

 Open access • Book Chapter • DOI:10.1007/11779148_19

Well-Founded semantics for boolean grammars — Source link

Vassilis Kountouriotis, Christos Nomikos, Panos Rondogiannis

Institutions: National and Kapodistrian University of Athens, University of Ioannina

Published on: 26 Jun 2006 - Developments in Language Theory

Topics: Boolean grammar, Context-sensitive grammar, Tree-adjoining grammar, L-attributed grammar and Indexed grammar

Related papers:

- [Time-bounded controlled bidirectional grammars](#)
- [Boolean grammars](#)
- [Two Level Grammars: CF-Grammars with Equation Schemes](#)
- [Generalized P-Systems](#)
- [Expressive power of LL\(k\) Boolean grammars](#)

Share this paper:    

View more about this paper here: <https://typeset.io/papers/well-founded-semantics-for-boolean-grammars-56v4b1cfxq>



Well-founded semantics for Boolean grammars[☆]

Vassilis Kountouriotis^a, Christos Nomikos^b, Panos Rondogiannis^{a,*}

^a Department of Informatics & Telecommunications, University of Athens, Panepistimiopolis, 157 84 Athens, Greece

^b Department of Computer Science, University of Ioannina, P.O. Box 1186, 45 110 Ioannina, Greece

ARTICLE INFO

Article history:

Received 1 August 2007

Revised 27 May 2009

Available online 9 June 2009

ABSTRACT

Boolean grammars [A. Okhotin, Boolean grammars, Information and Computation 194 (1) (2004) 19–48] are a promising extension of context-free grammars that supports conjunction and negation in rule bodies. In this paper, we give a novel semantics for Boolean grammars which applies to *all* such grammars, independently of their syntax. The key idea of our proposal comes from the area of *negation in logic programming*, and in particular from the so-called *well-founded semantics* which is widely accepted in this area to be the “correct” approach to negation. We show that for every Boolean grammar there exists a distinguished (three-valued) interpretation of the non-terminal symbols, which satisfies all the rules of the grammar and at the same time is the least fixed-point of an operator associated with the grammar. Then, we demonstrate that every Boolean grammar can be transformed into an equivalent (under the new semantics) grammar in normal form. Based on this normal form, we propose an $\mathcal{O}(n^3)$ algorithm for parsing that applies to any such normalized Boolean grammar. In summary, the main contribution of this paper is to provide a semantics which applies to *all* Boolean grammars while at the same time retaining the complexity of parsing associated with this type of grammars.

© 2009 Elsevier Inc. All rights reserved.

1. Introduction

Boolean grammars constitute a new and promising formalism, proposed by Okhotin in [8], which extends the class of conjunctive grammars introduced by the same author in [7]. The basic idea behind this new formalism is to augment context-free rules by allowing intersection and negation to appear in their right-hand sides. It is immediately obvious that the class of languages that can be produced by Boolean grammars is a proper superset of the class of context-free languages.

Despite their syntactical simplicity, Boolean grammars appear to be non-trivial from a semantical point of view. As we are going to see in the next section, the existing approaches for assigning meaning to Boolean grammars suffer from certain shortcomings (one of which is that they do not give a meaning to all such grammars).

In this paper, we propose a new semantics (the *well-founded semantics*) which applies to all Boolean grammars. More specifically, we demonstrate that for every Boolean grammar there exists a distinguished (*three-valued*, see below) interpretation of the non-terminal symbols, which satisfies all the rules of the grammar. This interpretation is the unique least fixed-point of an appropriate operator associated with the grammar. The language assigned by this interpretation to the start symbol of the grammar, can be taken as the intended meaning of the grammar.

[☆] This work is supported by the Ο3ΕΔ 330 research project, implemented within the framework of the “Reinforcement Programme of Human Research Manpower” (ΠΕΝΕΔ) and co-financed by National and Community Funds (75% from E.U.-European Social Fund and 25% from the Greek Ministry of Development—General Secretariat of Research and Technology and from the private sector).

* Corresponding author.

E-mail addresses: bk@di.uoa.gr (V. Kountouriotis), cnomikos@cs.uoi.gr (C. Nomikos), prondo@di.uoa.gr (P. Rondogiannis).

Our ideas originate from an important area of research in the theory of logic programming, that has been very active for more than two decades (references such as [1,9] provide nice surveys). In this area, there is nowadays an almost unanimous agreement that if we seek a unique model of a logic program with negation, then we have to search for a three-valued one. In other words, classical two-valued logic is not sufficient in order to assign a proper meaning to arbitrary logic programs with negation. Actually, it can be demonstrated that every logic program with negation has a distinguished three-valued model, which is usually termed the *well-founded model* [13].

We follow the same ideas here: we consider three-valued languages, namely languages in which the membership of strings may be characterized as true, false, or *unknown*. As we will see, this simple extension solves the semantic problems associated with negation in Boolean grammars. Actually we show that this extension to three values is in some sense necessary: we prove that the problem of whether a Boolean grammar defines under the well-founded semantics a classical (that is, two-valued) language, is undecidable. We then proceed by demonstrating that under this new semantics, every Boolean grammar has an equivalent grammar in normal form (similar to that of [8]). Finally, we show that for every such normalized grammar, there is an $\mathcal{O}(n^3)$ parsing algorithm under our new semantics. Our results indicate that there may be other fruitful connections between formal language theory and the theory of logic programming.

The rest of the paper is organized as follows: Section 2 presents the basic issues regarding Boolean grammars and discusses the existing approaches to their semantics. In Section 3 the notion of a *three-valued formal language* is proposed and the basic tools that will be used in our semantic investigations are developed. In Section 4 the well-founded semantics of Boolean grammars is defined and its basic properties are demonstrated. In Section 5 a normal form for Boolean grammars is introduced based on the well-founded semantics. In Section 6 a parsing algorithm for Boolean grammars is derived based on the normal form introduced in Section 5. Finally, Section 7 concludes the paper giving pointers to future work.

2. Why an alternative semantics for Boolean grammars?

In [8] Okhotin proposed the class of Boolean grammars. Formally:

Definition 1 [8]. A Boolean grammar is a quadruple $G = (\Sigma, N, P, S)$, where Σ and N are disjoint finite non-empty sets of terminal and non-terminal symbols, respectively, P is a finite set of rules, each of the form

$$A \rightarrow \alpha_1 \& \cdots \& \alpha_m \& \neg \beta_1 \& \cdots \& \neg \beta_n \quad (m + n \geq 1, \alpha_i, \beta_j \in (\Sigma \cup N)^*),$$

and $S \in N$ is the start symbol of the grammar. We will call the non-terminal A the *head* of the rule, the α_i 's *positive conjuncts* and the $\neg \beta_j$'s *negative ones*.

