of Π. Then $\mathcal{F}_S^D \downarrow 1 = \{p, s\}$ because there is no clause supporting q or r according to Definition 4.4 whereas p occurs in $\mathcal{F}_S^D \downarrow 1$ since e.g. the first clause contains q, and $s \in \mathcal{F}_S^D \downarrow 1$ because the forth clause contains s and both, q and s, are contained in $\mathcal{F}_S^D \downarrow 0$. Then $\mathcal{F}_S^D \downarrow 2 = \{s\}$ because neither q nor r are contained in $\mathcal{F}_S^D \downarrow 1$, thus removing p, and we obtained the fixed point. Hence, $\mathcal{F}_S^D = \bigcap_{n<\omega} \mathcal{F}_S \downarrow n = \{s\}$.

So given a state-pair S, $\mathcal{T}_S^D \cup \neg\mathcal{F}_S^D$ is the information derivable from Π. However, this is not the complete semantics: the derived false conjunctions are not really indefinite but always based on at least one false atom. Thus, two more operators defined for definite disjunctive programs were introduced in [3] and that is why disjunctive logic programs have to be transformed into definite disjunctive programs.

Definition 4.7 Given a disjunctive program Π and a state-pair S, $DIS(\Pi)$ is obtained by transferring all negated atoms in the body of each clause of Π as atoms to its head. $Dis(\Pi, S)$ is obtained from $DIS(\Pi)$ by reducing the clauses in $DIS(\Pi)$ as follows:

1. Remove atoms from the body of a clause if they are true is S.
2. Remove a clause if its head is true in S.
3. Remove atoms from the head of a clause if they are false in S.

This transformation has some similarities with the transformation used for the stable model semantics [14]. The second and the third condition directly match the conditions of that transformation only that we now apply it to the positive heads which are partially former negated atoms from the body.

We present now the operators for deriving information from the obtained definite disjunctive programs and we start with T^D_Π, a simplification of the operator $\mathcal{T}^D_S$, which was at first introduced in [23].

Definition 4.8 Let Π be a definite disjunctive program. Let T be a subset of EB_Π.

$T^D_\Pi(T) = \{D \in EB_\Pi \mid H \leftarrow A_1, \dots, A_n$ in $\mathsf{ground}(\Pi)$ such that for all i, $1 \le i \le n$, $(A_i \vee D_i) \in T$, D_i might be empty, and $(\mathcal{H} \cup \bigcup_i \mathcal{D}_i) \subseteq \mathcal{D}.\}$

We also have a definition of the iteration similar to Definition 4.5.

Definition 4.9

$$T^D_\Pi \uparrow 0 = \emptyset. \;\; T^D_\Pi \uparrow (n+1) = T^D_\Pi(T^D_\Pi \uparrow n). \;\; T^D_\Pi = \bigcup_{n<\omega} T^D_\Pi \uparrow n.$$

For deriving indefinite false conjunctions an extension of the Generalized Closed World Assumption (GCWA) [22], the Extended Generalized Closed World Assumption (EGCWA) due to Yahya and Henschen [37] is used.

Definition 4.10 Let Π be a disjunctive logic program and $C_1 \wedge \cdots \wedge C_r$ a ground conjunction. $C_1 \wedge \cdots \wedge C_r$ can be inferred to be false from Π by the *extended generalized closed world assumption* EGCWA if and only if $\forall K_1, \dots, K_r$ if $\Pi \vdash C_1 \vee K_1, \dots, \Pi \vdash C_r \vee K_r$ then $\Pi \vdash K_1 \vee \ldots \vee K_r$. EGCWA($\Pi$) is the set of conjuncts that can be assumed false from Π using the EGCWA.

In [37], an alternative characterization was presented which was shown to be equivalent: a conjunction can be inferred to be false from Π if and only if it is false in all minimal models of Π where a minimal model [22] is a two-valued model M of Π such that no subset of it is a model as well. However, minimal models seem to be difficult to capture with a level mapping characterization as we will show in Section 6 and we stay with the definition above. Nevertheless, as shown in [37], the derivability notion $\vdash$ means truth in all minimal models and we can thus also use this alternative definition when computing the extended generalized closed world assumption of some program.

Example 4.11

$$\begin{aligned} p &\leftarrow q, t \\ q \vee t &\leftarrow \end{aligned}$$

This program Π has two minimal models, $\{q\}$ an $\{t\}$. Thus p is false in all minimal models and, likewise, $q \wedge t$. Of course, we also have closure with respect to supersets: any conjunction containing p is also false. We can therefore use the minimal set and EGCWA(Π) $= \{p, q \wedge t\}$ including all supersets of the mentioned conjunctions.

We continue by presenting the definitions of the operators T_S^{ED} and F_S^{ED} which are in fact just defined by means of T_Π^D, respectively the EGCWA, only restricted to knowledge not being already contained in S.

Definition 4.12 Let Π be a disjunctive program and S be a state-pair.

- $T_S^{ED} = \{D \mid D \in T^D_{Dis(\Pi,S)} \text{ and } D \notin S\}$
- $F_S^{ED} = \{C \mid C \in \text{EGCWA}(Dis(\Pi, S) \cup S) \text{ and } C \notin S\}$

Now we can combine the constructions for obtaining the complete derived knowledge given a state pair S.

Definition 4.13 $\mathcal{S}^{ED}(S) = S \cup \mathcal{T}_S^D \cup \neg\mathcal{F}_S^D \cup T_S^{ED} \cup \neg F_S^{ED}$.

Finally, we generalize this as well and define an operator calculating state-pairs by means of $\mathcal{S}^{ED}(S)$.

Definition 4.14

$$\begin{aligned} M_0 &= \emptyset \\ M_{\alpha+1} &= \mathcal{S}^{ED}(M_\alpha) \\ M_\alpha &= \bigcup_{\beta<\alpha} M_\beta, \text{ for limit ordinal } \alpha. \end{aligned}$$

This operator $\mathcal{S}^{ED}$ has a fixed point ([3]) which corresponds to the generalized disjunctive well-founded model M_Π^{ED}.

Theorem 4.15 ([3]) The generalized disjunctive well-founded model M_Π^{ED} is a fixpoint of operator $\mathcal{S}^{ED}$.

Moreover, It has been shown that M_Π^{ED} is consistent.

Corollary 4.16 ([3]) The generalized disjunctive well-founded model M_Π^{ED} is a consistent state-pair.

Example 4.17 Consider the program Π taken from [3].

$$\begin{aligned} p \leftarrow & \quad t, q \\ q \leftarrow & \quad \neg a \\ t \leftarrow & \quad \neg b \\ a \vee b \leftarrow & \\ e \leftarrow & \quad \neg f, p \\ f \leftarrow & \quad \neg e \end{aligned}$$

$M_0 = \emptyset$, and we calculate M_1. Since M_0 is empty we only derive $\mathcal{T}^D_{M_0} = \{a \vee b\}$ and $\mathcal{F}^D_{M_0} = \emptyset$. For the same reason $Dis(\Pi, M_0) = \{p \leftarrow t, q; q \vee a \leftarrow; t \vee b \leftarrow; a \vee b \leftarrow; e \vee f \leftarrow p; e \vee f \leftarrow\}$ which is identical to $DIS(\Pi)$. Then we can apply T^S_Π and the EGCWA to $Dis(\Pi, M_0)$ and obtain $T^{ED}_{M_0} = \{q \vee a, t \vee b, a \vee b, e \vee f\}$ and $F^{ED}_{M_0} = \{p, q \wedge a, t \wedge b, e \wedge f, a \wedge b, q \wedge t\}$. Then $M_1 = \mathcal{S}^{ED}(M_0) = \{q \vee a, t \vee b, a \vee b, e \vee f, \neg p, \neg(q \wedge a), \neg(t \wedge b), \neg(e \wedge f), \neg(a \wedge b), \neg(q \wedge t)\}$. $\mathcal{T}^D_{M_1} = \emptyset$ because we know already $(a \vee b) \in M_1$ and nothing else is deriveable. $\mathcal{F}^D_{M_1} = \{e\}$ is obtained by clause $e \leftarrow \neg f, p$ and because $\neg p \in M_1$. Furthermore $Dis(P, M_1) = \emptyset$ since all clauses with a true, respectively empty, head can be deleted, so $T^{ED}_{M_0} = \emptyset$ and $F^{ED}_{M_0} = \emptyset$. Consequently, $M_2 = M_1 \cup \{\neg e\}$. In the next iteration we obtain $f \in \mathcal{T}^D_{M_2}$ because $\neg e \in M_2$. All the other operators derive nothing new and this holds also for the entire next derivation so that $M_3 = M_2 \cup \{f\}$ is the fixpoint for this program and thus $M^{ED}_\Pi = \{q \vee a, t \vee b, a \vee b, f, \neg p, \neg(q \wedge a), \neg(t \wedge b), \neg(a \wedge b), \neg(q \wedge t), \neg e\}$ is the generalized well-founded model.

However, we do not have a statement corresponding to Theorem 3.17 for the generalized disjunctive well-founded semantics. For normal logic programs in general, the well-founded semantics and the generalized disjunctive well-founded semantics do not coincide. Consider the program Π just consisting of one clause $p \leftarrow \neg p$. In the well-founded model p is undefined whereas in the generalized disjunctive well-founded model p is true because $Dis(\Pi, M_0) = \{p \leftarrow\}$ and thus we can derive p to be true.

Since the generalized disjunctive well-founded semantics is based on operators we can directly continue with presenting the alternative level mapping characterization. The only thing remaining to be done beforehand is to define level mappings for state pairs.

Definition 4.18 For a disjunctive program Π and a state-pair I a disjunctive I-partial level mapping for Π is a partial mapping $l : (EB_\Pi \cup CB_\Pi) \to \alpha$ with domain $\mathsf{dom}(l) = \{D \mid D \in I \text{ or } \neg C \in I\}$, where α is some (countable) ordinal.

Note that now we do not extend the mapping to identify $l(D) = l(\neg D)$ since in a state-pair D is a disjunction and $\neg D$ a negated conjunction. We also recall from Definition 2.18 that we may use a lexicographic order on ordinals and that we can apply the components of that order.

Definition 4.19 Let Π be a disjunctive logic program, let the state-pair I be a model for Π, and let l_1, l_2 be disjunctive I-partial level mappings for Π. We say that Π satisfies (GDWF) with respect to I, l_1, and l_2 if each $D \in \mathsf{dom}(l_1)$ and each $\neg C \in \mathsf{dom}(l_1)$ satisfies one of the following conditions:

> (GDWFi) $D \in I$ and there is a clause $H \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$ with $\mathcal{H} \subseteq \mathcal{D}$ such that there is $\mathcal{D}_i \subseteq \mathcal{D}$ with $(D_i \vee A_i) \in I$ and $l_1(D) > l_1(D_i \vee A_i)$ or $l_1(D) >_1 l_2(D_i \vee A_i)$ for all $i = 1, \ldots, n$, and $\neg B_j \in I$ and $l_1(D) >_1 l_t(\neg B_j)$, $t \in \{1, 2\}$, for all $j = 1, \ldots, m$.
>
> (GDWFii) $\neg C \in I$ with atom $A \in C$ and for each clause $H \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$ with $A \in H$ (at least) one of the following conditions holds:
>
> > (GDWFiia) $\neg(A_1 \wedge \ldots \wedge A_n) \in I$ and $l_1(\neg C) \geq l_1(\neg(A_1 \wedge \ldots \wedge A_n))$.
> >
> > (GDWFiia') $\neg(A_1 \wedge \ldots \wedge A_n) \in I$ and $l_1(\neg C) >_1 l_2(\neg(A_1 \wedge \ldots \wedge A_n))$.
> >
> > (GDWFiib) $(B_1 \vee \ldots \vee B_m) \in I$ and $l_1(\neg C) >_1 l_t(B_1 \vee \ldots \vee B_m)$ for $t \in \{1, 2\}$.

and each $D \in \mathsf{dom}(l_2)$ and each $\neg C \in \mathsf{dom}(l_2)$ satisfies one of the following conditions:

> (GDWFi') $D \in I$ and there is a clause $H_1 \vee \cdots \vee H_l \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$ such that there is $\mathcal{D}_i \subseteq \mathcal{D}$ with $(D_i \vee A_i) \in I$ and $l_2(D) >_2 l_2(D_i \vee A_i)$ or $A_i \in I$ and $l_2(D) >_1 l_s(A_i)$, $s \in \{1, 2\}$, for all $i = 1, \ldots, n$, $\emptyset \neq ((\mathcal{H} \cup \mathcal{B}) \setminus \mathcal{D}') \subseteq \mathcal{D}$, $H_k \in \mathcal{D}'$ for each $\neg H_k \in I$ with $l_2(D) >_1 l_t(\neg H_k)$, $t \in \{1, 2\}$, and $B_j \in \mathcal{D}'$ for each $\neg B_j \in I$ with $l_2(D) >_1 l_t(\neg B_j)$, $t \in \{1, 2\}$, for all $k = 1, \ldots, l$ and all $j = 1, \ldots, m$.
>
> (GDWFii') $\neg C \in I$ and $C \in \mathrm{ECGWA}(Dis(\Pi, S) \cup S)$, $C \notin S$ and $l_2(\neg C) >_1 l_t(L)$, $t \in \{1, 2\}$, if and only if $L \in S$.

One might ask why we need two level mappings in this characterization but we postpone the answer and show like in the previous section that this characterization is equivalent to the generalized disjunctive well-founded model.

Theorem 4.20 Let Π be a disjunctive program with generalized disjunctive well-founded model M. Then, in the disjunctive knowledge ordering, M is the greatest model amongst all models I for which there exist disjunctive I-partial level mappings l_1 and l_2 for Π such that Π satisfies (GDWF) with respect to I, l_1, and l_2.

Proof: Let M be the generalized disjunctive well-founded model of Π. We define the disjunctive M-partial level mappings l_1 and l_2 in the following way: If $D \in \mathcal{T}^D_{M_\alpha}$ then β is the least ordinal such that $D \in \mathcal{T}^D_{M_\alpha} \uparrow (\beta + 1)$ and $l_1(D) = (\alpha, \beta)$. If $D \in T^{ED}_{M_\alpha}$ then β is the least ordinal such that $D \in T^D_{Dis(\Pi, M_\alpha)} \uparrow (\beta + 1)$ and $l_2(D) = (\alpha, \beta)$. If $C \in \mathcal{F}^D_{M_\alpha}$ then $l_1(\neg C) = (\alpha, 0)$. If $C \in F^{ED}_{M_\alpha}$ then $l_2(\neg C) = (\alpha, 0)$. All other values remain undefined. By Definition 4.13 and 4.14, $M = \bigcup_{\beta<\alpha} M_\beta$ for limit ordinal α, $M_{\alpha+1} = \mathcal{S}^{ED}(M_\alpha)$, and $\mathcal{S}^{ED}(S) = S \cup \mathcal{T}^D_S \cup \neg\mathcal{F}^D_S \cup T^{ED}_S \cup \neg F^{ED}_S$, and we know that any $D \in M$, respectively $\neg C \in M$, is at least contained in the domain of either l_1 or l_2. We show that Π satisfies (GDWF) with respect to M, l_1, and l_2.