We will often use the short notation $A \rightarrow \varphi_1 | \cdots | \varphi_k$ to represent k rules of the form $A \rightarrow \varphi_i$.

To illustrate the use of Boolean grammars, consider the following example from [8] (presented here in a slightly modified form):

Example 2. Let $\Sigma = \{a, b\}$. We define:

$$\begin{aligned} S &\rightarrow \neg(AB) \& \neg(BA) \& \neg A \& \neg B \\ A &\rightarrow a|CAC \\ B &\rightarrow b|CBC \\ C &\rightarrow a|b \end{aligned}$$

The above grammar defines the language $L_{ww} = \{ww | w \in \{a, b\}^*\}$, which is well-known to be non-context-free. This can be justified as follows: first, it is easy to see that the language $L(A)$ (respectively, the language $L(B)$) produced by the non-terminal A (respectively, the non-terminal B) contains the strings of odd length in which the symbol in the middle is a (respectively, b). Consider now any string y of length $2n$ for some n , that is not in L_{ww} . This implies that there exists some i , $1 \leq i \leq n$, such that the i th symbol of y is different from the $(n + i)$ th symbol of y . Suppose that the i th symbol of y is a and the $(n + i)$ th symbol of y is b (the other case is completely symmetric). Then, $y = y_a y_b$, where $|y_a| = 2i - 1$, $|y_b| = 2(n - i) + 1$, $y_a \in L(A)$ and $y_b \in L(B)$ (since the i th and the $(n + i)$ th symbol of y are the symbols in the middle of y_a and y_b , respectively).

Therefore, a string that is not in L_{ww} , belongs to $L(A) \cup L(B)$ if it has odd length, and belongs to $L(A) \circ L(B) \cup L(B) \circ L(A)$ if it has even length. Using De Morgan's law, we obtain the first rule, which defines the language produced by the grammar.

Okhotin proposed two semantics intended to capture the meaning of Boolean grammars. In this section, we demonstrate some deficiencies of these two approaches, which led us to the definition of the well-founded semantics. Both semantics proposed in [8] are defined using a system of equations, which is obtained from the given grammar as follows: consider a Boolean grammar $G = (\Sigma, N, P, S)$, where $N = \{X_1, X_2, \dots, X_k\}$. The equation for the non-terminal X_i is

$$X_i = \bigcup_{X_i \rightarrow \alpha_1 \& \cdots \& \alpha_m \& \neg \beta_1 \& \cdots \& \neg \beta_n \in P} \left(\bigcap_{j=1}^m \alpha_j \cap \bigcap_{j=1}^n \overline{\beta_j} \right)$$

We denote the formula in the right-hand side of the above rule (which in general involves the non-terminal symbols in N) by $\phi_i(X_1, \dots, X_k)$. An interpretation I of G (i.e., an assignment of a language from Σ to every non-terminal symbol in N) is said to be a solution of the system of equations

$$\begin{aligned} X_1 &= \phi_1(X_1, \dots, X_k) \\ &\dots \\ X_k &= \phi_k(X_1, \dots, X_k) \end{aligned}$$

if for every i , $1 \leq i \leq k$, it holds $I(X_i) = \widehat{I}(\phi_i(X_1, \dots, X_k))$, where \widehat{I} is the extension of I to expressions that may appear in the right-hand sides of equations, which can be obtained in a straightforward manner (for more details see Definition 7 of Section 3, where the extension of three-valued interpretations is defined).

In the first approach proposed in [8], the semantics of a Boolean grammar is defined only in the case that the corresponding system of equations has a *unique solution*. This is a restrictive choice: actually many interesting grammars do not correspond to systems of equations having a unique solution. For example, even simple context-free grammars (such as for example the grammar with a single rule $S \rightarrow S$), may give systems of equations which have infinitely many solutions. For such grammars, it seems that the desired property is a form of *minimality* rather than uniqueness of the solution.

Apart from its limited applicability, the unique solution semantics also exhibits a kind of instability. For example, let $\Sigma = \{0, 1\}$ and consider the Boolean grammar consisting of the two rules $A \rightarrow \neg A \& \neg B$ and $B \rightarrow 0 \& 1$. The corresponding system of equations has no solution and therefore the unique solution semantics for this grammar is not defined. Suppose that we augment the above grammar with the rule $B \rightarrow B$. Seen from a constructive point of view, the new rule does not offer to the grammar any additional information. It is reasonable to expect that such a rule would not change the semantics of the grammar. However, the augmented grammar has unique solution semantics, namely $(A, B) = (\emptyset, \Sigma^*)$. On the other hand, suppose that we augment the initial grammar with the rule $A \rightarrow A$. Then, the unique solution semantics is also defined, but now the solution is $(A, B) = (\Sigma^*, \emptyset)$. Consequently, by adding to an initially meaningless grammar two different information-free rules, we get two grammars defining complementary languages. To put it another way, three grammars that look equivalent, have completely different semantics.

Let us now turn to the second approach proposed in [8], namely the *naturally reachable solution semantics* defined as follows (for convenience, given an interpretation I of G and a finite language M we denote by $I^{\cap M}$ the interpretation with $I^{\cap M}(A) = I(A) \cap M$ for every $A \in N$):

Definition 3. Let $X_1 = \phi_1(X_1, \dots, X_k), \dots, X_k = \phi_k(X_1, \dots, X_k)$ be a system of equations which corresponds to a Boolean grammar $G = (\Sigma, N, P, S)$, with $N = \{X_1, \dots, X_k\}$. An interpretation I is called a *naturally reachable solution* of the system if for every finite language M closed under substringing and for every string $u \notin M$ such that all proper substrings of u are in M , every sequence of interpretations of the form: $I^{(0)}, I^{(1)}, \dots, I^{(i)}, \dots$ which satisfies the properties

- $I^{(0)} = I^{\cap M}$
- $I^{(i+1)} \neq I^{(i)}$ and
- there exists some j such that $I^{(i+1)}(X_j) = \widehat{I^{(i)}}(\phi_j(X_1, \dots, X_k)) \cap (M \cup \{u\})$ and $I^{(i+1)}(X_\ell) = I^{(i)}(X_\ell)$ for all $\ell \neq j$

converges to $I^{\cap(M \cup \{u\})}$ in finitely many steps.

Contrary to the unique solution semantics, the naturally reachable solution semantics generalizes the semantics of context-free and conjunctive languages (see [8] [Theorem 3]). However, when negation appears, there are cases that this approach does not behave in an expected manner. Consider for example the Boolean grammar with rules:

$$A \rightarrow \neg B, \quad B \rightarrow C \& \neg D, \quad C \rightarrow D, \quad D \rightarrow A$$

This grammar has the naturally reachable solution $(A, B, C, D) = (\Sigma^*, \emptyset, \Sigma^*, \Sigma^*)$. It is reasonable to expect that composing two rules would not affect the semantics of the grammar. For example, in context-free grammars such a composition is a natural transformation rule that simply allows to perform two steps of the production in a single step. However, if we add $C \rightarrow A$ to the above set of rules, then the naturally reachable solution semantics of the resulting grammar is not defined. On the other hand, the technique we will define shortly, does not suffer from this shortcoming.

Furthermore, there exist grammars for which the naturally reachable solution semantics is undefined, although they may have a clear intuitive meaning. For example, let $\Sigma = \{a\}$ and suppose that the grammar contains the following rules:

$$A \rightarrow \neg B|D, \quad B \rightarrow \neg C|D, \quad C \rightarrow \neg A|D, \quad D \rightarrow aD|\epsilon$$

The semantics of this grammar is clearly $(A, B, C, D) = (\Sigma^*, \Sigma^*, \Sigma^*, \Sigma^*)$, and actually this is what the well-founded semantics will produce. On the other hand the naturally reachable solution semantics is undefined.

The problem of giving semantics to recursive formalisms in the presence of negation has been extensively studied in the context of logic programming. Actually, the unique solution semantics can be paralleled with one of the early attempts to give semantics to logic programs with negation, namely what is now called *the Clark's completion semantics* (which actually presents similar shortcomings as the unique solution approach). On the other hand, the naturally reachable solution can be thought of as a first approximation to the procedure of constructing the intended minimal model of a logic program with negation (see also Theorem 28 that will follow). Since the most broadly accepted semantic approach for logic programs with

negation is the well-founded semantics, in this paper we investigate the possibility of applying such an approach to Boolean grammars.

At this point we should also mention two other recent works on the semantics of Boolean grammars, namely the *stratified semantics* [14] and the *locally stratified* one [5,6]. Both of these approaches also have their roots in the theory of non-monotonic logic programming. However, these two semantics differ from the well-founded one in the sense that they aim to identify interesting (syntactic) subclasses of Boolean grammars that have a well-defined meaning (while the present approach aims at providing a formal framework for the whole class of Boolean grammars).

3. Interpretations and models for Boolean grammars

In this section, we initiate our study of the semantics of Boolean grammars. We begin by defining the notions of *interpretation* and *model* for Boolean grammars, two concepts that have been borrowed from mathematical logic (see for example [3]). In context-free grammars, an interpretation is a function that assigns to each non-terminal symbol of the grammar a set of strings over the set of terminal symbols of the grammar. An interpretation of a context-free grammar is a model of the grammar if it satisfies all the rules of the grammar. The usual semantics of context-free grammars dictate that every such grammar has a minimum model, which is taken to be as its intended meaning.

When one considers Boolean grammars, the situation becomes much more complicated. For example, a grammar with the unique rule $S \rightarrow \neg S$ appears to be meaningless. More generally, in many cases where negation is used in a circular way, the corresponding grammar looks problematic. These difficulties arise because we are trying to find *classical* models of Boolean grammars, which are based on classical two-valued logic. If however we shift to *three-valued* models, every Boolean grammar has a well-defined meaning. We need of course to redefine many notions, starting even from the notion of a language:

Definition 4. Let Σ be a finite non-empty set of symbols. Then, a (three-valued) language over Σ is a function from Σ^* to the set $\{0, \frac{1}{2}, 1\}$.

Intuitively, given a three-valued language L and a string w over the alphabet of L , there are three cases: either $w \in L$ (i.e., $L(w) = 1$), or $w \notin L$ (i.e., $L(w) = 0$), or finally, the membership of w in L is unclear (i.e., $L(w) = \frac{1}{2}$). Given this extended notion of language, it is now possible to interpret the grammar $S \rightarrow \neg S$: its meaning is the language which assigns to every string the value $\frac{1}{2}$.

The following definition, which generalizes the familiar notion of concatenation of languages, will be used in the rest of the paper:

Definition 5. Let Σ be a finite non-empty set of symbols and let L_1, \dots, L_n be (three-valued) languages over Σ . We define the *three-valued concatenation* of the languages L_1, \dots, L_n to be the language L such that for every $w \in \Sigma^*$:

$$L(w) = \max_{\substack{(w_1, \dots, w_n): \\ w = w_1 \dots w_n}} \left(\min_{1 \leq i \leq n} L_i(w_i) \right)$$

The concatenation of L_1, \dots, L_n will be denoted by $L_1 \circ \dots \circ L_n$.

The above definition can be explained as follows:

- A string belongs to $L_1 \circ \dots \circ L_n$ (truth value 1) if it can be partitioned into n parts so that for every $i \leq n$, the i 'th part belongs to L_i .
- A string is excluded from the concatenation (truth value 0) if in every partition, there exists some i such that the i 'th part is excluded from the language L_i .
- The membership of a string w is undefined in the concatenation (truth value $\frac{1}{2}$) if there exists a partition of w such that no part is excluded from the corresponding language, and there does not exist a partition of w such that every part belongs to the corresponding language.

It can be easily checked that when the languages involved are total (i.e., with no $\frac{1}{2}$ values assigned to strings) then the above definition reduces to the familiar definition of concatenation.

We can now define the notion of *interpretation* of a given Boolean grammar:

Definition 6. An *interpretation* I of a Boolean grammar $G = (\Sigma, N, P, S)$ is a function $I : N \rightarrow (\Sigma^* \rightarrow \{0, \frac{1}{2}, 1\})$.