Let $D \in \mathsf{dom}(l_1)$ and $l_1(D) = (\alpha, \beta)$. By Definition of l_1 and since D is not a negated conjunction, we have that $D \in \mathcal{T}^D_{M_\alpha}$. Since $\mathcal{T}^D_{M_\alpha} = \bigcup_{n<\omega} \mathcal{T}^D_{M_\alpha} \uparrow n$ we have $D \in \bigcup_{n<\omega} \mathcal{T}^D_{M_\alpha} \uparrow n$. We know that β is the least ordinal such that $D \in \mathcal{T}^D_{M_\alpha} \uparrow (\beta + 1)$ and that $\mathcal{T}^D_{M_\alpha} \uparrow (\beta + 1) = \mathcal{T}^D_{M_\alpha}(\mathcal{T}^D_{M_\alpha} \uparrow \beta)$ and thus $D \in \mathcal{T}^D_{M_\alpha}(\mathcal{T}^D_{M_\alpha} \uparrow \beta)$. Then, by Definition 4.4, D is undefined in M_α and $H \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$ such that for all i, $1 \le i \le n$, $(A_i \vee D_i) \in M_\alpha$ or $(A_i \vee D_i) \in (\mathcal{T}^D_{M_\alpha} \uparrow \beta)$, D_i might be empty, $\neg B_j \in M_\alpha$ for all j, $1 \le j \le m$, and $(\mathcal{H} \cup \bigcup_i \mathcal{D}_i) \subseteq \mathcal{D}$. If $(A_i \vee D_i) \in (\mathcal{T}^D_{M_\alpha} \uparrow \beta)$ then $(A_i \vee D_i) \in \mathcal{T}^D_{M_\alpha}$, thus $(A_i \vee D_i) \in M_{\alpha+1}$ and $l_1(A_i \vee D_i) = (\alpha, \beta')$ with $\beta' < \beta$. Then $(A_i \vee D_i) \in M$ and $l_1(D) > l_1(A_i \vee D_i)$. If $(A_i \vee D_i) \in M_\alpha$ then $(A_i \vee D_i) \in M$ and $l_1(D) >_1 l_t(A_i \vee D_i)$, $t \in \{1, 2\}$ and, likewise, for all $\neg B_j \in M_\alpha$, we have $\neg B_j \in M$ and $l_1(D) >_1 l_t(\neg B_j)$, $t \in \{1, 2\}$. Then D satisfies (GDWFi).

Let $\neg C \in \mathsf{dom}(l_1)$. By Definition of l_1 and since $\neg C$ is a negated conjunction, we have that $C \in \mathcal{F}^D_{M_\alpha}$ and $l_1(\neg C) = (\alpha, 0)$. We know that $\mathcal{F}^D_{M_\alpha} = \bigcap_{n<\omega} \mathcal{F}^D_{M_\alpha} \downarrow n$ and thus, for all n, $C \in \mathcal{F}^D_{M_\alpha} \downarrow n$. Consider any $n = n' + 1$. Then, by Definition 4.5, $\mathcal{F}^D_{M_\alpha} \downarrow (n' + 1) = \mathcal{F}^D_{M_\alpha}(\mathcal{F}^D_{M_\alpha} \downarrow n')$. By Definition 4.4, C is undefined in M_α, $A \in C$, and for all clauses $H \leftarrow A_1, \ldots, A_n$, $\neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$, with $A \in H$, at least one of the following three cases holds:

- We have $(B_1 \vee \cdots \vee B_m) \in M_\alpha$. Then $(B_1 \vee \cdots \vee B_m) \in M$, by Definition 4.14, and $l_1(\neg C) >_1 l_t(B_1 \vee \cdots \vee B_m)$, $t \in \{1, 2\}$, and C satisfies (GDWFiib).

- Or $\neg(A_1 \wedge \cdots \wedge A_n) \in M_\alpha$ holds. Then $\neg(A_1 \wedge \cdots \wedge A_n) \in M$ and $l_1(\neg C) >_1 l_1(\neg(A_1 \wedge \cdots \wedge A_n))$ and C satisfies (GDWFiia) or $l_1(\neg C) >_1 l_2(\neg(A_1 \wedge \cdots \wedge A_n))$and C satisfies (GDWFiia').

- Otherwise $\neg(A_1 \wedge \cdots \wedge A_n) \in (\mathcal{F}^D_{M_\alpha} \downarrow n')$. Since $C \in (\mathcal{F}^D_{M_\alpha} \downarrow n')$ for all n', we know that $\neg(A_1 \wedge \cdots \wedge A_n) \in (\mathcal{F}^D_{M_\alpha} \downarrow n')$ for all n', otherwise there had to be another reason for C occurring in all iterations of $\mathcal{F}^D_{M_\alpha}$. Thus $\neg(A_1 \wedge \cdots \wedge A_n) \in \mathcal{F}^D_{M_\alpha}$ and $\neg(A_1 \wedge \cdots \wedge A_n) \in M_{\alpha+1}$ and $l_1(\neg(A_1 \wedge \cdots \wedge A_n)) = (\alpha, 0)$ Therefore $l_1(\neg C) = l_1(\neg(A_1 \wedge \cdots \wedge A_n))$ and $\neg(A_1 \wedge \cdots \wedge A_n) \in M$. Hence, C satisfies (GDWFiia), as well.

Alternatively, let $D \in \mathsf{dom}(l_2)$ and $l_2(D) = (\alpha, \beta)$. By Definition of l_2 and since D is not a negated conjunction, we have that $D \in T^{ED}_{M_\alpha}$. We know that if $D \in T^{ED}_{M_\alpha}$ then $D \in T^D_{Dis(\Pi,M_\alpha)}$ and $D \notin M_\alpha$ by Definition 4.12. Since $T^D_{Dis(\Pi,M_\alpha)} = \bigcup_{n<\omega} \mathcal{T}^D_{Dis(\Pi,M_\alpha)} \uparrow n$ we have $D \in \bigcup_{n<\omega} \mathcal{T}^D_{Dis(\Pi,M_\alpha)} \uparrow n$. By definition of l_2, β is the least ordinal such that $D \in T^D_{Dis(\Pi,M_\alpha)} \uparrow (\beta + 1)$. We know that $T^D_{Dis(\Pi,M_\alpha)} \uparrow (\beta + 1) = T^D_{Dis(\Pi,M_\alpha)}(T^D_{Dis(\Pi,M_\alpha)} \uparrow \beta)$ and thus $D \in T^D_{Dis(\Pi,M_\alpha)}(T^D_{Dis(\Pi,M_\alpha)} \uparrow \beta)$. Then $H \leftarrow A_1, \ldots, A_n$ in $\mathsf{ground}(Dis(\Pi, M_\alpha))$ such that for all i, $1 \leq i \leq n$, $(A_i \vee D_i) \in T^D_{Dis(\Pi,M_\alpha)} \uparrow \beta$, D_i might be empty, and $(\mathcal{H} \cup \bigcup_i D_i) \subseteq \mathcal{D}$ by Definition 4.8. If $(A_i \vee D_i) \in (T^D_{Dis(\Pi,M_\alpha)} \uparrow \beta)$ then $(A_i \vee D_i) \in T^D_{Dis(\Pi,M_\alpha)}$ and thus $(A_i \vee D_i) \in T^{ED}_{M_\alpha}$. Then $(A_i \vee D_i) \in M_{\alpha+1}$ and $l_2(A_i \vee D_i) = (\alpha, \beta')$ with $\beta' < \beta$, so $(A_i \vee D_i) \in M$ and $l_2(D) >_2 l_2(A_i \vee D_i)$. Furthermore, we also have a clause $H \vee D' \leftarrow A_1, \ldots, A_n, A_{n+1}, \ldots, A_r$ in $\mathsf{ground}(Dis(\Pi))$. By Definition 4.7, all A_q, $q = n+1, \ldots, r$, occur in M_α and thus $A_q \in M$ and $l_2(D) >_1 l_t(A_q)$, $t \in \{1, 2\}$. By the same definition, for all elements $E \in D'$ we have $\neg E \in M_\alpha$. Then $\neg E \in M$ and $l_2(D) >_1 l_t(\neg E)$, $t \in \{1, 2\}$. We also have a clause $H' \leftarrow A_1, \ldots, A_n, A_{n+1}, \ldots, A_r, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$ with $\mathcal{H}' \cup \{B_1, \ldots, B_m\} = \mathcal{H} \cup \mathcal{D}'$. Then $((\mathcal{H}' \cup \mathcal{B}) \setminus \mathcal{D}') \subseteq \mathcal{D}$ with $l_2(D) >_1 l_t(\neg E)$, $t \in \{1, 2\}$, for each $E \in D'$. Hence, (GDWFi') is satisfied. Finally, let $C \in \mathsf{dom}(l_2)$. By Definition of l_2 and since $\neg C$ is a negated conjunction we have that $C \in F^D_{M_\alpha}$ and $l_2(\neg C) = (\alpha, 0)$. By Definition 4.12, $C \in \mathrm{EGCWA}(Dis(\Pi, M_\alpha) \cup M_\alpha)$ and $C \notin M_\alpha$. By definition of l_1 and l_2, for all $L \in M_\alpha$ we have $l_2(C) >_1 l_t(L)$, $t \in \{1, 2\}$. Then, for all L with $l_2(C) >_1 l_t(L)$, $t \in \{1, 2\}$, we have $L \in M_\alpha$. Hence, C satisfies (GDWFii').

Alternatively, we show that if I is a model of Π and l_1 and l_2 are disjunctive I-partial level mappings such that Π satisfies (GDWF) with respect to I, l_1, and l_2 then $I \subseteq M^{ED}_\Pi$. We show via transfinite induction on α that whenever $D \in I$ with $l_1(D) = (\alpha, \beta)$ or $l_2(D) = (\alpha, \beta)$ then $D \in M_{\alpha+1}$ and whenever $\neg C \in I$ with $l_1(C) = (\alpha, \beta)$ or $l_2(C) = (\alpha, \beta)$ then $\neg C \in M_{\alpha+1}$. By Definition 4.14, this suffices to show that $D \in M^{ED}_\Pi$ and $\neg C \in M^{ED}_\Pi$.

Let $\alpha = 0$.

If $D \in I$ and $D \in \mathsf{dom}(l_1)$ then D satisfies (GDWFi) and there is a clause $H \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$ with $\mathcal{H} \subseteq \mathcal{D}$ such that there is $\mathcal{D}_i \subseteq \mathcal{D}$ with $(D_i \vee A_i) \in I$ and $l_1(D) > l_1(D_i \vee A_i)$ or $l_1(D) >_1 l_2(D_i \vee A_i)$ for all $i = 1, \ldots, n$, and $\neg B_j \in I$ and $l_1(D) >_1 l_t(\neg B_j)$, $t \in \{1, 2\}$, for all $j = 1, \ldots, m$. Since there is no ordinal smaller than 0, we know that all conditions including $>_1$ cannot be satisfied. Thus (GDWFi) simplifies to that there is a clause $H \leftarrow A_1, \ldots, A_n$ in $\mathsf{ground}(\Pi)$ with $\mathcal{H} \subseteq \mathcal{D}$ such that there is $\mathcal{D}_i \subseteq \mathcal{D}$ with $(D_i \vee A_i) \in I$ and $l_1(D) > l_1(D_i \vee A_i)$ for all $i = 1, \ldots, n$. We show by induction on β that $D \in \mathcal{T}^D_{M_0} \uparrow (\beta + 1)$, thus $D \in \mathcal{T}^D_{M_0}$ and therefore $D \in M_1$. Let $\beta = 0$. Since again there is no ordinal smaller than 0, the considered clause is a fact. Then $D \in (\mathcal{T}^D_{M_0} \uparrow 1)$, by Definition 4.4. Suppose the property holds for all D with $\beta' < \beta$ and let $l(D) = (0, \beta)$. We know that D satisfies the simplified (GDWFi) and, by assumption, for all $(D_i \vee A_i) \in I$ with $l_1(D) > l_1(D_i \vee A_i)$ we have $(D_i \vee A_i) \in (\mathcal{T}^D_{M_0} \uparrow \beta)$. Then, by Definition 4.4 and 4.5, $D \in \mathcal{T}^D_{M_0} \uparrow (\beta + 1)$.
If $\neg C \in I$ and $\neg C \in \mathsf{dom}(l_1)$ then C satisfies (GDWFii) and there is an atom $A \in D$ such that for each clause $H \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$ with $A \in H$ (at least) one of the conditions (GDWFiia), (GDWFiia'), or (GDWFiib) holds. Since there is no ordinal smaller than 0, (GDWFiia') and (GDWFiib) cannot hold by definition of $>_1$. Thus (GDWFiia) holds for all clauses and there is $\neg(A_1 \wedge \ldots \wedge A_n) \in I$, $l_1(\neg D) \geq l_1(\neg(A_1 \wedge \ldots \wedge A_n))$, and for the same reason $l_1(\neg(A_1 \wedge \ldots \wedge A_n)) = (0, \beta')$ with $\beta' \leq \beta$. Thus $\neg(A_1 \wedge \ldots \wedge A_n)$ satisfies (GDWFiia) as well. We can apply the argument again and obtain eventually that each $\neg C$ which satisfies (GDWFiia) does this by means of a negated conjunction which satisfies also (GDWFiia). But then for all clauses $H \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$, with $A \in H$, $\neg(A_1 \wedge \cdots \wedge A_n) \in \mathcal{F}^D_{M_0} \downarrow n'$ for all n' and thus $C \in \bigcap_{n' < \omega} \mathcal{F}^D_{M_0} \downarrow n'$ and $C \in \mathcal{F}^D_{M_0}$. Hence, $\neg C \in M_1$.
If $D \in I$ and $D \in \mathsf{dom}(l_2)$ then D satisfies (GDWFi') and there is a clause $H_1 \vee \cdots \vee H_l \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$ such that there is $\mathcal{D}_i \subseteq \mathcal{D}$ with $(D_i \vee A_i) \in I$ and $l_2(D) >_2 l_2(D_i \vee A_i)$ or $A_i \in I$ and $l_2(D) >_1 l_s(A_i)$, $s \in \{1, 2\}$, for all $i = 1, \ldots, n$, $\emptyset \neq ((\mathcal{H} \cup \mathcal{B}) \setminus \mathcal{D}') \subseteq \mathcal{D}$, $H_k \in \mathcal{D}'$ for each $\neg H_k \in I$ with $l_2(D) >_1 l_t(\neg H_k)$, $t \in \{1, 2\}$, and $B_j \in \mathcal{D}'$ for each $\neg B_j \in I$ with $l_2(D) >_1 l_t(\neg B_j)$, $t \in \{1, 2\}$, for all $k = 1, \ldots, l$ and all $j = 1, \ldots, m$. Since there is no ordinal smaller than 0 we know that all conditions including $>_1$ cannot be satisfied. Thus (GDWFi') simplifies to that there is a clause $H_1 \vee \cdots \vee H_l \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$ such that there is $\mathcal{D}_i \subseteq \mathcal{D}$ with $(D_i \vee A_i) \in I$ and $l_2(D) >_2 l_2(D_i \vee A_i)$ for all $i = 1, \ldots, n$, $(\mathcal{H} \cup \mathcal{B}) \subseteq \mathcal{D}$. Note in particular that the set $\mathcal{D}'$ has to be empty and that $((\mathcal{H} \cup \mathcal{B}) \setminus \mathcal{D}') \neq \emptyset$ holds automatically. Then there is also a clause $H' \leftarrow A_1, \ldots, A_n$ with $\mathcal{H}' = \mathcal{H} \cup \mathcal{B}$ in $DIS(\Pi)$ and, since M_0 is empty, also