An interpretation I can be recursively extended to apply to expressions that appear in the right-hand sides of Boolean grammar rules:

Definition 7. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar and let I be an interpretation of G . Then, the extension \hat{I} of I is defined as follows:

- For every $w \in \Sigma^*$, it is $\widehat{I}(\epsilon)(w) = 1$ if $w = \epsilon$, and $\widehat{I}(\epsilon)(w) = 0$ otherwise.
- Let $A \in N$. Then, for every $w \in \Sigma^*$, it is $\widehat{I}(A)(w) = I(A)(w)$.
- Let $a \in \Sigma$. Then, for every $w \in \Sigma^*$, it is $\widehat{I}(a)(w) = 1$ if $w = a$, and $\widehat{I}(a)(w) = 0$ otherwise.
- Let $\alpha = \alpha_1 \cdots \alpha_n$, $n \geq 2$, $\alpha_i \in \Sigma \cup N$. Then, for every $w \in \Sigma^*$, it is $\widehat{I}(\alpha)(w) = (\widehat{I}(\alpha_1) \circ \cdots \circ \widehat{I}(\alpha_n))(w)$.
- Let $\alpha \in (\Sigma \cup N)^*$. Then, for every $w \in \Sigma^*$, it is $\widehat{I}(\neg\alpha)(w) = 1 - \widehat{I}(\alpha)(w)$.
- Let l_1, \dots, l_n be conjuncts. Then, for every $w \in \Sigma^*$, it is $\widehat{I}(l_1 \& \cdots \& l_n)(w) = \min\{\widehat{I}(l_1)(w), \dots, \widehat{I}(l_n)(w)\}$.

We are now in a position to define the notion of a model of a Boolean grammar:

Definition 8. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar and I an interpretation of G . Then, I is a *model* of G if for every rule $A \rightarrow l_1 \& \cdots \& l_n$ in P and for every $w \in \Sigma^*$, it is $I(A)(w) \geq \widehat{I}(l_1 \& \cdots \& l_n)(w)$.

Certain explanations regarding the notion of model are needed, since this concept is not broadly used in formal language theory – despite its fundamental applicability in mathematical logic. A model of a set of formulas in logic, is an interpretation that satisfies all the formulas in the set. In the context of Boolean grammars, each rule can be thought of as a formula which states that the membership value of a string in the language that corresponds to the head of the rule, is greater than or equal to the membership value of the string in the language that corresponds to the body of the rule. This idea restricted to total languages states that, for every rule, the language that corresponds to the head of a rule is a superset of the language that corresponds to the body. Clearly, a model of a grammar does not necessarily capture the meaning of a grammar (for example, an interpretation that assigns Σ^* to every non-terminal of a grammar, is a model of the grammar). However, the interpretation that captures the intended meaning of a grammar, *has* to be a model of the grammar. In other words, the first basic property that an interpretation has to satisfy in order to be eligible as a candidate for the correct meaning of a Boolean grammar, is to be a model of the grammar.¹

In the definition of the well-founded model, two orderings on interpretations play a crucial role (see [9] for the corresponding ordering in the case of logic programming). Given two interpretations, the first ordering (usually called the *standard ordering*) compares their *degree of truth*:

Definition 9. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar and I, J be two interpretations of G . Then, we write $I \preceq J$ if for all $A \in N$ and for all $w \in \Sigma^*$, $I(A)(w) \leq J(A)(w)$.

The following lemma is easy to establish:

Lemma 10. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar and I, J be two interpretations of G such that $I \preceq J$. Then, for all $\alpha \in (\Sigma \cup N)^*$ and for all $w \in \Sigma^*$, $\widehat{I}(\alpha)(w) \leq \widehat{J}(\alpha)(w)$.

Proof. The statement is obvious when $\alpha = \epsilon$ or when $\alpha \in \Sigma \cup N$. For $\alpha = \alpha_1 \cdots \alpha_n$, $n \geq 2$, $\alpha_i \in \Sigma \cup N$, it is:

$$\begin{aligned}
 \widehat{I}(\alpha)(w) &= (\widehat{I}(\alpha_1) \circ \cdots \circ \widehat{I}(\alpha_n))(w) \\
 &= \max_{\substack{(w_1, \dots, w_n): \\ w = w_1 \cdots w_n}} (\min_{1 \leq i \leq n} \widehat{I}(\alpha_i)(w_i)) \\
 &\leq \max_{\substack{(w_1, \dots, w_n): \\ w = w_1 \cdots w_n}} (\min_{1 \leq i \leq n} \widehat{J}(\alpha_i)(w_i)) \\
 &= (\widehat{J}(\alpha_1) \circ \cdots \circ \widehat{J}(\alpha_n))(w) \\
 &= \widehat{J}(\alpha)(w)
 \end{aligned}$$

This completes the proof of the lemma. \square

Among the interpretations of a given Boolean grammar, there is one which is the least with respect to the \preceq ordering and is denoted by \perp . That is, for all A and all w , $\perp(A)(w) = 0$.

The second ordering (usually called the *Fitting ordering*) compares the *degree of information* of two interpretations. We first need to define the corresponding numerical ordering:

Definition 11. Let $v_1, v_2 \in \{0, \frac{1}{2}, 1\}$. We write $v_1 \leq_F v_2$ if and only if either $v_1 = v_2$ or $v_1 = \frac{1}{2}$.

Definition 12. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar and I, J be two interpretations of G . Then, we write $I \leq_F J$ if for all $A \in N$ and for all $w \in \Sigma^*$, $I(A)(w) \leq_F J(A)(w)$.

We now establish a lemma regarding \leq_F which is similar to Lemma 10 for \preceq :

¹ One could avoid the use of models by first transforming a Boolean grammar into a set of equations (see [8] or the corresponding definition in Section 2), and then looking for a solution to this set of equations. We prefer to follow the model-based approach, which is closer to the logical background of Boolean grammars.

Lemma 13. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar and I, J be two interpretations of G such that $I \leq_F J$. Then, for any conjunct l (either positive or negative) and for any $w \in \Sigma^*$, $\widehat{I}(l)(w) \leq_F \widehat{J}(l)(w)$.

Proof. Assume first that l is a positive conjunct. When $l = \epsilon$ or $l \in \Sigma \cup N$, the result holds trivially. For $l = \alpha_1 \cdots \alpha_n$, $n \geq 2$, $\alpha_i \in \Sigma \cup N$, we distinguish two cases:

Case 1: $\widehat{I}(l)(w) = 0$ or equivalently $(\widehat{I}(\alpha_1) \circ \cdots \circ \widehat{I}(\alpha_n))(w) = 0$. From Definition 5, this means that $\max_{\substack{(w_1, \dots, w_n): \\ w = w_1 \cdots w_n}} (\min_{1 \leq i \leq n} \widehat{I}(\alpha_i)(w_i)) = 0$, or equivalently that for all (w_1, \dots, w_n) such that $w = w_1 \cdots w_n$ there exists α_i such that $\widehat{I}(\alpha_i)(w_i) = 0$. But for every such α_i it is also $\widehat{J}(\alpha_i)(w_i) = 0$, which implies that $\max_{\substack{(w_1, \dots, w_n): \\ w = w_1 \cdots w_n}} (\min_{1 \leq i \leq n} \widehat{J}(\alpha_i)(w_i)) = 0$. Therefore, $(\widehat{J}(\alpha_1) \circ \cdots \circ \widehat{J}(\alpha_n))(w) = 0$ or equivalently $\widehat{J}(l)(w) = 0$.

Case 2: $\widehat{I}(l)(w) = 1$ and therefore $(\widehat{I}(\alpha_1) \circ \cdots \circ \widehat{I}(\alpha_n))(w) = 1$. Therefore, from Definition 5, there exists (w_1, \dots, w_n) with $w = w_1 \cdots w_n$ such that for all α_i it is $\widehat{I}(\alpha_i)(w_i) = 1$. This implies that for all α_i , it is also $\widehat{J}(\alpha_i)(w_i) = 1$, from which it follows that $\widehat{J}(l)(w) = 1$.

When $l = \neg\alpha$ is a negative conjunct the result follows from the fact that $\widehat{I}(\neg\alpha)(w) = 1 - \widehat{I}(\alpha)(w)$. This completes the proof of the lemma. \square

Among the interpretations of a given Boolean grammar, there is one which is the least with respect to the \leq_F ordering and is denoted by \perp_F . That is, for all A and all w , $\perp_F(A)(w) = \frac{1}{2}$.

Given a set U of interpretations, we will write $\text{lub}_{\leq} U$ for the least upper bound of the members of U under the standard ordering. Formally:

$$(\text{lub}_{\leq} U)(A)(w) = \begin{cases} 1, & \text{if there exists } I \in U \text{ such that } I(A)(w) = 1 \\ 0, & \text{if for all } I \in U, I(A)(w) = 0 \\ \frac{1}{2}, & \text{otherwise} \end{cases}$$

The situation changes when one wants to define $\text{lub}_{\leq_F} U$, that is, the least upper bound of the members of U under the Fitting ordering, since this notion cannot in general be defined for an arbitrary set of interpretations U . However, $\text{lub}_{\leq_F} U$ can be defined if U is a directed set of interpretations, i.e., if for every $I_1, I_2 \in U$ there exists $J \in U$ such that $I_1 \leq_F J$ and $I_2 \leq_F J$. In this case $\text{lub}_{\leq_F} U$ is defined as follows:

$$(\text{lub}_{\leq_F} U)(A)(w) = \begin{cases} 1, & \text{if there exists } I \in U \text{ such that } I(A)(w) = 1 \\ 0, & \text{if there exists } I \in U \text{ such that } I(A)(w) = 0 \\ \frac{1}{2}, & \text{otherwise} \end{cases}$$

Obviously, an increasing sequence $U = I_1 \leq_F I_2 \leq_F \cdots$ of interpretations constitutes a directed set of interpretations, and therefore in this case $\text{lub}_{\leq_F} U$ is well-defined.

4. Well-founded semantics for Boolean grammars

In this section, we define the well-founded semantics of Boolean grammars. The basic idea behind the well-founded semantics is that the intended model of the grammar is constructed in stages that are related to the levels of negation used by the grammar. At each step of this process and for every non-terminal symbol, the values of certain strings are computed and fixed (as either true or false); at each new level, the values of more and more strings become fixed (and this is a monotonic procedure in the sense that values of strings that have been fixed for a given non-terminal in a previous stage, are not altered by the next stages). At the end of all the stages, certain strings for certain non-terminals may have not managed to get the status of either true or false (this will be due to circularities through negation in the grammar). Such strings are classified as unknown (i.e., $\frac{1}{2}$).

Consider the Boolean grammar $G = (\Sigma, N, P, S)$. Then, for any interpretation J of G we define the operator $[\Theta_G]_J : \mathcal{I} \rightarrow \mathcal{I}$ on the set \mathcal{I} of all three-valued interpretations of G . Intuitively, J represents information that we have already derived and is considered stable (and therefore it can be safely used to compute the value of negative conjuncts). More specifically, given $I \in \mathcal{I}$, $A \in N$ and $w \in \Sigma^*$, $[\Theta_G]_J(I)(A)(w)$ is the value that w gets in *one step* when using J in order to evaluate the negative conjuncts in rules defining A in G and I to evaluate the positive ones. More formally:

Definition 14. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar, let \mathcal{I} be the set of all three-valued interpretations of G and let $J \in \mathcal{I}$. The operator $[\Theta_G]_J : \mathcal{I} \rightarrow \mathcal{I}$ is defined as follows. For every $I \in \mathcal{I}$, for all $A \in N$ and for all $w \in \Sigma^*$:

1. $[\Theta_G]_J(I)(A)(w) = 1$, if there exists a rule $A \rightarrow l_1 \& \cdots \& l_r$ in P such that for every positive l_i it is $\widehat{I}(l_i)(w) = 1$, and for every negative l_i it is $\widehat{J}(l_i)(w) = 1$;
2. $[\Theta_G]_J(I)(A)(w) = 0$, if for every rule $A \rightarrow l_1 \& \cdots \& l_r$ in P , either there exists a positive l_i such that $\widehat{I}(l_i)(w) = 0$, or there exists a negative l_i such that $\widehat{J}(l_i)(w) = 0$;
3. $[\Theta_G]_J(I)(A)(w) = \frac{1}{2}$, otherwise.

Some remarks are in order. The operator $[\Theta_G]_J$ is analogous to the ones that have been used in the logic programming domain, but has some important differences from them. More specifically, in [10] two operators are used which produce two sets of atoms corresponding to true and false conclusions of the program, respectively. When applied to arbitrary interpretations, these operators may produce inconsistent sets of atoms. However, it is demonstrated in [10] that these operators when used appropriately, never give rise to inconsistent sets of atoms. In [9], one operator Θ_J is introduced whose definition however is not precise in the sense that it is not truth-functional: given arbitrary interpretations I, J and atom A it is possible that $\Theta_J(I)(A)$ can be assigned both the values 0 and 1. Note however that this problematic case never arises in the construction of the well-founded model. This imprecise definition was also present in the original conference version of our paper [4]. The above functional definition of $[\Theta_G]_J$ remedies this deficiency.