in $Dis(\Pi, M_0)$. We show by induction on β that $D \in T^D_{Dis(\Pi,M_0)} \uparrow (\beta+1)$, thus $D \in T^{ED}_{M_0}$ and therefore $D \in M_1$. Let $\beta = 0$. Since there is no ordinal smaller than 0, the considered clause is a fact. Then $D \in (T^D_{Dis(\Pi,M_0)} \uparrow 1)$, by Definition 4.8. Suppose the property holds for all D with $\beta' < \beta$ and let $l(D) = (0, \beta)$. We know that D satisfies the simplified (GDWFi') and by assumption for all $(D_i \vee A_i) \in I$ with $l_2(D) >_2 l_2(D_i \vee A_i)$ we have $(D_i \vee A_i) \in (T^D_{Dis(\Pi,M_0)} \uparrow \beta)$. Then, by Definitions 4.8 and 4.9, $D \in T^D_{Dis(\Pi,M_0)} \uparrow (\beta+1)$.
If $\neg C \in I$ and $\neg C \in \mathsf{dom}(l_2)$ then C satisfies (GDWFii') so that we have $C \in \text{ECGWA}(Dis(\Pi, S) \cup S)$, $C \notin S$ and $l_2(\neg C) >_1 l_t(L)$, $t \in \{1,2\}$, if and only if $L \in S$. Since there is no ordinal smaller than 0, we know that S is empty. Then $C \in \text{ECGWA}(Dis(\Pi, M_0) \cup M_0)$, $C \notin M_0$ and $C \in F^{ED}_{M_0}$ by Definition 4.12. Thus $\neg C \in M_1$.

So suppose for all $\alpha' < \alpha$ that if $D \in I$ with $l_1(D) = (\alpha', \beta)$ or $l_2(D) = (\alpha', \beta)$ then $D \in M_{\alpha'+1}$ and if $\neg C \in I$ with $l_1(\neg C) = (\alpha', \beta)$ or $l_2(\neg C) = (\alpha', \beta)$ then $\neg C \in M_{\alpha'+1}$. We show that the property also holds for all $D \in I$ with $l_1(D) = (\alpha, \beta)$ or $l_2(D) = (\alpha, \beta)$ and all $\neg C \in I$ with $l_1(\neg C) = (\alpha, \beta)$ or $l_2(\neg C) = (\alpha, \beta)$.
If $D \in I$ and $D \in \mathsf{dom}(l_1)$ then D satisfies (GDWFi) and there is a clause $H \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$ with $\mathcal{H} \subseteq \mathcal{D}$ such that there is $\mathcal{D}_i \subseteq \mathcal{D}$ with $(D_i \vee A_i) \in I$ and $l_1(D) > l_1(D_i \vee A_i)$ or $l_1(D) >_1 l_2(D_i \vee A_i)$ for all $i = 1, \ldots, n$, and $\neg B_j \in I$ and $l_1(D) >_1 l_t(\neg B_j)$, $t \in \{1,2\}$, for all $j = 1, \ldots, m$. We show by induction on β that $D \in \mathcal{T}^D_{M_\alpha} \uparrow (\beta+1)$ and thus $D \in \mathcal{T}^D_{M_\alpha}$. Then we have $D \in M_{\alpha+1}$ which finishes this case. Let $\beta = 0$. Since there is no ordinal smaller than 0, we know that $l_1(D) > l_1(D_i \vee A_i)$ can only hold if $l_1(D) >_1 l_1(D_i \vee A_i)$. Then all $(A_i \vee D_i) \in M_\alpha$ and all $\neg B_j \in M_\alpha$ and thus $D \in (\mathcal{T}^D_{M_\alpha} \uparrow 1)$, by Definition 4.4. Suppose the property holds for all D with $l(D) = (\alpha, \beta')$, $\beta' < \beta$, and let $l(D) = (\alpha, \beta)$. We know that D satisfies (GDWFi) and, by assumption, for all $(D_i \vee A_i) \in I$ with $l_1(D) > l_1(D_i \vee A_i)$ we have $(D_i \vee A_i) \in (\mathcal{T}^D_{M_\alpha} \uparrow \beta)$ or $(A_i \vee D_i) \in M_\alpha$. Together with $(A_i \vee D_i) \in M_\alpha$ if $l_1(D) >_1 l_2(D_i \vee A_i)$ and $\neg B_j \in M_\alpha$, for all $i = 1, \ldots, n$ and all $j = 1, \ldots, m$, we conclude, by Definition 4.4 and 4.5, that $D \in \mathcal{T}^D_{M_\alpha} \uparrow (\beta+1)$.
If $\neg C \in I$ and $\neg C \in \mathsf{dom}(l_1)$ then C satisfies (GDWFii) and there is an atom $A \in D$ such that for each clause $H \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$ with $A \in H$ (at least) one of the conditions (GDWFiia), (GDWFiia'), or (GDWFiib) holds. Consider such a clause. If (GDWFiib) holds then $(B_1 \vee \ldots \vee B_m) \in I$ and $l_1(\neg C) >_1 l_t(B_1 \vee \ldots \vee B_m)$, $t \in \{1,2\}$. Then $(B_1 \vee \ldots \vee B_m) \in M_\alpha$. If (GDWFiia') holds then $\neg(A_1 \wedge \ldots \wedge A_n) \in I$ and $l_1(\neg D) >_1 l_2(\neg(A_1 \wedge \ldots \wedge A_n))$. Then also $\neg(A_1 \wedge \ldots \wedge A_n) \in M_\alpha$. If (GDWFiia) holds then there is $\neg(A_1 \wedge \ldots \wedge A_n) \in I$ and $l_1(\neg D) \geq l_1(\neg(A_1 \wedge \ldots \wedge A_n))$. If even

$l_1(\neg D) >_1 l_1(\neg(A_1 \wedge \ldots \wedge A_n))$ then $\neg(A_1 \wedge \ldots \wedge A_n) \in M_\alpha$ as well. Otherwise $l_1(\neg(A_1 \wedge \ldots \wedge A_n)) = (\alpha, \beta')$ with $\beta' \leq \beta$. Then $\neg(A_1 \wedge \ldots \wedge A_n)$ satisfies (GDWFii) as well. We can apply the argument again and obtain for each clause that we either have eventually a dependence on an element contained in M_α or an infinite chain of negated conjunctions satisfying (GDWFiia). But then, by Definition 4.4, $C \in \mathcal{F}^D_{M_\alpha} = \bigcap_{n<\omega} \mathcal{F}^D_{M_\alpha} \downarrow n$ and thus $\neg C \in M_{\alpha+1}$.
If $D \in I$ and $D \in \mathsf{dom}(l_2)$ then D satisfies (GDWFi') and there is a clause $H_1 \vee \cdots \vee H_l \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$ such that there is $\mathcal{D}_i \subseteq \mathcal{D}$ with $(D_i \vee A_i) \in I$ and $l_2(D) >_2 l_2(D_i \vee A_i)$ or $A_i \in I$ and $l_2(D) >_1 l_s(A_i)$, $s \in \{1,2\}$, for all $i = 1, \ldots, n$, $\emptyset \neq ((\mathcal{H} \cup \mathcal{B}) \backslash \mathcal{D}') \subseteq \mathcal{D}$, $H_k \in \mathcal{D}'$ for each $\neg H_k \in I$ with $l_2(D) >_1 l_t(\neg H_k)$, $t \in \{1,2\}$, and $B_j \in \mathcal{D}'$ for each $\neg B_j \in I$ with $l_2(D) >_1 l_t(\neg B_j)$, $t \in \{1,2\}$, for all $k = 1, \ldots, l$ and each $j = 1, \ldots, m$. Then there is also a clause $H' \leftarrow A_1, \ldots, A_n$ with $\mathcal{H}' = \mathcal{H} \cup \mathcal{B}$ in $DIS(\Pi)$ by Definition 4.7. By the same definition, we obtain a clause $H'' \leftarrow A_1, \ldots, A_{n'}$ in $\mathsf{ground}(Dis(\Pi, M_\alpha))$. All $A_i \in M_\alpha$ are removed from the body and we only have $l(D) >_2 l_2(D_{i'} \vee A_{i'}) = (\alpha, \beta')$ with $\beta' < \beta$ for $i' = 1, \ldots, n'$. We cannot have removed the clause by 2. of Definition 4.7 since otherwise D would already occur in M_α in contradiction to our initial assumption. Finally $\mathcal{H}'' = \mathcal{H}' \setminus \mathcal{D}'$ by 3. of the same definition and H'' is not empty by (GDWFii'). We show by induction on β that $D \in T^D_{Dis(\Pi,M_0)} \uparrow (\beta + 1)$, thus $D \in T^{ED}_{M_0}$ and therefore $D \in M_1$. Let $\beta = 0$. Since there is no ordinal smaller than 0 the considered clause is a fact. Then $D \in (T^D_{Dis(\Pi,M_0)} \uparrow 1)$, by Definition 4.8. Suppose the property holds for all D with $l(D) = (\alpha, \beta')$, $\beta' < \beta$, and let $l(D) = (\alpha, \beta)$. We know by assumption for all $(D_i \vee A_i) \in I$ with $l_2(D) >_2 l_2(D_i \vee A_i)$ that $(D_i \vee A_i) \in (T^D_{Dis(\Pi,M_\alpha)} \uparrow \beta)$. Then, by Definition 4.8 and 4.9, $D \in T^D_{Dis(\Pi,M_\alpha)} \uparrow (\beta + 1)$.
If $\neg C \in I$ and $\neg C \in \mathsf{dom}(l_2)$ then C satisfies (GDWFii') so that $\neg C \in I$ and $C \in \mathrm{ECGWA}(Dis(\Pi, S) \cup S)$, $C \notin S$ and $l_2(\neg C) >_1 l_t(L)$, $t \in \{1,2\}$, if and only if $L \in S$. Thus by definition of $>_1$, $l_t(L) = (\alpha', \beta')$ with $\alpha' < \alpha$ and some β'. If $L \in S$ with $l_t(L) = (\alpha', \beta')$, by induction hypothesis, we know that $L \in M_\alpha$. If $L \in M_\alpha$ then $l(L) = (\alpha', \beta')$ with $\alpha' < \alpha$ and $L \in S$. So we have that $S = M_\alpha$. Then $C \in \mathrm{ECGWA}(Dis(\Pi, M_\alpha) \cup M_\alpha)$ and $C \notin M_\alpha$. Thus $C \in F^{ED}_{M_\alpha}$ by Definition 4.12 and so $\neg C \in M_{\alpha+1}$. ■

Since the generalized disjunctive well-founded semantics and the well-founded semantics do not coincide for normal logic programs the corresponding level mapping characterization should not be equivalent either although the definition of l_1 appears to be at first sight. However, combined with l_2 this equivalence does not hold. Recall the program Π consisting just of one clause $p \leftarrow \neg p$. The well-founded model is and the mapping has therefore an empty domain. We already know that $M^{ED}_\Pi = \{p\}$ and obtain $l_2(p) = (0,0)$.

We finish the section by presenting two more examples which in particular will explain why the characterization given in Definition 4.19 is so complicated.

Example 4.21 Consider the program Π which is an extended variant of Example 4.17.

$$
\begin{array}{rcl}
a & \leftarrow & \\
b & \leftarrow & \neg a, c \\
c & \leftarrow & \neg b \\
d & \leftarrow & a, \neg d \\
e & \leftarrow & \neg c, f \\
f & \leftarrow & e \\
g & \leftarrow & c, \neg e, \neg h \\
h & \leftarrow & g
\end{array}
$$

We calculate at first M_Π^{ED} starting with $M_0 = \emptyset$. Then $\mathcal{T}_{M_0}^D = \{a\}$ and $\mathcal{F}_{M_0}^D = \{e, f\}$. We then transform Π into $Dis(\Pi, M_0)$ which is identical to $DIS(\Pi)$ since M_0 is empty.

$$
\begin{array}{rcl}
a & \leftarrow & \\
a \vee b & \leftarrow & c \\
b \vee c & \leftarrow & \\
d & \leftarrow & a \\
c \vee e & \leftarrow & f \\
f & \leftarrow & e \\
e \vee g \vee h & \leftarrow & c \\
h & \leftarrow & g
\end{array}
$$

We start with $T_{Dis(\Pi,M_0)}^D \uparrow 0 = \emptyset$. Then $T_{Dis(\Pi,M_0)}^D \uparrow 1 = \{a, b \vee c\}$ and $T_{Dis(\Pi,M_0)}^D \uparrow 2 = \{a, b \vee c, d, b \vee e \vee g \vee h\}$. We also obtain $T_{Dis(\Pi,M_0)}^D \uparrow 3 = \{a, b \vee c, d, b \vee e \vee h, b \vee f \vee g \vee h\}$ and the fixed point $T_{Dis(\Pi,M_0)}^D \uparrow 4 = \{a, b \vee c, d, b \vee e \vee h, b \vee f \vee h\}$ which is also $T_{M_0}^{ED}$. Note how $(b \vee f \vee h)$ is derived by this operator. The minimal models of $Dis(\Pi, M_0)$ are $\{a, d, b\}$, $\{a, d, c, h\}$, and $\{a, d, c, e, f\}$. Then $F_{M_0}^{ED} = \{g, b \wedge c, b \wedge e, b \wedge f, b \wedge h, e \wedge h, f \wedge h\}$. We combine all the results and obtain $M_1 = \{a, b \vee c, d, b \vee e \vee h, b \vee f \vee h, \neg e, \neg f, \neg g, \neg(b \wedge c), \neg(b \wedge h)\}$. We continue with M_2 and compute therefore $\mathcal{T}_{M_1}^D = \emptyset$ and $\mathcal{F}_{M_1}^D = \{b\}$. We obtain $Dis(\Pi, M_1)$ from $DIS(\Pi)$ by removing the first four clauses because of 2. of Definition 4.7 and all e, f, and g from the heads by 3. of the same definition. Then $T_{M_1}^{ED} = \emptyset$. The minimal models of $Dis(\Pi, M_1)$ and M_1 are $\{a, d, b\}$ and $\{a, d, c, h\}$ and we obtain no new false conjunction thus $F_{M_1}^{ED} = \emptyset$. Then $M_2 = \{a, b \vee c, d, b \vee e \vee h, b \vee f \vee h, \neg e, \neg f, \neg g, \neg b\}$. We then derive $\mathcal{T}_{M_2}^{D} = \{c\}$ where all the other operators yield nothing new so $M_3 = \{a, c, d, b \vee e \vee h, b \vee f \vee h, \neg e, \neg f, \neg g, \neg b\}$. Then $Dis(\Pi, M_3)$ contains a fact $h \leftarrow$ and $T_{M_3}^{ED} = \{h\}$ whereas all the other operators derive nothing new. Thus $M_4 = \{a, c, d, h, \neg e, \neg f, \neg g, \neg b\}$ which is also the fixed point. For these elements in the minimal representation we obtain the following level mappings according to the proof of Theorem 4.20: $l_1(a) = (0,0)$, $l_2(a) = (0,0)$, $l_1(c) = (2,0)$, $l_2(d) = (0,1)$, $l_2(h) = (3,0)$, $l_1(\neg e) = (0,0)$, $l_1(\neg f) = (0,0)$, $l_2(\neg g) = (0,0)$, and $l_1(\neg b) = (1,0)$. We can see that a is in the domain of both mappings but this does not hold in general as we see for all the other literals. It can also be noticed that supersets do not necessarily have the same level. In case of a we also have $l_t(a \vee b) = (0,0)$, $t \in \{1,2\}$, but for $b \vee e \vee h$ we have $l_2(b \vee e \vee h) = (0,2)$ even though $l_2(h) = (3,0)$. Even the mapping may differ: $l_1(c) = (2,0)$ but $l_2(b \vee c) = (0,0)$.