An important fact regarding the operator $[\Theta_G]_J$ is that it is monotonic with respect to the \leq ordering of interpretations:

Lemma 15. *Let $G = (\Sigma, N, P, S)$ be a Boolean grammar and let J be an interpretation of G . Then, the operator $[\Theta_G]_J$ is monotonic with respect to the \leq ordering of interpretations.*

Proof. Let I_1, I_2 be interpretations of G such that $I_1 \leq I_2$ and let $A \in N$ and $w \in \Sigma^*$. We show by a case analysis on the value of $[\Theta_G]_J(I_1)(A)(w)$ that $[\Theta_G]_J(I_1)(A)(w) \leq [\Theta_G]_J(I_2)(A)(w)$. The case $[\Theta_G]_J(I_1)(A)(w) = 0$ is immediate.

Consider now the case $[\Theta_G]_J(I_1)(A)(w) = 1$. Then, from Definition 14, there is a rule $A \rightarrow l_1 \& \dots \& l_r$ in P such that for all positive l_i it is $\widehat{I}_1(l_i)(w) = 1$ and for all negative l_i it is $\widehat{J}(l_i)(w) = 1$. But since $I_1 \leq I_2$, using Lemma 10 we get that $\widehat{I}_2(l_i)(w) = 1$ for all positive l_i , which implies that $[\Theta_G]_J(I_2)(A)(w) = 1$.

Consider now the remaining case $[\Theta_G]_J(I_1)(A)(w) = \frac{1}{2}$ and assume for the sake of contradiction that $[\Theta_G]_J(I_2)(A)(w) = 0$. This implies that for every rule $A \rightarrow l_1 \& \dots \& l_r$ in P , either there exists a positive l_i such that $\widehat{I}_2(l_i)(w) = 0$, or there exists a negative l_i such that $\widehat{J}(l_i)(w) = 0$. But since $I_1 \leq I_2$, using Lemma 10 we get that $[\Theta_G]_J(I_1)(A)(w) = 0$ (contradiction). Therefore, in any case $[\Theta_G]_J(I_1)(A)(w) \leq [\Theta_G]_J(I_2)(A)(w)$. \square

The following definition will be useful in the subsequent discussion:

Definition 16. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar, let I be an interpretation of G and let $w \in \Sigma^*$. We denote by I/w the interpretation defined as follows:

$$(I/w)(A)(u) = \begin{cases} I(A)(u), & \text{if } u \text{ is a substring of } w \\ 0, & \text{otherwise} \end{cases}$$

We now have the following lemmata:

Lemma 17. *Let $G = (\Sigma, N, P, S)$ be a Boolean grammar, w be a string in Σ^* , and $\{I_n\}_{n < \omega}$ be an increasing sequence of interpretations with respect to the ordering \leq (respectively, \leq_F). Then there exists some m such that $(\text{lub}_{\leq} \{I_n\}_{n < \omega})/w = I_m/w$ (respectively, $(\text{lub}_{\leq_F} \{I_n\}_{n < \omega})/w = I_m/w$).*

Proof. We give the proof for \leq ; the proof for \leq_F is similar.

Let $J = \text{lub}_{\leq} \{I_n\}_{n < \omega}$. It is easy to verify that the sequence $\{I_n/w\}_{n < \omega}$ is also increasing with respect to \leq and that $\text{lub}_{\leq} \{I_n/w\}_{n < \omega} = J/w$. Moreover, the set $\{I/w \mid I \text{ is an interpretation of } G\}$ is finite, since N is finite and there is a finite number of substrings of w . The above facts imply that there exists some $m < \omega$ such that $I_i/w \leq I_m/w$ for every $i < \omega$, that is, I_m/w is an upper bound for $\{I_n/w\}_{n < \omega}$. Since J/w is the least upper bound of this sequence, we obtain that $J/w \leq I_m/w$, and since I_m/w belongs to the sequence it holds $I_m/w \leq J/w$. The last two inequalities imply that $J/w = I_m/w$. \square

Lemma 18. *Let $G = (\Sigma, N, P, S)$ be a Boolean grammar and let I_1, I_2, J_1, J_2 be interpretations of G . Let $w \in \Sigma^*$ and assume that $I_1/w = I_2/w$ and $J_1/w = J_2/w$. Then, for every $A \in N$, $[\Theta_G]_{J_1}(I_1)(A)(w) = [\Theta_G]_{J_2}(I_2)(A)(w)$.*

Proof. We perform a case analysis on the value of $[\Theta_G]_{J_1}(I_1)(A)(w)$.

Case 1: $[\Theta_G]_{J_1}(I_1)(A)(w) = 0$. But this is equivalent to saying that for every rule $A \rightarrow l_1 \& \dots \& l_r$ in P , either there exists a positive l_i such that $\widehat{I}_1(l_i)(w) = 0$, or there exists a negative l_i such that $\widehat{J}_1(l_i)(w) = 0$. But using the fact that $I_1/w = I_2/w$ and $J_1/w = J_2/w$, this again is equivalent to the statement that for every rule $A \rightarrow l_1 \& \dots \& l_r$ in P , either there exists a positive l_i such that $\widehat{I}_2(l_i)(w) = 0$, or there exists a negative l_i such that $\widehat{J}_2(l_i)(w) = 0$. Equivalently, $[\Theta_G]_{J_2}(I_2)(A)(w) = 0$.

Case 2: $[\Theta_G]_{J_1}(I_1)(A)(w) = 1$. Entirely analogous to the proof of Case 1. \square

The next definition and theorem demonstrate that in addition, $[\Theta_G]_J$ has a unique least fixed-point:

Definition 19. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar and let J be an interpretation of G . Define:

$$\begin{aligned}
[\Theta_G]_J^{\uparrow 0} &= \perp \\
[\Theta_G]_J^{\uparrow n+1} &= [\Theta_G]_J([\Theta_G]_J^{\uparrow n}) \\
[\Theta_G]_J^{\uparrow \omega} &= \text{lub}_{\leq} \{[\Theta_G]_J^{\uparrow n} \mid n < \omega\}.
\end{aligned}$$

Theorem 20. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar and let J be an interpretation of G . Then, the sequence $\{[\Theta_G]_J^{\uparrow n}\}_{n < \omega}$ is increasing with respect to \leq and $[\Theta_G]_J^{\uparrow \omega}$ is the unique least fixed-point of the operator $[\Theta_G]_J$ with respect to \leq .

Proof. We first show by induction that the sequence $\{[\Theta_G]_J^{\uparrow n}\}_{n < \omega}$ is increasing with respect to \leq . Obviously $[\Theta_G]_J^{\uparrow 0} = \perp \leq [\Theta_G]_J^{\uparrow 1}$. Moreover, assuming that $[\Theta_G]_J^{\uparrow k} \leq [\Theta_G]_J^{\uparrow k+1}$ and using the monotonicity of $[\Theta_G]_J$ with respect to \leq (Lemma 15), we get that $[\Theta_G]_J^{\uparrow k+1} \leq [\Theta_G]_J^{\uparrow k+2}$.

Next we show that $[\Theta_G]_J^{\uparrow \omega}$ is a fixed-point of $[\Theta_G]_J$, i.e., that $[\Theta_G]_J([\Theta_G]_J^{\uparrow \omega}) = [\Theta_G]_J^{\uparrow \omega}$. We first demonstrate that $[\Theta_G]_J^{\uparrow \omega} \leq [\Theta_G]_J([\Theta_G]_J^{\uparrow \omega})$. Since $[\Theta_G]_J^{\uparrow \omega}$ is the least upper bound of the sequence $\{[\Theta_G]_J^{\uparrow n}\}_{n < \omega}$, we have that for every $n \geq 0$, $[\Theta_G]_J^{\uparrow n} \leq [\Theta_G]_J^{\uparrow \omega}$. Using the monotonicity of $[\Theta_G]_J$ with respect to \leq , we get that for every $n \geq 0$, $[\Theta_G]_J^{\uparrow n+1} = [\Theta_G]_J([\Theta_G]_J^{\uparrow n}) \leq [\Theta_G]_J([\Theta_G]_J^{\uparrow \omega})$, or equivalently that $[\Theta_G]_J([\Theta_G]_J^{\uparrow \omega})$ is an upper bound of the sequence $\{[\Theta_G]_J^{\uparrow n+1}\}_{n < \omega}$. Then $[\Theta_G]_J([\Theta_G]_J^{\uparrow \omega})$ is also an upper bound of the sequence $\{[\Theta_G]_J^{\uparrow n}\}_{n < \omega} = \{\perp\} \cup \{[\Theta_G]_J^{\uparrow n+1}\}_{n < \omega}$. But we know that $[\Theta_G]_J^{\uparrow \omega}$ is the least upper bound of this sequence, and therefore $[\Theta_G]_J^{\uparrow \omega} \leq [\Theta_G]_J([\Theta_G]_J^{\uparrow \omega})$.

We now demonstrate that $[\Theta_G]_J([\Theta_G]_J^{\uparrow \omega}) \leq [\Theta_G]_J^{\uparrow \omega}$, or equivalently that for all $A \in N$ and for every $w \in \Sigma^*$, $[\Theta_G]_J([\Theta_G]_J^{\uparrow \omega})(A)(w) \leq [\Theta_G]_J^{\uparrow \omega}(A)(w)$. Consider arbitrary $A \in N$ and $w \in \Sigma^*$. Since the sequence $\{[\Theta_G]_J^{\uparrow n}\}_{n < \omega}$ is increasing with respect to \leq from Lemma 17 there exists some $m < \omega$ such that $[\Theta_G]_J^{\uparrow \omega}/w = [\Theta_G]_J^{\uparrow m}/w$. From Lemma 18, it is $[\Theta_G]_J([\Theta_G]_J^{\uparrow \omega})(A)(w) = [\Theta_G]_J([\Theta_G]_J^{\uparrow m})(A)(w) = [\Theta_G]_J^{\uparrow m+1}(A)(w) \leq [\Theta_G]_J^{\uparrow \omega}(A)(w)$. Therefore, it holds that $[\Theta_G]_J([\Theta_G]_J^{\uparrow \omega}) \leq [\Theta_G]_J^{\uparrow \omega}$.

It remains to show that $[\Theta_G]_J^{\uparrow \omega}$ is the least fixed-point of $[\Theta_G]_J$ with respect to \leq . Suppose that Q is another fixed-point of $[\Theta_G]_J$. It suffices to show that $[\Theta_G]_J^{\uparrow \omega} \leq Q$. We show by induction that $[\Theta_G]_J^{\uparrow n} \leq Q$ for every $n \geq 0$. Obviously, $\perp = [\Theta_G]_J^{\uparrow 0} \leq Q$. Assume that $[\Theta_G]_J^{\uparrow n} \leq Q$. Then, $[\Theta_G]_J^{\uparrow n+1} \leq [\Theta_G]_J(Q) = Q$, since we have assumed that Q is a fixed-point of $[\Theta_G]_J$. Consequently, $[\Theta_G]_J^{\uparrow n} \leq Q$ for every $n \geq 0$, i.e., Q is an upper bound of the sequence $\{[\Theta_G]_J^{\uparrow n}\}_{n < \omega}$. Now, since $[\Theta_G]_J^{\uparrow \omega}$ is the least upper bound of the sequence $\{[\Theta_G]_J^{\uparrow n}\}_{n < \omega}$, we get that $[\Theta_G]_J^{\uparrow \omega} \leq Q$, which proves $[\Theta_G]_J^{\uparrow \omega}$ to be the least fixed-point of $[\Theta_G]_J$. \square

We will denote by $\Omega_G(J)$ the least fixed-point $[\Theta_G]_J^{\uparrow \omega}$ of $[\Theta_G]_J$. Given a grammar G , we can use the Ω_G operator to construct a sequence of interpretations whose least upper bound M_G (with respect to \leq_F) will prove to be a distinguished model of G . Notice that here we have an essential difference with respect to the well-founded semantics of logic programming: there, the construction of the well-founded model may require a transfinite number of iterations which is greater than ω . An undesirable consequence of this fact is that the well-founded semantics of logic programs is not computable in the general case. However, in the case of Boolean grammars, the model is constructed in at most ω iterations. Intuitively, this is due to the following reasons: (i) Boolean grammars are finite, and (ii) the membership of a string w in the language defined by a non-terminal, depends only on the memberships of a finite number of strings (namely the substrings of w) in finitely many languages (corresponding to the non-terminal symbols of the grammar).

The definition of M_G has as follows:

Definition 21. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar. Define:

$$\begin{aligned}
M_{G,0} &= \perp_F \\
M_{G,n+1} &= \Omega_G(M_{G,n}) \\
M_G &= \text{lub}_{\leq_F} \{M_{G,n} \mid n < \omega\}
\end{aligned}$$

From the above definition, it is not immediately obvious that M_G is well-defined (since as we have remarked at the end of Section 3, lub_{\leq_F} is not always well-defined). However, as we are going to see shortly, the operator Ω_G is monotonic with respect to \leq_F and this ensures that the sequence $\{M_{G,n}\}_{n < \omega}$ is increasing (which ensures that lub_{\leq_F} is well-defined).