Considering this example we now give some arguments for the way the alternative characterization is done. Clearly, the lexicographic order seems reasonable to represent the iterations of M_α and the (possible) iteration of the specific operator, e.g. $\mathcal{T}_{M_\alpha}^{D}$. However, one might ask, why we split the lexicographic order and do also allow $>_1$ and $>_2$. If we would in case of (GDWFi) for the negative literals use $>$ instead of $>_1$ then e.g. in the example $l(c) = (1,1)$ would also suffice to have $l(c) > l(\neg b)$. But this does no longer correspond to the construction by means of the operators since we only want to derive knowledge by means of $\mathcal{T}_{M_\alpha}^{D}$ if all negative literals are contained in M_α and results from $\mathcal{F}_{M_\alpha}^{D}$ should not interfere. For the same reason we also need two mappings to separate the operators for Π and for $Dis(\Pi, M_\alpha)$. Otherwise $l_1(h) = (2,1) > l_1(c)$ which does again not correspond to the construction of $T_{Dis(\Pi, M_\alpha)}^{D}$. The order $>_2$ in (GDWFi') is used in the same spirit to distinguish between atoms already known in M_α and the prior consequences of the operator.

There is one other thing to be explained in a more detailed manner: the treatment of the EGCWA in the characterization. We simply kept this because the characterization of the minimal model semantics in the level mapping approach is difficult and unsolved up to now as we will see in Section 6. Anyway, we present a small example to demonstrate some problems regarding a level mapping characterization of EGCWA.

Example 4.22 Consider the program Π:

$$\begin{aligned} q \vee r &\leftarrow \\ q \vee p &\leftarrow \\ r \vee s &\leftarrow \end{aligned}$$

We compute M_Π^{ED}. We have $\mathcal{T}_{M_0}^D = \{q \vee r, q \vee p, r \vee s\}$ and $\mathcal{F}_{M_0}^D = \emptyset$. Furthermore, $T_{M_0}^{ED} = \mathcal{T}_{M_0}^D$. The minimal models of $\Pi = DIS(\Pi)$ are $\{q, r\}$, $\{q, s\}$, and $\{r, p\}$. Then $F_{M_0}^{ED} = \{p \wedge q, r \wedge s, p \wedge s\}$. We could try to derive the same directly from the clauses but it is not obvious how this could be done. One attempt could be to take the disjunctive consequences like $p \vee q$ and to conclude that then one should be true and the other one false even though we do not know which one. Then $p \wedge q$ should be false. But this argument does not hold for $q \vee r$ in the example. Additionally, we obtain $p \wedge s$ to be false and there is no evident argument given in the program, at least not in a single clause which allows to draw this conclusion.

For the sake of completeness we finish this section with an example which is not propositional.

Example 4.23

$$\begin{aligned} p(0) &\leftarrow \\ p(s(X)) &\leftarrow p(X) \\ q(0) &\leftarrow \neg p(X) \\ r(X) &\leftarrow \neg q(X) \\ q(s(X)) &\leftarrow \neg r(X) \\ t &\leftarrow \neg r(X) \\ a \vee b &\leftarrow \neg t \end{aligned}$$

The generalized disjunctive well-founded model M_Π^{ED} is $\{p(s^n(0)) \mid n \geq 0\} \cup \{\neg q(s^n(0)) \mid n \geq 0\} \cup \{r(s^n(0)) \mid n \geq 0\} \cup \{\neg t, a \vee b, \neg(a \wedge b)\}$. The level mappings for the elements in the minimal set are the following: We have $l_t(p(s^n(0))) = (0, n)$, $t \in \{1, 2\}$. Then $l_1(\neg q(s^n(0))) = (2n + 1, 0)$ and $l_1(r(s^n(0))) = (2n + 2, 0)$ by the stepwise iteration of $\mathcal{T}_\Pi^D$ and $\mathcal{F}_\Pi^D$. Furthermore, $l_1(\neg t) = (\omega, 0)$ and $l_1(a \vee b) = (\omega + 1, 0)$. We also have $l_2(\neg(a \wedge b)) = (0, 0)$.

5 Disjunctive Well-founded Semantics

The third approach we study is the disjunctive well-founded semantics which was presented by Brass and Dix in [6]. This semantics is used to obtain the truth of pure disjunctions, i.e. disjunctions which consist either only of atoms or of negative literals. Thus, the derived negative information is given in form of disjunctions of negative literals which obviously differs from the previously presented approaches. However, as defined in [6], a disjunction of negative literals is only true if at least one of the contained literals is explicitly false. In other words, there is no indefinite negative knowledge and we could restrict the negative information to atoms. Instead, we use disjunctive three-valued interpretations: a negated disjunction is true whenever all elements of that disjunction are false. We will see that in this way we obtain the closure properties given in Definition 3.1 and that consistency of the disjunctive well-founded model comes for free. We note that in the original paper which addresses a wider framework of semantics the interpretations are not necessarily consistent but since our interest is restricted to the disjunctive well-founded semantics we ignore that.

There is another general difference to the two previous approaches: the semantics is only defined for (disjunctive) DATALOG programs - a subclass of disjunctive logic programs. A DATALOG program is a finite ground disjunctive program, i.e. a program whose corresponding language does not have any function symbols apart from (nullary) constants. Thus it can be identified with a propositional program and we use the notation Φ from [6] to distinguish a DATALOG program from a disjunctive logic program Π. We then refer to the Definitions 3.2 to 3.5 only changing Π to Φ and the notion of a disjunctive knowledge ordering can be used without change.

We continue recalling the operator T_Φ from [6] which given a logic program Φ derives a set of specific clauses, called conditional facts.

Definition 5.1 A *conditional fact* is a disjunctive clause without any positive atoms in the body, i.e. a clause of the form $H_1 \vee \cdots \vee H_l \leftarrow \neg B_1, \ldots, \neg B_m$.

We recall T_Φ in the following.

Definition 5.2 Let Φ be a DATALOG program and Γ be a set of conditional facts. We define:

> $T_\Phi(\Gamma) = \{(\mathcal{H} \cup \bigcup_i (\mathcal{H}_i \setminus \{A_i\})) \leftarrow \neg(\mathcal{B} \cup \bigcup_i \mathcal{B}_i) \mid$ there is $H \leftarrow A_1, \ldots, A_n, \neg B$ in $\mathsf{ground}(\Phi)$ and conditional facts $H_i \leftarrow \neg B_i \in \Gamma$ with $A_i \in \mathcal{H}_i$ for all $i = 1, \ldots, n.\}$

The iteration of T_Φ is defined in the usual way restricted to ordinals below ω since we only deal with DATALOG programs.

Definition 5.3

$$T_\Phi \uparrow 0 = \emptyset. \;\; T_\Phi \uparrow (n+1) = T_\Phi(T_\Phi \uparrow n). \;\; T_\Phi = \bigcup_{n<\omega} T_\Phi \uparrow n.$$

We use the following example to demonstrate the operator(s).

Example 5.4 Let Φ be the following program.

$$\begin{array}{rcl} a & \leftarrow & \\ b & \leftarrow & \neg a, c \\ c & \leftarrow & \neg b \\ d \vee e & \leftarrow & a, \neg b \\ f & \leftarrow & \neg d, \neg e \\ d \vee g & \leftarrow & d \end{array}$$

We start with $T_\Phi \uparrow 0 = \emptyset$ and compute $T_\Phi \uparrow 1 = T_\Phi(T_\Phi \uparrow 0) = \{a \leftarrow, c \leftarrow \neg b, f \leftarrow \neg d, \neg e\}$ where we just collect all conditional facts already contained in Φ. Then we have $T_\Phi \uparrow 2 = \{a \leftarrow, c \leftarrow \neg b, f \leftarrow \neg d, \neg e, b \leftarrow \neg a, \neg b, d \vee e \leftarrow \neg b\}$ and $T_\Phi \uparrow 3 = \{a \leftarrow, c \leftarrow \neg b, f \leftarrow \neg d, \neg e, b \leftarrow \neg a, \neg b, d \vee e \leftarrow \neg b, d \vee e \vee g \leftarrow \neg b\}$ which is also the fixed point. Note how we obtained e.g. $d \vee e \vee g \leftarrow \neg b$ from the clause $d \vee g \leftarrow d$ and the conditional fact $d \vee e \leftarrow \neg b$ contained in $T_\Phi \uparrow 2$.

The fixed point of T_Φ is of course not the semantics of the program. Instead, it is used as the starting point for another operator. But beforehand we have to recall one more notion from [6].

Definition 5.5 Let S be a set of ground clauses. Then $\mathsf{heads}(S)$ is the set of all atoms occurring in some head of a clause contained in S.

The reason for introducing this notion will be that given a set of clauses, an atom not occurring in any head cannot become true and thus can assumed to be false. This will be used in the definition of the following operator R.

Definition 5.6 Let Γ be a set of conditional facts. We define the operator R as follows:

$$R(\Gamma) = \{H \leftarrow \neg(\mathcal{B} \cap \mathsf{heads}(\Gamma)) \mid H \leftarrow \neg B \in \Gamma, \text{ and}$$

(1) there is no $H' \leftarrow$ in Γ with $\mathcal{H}' \subseteq \mathcal{B}$,

(2) there is no $H' \leftarrow \neg B'$ in Γ with $\mathcal{H}' \subseteq \mathcal{H}$ and $\mathcal{B}' \subseteq \mathcal{B}$ where at least one $\subseteq$ is proper.}

Note that the additional condition in (2) forcing one $\subseteq$ to be proper is necessary since otherwise we could remove each conditional fact by means of itself.
The iteration of this operator is defined in a different way.

Definition 5.7

$$R \uparrow 0 = T_\Phi. \;\; R \uparrow (n+1) = R(R \uparrow n).$$

The fixed point of this operator is called the residual program of Φ.

Definition 5.8 The residual program of a DATALOG program Φ is $\mathsf{res}(\Phi) = R \uparrow \omega$.

Thus we take a DATALOG program, compute T_Φ, and then calculate fixed point of R, i.e. the residual program, starting with T_Φ.

Example 5.9 We continue Example 5.4 where we obtained the fixed point $T_\Phi = \{a \leftarrow, c \leftarrow \neg b, f \leftarrow \neg d, \neg e, b \leftarrow \neg a, \neg b, d \vee e \leftarrow \neg b, d \vee e \vee g \leftarrow \neg b\}$. Then $R \uparrow 0 = T_\Phi$ and we compute $R \uparrow 1 = \{a \leftarrow, c \leftarrow \neg b, f \leftarrow \neg d, \neg e, d \vee e \leftarrow \neg b\}$. We removed the conditional fact $b \leftarrow \neg a$ because of (1) and $a \leftarrow$ and $d \vee e \vee g \leftarrow \neg b$ by (2) and $d \vee e \leftarrow \neg b$. Then $R \uparrow 2 = \{a \leftarrow, c \leftarrow, f \leftarrow \neg d, \neg e, d \vee e \leftarrow\}$ where we just removed all $\neg b$ from the clauses because b is not contained in $\mathsf{heads}(R \uparrow 1)$. Finally, $\mathsf{res}(\Phi) = R \uparrow 3 = \{a \leftarrow, c \leftarrow, d \vee e \leftarrow\}$ since we removed the conditional fact $f \leftarrow \neg d, \neg e$ by (1) and $d \vee e \leftarrow$ in $R \uparrow 2$.

The disjunctive well-founded semantics is then derived from the residual program.

Definition 5.10 Let Φ be a DATALOG program with residual program $\mathsf{res}(\Phi)$. The disjunctive well-founded model M_Φ is the following.

$M_\Phi = \{D \in EB_\Phi \mid$ there is $H \leftarrow$ in $\mathsf{res}(\Phi)$ with $\mathcal{H} \subseteq \mathcal{D}\} \cup \{\neg D \mid D \in EB_\Phi$ and $\forall D' \in D : D' \notin \mathsf{heads}(\mathsf{res}(\Phi))\}$.

As already mentioned, this differs from the way the semantics is defined in [6] but since it only contains definite negative information we do not care. This definition also shows why the disjunctive well-founded model is automatically closed with respect to definition 3.1. Thus we can apply the minimal representation also to this semantics and mention only that disjunctions without a true subset and all negative literals. All supersets are contained implicitly.

We can finish Example 5.9 by calculating the disjunctive well-founded model $M_\Phi = \{a, c, d \vee e, \neg b, \neg f\}$. Furthermore d and e remain undefined.

In [6], Brass and Dix also introduced program transformations which correspond to the computation of these two operators and in [7] they showed that these transformation rules form a confluent and terminating calculus, i.e. no matter in which order the rules are applied to Φ, the process always ends and yields a program identical to $\mathsf{res}(\Phi)$. The main argument for the termination is the same they used for showing that T_Φ and R have a least fixed point: the program Φ is finite. However, we do not need the program transformations to obtain $\mathsf{res}(\Phi)$, thus we can try to generalize the operators to disjunctive logic programs Π by showing that in general the two operators are monotonic which allows us to apply Theorem 2.20 yielding also the existence of a least fixed point.

Proposition 5.11 The operator T_Φ is monotonic.

Proof: Let $\Gamma_1 \subseteq \Gamma_2$ and let $c \in T_\Phi(\Gamma_1)$. Then by Definition 5.2, $c = (\mathcal{H} \cup \bigcup(\mathcal{H}_i \setminus \{A_i\})) \leftarrow \neg(\mathcal{B} \cup \bigcup \mathcal{B}_i)$ and there is $\mathcal{H} \leftarrow A_1, \ldots, A_n, \neg\mathcal{B}$ in $\mathsf{ground}(\Phi)$ and conditional facts $H_i \leftarrow \neg B_i$ in Γ_1 with $A_i \in \mathcal{H}_i$ for all $i = 1, \ldots, n$. Since all conditional facts contained in Γ_1 are also contained in Γ_2 we have that $c \in T_\Phi(\Gamma_2)$. ■

This result does in no way rely on DATALOG programs thus we can generalize the operator also to disjunctive logic programs. Unfortunately, this does not hold for the operator R. Let $\Gamma_1 = \{a \leftarrow \neg b\}$ and $\Gamma_2 = \{a \leftarrow \neg b, b \leftarrow\}$. Then $\Gamma_1 \subseteq \Gamma_2$ but $R(\Gamma_1) = \Gamma_1$ whereas $R(\Gamma_2) = \{b \leftarrow\}$ and we have $R(\Gamma_1) \not\subseteq R(\Gamma_2)$. This example also shows that R is not antitonic either, i.e. $R(\Gamma_2) \subseteq R(\Gamma_1)$ does not hold.

Thus we have to keep the restriction to DATALOG programs and continue with the alternative level mapping characterization. We should note that in [11] the approach was extended to disjunctive logic programs by combining the transformation rules with constraint logic programming. But the operators are not extended as well and we remain with that restriction.

We can use disjunctive I-partial mappings since we modified the semantics to disjunctive three-valued interpretations, the only difference is that like in the previous section ordinals are lexicographically ordered pairs including the possibility of applying the components of that order. Additionally we allow the order $\succ$:

Definition 5.12 Let $\alpha \times \beta$ be an ordinal and (a, b), $(a', b') \in (\alpha \times \beta)$. we define: $(a, b) \succ (a', b')$ if and only if $b > b'$.

This order ignores the first component of the ordinal and compares only the second one. We will need it specifically in the following characterization.

Definition 5.13 Let Φ be a DATALOG program, let I be a model for Φ, and let l be a disjunctive I-partial level mapping for Φ. We say that Φ satisfies (DWF) with respect to I and l if each $D \in \mathsf{dom}(l)$ satisfies one of the following conditions:

(DWFi) $D \in I$ and there is a clause $H \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Phi)$ with $\mathcal{H} \subseteq \mathcal{D}$ such that there is $\mathcal{D}_i \subseteq \mathcal{D}$ with $(D_i \vee A_i) \in I$, $l(D) > l(D_i \vee A_i)$, and $l(D) \succ l(D_i \vee A_i)$ if $l(D) >_1 l(D_i \vee A_i)$, for all $i = 1, \ldots, n$, and $\neg B_j \in I$ and $l(D) >_1 l(B_j)$ for all $j = 1, \ldots, m$.

(DWFii) $\neg D \in I$ and for each clause $H \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Phi)$ with $A \in H$ and $A \in D$ (at least) one of the following conditions holds:

(DWFiia) $\neg A_i \in I$ and $l(D) \geq l(A_i)$.

(DWFiib) $D' \in I$ with $\mathcal{D}' \subseteq \mathcal{B}$ and $l(D) >_1 l(D')$.

(DWFii') $\neg D \in I$ and for each conditional fact $H \leftarrow \neg B$ in T_Φ with $A \in H$ and $A \in D$ (at least) one of the following conditions holds:

(DWFiia') there is $H' \leftarrow \neg B'$ in $R \uparrow \alpha$ with $\mathcal{H}' \subset \mathcal{H}$ and $\mathcal{B}' \subseteq (\mathcal{B} \setminus \mathcal{D}')$ where $A \notin H'$, $B_j \in B$, $\neg B_j \in I$, and $l(D) >_1 (l(B_j) + 1)$ for all $B_j \in \mathcal{D}'$, and $l(D) >_1 (\alpha, \beta)$ for some β.

(DWFiib') $D' \in I$ with $\mathcal{D}' \subseteq \mathcal{B}$ and $l(D) >_1 l(D')$.

Then we show the following equivalence.

Theorem 5.14 Let Φ be a (disjunctive) DATALOG program with disjunctive well-founded model M. Then, in the disjunctive knowledge ordering, M is the greatest model amongst all models I for which there exists a disjunctive I-partial level mapping l for Φ such that Φ satisfies (DWF) with respect to I and l.

Proof: Let M be the disjunctive well-founded model of Φ. We define the disjunctive M-partial level mapping l in the following way: If $D \in M$ then $l(D) = (\alpha, \beta)$ where α is the least ordinal such that $H \leftarrow$ in $R \uparrow \alpha$ with $\mathcal{H} \subseteq \mathcal{D}$ and β is the least ordinal such that the corresponding conditional

fact $H \leftarrow \neg\mathcal{B}$ in $T_\Phi \uparrow (\beta+1)$. If $\neg D \in M$ then $l(D) = (\alpha, 0)$ where α is the least ordinal such that for each $A \in D$ there is no conditional fact $H \leftarrow \neg\mathcal{B}$ in $R \uparrow \alpha$ with $A \in H$. All other values remain undefined. Note that we do not assign any limit ordinals apart from 0 by means of the definitions of the operators and the restriction to DATALOG programs. By Definition 5.8, we know that the residual program of a DATALOG program Φ is $\mathsf{res}(\Phi) = R \uparrow \omega$. By Definition 5.10, $M = \{D \in EB_\Phi \mid$ there is $H \leftarrow$ in $\mathsf{res}(\Phi)$ with $\mathcal{H} \subseteq \mathcal{D}\} \cup \{\neg D \mid D \in EB_\Phi$ and $\forall D' \in D : D' \notin \mathsf{heads}(\mathsf{res}(\Phi))\}$ and thus we conclude that l is well-defined, i.e. if $D \in M$ or $\neg D \in M$ then $D \in \mathsf{dom}(l)$. We show that Φ satisfies (DWF) with respect to M and l.
Let D be in M and $l(D) = (\alpha, \beta)$. By Definition 5.10, we know that there is $H \leftarrow$ in $\mathsf{res}(\Phi)$ with $\mathcal{H} \subseteq \mathcal{D}$, i.e. $H \leftarrow$ in $R \uparrow \omega$, by Definition 5.8, and $H \leftarrow$ in $R \uparrow \alpha$ by definition of l. Then, by Definition 5.6, we have $H \leftarrow \neg\mathcal{B}$ in T_Φ and for all $B'_j \in \mathcal{B}$ we have $\neg B'_j \in M$ with $l(B'_j) = (\alpha', 0)$, $\alpha' < \alpha$ by definition of l. Then $H \leftarrow \neg\mathcal{B}$ in $T_\Phi \uparrow \beta'$ for some β' and thus $H \leftarrow \neg\mathcal{B}$ in $T_\Phi \uparrow (\beta+1)$ by definition of l. Thus, by Definition 5.2, we can unfold the derivation of that conditional fact, so that there are $H' \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Phi)$ and conditional facts $H_i \leftarrow \neg\mathcal{B}'_i$ in $(T_\Phi \uparrow \beta)$ with $A_i \in \mathcal{H}_i$ for all $i = 1, \ldots, n$ with $\mathcal{B} = (\bigcup_i \mathcal{B}'_i \cup \{B_1, \ldots, B_m\})$. Thus $l(D) >_1 l(B_j)$ for all $j = 1, \ldots, m$ since all $B_j \in \mathcal{B}$. Furthermore, $H_i = D_i \vee A_i$ for some D_i and since $H_i \leftarrow \neg\mathcal{B}'_i$ in $(T_\Phi \uparrow \beta)$, we know that $l(D) \succ l(D_i \vee A_i)$. Since $l(D) > l(B')$ for all $B' \in \mathcal{B}'_i$, $i = 1, \ldots, n$, we know that $l(D) > l(D_i \vee A_i)$ which shows that (DWFi) is satisfied.
Alternatively, $\neg D \in M$ and $l(D) = (\alpha, 0)$. We know that $\forall D' \in D : D' \notin \mathsf{heads}(\mathsf{res}(\Phi))$, by Definition 5.10, and thus $\forall D' \in D : D' \notin \mathsf{heads}(R \uparrow \omega)$ by Definition 5.8. By definition of l, α is the least ordinal such that for each $A \in D$ there is no conditional fact $H \leftarrow \neg\mathcal{B}$ in $R \uparrow \alpha$ with $A \in H$. Thus consider any conditional fact $H \leftarrow \neg\mathcal{B}$ in T_Φ with $A \in H$ and $A \in D$. Then this conditional fact also occurs in $R \uparrow 0$, by Definition 5.7, and is thus removed by an iteration step $R \uparrow \alpha'$, $\alpha' < \alpha$. By definition of R, a conditional fact $H \leftarrow \neg\mathcal{B}$ is removed from a set Γ if there is $H' \leftarrow$ in Γ with $\mathcal{H} \subseteq \mathcal{B}$, or there is $H' \leftarrow \neg\mathcal{B}'$ in Γ with $\mathcal{H}' \subseteq \mathcal{H}$ and $\mathcal{B}' \subseteq \mathcal{B}$ where at least one $\subseteq$ is proper. Suppose that $H \leftarrow \neg\mathcal{B}$ is removed in the iteration step $R \uparrow \alpha'$ because there is $H' \leftarrow$ in $R \uparrow (\alpha' - 1)$ with $\mathcal{H}' \subseteq \mathcal{B}$. Then $H' \in M$ by Definition 5.10. By definition of l, we have $l(H') = (\alpha'', \beta')$ for some β' and $\alpha'' < \alpha'$ and thus $l(D) >_1 l(H')$. Hence (DWFiib') is satisfied. Otherwise, if $H \leftarrow \neg\mathcal{B}$ is removed in the iteration step $R \uparrow \alpha'$, because there is $H' \leftarrow \neg\mathcal{B}'$ in $R \uparrow (\alpha'-1)$ with $\mathcal{H}' \subseteq \mathcal{H}$ and $\mathcal{B}' \subseteq \mathcal{B}$ where at least one $\subseteq$ is proper, then we also have a conditional fact $H \leftarrow \neg(\mathcal{B} \cup \mathcal{D}')$ in $R \uparrow 0$ with $\neg B'_j \in I$ and $l(B'_j) = (\alpha'', \beta'')$, $\alpha'' < \alpha'$, for all $B'_j \in \mathcal{D}'$. Then $l(D) >_1 (l(B'_j) + 1)$ and $l(D) >_1 (\alpha', \beta')$ for some β'. Without loss of generality we assume that $A \notin H'$, otherwise

there still would be a conditional fact contained in $R \uparrow \alpha'$ with A in the head which is also eliminated either by (1) or (2) of the definition of R, and in both cases this implies that either (DWFiia') or (DWFiib') is satisfied. We conclude that (DWFiia') is satisfied.

Conversely, we show that if I is a model of Φ and l a disjunctive I-partial level mapping such that Φ satisfies (DWF) with respect to I and l then $I \subseteq M_\Phi$. We show via induction on α for $l(D) = (\alpha, \beta)$ that whenever $D \in I$ then $H \leftarrow$ in $R \uparrow \alpha$ with $\mathcal{H} \subseteq \mathcal{D}$ and whenever $\neg D \in I$ then for all $A \in D$ there is no conditional fact $H \leftarrow \neg B$ in $R \uparrow \alpha$ with $A \in H$. By Definition 5.10, this suffices to show that $D \in M_\Phi$, $\neg D \in M_\Phi$ respectively.

Let $\alpha = 0$. We have to consider three cases.
Let $D \in I$. By (DWFi) there is a clause $H \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Phi)$ with $\mathcal{H} \subseteq \mathcal{D}$ such that there is $\mathcal{D}_i \subseteq \mathcal{D}$ with $(D_i \vee A_i) \in I$, $l(D) > l(D_i \vee A_i)$, and $l(D) \succ l(D_i \vee A_i)$ if $l(D) >_1 l(D_i \vee A_i)$, for all $i = 1, \ldots, n$, and $\neg B_j \in I$ and $l(D) >_1 l(B_j)$ for all $j = 1, \ldots, m$. Since there is no ordinal smaller than 0, we know that $>_1$ cannot hold and (DWFi) simplifies to that there is a clause $H \leftarrow A_1, \ldots, A_n$ in $\mathsf{ground}(\Phi)$ with $\mathcal{H} \subseteq \mathcal{D}$ such that there is $\mathcal{D}_i \subseteq \mathcal{D}$ with $(D_i \vee A_i) \in I$ and $l(D) >_2 l(D_i \vee A_i)$ for all $i = 1, \ldots, n$. We show by induction on β that $(\mathcal{H} \cup \bigcup_i \mathcal{D}_i) \leftarrow$ in $T_\Phi \uparrow (\beta + 1)$ and thus $(\mathcal{H} \cup \bigcup_i \mathcal{D}_i) \leftarrow$ in T_Φ by Definition 5.3. Then $(H \cup \bigcup_i \mathcal{D}_i) \leftarrow$ in $R \uparrow 0$ by Definition 5.7. This suffices since $(\mathcal{H} \cup \bigcup_i \mathcal{D}_i) \subseteq \mathcal{D}$.

Let β be 0. Since there is no ordinal smaller than 0, the considered clause is a fact, $H \leftarrow$ in $T_\Phi \uparrow 1$ and all D_i are empty.

Suppose that the claim holds for all $D \in I$ with $l(D) = (0, \beta')$, $\beta' \leq \beta$ and let $D \in I$ with $l(D) = (0, \beta + 1)$. We know that D satisfies the simplified (DWFi), so there is a clause $H \leftarrow A_1, \ldots, A_n$ in $\mathsf{ground}(\Phi)$ with $\mathcal{H} \subseteq \mathcal{D}$ such that there is $\mathcal{D}_i \subseteq \mathcal{D}$ with $(D_i \vee A_i) \in I$ and $l(D) >_2 l(D_i \vee A_i)$ for all $i = 1, \ldots, n$. By assumption, for all $A_i \vee D_i$ we have a conditional fact $H'_i \leftarrow$ in $T_\Phi \uparrow (\beta + 1)$ with $\mathcal{H}'_i \subseteq (\mathcal{D}_i \cup \{A_i\})$. If $A_i \notin H'_i$ for one i then we have $\mathcal{H}'_i \subseteq \mathcal{D}_i$ and since $\mathcal{D}_i \subseteq \mathcal{D}$ we obtain that $H'_i \leftarrow$ in $T_\Phi \uparrow (\beta + 1)$ and thus also in $T_\Phi \uparrow (\beta + 2)$. Otherwise $A_i \in H_i$ for all $i = 1, \ldots, n$ and $H'_i = (\mathcal{D}'_i \cup \{A_i\})$ with $\mathcal{D}'_i \subseteq \mathcal{D}_i$. By Definition 5.2, then we have a fact $(\mathcal{H} \cup \bigcup_i D_i) \leftarrow$ in $T_\Phi \uparrow (\beta + 2)$ with $(\mathcal{H} \cup \bigcup_i D_i) \subseteq \mathcal{D}$ which finishes this case.
Let $\neg D \in I$ and D satisfies (DWFii'). Then for each conditional fact $H \leftarrow \neg B$ in T_Φ with $A \in H$ and $A \in D$ we have that (DWFiia') or (DWFiib') holds. Since we know that there is no ordinal smaller than 0, neither (DWFiia') nor (DWFiib') can hold and thus there is no conditional fact $H \leftarrow \neg B$ in T_Φ with $A \in H$ and $A \in D$. Then there is also no such conditional fact in $R \uparrow 0$, by Definition 5.7, which finishes this case.
Let $\neg D \in I$ and D satisfies (DWFii). Since we know that there is no ordinal smaller than 0, (DWFiib) cannot hold and by (DWFiia) for each clause $H \leftarrow$

$A_1, \dots, A_n, \neg B_1, \dots, \neg B_m$ in $\mathsf{ground}(\Phi)$ with $A \in H$ and $A \in D$ we have $\neg A_i \in I$ and $l(D) >_2 l(A_i)$. Consider such a clause. By definition of l, we know that A_i also satisfies either (DWFii) or (DWFii'). If A_i satisfies (DWFii') then there is no conditional fact in $R \uparrow 0$ with A_i in the head and thus also not in T_Φ. Then, according to Definition 5.2, the considered clause cannot be used to derive a conditional fact in T_Φ. Alternatively, if A_i satisfies (DWFii) then we can re-apply the argument. Thus for each clause we derive either eventually a dependency on an atom which satisfies (DWFii') or we have an infinite chain of atoms which are false in I and satisfy (DWFii), but in both cases we will not have a conditional fact in T_Φ with A in the head. Thus, by Definition 5.7, there is no such conditional fact in $R \uparrow 0$ which also finishes this case and therefore the induction base.

Assume for all D with $l(D) = (\alpha', \beta')$, for arbitrary β' and $\alpha' \leq \alpha$, that whenever $D \in I$ then $H \leftarrow$ in $R \uparrow \alpha'$ with $\mathcal{H} \subseteq \mathcal{D}$ and whenever $\neg D \in I$ then for all $A \in D$ there is no conditional fact $H \leftarrow \neg B$ in $R \uparrow \alpha'$. We are going to show this property for D with $l(D) = (\alpha + 1, \beta)$ for some β, and we consider three cases again.
Let $D \in I$. By (DWFi) there is a clause $H \leftarrow A_1, \dots, A_n, \neg B_1, \dots, \neg B_m$ in $\mathsf{ground}(\Phi)$ with $\mathcal{H} \subseteq \mathcal{D}$ such that there is $\mathcal{D}_i \subseteq \mathcal{D}$ with $(D_i \vee A_i) \in I$, $l(D) > l(D_i \vee A_i)$, and $l(D) \succ l(D_i \vee A_i)$ if $l(D) >_1 l(D_i \vee A_i)$, for all $i = 1, \dots, n$, and $\neg B_j \in I$ and $l(D) >_1 l(B_j)$ for all $j = 1, \dots, m$. We show by induction on β that $(\mathcal{H} \cup \bigcup_i \mathcal{D}_i) \leftarrow \neg B_1, \dots, \neg B_m, \neg B_{m+1}, \dots, \neg B_r$ in $T_\Phi \uparrow (\beta + 1)$ with $(\mathcal{H} \cup \bigcup_i \mathcal{D}_i) \subseteq \mathcal{D}$ and $\neg B_j \in M_\Phi$ and $l(D) >_1 l(\neg B_j)$ for all $j = 1, \dots, r$. Then $(\mathcal{H} \cup \bigcup_i \mathcal{D}_i) \leftarrow \neg B_1, \dots, \neg B_m, \neg B_{m+1}, \dots, \neg B_r$ in T_Φ by Definition 5.3. Since $l(D) >_1 l(\neg B_j)$, for all $j = 1, \dots, r$, we know by induction hypothesis that all $B_j \notin \mathsf{heads}(R \uparrow \alpha)$ and thus, by Definition 5.2, none of these appear in the body of the conditional fact occurring in $R \uparrow (\alpha + 1)$. Then $(H \cup \bigcup_i \mathcal{D}_i) \leftarrow$ in $R \uparrow (\alpha + 1)$ which finishes this case because $(\mathcal{H} \cup \bigcup_i \mathcal{D}_i) \subseteq \mathcal{D}$.

Let β be 0. Consider there are positive atoms in the body of the clause. Since there is no ordinal smaller than 0, we know that $l(D) > l(D_i \vee A_i)$ for all $i = 1, \dots, n$, can only be satisfied by $l(D) >_1 l(D_i \vee A_i)$. Thus for all these atoms the additional condition $l(D) \succ l(D_i \vee A_i)$ from (DWFi) has to hold. Since we have 0 as smallest ordinal, this cannot hold either and we conclude that there cannot be any positive atoms in the body. Then $H \leftarrow \neg B_1, \dots, \neg B_m$ in $\mathsf{ground}(\Phi)$ and also in $T_\Phi \uparrow 1$, and for all B_j, $j = 1, \dots, m$, we have $l(D) >_1 l(B_j)$ and thus, by induction hypothesis, $\neg B_j \in M_\Phi$.

Suppose that the property holds for all D with $l(D) = (\alpha, \beta')$, $\beta' \leq \beta$. We show that it holds for D with $l(D) = (\alpha, \beta + 1)$. For all $D_i \vee A_i$ we know $l(D) > l(D_i \vee A_i)$. If $l(D) >_1 l(D_i \vee A_i)$ then $l(D) \succ l(D_i \vee A_i)$. Otherwise $l(D) >_2 l(D_i \vee A_i)$ and by assumption, respectively by induction hypothesis in the former case, there is a conditional fact $H_i' \leftarrow \neg B_i'$ in $T_\Phi \uparrow (\beta+1)$ with $\mathcal{H}_i' \subseteq (\mathcal{D}_i \cup \{A_i\})$. If $A_i \notin H_i'$ for one i then we have $\mathcal{H}_i' \subseteq \mathcal{D}_i$. Since $\mathcal{D}_i \subseteq \mathcal{D}$ we obtain that $H_i' \leftarrow \neg B_i'$ in $T_\Phi \uparrow (\beta+1)$ and thus also in $T_\Phi \uparrow (\beta+2)$ where all $\neg B \in \neg B_i'$ are contained in M_Φ by induction hypothesis. Otherwise $A_i \in H_i$ for all $i = 1, \ldots, n$ and $H_i' = (\mathcal{D}_i' \cup \{A_i\})$ with $\mathcal{D}_i' \subseteq \mathcal{D}_i$. By Definition 5.2, then we have a conditional fact $(\mathcal{H} \cup \bigcup_i D_i) \leftarrow \neg(\bigcup_i \mathcal{B}_i' \cup \{B_1, \ldots, B_m\})$ in $T_\Phi \uparrow (\beta+2)$ with $(\mathcal{H} \cup \bigcup_i D_i) \subseteq \mathcal{D}$ and, by induction hypothesis, $\neg B_j \in M_\Phi$ and $\neg B' \in M_\Phi$ for all $B' \in B_i'$, for each i, which finishes this case.

Let $\neg D \in I$ and D satisfies (DWFii'). Then for each conditional fact $H \leftarrow \neg B$ in T_Φ with $A \in H$ and $A \in D$ we know that (DWFiia') or (DWFiib') holds. Consider such a conditional fact. If (DWFiia') holds then there is $H' \leftarrow \neg B'$ in $R \uparrow \alpha'$ with $\mathcal{H}' \subset \mathcal{H}$ and $\mathcal{B}' \subseteq (\mathcal{B} \setminus \mathcal{D}')$ where $A \notin H'$, $B_j \in B$, $\neg B_j \in I$, and $l(D) >_1 (l(B_j) + 1)$ for all $B_j \in \mathcal{D}'$, and $l(D) >_1 (\alpha', \beta')$ for some β'. Then for all $B_j \in \mathcal{D}'$, $B_j \notin \mathsf{heads}(R \uparrow (\alpha-1))$. Thus $H \leftarrow \neg(\mathcal{B} \setminus \mathcal{D}')$ in $R \uparrow \alpha$. Then by Definition 5.6, $H \leftarrow \neg(\mathcal{B} \setminus \mathcal{D}')$ is not contained in $R \uparrow (\alpha+1)$ since $A_i \notin H'$. If (DWFiib') holds then $D' \in I$ with $\mathcal{D}' \subseteq \mathcal{B}$ and $l(D) >_1 l(D')$. By induction hypothesis, we know that $H \leftarrow$ in $R \uparrow \alpha'$ with $\mathcal{H} \subseteq \mathcal{D}'$ and $\alpha' \leq \alpha$ and therefore also in $R \uparrow \alpha$. Then by Definition 5.6, $H \leftarrow \neg B$ is not contained in $R \uparrow (\alpha+1)$ as well. Thus there is no conditional fact $\mathcal{H} \leftarrow \neg\mathcal{B}$ with $A \in H$ and $A \in D$ in $R \uparrow (\alpha+1)$ which finishes this case.

Let $\neg D \in I$ and D satisfies (DWFii). Then for each clause $H \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Phi)$ with $A \in H$ and $A \in D$ we have that (DWFiia) or (DWFiib) holds. Consider such a clause. If (DWFiib) holds then $D' \in I$ with $\mathcal{D}' \subseteq \mathcal{B}$ and $l(D) >_1 l(D')$. Thus if this clause is used to add a conditional fact to T_Φ then this conditional fact contains still all $\neg B_j$, $j = 1, \ldots, m$. By induction hypothesis, we know that there is $H \leftarrow$ in $R \uparrow \alpha'$, $\alpha' \leq \alpha$, with $\mathcal{H} \subseteq \mathcal{D}'$ and thus also in $R \uparrow \alpha$. Then this conditional fact with A in the head does not occur in $R \uparrow (\alpha+1)$ by definition of R. Otherwise this clause is not used to add a conditional fact to T_Φ and there is no resulting conditional fact in $R \uparrow (\alpha+1)$ either. If (DWFiia) holds then $\neg A_i \in I$ and $l(D) \geq l(A_i)$. We consider two cases. A_i satisfies (DWFii'). Then for each conditional fact $H \leftarrow \neg B$ in T_Φ with $A_i \in H$ we know that (DWFiia') or (DWFiib') holds. Consider such a conditional fact. If (DWFiia') holds then there is $H' \leftarrow \neg B'$ in $R \uparrow \alpha'$ with $\mathcal{H}' \subset \mathcal{H}$ and $\mathcal{B}' \subseteq (\mathcal{B} \setminus \mathcal{D}')$ where $A \notin H'$, $B_j \in B$, $\neg B_j \in I$, and $l(D) >_1 (l(B_j)+1)$ for all $B_j \in \mathcal{D}'$, and $l(D) >_1 (\alpha', \beta')$ for some β'. If we apply this conditional fact in one step of T_Φ to the clause mentioned above then the resulting conditional fact in T_Φ has to contain $\mathcal{H} \setminus \{A_i\}$ as a subset of the head and $\neg\mathcal{B}$ as a subset of the negative literals in the resulting body. But then this resulting conditional fact is removed by (2) of Definition 5.6 and $H' \leftarrow \neg B'$ in $R \uparrow \alpha$ and thus does not occur in $R \uparrow (\alpha+1)$. We then derive that in this case (DWFiia') is satisfied for A.

If (DWFiib') holds then $D' \in I$ with $\mathcal{D}' \subseteq \mathcal{B}$ and $l(A_i) >_1 l(D')$. If we use this conditional fact $H \leftarrow \neg B$ in T_Φ with $A_i \in H$ to derive a conditional fact then, by means of T_Φ, the clause mentioned above, and by substituting A_i, $\neg B$ is a subset of the body of the resulting conditional fact. But since $D' \in I$ with $\mathcal{D}' \subseteq \mathcal{B}$ and $l(A_i) >_1 l(D')$ we know by induction hypothesis that there is $H' \leftarrow$ in $R \uparrow \alpha$ with $\mathcal{H}' \subseteq \mathcal{D}'$ and thus the resulting conditional fact does not occur $R \uparrow (\alpha+1)$ and therefore A satisfies also (DWFiib'). Alternatively, A_i satisfies (DWFii). Then for each clause $H \leftarrow A'_1, \ldots, A'_n, \neg B'_1, \ldots, \neg B'_m$ in $\mathsf{ground}(\Phi)$ with $A_i \in H$ we have that (DWFiia) or (DWFiib) holds. Consider such a clause. If (DWFiib) holds then $D' \in I$ with $\mathcal{D}' \subseteq \mathcal{B}'$ and $l(A_i) >_1 l(D')$. If this clause is used to derive a conditional fact in T_Φ then we can apply the very same argument as in case of (DWFiib') and obtain that A also satisfies (DWFiib'). Otherwise, the clause does not allow for having a conditional fact in T_Φ with A in the head. Alternatively, (DWFiia) holds and $\neg A'_{i'} \in I$ and $l(A_i) \geq l(A'_{i'})$. Then we can re-apply the argument and for each clause we derive either eventually a dependency on an atom which satisfies (DWFii') or we have an infinite chain of atoms which are false in I and satisfy (DWFiia), but in both cases we will not have a conditional fact in $R \uparrow (\alpha + 1)$ with A in the head which finishes the case and the induction step. ■

Example 5.15 We continue Example 5.9, where we obtained the disjunctive well-founded model $M_\Phi = \{a, c, d \vee e, \neg b, \neg f\}$. For the exact iteration of T_Φ we refer to Example 5.4. Then, following the definition of l in the first part of the previous proof, we obtain the levels of all elements contained in M_Φ: $l(a) = (0,0)$, $l(c) = (2,0)$, $l(d \vee e) = (2,1)$, $l(b) = (1,1)$, and $l(f) = (3,0)$.

Similarly to Section 3, there is a possible simplification of the alternative characterization. In the part of the proof where we showed that M_Φ together with the defined l satisfies (DWF), we only needed (DWFii'), not (DWFii). Thus the following corollary is straightforward.

Corollary 5.16 Let Φ be a (disjunctive) DATALOG program with disjunctive well-founded model M. Then, in the disjunctive knowledge ordering, M is the greatest model amongst all models I for which there exists a disjunctive I-partial level mapping l for Φ such that Φ satisfies (DWF') with respect to I and l where (DWF') is (DWF) without (DWFii).

Since the levels are pairs of ordinals, a statement of the identity of the characterizations of the well-founded semantics and the disjunctive well-founded semantics is impossible even though it was shown in [6] that the semantics coincide for normal programs. We note again that for disjunctive three-valued interpretations this can only hold if we restrict the latter to literals.

Example 5.17 Let Φ be the following program.

$$\begin{array}{rcl} a & \leftarrow & \\ b & \leftarrow & a \\ c & \leftarrow & \neg b \end{array}$$

We have $M_\Phi = \{a, b, \neg c\}$ which is also the well-founded model. The mapping from Definition 2.46 gives $l(a) = 0$, $l(b) = 1$, and $l(c) = 2$. It is not obvious how this can be related to the mapping we obtain by Definition 5.13: $l(a) = (0, 0)$, $l(b) = (0, 1)$, and $l(c) = (1, 0)$. Of course, we keep the relation in the corresponding order but nothing more specific seems to be possible.

We continue like in the previous section with some examples for showing why the characterization is defined in that way.

Example 5.18 We start with the following program Φ.

$$\begin{array}{rl} p \leftarrow & r, \neg q \\ r \leftarrow & \neg s \\ q \vee s \leftarrow & \end{array}$$

We obtain $T_\Phi = \{p \vee q \leftarrow \neg s, \neg q, r \leftarrow \neg s, q \vee s \leftarrow\}$. Then $R \uparrow 1 = \{r \leftarrow \neg s, q \vee s \leftarrow\}$ which is also the fixed point and we have $M_\Phi = \{q \vee s, \neg p\}$. The problem is the very same as with the strong well-founded semantics. We derive p to be false but both, q and r, remain undefined and (DWFii) alone is not sufficient for the characterization. Instead, we need a more general case which is (DWFiib') covering the combination of the negative literals from several clauses.

Unfortunately, there is still another option available.

Example 5.19 Let Φ be given with these clauses:

$$\begin{array}{rcl} p \vee q & \leftarrow & \neg q \\ q & \leftarrow & \neg q, \neg e \end{array}$$

We obviously derive $T_\Phi = \Phi$ and the same holds for $R \uparrow 0$ by Definition 5.7. Then $R \uparrow 1 = \{p \vee q \leftarrow \neg q, q \leftarrow \neg q\}$, i.e. we removed $\neg e$ from one body and thus $R \uparrow 2 = \{q \leftarrow \neg q\}$. Then $M_\Phi = \{\neg p, \neg e\}$ and $l(p) = (2, 0)$ and $l(e) = (0, 0)$. We use this example to explain why (DWFiia') is defined in such a complicated way. First of all, p is false but q (and thus also $\neg q$) is undefined, i.e. (DWFii) is not applicable. Furthermore, (DWFiib') does not help since we do not substitute anything in the body of the first clause. So we added a special case (DWFiia') corresponding to the treatment of conditional facts in (2) of the operator R (cf. Definition 5.6). The first problem is that in general we do not assign level to clauses or conditional facts. That is why we put the explicit reference to the iteration of R, something which is not occurring in any other characterization up to now. Moreover, this example shows the reason for the definition of the level in (DWFiia'). Only after $\neg e$ is removed from the body, we can apply (2) of R in the next step which eliminates the conditional fact containing p in the head.

Finally, we will present arguments for the introduction of $\succ$ as additional relation.

Example 5.20 Let Φ be the following.

$$\begin{array}{rcl} a & \leftarrow & \\ b & \leftarrow & a \\ c & \leftarrow & b, \neg d \end{array}$$

We have $T_\Phi = \{a \leftarrow, b \leftarrow, c \leftarrow \neg d\}$ and $R \uparrow 1 = \{a \leftarrow, b \leftarrow, c \leftarrow\}$. Then $M_\Phi = \{a, b, c, \neg d\}$ and we obtain $l(a) = (0, 0)$, $l(b) = (0, 1)$, $l(c) = (1, 2)$, and $l(d) = (0, 0)$. Obviously, c is true and satisfies (DWFi). However, we could set $l(c) = (1, 0)$ and remove the part of the condition referring to $\succ$ and this would hold as well. But then the mapping does no longer correspond to the construction used to define it in the first part of the proof of Theorem 5.14 and it would not be possible to show the other direction of the equivalence which is based on this construction.

6 Discussions

6.1 Earlier Comparisons

As already said, there is a large number of different semantics defined for disjunctive logic programs during the last 25 years. Most of them are based on completely different constructions and differ in the desired application area. Some semantics treat the disjunctions exclusively whenever possible, others inclusively. Various approaches are only defined for definite disjunctive programs or stratified ones. Quoting from [10], '...the starting point of most of these semantics was a particular program that was, according to some researchers, not handled correctly by the existing semantics. The new semantics was then defined to give the desired conclusions.' We have to add that comparisons to previous approaches emphasize on the advantages while the disadvantages are only rarely mentioned or not at all.

Dix introduced in [8] and [9] general properties for semantics of normal logic programs to allow comparisons of the semantics and alternative characterizations by means of the properties. They are divided into strong ([8]) and weak ([9]) ones. While the latter are mainly introduced for specifically dealing with negation, the former are adaptations of more general already existing principles. In [10], Dix and Müller extended this framework to disjunctive logic programs lifting some of the weak and strong properties and adding new properties specifically for disjunctive programs. Most of these properties are nowadays widely accepted to be necessarily fulfilled. Referring to [10] for the complete overview of all properties and a comparison of a large variety of semantics, we restrict our interest to the three semantics we studied in the previous sections and the properties which are used for the program transformation rules forming a terminating and confluent calculus ([7]). In fact, it was shown in [6] that the disjunctive well-founded semantics is the weakest semantics satisfying the following five properties. All definitions consider a semantics to be a relation between programs and their derived consequences and we start with recalling the *generalized principle of partial evaluation*, abbreviated with GPPE.

Definition 6.1 A semantics $\vdash$ satisfies GPPE if and only if the following transformation on instantiated logic programs is a $\vdash$-equivalence transformation:

Replace a clause $H \leftarrow A, \neg B$ where A contains an atom A' by the clauses
$\mathcal{H} \cup (\mathcal{H}_i - \{A'\}) \leftarrow (\mathcal{A} - \{A'\}) \cup \mathcal{A}_i \cup \neg(\mathcal{B} \cup \mathcal{B}_i)$
where $H_i \leftarrow A_i, \neg B_i$ are all ground clauses with $A \in \mathcal{A}_i$.

Intuitively, this allows to substitute positive atoms in the body of a clause by clauses with that atom in the head. This usually increases the number of clauses but the following two rules will reduce it again.

Definition 6.2 A semantics $\vdash$ allows 1.) the *elimination of tautologies*, respectively 2.) the *elimination of non-minimal clauses*, if and only if the following transformations on instances of logic programs are $\vdash$-equivalence transformations:

1. Delete a clause $H \leftarrow A, \neg B$ with $\mathcal{H} \cap \mathcal{A} \neq \emptyset$.
2. Delete a clause $H \leftarrow A, \neg B$ if there is another clause $H' \leftarrow A', \neg B'$ with $\mathcal{H}' \subseteq \mathcal{H}$, $\mathcal{A}' \subseteq \mathcal{A}$ and $\mathcal{B}' \subseteq \mathcal{B}$.

Both transformation rules basically remove clauses which are redundant and do not yield any additional information.
There are two more rules to reduce the programs.

Definition 6.3 A semantics allows 1.) *positive*, respectively 2.) *negative reduction* if and only if the following are $\vdash$-equivalence transformations on instantiated logic programs Π:

1. Replace a clause $H \leftarrow A, \neg B$ by $H \leftarrow A, \neg(\mathcal{B} \cap \mathsf{heads}(\Pi))$.
2. Delete a clause $H \leftarrow A, \neg B$ if there is a clause $H' \leftarrow$ with $\mathcal{H}' \subseteq \mathcal{B}$.

We note the striking similarity to the definitions of T_Φ and R in the previous section.

As already said, the disjunctive well-founded semantics satisfies these five concepts and one result of [10] is that the same holds for the generalized disjunctive well-founded semantics. The strong well-founded semantics is not mentioned in this article but in [6] and [7] the corresponding results are presented. SWFS allows positive and negative reduction but neither GPPE nor elimination of tautologies or non-minimal clauses. In [6], the counterexamples for elimination of non-minimal clauses and GPPE are presented. Unfortunately, the counterexample for GPPE is not correct.

Example 6.4 ([6]) Let the following program Π be given.

$$\begin{array}{rcl} p \vee q & \leftarrow & r \\ p \vee r & \leftarrow & \\ r & \leftarrow & p \end{array}$$

They stated that $\neg p$ is contained in the strong well-founded model M^S_{WF} which violates two equivalence results in [29] and is no longer derivable if we add the consequence $p \vee q$ to the program. But this is not true.

We determine M^S_{WF}. There is a derivation sequence $\{r\} \Leftarrow \{p \vee r\} \Leftarrow \{\}$ in Γ^D_r and thus r is true in M^S_{WF}. We also have $\{p \vee q\} \Leftarrow \{p \vee q \vee r\} \Leftarrow \{\}$ and $p \vee q$ is true as well. However, for showing that p is false, every child in Γ^D_p has to be true. In particular, there is one derivation sequence $\{p\} \Leftarrow \{r, \neg q\} \Leftarrow \{p \vee r, \neg q\} \Leftarrow \{\neg q\}$ with child q of which we have to show that it is true. Thus, we have to find a derivation sequence in Γ^D_q which has a false child. The only possibility for the first derivation of any sequence starting with q is $\{q\} \Leftarrow \{r, \neg p\}$. Then, no matter how we continue, the child contains p. We know that $p \vee r$ and $p \vee q$ are true thus the only possible false child is p itself and we obtain an infinite loop, yielding that both, p and q are undefined in M^S_{WF}. Thus, the argument in [6] is not correct.

Note that this does not mean that the strong well-founded semantics satisfies GPPE, we only know that the presented counterexample is not sufficient and it thus remains an open question whether GPPE is satisfied by the strong well-founded semantics or not.

6.2 Generalization of the Well-founded Semantics

We have seen that both, D-WFS and GDWFS, satisfy the five program transformation rules whereas SWFS does not. Furthermore, in [10] it was shown that GDWFS always derives more knowledge than D-WFS which is no surprise since as already mentioned D-WFS is the weakest semantics satisfying these properties. However, we do not have a similar result for D-WFS and SWFS since they are incomparable with respect to the derived knowledge.

Example 6.5 ([6]) Let the following program Π be given.

$$\begin{aligned} p \vee q &\leftarrow \neg q \\ q &\leftarrow \neg q \end{aligned}$$

This is the counterexample which shows that SWFS does not satisfy the principle of elimination of non-minimal clauses. The first clause is non-minimal, but removing it allows to derive p to be false. For the entire program this is not possible since we have the undefined child q for p. Clearly, since D-WFS satisfies elimination of non-minimal clauses we obtain that p is false.

The next example shows that there are also programs for which SWFS derives more knowledge than D-WFS.

Example 6.6 ([36]) Let the following program Π be given which was originally presented to show that the advantage of Wang's well-founded disjunctive semantics WFDS compared to D-WFS.

$$\begin{aligned} b \vee l &\leftarrow \neg p \\ l \vee p &\leftarrow \end{aligned}$$

In D-WFS, we obtain that b remains undefined because $\mathsf{res}(\Pi) = \Pi$. To the contrary, we have a derivation sequence $\{b\} \Leftarrow \{\neg p, \neg l\}$ in Γ_b^S and thus one true child, i.e. b is false in SWFS.

As we have seen, there is a semantics not satisfying all the five principles but deriving more knowledge for some programs. We conclude that those principles may not be the only criteria. We e.g. also want that the semantics coincide with the well-founded semantics for normal logic programs. But this also holds for the two semantics SWFS and D-WFS, and we come to the results of our alternative characterizations which give more arguments for the comparison.

At first, a main advantage of the level mapping characterizations is that we separate the derivation of positive and negative information. Our main result of the undertaken investigations is that any characterization basically yields that a true disjunction D satisfies the following scheme with respect to the model I and the program Π.

> $D \in I$ and there is a clause $H \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in $\mathsf{ground}(\Pi)$ with $\mathcal{H} \subseteq \mathcal{D}$ such that there is $\mathcal{D}_i \subseteq \mathcal{D}$ with $(D_i \vee A_i) \in I$, $l(D) > l(D_i \vee A_i)$, for all $i = 1, \ldots, n$, and $\neg B_j \in I$ and $l(D) > l(B_j)$ for all $j = 1, \ldots, m$.

We can see that this corresponds in general to (SWFia) from Definition 3.39, to (GDWFi) from Definition 4.19, and to (DWFi) from 5.13 and thus to one condition of all the three characterizations. We only have to consider that the relation $>$ is literally not exact and that, due to the construction, we sometimes need a more specific order. Nevertheless, in all cases we obtain levels such that $l(D)$ is greater with respect to the specific ordering. Theoretically, there are further differences. For (SWF), we have to abstract additionally from the notion of derivation sequences and their children, and there is also (SWFib). But we argumented in Section 3, that these are only proof-theoretical problems which arise from the difficulty of showing some natural closure properties of the strong well-founded model. Moreover, instead of allowing some subset D_i of D to form a true disjunction with some A_i, the approach only considers the entire disjunction D, as a consequence of the approach itself. However, it is straightforward to see that if $D \vee A_i$ is true then there is also a subset D_i of D, namely D itself, such that $D_i \vee A_i$

is true and likewise if $D_i \vee A_i$ is true then also $D \vee A_i$. In case of (GDWF) we have additionally a condition (GDWFi') but it is easy to see that this is the part (corresponding to T^D_Π) which allows GDWFS to derive more knowledge than the well-founded semantics and is thus not an intended result for a well-founded semantics for disjunctive programs. We claim that the condition given above is the 'disjunctive' version of (WFi) from Definition 2.46 and we propose it to be a necessarily satisfied condition for any semantics aiming to extend the well-founded semantics to disjunctive logic programs.

If we continue looking for common adequate extensions of (WFii) to disjunctive logic programs including (WFiia) and (WFiib) then we see that the conditions for negative information obtained for the three semantics differ a lot. A straightforward extension to disjunctive programs is given in Definition 5.13 with (DWFii) containing appropriate cases (DWFiia) and (DWFiib). For SWFS we have (SWFiia') (cf. Definition 3.39) which somehow approximates that, ignoring derivation sequences, but we also have (SWFiia) which refers to $A_i \vee D$ being false. We know that then A_i is false as well but we cannot derive the contrary, again because of the missing closure properties. GDWFS is based on false conjunctions and we have (GDWFiia) and (GDWFiia') corresponding to (DWFiia) but possibly being based on indefinite information. We just have two cases because of the construction of the operators defining GDWFS, from a more general point of view one statement suffices to guarantee that $\geq$ holds with respect to some order. For (DWFiib) it seems to be easier, because we have (SWFiib') and (GDWFiib) as corresponding statements, ignoring some minor proof-theoretical details. Unfortunately, since positive information may be indefinite, it is also possible to obtain a correspondence to (WFiib) which results from several clauses (see e.g. Example 5.18). This is covered by (SWFiic), (GDWFii'), and (DWFiib'). But this is not the whole characterization for any of the cases. (SWFiib") extends (SWFiib') to include particular atoms from the head. (GDWFii') is in fact much more powerful by means of the EGCWA and allows for deriving more knowledge difficult to characterize in a clause-based approach. In case of D-WFS we also have (DWFiia') which resolves the elimination of non-minimal clauses, a feature not contained in SWFS and also covered by (GDWFii') for GDWFS.

Altogether, we conclude that the derivation of negative information is in fact the problem which makes comparisons of the various semantics so difficult. All characterizations contain more or less straightforward extensions of (WFii) but additionally non-trivial conditions some of which are even difficult to capture within a clause-based approach like the level mapping characterizations. It remains an open question which of the conditions to include in an appropriate characterization for a well-founded semantics of disjunctive programs.

6.3 Minimal Model Semantics

Finally, we want to show the problems we had with the level mapping characterization of the minimal model semantics ([22]) which is in fact needed for the EGCWA in case of the GDWFS. We mentioned in Section 4 that we could not resolve the problem and stayed with a rather unspecific condition not based on clauses. We recall in the following the characterization of minimal models from [18].

Theorem 6.7 ([18]) Let Π be a definite disjunctive program. Then a model M of Π is a minimal model of Π if and only if there exists a total level mapping $l : B_\Pi \rightarrow \alpha$ such that for each $A \in M$ exists a clause $A \vee H_1 \vee \cdots \vee H_l \leftarrow A_1, \ldots, A_n$ in $\mathsf{ground}(\Pi)$ with $A_i \in M$, $H_k \notin M$, and $l(A) > l(A_i)$ for all $i = 1, \ldots, n$ and all $k = 1, \ldots, l$.

Unfortunately, as it turned out, this characterization is too strong, meaning that there are programs with a minimal model that does not satisfy this condition.

Example 6.8

$$\begin{aligned} a \vee b &\leftarrow \\ a &\leftarrow b \\ b &\leftarrow a \end{aligned}$$

This program has only one minimal model $\{a, b\}$, so according to the condition above, the first clause cannot be used since both atoms in the head are true. With the remaining two clauses we cannot have a level mapping satisfying the given condition since we must have $l(a) > l(b)$ and $l(b) > l(a)$ which is not possible.

A first idea to fix the problem was to modify the condition to let all atoms in the head of the specific clause for A either to be false or of level strictly greater than A. But this condition is too weak.

Example 6.9

$$\begin{aligned} a \vee b &\leftarrow \\ a \vee c &\leftarrow \end{aligned}$$

This program has two minimal models: $\{a\}$ and $\{b, c\}$. But $\{a, b\}$ is also a model and with $l(a) = 0$ and $l(b) = 1$ we have a mapping satisfying the modified condition, even though $\{a, b\}$ is not minimal.

The next attempt contains additionally a further clause for each $A \in M$ with A being the only true atom in the head but without any restrictions regarding the level.

Example 6.10 We show with the following program Π that this does not work either.

$$\begin{array}{rcl} p(0) & \leftarrow & \\ p(s(X)) \vee p(X) & \leftarrow & \\ p(X) & \leftarrow & p(s(s(X))) \end{array}$$

A minimal model is e.g. $\{p(s^{2n}(0)) \mid n \geq 0\}$ with $l(p(s^n(0))) = 0$. However, B_Π is also a model and, by $l(p(s^n(0))) = n$, satisfies this further modified condition which is therefore also too weak.

Since the levels for the minimal model are smaller than for the Herbrand basis we finally try to allow instead only minimal mappings which is natural in so far that the assignments given in the proofs of all characterizations presented up to now are minimal. Then the Herbrand basis cannot be a model satisfying the condition in the example above. But in the following we also present a counterexample to this idea.

Example 6.11

$$\begin{array}{rcl} b \vee c & \leftarrow & \\ c & \leftarrow & b \\ d & \leftarrow & c \\ e & \leftarrow & \\ f & \leftarrow & e \\ d & \leftarrow & f \end{array}$$

Given that program Π, we have a model $\{b, c, d, e, f\}$ with $l(b) = 0$, $l(c) = 1$, $l(d) = 1$, $l(e) = 0$, and $l(f) = 1$ and there is no model with a mapping such that all values are smaller or equal with one value being strictly smaller. However, we have a minimal model $\{c, d, e, f\}$ with $l_1(b) = 0$, $l_1(c) = 0$, $l_1(d) = 2$, $l_1(e) = 0$, and $l_1(f) = 1$ even though $l(c) > l_1(c)$ and $l(d) < l_1(d)$ and the mapping is not smaller.

Thus, the best possible result we may get is the following.

Corollary 6.12 Let Π be a definite disjunctive program. If there exists a total level mapping $l : B_\Pi \to \alpha$ such that for each $A \in M$ exists a clause $A \vee H_1 \vee \cdots \vee H_l \leftarrow A_1, \ldots, A_n$ in $\mathsf{ground}(\Pi)$ with $A_i \in M$, $H_k \notin M$ or $l(H_k) > l(A)$, and $l(A) > l(A_i)$, for all $i = 1, \ldots, n$ and all $k = 1, \ldots, l$, then M is a minimal model of Π.

This is of course no equivalent characterization. It just states that minimal models induce a level mapping which does not help us to simplify a characterization involving minimal models. The same problem occurs for disjunctive stable models ([26]), the extension of minimal model semantics to non-definite programs. Summing up, this also shows that level mapping characterizations are not easy to extend to disjunctive programs in general.

7 Conclusions and Further Work

We have characterized three of the extensions of the well-founded semantics to disjunctive logic programs. It has been revealed that these characterizations and their proofs are non-trivial and in the previous section we have seen that all the approaches share a common derivability for true disjunctions. The conditions for deriving negative information however vary a lot. Some parts of the characterizations are common extensions of conditions presented for the well-founded semantics to disjunctive programs. Others are completely different and cover a specific deduction mechanism occurring only in one of the semantics.

One idea for continuing the work is to find a characterization which defines the well-founded semantics for disjunctive logic programs in a systematic way. For figuring out which of the various deduction mechanisms to take into account and which not, it is important to study also the other approaches introduced for extending the well-founded semantics to disjunctive logic programs. Thus further studies should include the well-founded disjunctive semantics WFDS ([35]) and the well-founded semantics with disjunction WFS_d ([1]). This also refers to the stationary semantics ([25]) and the static semantics ([27]) both introduced by Przymusinski where the latter is closely related to D-WFS. We may also consider the semantics WF 3 ([5]) which was presented by Baral, Lobo, and Minker as an extension to GDWFS. In all three characterizations there are details which might be solvable in a more appropriate way regarding the characterization itself. Further studies may reveal whether this is possible, which particularly includes the problems we obtained when dealing with the minimal model semantics and thus also with disjunctive stable models.

All our considered semantics interpret disjunctions exclusively whenever possible. There is also a large amount of semantics using inclusive disjunctions (see [10]) and some of them are also just particular slight modifications of the approaches we studied, like the weak well-founded semantics ([29]) for SWFS or the weak disjunctive well-founded semantics ([6]) in case of D-WFS. It is reasonable to expect that characterizations for these variants are similar to the ones we presented which together with other approaches for inclusive semantics also might result in a deeper understanding of the desired deduction mechanisms of a well-founded semantics for logic programs with disjunctions.

References

[1] J. Alcântara, C. V. Damásio, and L. M. Pereira. A well-founded semantics with disjunction. In G. G. M. Gabbrielli, editor, *Proceedings of the 21st Intl.Conf. on Logic Programming (ICLP '05)*, LNCS 3668, pages 341–355. Springer, 2005.

[2] J. J. Alferes and L. M. Pereira. *Reasoning with Logic Programming*, volume 1111. Springer, 1996.

[3] C. Baral, J. Lobo, and J. Minker. Generalized disjunctive well-founded semantics for logic programs. In M. Z. Z. W. Ras and M. Emrich, editors, *Proceedings of the Fifth International Symposium on Methodologies for Intelligent Systems*, ACM SIGMOD-SIGACT-SIGART, North Holland, pages 465–473, 1989.

[4] C. Baral, J. Lobo, and J. Minker. Generalized well-founded semantics for logic programs. In M. E. Stickel, editor, *10th International Conference on Automated Deduction, LNAI 449, subseries LNCS*, pages 102–116. J. Siekmann, 1990.

[5] C. Baral, J. Lobo, and J. Minker. Wf 3 : A semantics for negation in normal disjunctive logic programs. In Z. W. Ras and M. Zemankova, editors, *Methodologies for Intelligent Systems, LNAI 542*, pages 459–468. Springer, 1991.

[6] S. Brass and J. Dix. Disjunctive semantics based upon partial and bottom-up evaluation. In *Proceedings of the 12th International Logic Programming Conference, Tokyo*, pages 199–213. MIT Press, 1995.

[7] S. Brass and J. Dix. Characterizations of the disjunctive well-founded semantics: Confluent calculi and iterated GCWA. *Journal of Automated Reasoning*, 20(1):143–165, 1998.

[8] J. Dix. A classification theory of semantics of normal logic programs: I. strong properties. *Fundamenta Informaticae*, 22(3):227–255, 1995.

[9] J. Dix. A classification theory of semantics of normal logic programs: II. weak properties. *Fundamenta Informaticae*, 22(3):257–288, 1995.

[10] J. Dix and M. Müller. An axiomatic approach to semantics of disjunctive programs. In P. van Hentenryck, editor, *Logic Programming, Proc. of the 11th International Conference*, pages 303–320. MIT Press, 1994.

[11] J. Dix and F. Stolzenburg. Computation of non-ground disjunctive well-founded semantics with constraint logic programming. In J. Dix, L. M. Pereira, and T. C. Przymusinski, editors, *Selected Papers of the Workshop on Non-Monotonic Extensions of Logic Programming in Conjunction with Joint International Conference and Symposium on Logic Programming 1996*, pages 202–224. Springer, Berlin, Heidelberg, New York, 1997.

[12] F. Fages. Consistency of Clark's completion and existence of stable models. *Journal of Methods of Logic in Computer Science*, 1:51–60, 1994.

[13] M. Fitting. Fixpoint semantics for logic programming — A survey. *Theoretical Computer Science*, 278(1–2):25–51, 2002.

[14] M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. In R. A. Kowalski and K. A. Bowen, editors, *Logic Programming. Proceedings of the 5th International Conference and Symposium on Logic Programming*, pages 1070–1080. MIT Press, 1988.

[15] P. Hitzler. Towards a systematic account of different semantics for logic programs. *Journal of Logic and Computation*, 15(3):391–404, 2005.

[16] P. Hitzler and S. Schwarz. Level mapping characterizations of selector-generated models for logic programs. In A. Wolf, T. W. Frühwirth, and M. Meister, editors, *19th Workshop on (Constraint) Logic Programming, Ulm, Germany, February 2005*, volume 2005-01 of *Ulmer Informatik-Berichte*, pages 65–75. Universität Ulm, 2005.

[17] P. Hitzler and M. Wendt. A uniform approach to logic rogramming semantics. *Theory and Practice of Logic Programming*, 5(1–2):123–159, 2005.

[18] M. Knorr. Level mapping characterizations of quantitative and disjunctive logic programs. bachelors thesis, Technische Universität Dresden, 2003.

[19] V. Lifschitz. Answer set programming and plan generation. *Journal of Artificial Intelligence*, 138:39–54, 2002.

[20] J. W. Lloyd. *Foundations of Logic Programming.* Springer, Berlin, 1988.

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

[22] J. Minker. On indefinite databases and the closed world assumption. In *Lecture Notes in Computer Science 138*, pages 292–308. Springer, Berlin, 1982.

[23] J. Minker and A. Rajasekar. A fixpoint semantics for disjunctive logic programs. *Journal of Logic Programming*, 9(1):45–74, 1990.

[24] J. Minker and D. Seipel. Disjunctive logic programming: A survey and assessment. In *Essays in Honour of Robert A. Kowalski, Part I, LNAI 2407*. Springer, 2002.

[25] T. Przymusinski. Stationary Semantics for Normal and Disjunctive Logic Programs. In C. Delobel, M. Kifer, and Y. Masunaga, editors, *DOOD '91, Proceedings of the 2nd International Conference*. Springer, LNCS 566, 1991.

[26] T. C. Przymusinski. Stable semantics for disjunctive programs. *New Generation Computing*, 9(3–4):401–424, 1991.

[27] T. C. Przymusinski. Static semantics for normal and disjunctive logic programs. *Annals of Mathematics and Artificial Intelligence*, 14(2-4):323–357, 1995.

[28] K. A. Ross. A procedural semantics for well founded negation in logic programs. In *Proceedings of the eighth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*, pages 22–33, New York, NY, USA, 1989. ACM Press.

[29] K. A. Ross. The well founded semantics for disjunctive logic programs. In W. Kim, J.-M. Nicolas, and S. Nishio, editors, *First International Conference on Deductive and Object Oriented Databases*, pages 385–402. Elsevier Science Publishers B.V. (North-Holland), 1990.

[30] S. Schwarz. *Selektor-erzeugte Modelle verallgemeinerter logischer Programme*. PhD thesis, Universität Leipzig, 2004.

[31] V. Stoltenberg-Hansen, I. Lindström, and E. R. Griffor. *Mathematical Theory of Domains*. Cambridge University Press, 1994.

[32] A. Tarski. Lattice-theoretic fixpoint theorem and its applications. *Pacific Journal of Mathematics*, pages 285–309, 1955.

[33] A. van Gelder. The alternating fixpoint of logic programs with negation. In *Eighth ACM SIGACT-SIGMOD-SIGART Symposium on Principles*

of Database Systems, Philadelphia, Pennsylvania, pages 1–10. ACM Press, 1989.

[34] A. van Gelder, K. A. Ross, and J. S. Schlipf. The well-founded semantics for general logic programs. *Journal of the ACM*, 38(3):620–650, 1991.

[35] K. Wang. Argumentation-based abduction in disjunctive logic programming. *Journal of Logic Programming*, 45(1-3):105–141, 2000.

[36] K. Wang. A comparative study of well-founded semantics for disjunctive logic programs. In T. Eiter, W. Faber, and M. Truszczynski, editors, *Lecture Notes in Artificial Intelligence*, volume 2173, pages 133–146. Springer, 2001.

[37] A. Yahya and L. Henschen. Deduction in non-horn databases. *Journal Automated Reasoning*, 1(2):141–160, 1985.