Lemma 22. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar. Then, Ω_G is monotonic with respect to the \leq_F ordering of interpretations.

Proof. Let J_1, J_2 be two interpretations of G such that $J_1 \preceq_F J_2$. We show that $\Omega_G(J_1) \preceq_F \Omega_G(J_2)$, or equivalently that $[\Theta_G]_{J_1}^{\uparrow\omega} \preceq_F [\Theta_G]_{J_2}^{\uparrow\omega}$. We first prove that for all $n \geq 0$, $[\Theta_G]_{J_1}^{\uparrow n} \preceq_F [\Theta_G]_{J_2}^{\uparrow n}$. The proof is by induction on n . The basis case obviously holds. Assume the statement holds for n ; we demonstrate the case $n + 1$. Let $A \in N$ and $w \in \Sigma^*$. We distinguish two cases regarding the value of $[\Theta_G]_{J_1}^{\uparrow n+1}(A)(w)$.

Case 1: $[\Theta_G]_{J_1}^{\uparrow n+1}(A)(w) = 0$, or equivalently $[\Theta_G]_{J_1}([\Theta_G]_{J_1}^{\uparrow n})(A)(w) = 0$. From Definition 14, this implies that for every rule $A \rightarrow l_1 \& \dots \& l_r$ in P , either there exists a positive l_i such that $([\Theta_G]_{J_1}^{\uparrow n})(l_i)(w) = 0$, or there exists a negative l_i such that $\widehat{J_1}(l_i)(w) = 0$. In the former case, from Lemma 13 and the induction hypothesis, we obtain that there exists a positive l_i such that $([\Theta_G]_{J_2}^{\uparrow n})(l_i)(w) = 0$. In the latter case, from Lemma 13 and the fact that $J_1 \preceq_F J_2$, we obtain that there exists a negative l_i such that $\widehat{J_2}(l_i)(w) = 0$. Therefore, $[\Theta_G]_{J_2}^{\uparrow n+1}(A)(w) = 0$.

Case 2: Entirely analogous to the proof of Case 1.

We can now prove that $[\Theta_G]_{J_1}^{\uparrow\omega} \preceq_F [\Theta_G]_{J_2}^{\uparrow\omega}$. Suppose first that $[\Theta_G]_{J_1}^{\uparrow\omega}(A)(w) = 1$. Then there exists some m such that $[\Theta_G]_{J_1}^{\uparrow m}(A)(w) = 1$. Thus, it is also $[\Theta_G]_{J_2}^{\uparrow m}(A)(w) = 1$, which implies that $[\Theta_G]_{J_2}^{\uparrow\omega}(A)(w) = 1$. Suppose now that $[\Theta_G]_{J_1}^{\uparrow\omega}(A)(w) = 0$. Then $[\Theta_G]_{J_1}^{\uparrow n}(A)(w) = 0$ for every n . Thus, it is also $[\Theta_G]_{J_2}^{\uparrow n}(A)(w) = 0$ for every n , which implies that $[\Theta_G]_{J_2}^{\uparrow\omega}(A)(w) = 0$. \square

Apart from its monotonicity, Ω_G has another important property (which is the analogue of the property described in Lemma 18 for the $[\Theta_G]$ operator):

Lemma 23. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar and let J_1, J_2 be interpretations of G . Let $w \in \Sigma^*$ and assume that $J_1/w = J_2/w$. Then, for every $A \in N$, $\Omega_G(J_1)(A)(w) = \Omega_G(J_2)(A)(w)$.

Proof. We first prove by induction on n that for every $n \geq 0$, $[\Theta_G]_{J_1}^{\uparrow n}/w = [\Theta_G]_{J_2}^{\uparrow n}/w$.

The basis case is trivial. For the induction hypothesis, let us assume that $[\Theta_G]_{J_1}^{\uparrow n}/w = [\Theta_G]_{J_2}^{\uparrow n}/w$. This implies that for every substring u of w , it is also $[\Theta_G]_{J_1}^{\uparrow n}/u = [\Theta_G]_{J_2}^{\uparrow n}/u$. Moreover, $J_1/u = J_2/u$. From Lemma 18 we obtain that $[\Theta_G]_{J_1}^{\uparrow n+1}(A)(u) = [\Theta_G]_{J_2}^{\uparrow n+1}(A)(u)$, for every $A \in N$ and every substring u of w . Thus, $[\Theta_G]_{J_1}^{\uparrow n+1}/w = [\Theta_G]_{J_2}^{\uparrow n+1}/w$, which completes the inductive proof.

Therefore, for every n and every $A \in N$, $[\Theta_G]_{J_1}^{\uparrow n}(A)(w) = [\Theta_G]_{J_2}^{\uparrow n}(A)(w)$. The lemma follows from the definition of Ω_G . \square

Theorem 24. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar. Then, the sequence $\{M_{G,n}\}_{n < \omega}$ is increasing with respect to the \preceq_F ordering of interpretations. Moreover, M_G is the least fixed-point of the operator Ω_G .

Proof. Using the monotonicity of Ω_G with respect to the \preceq_F (Lemma 22), it can be proved (by similar arguments as in Theorem 20) that the sequence $\{M_{G,n}\}_{n < \omega}$ is increasing with respect to \preceq_F and that $M_G \preceq_F \Omega_G(M_G)$.

In order to prove that M_G is a fixed-point, it remains to prove that $\Omega_G(M_G) \preceq_F M_G$. Consider arbitrary $A \in N$ and $w \in \Sigma^*$. Since the sequence $\{M_{G,n}\}_{n < \omega}$ is increasing with respect to \preceq_F from Lemma 17 there exists some $m < \omega$ such that $M_G/w = M_{G,m}/w$. From Lemma 23, $\Omega_G(M_G)(A)(w) = \Omega_G(M_{G,m})(A)(w) = M_{G,m+1}(A)(w) \preceq_F M_G(A)(w)$. In other words, $\Omega_G(M_G) \preceq_F M_G$.

Therefore, M_G is a fixed-point of Ω_G . Using a similar reasoning as in Theorem 20, we can show that M_G is actually the least fixed-point of Ω_G with respect to the \preceq_F ordering. \square

The above results lead to the following theorem, which demonstrates that M_G satisfies all the rules of the grammar G :

Theorem 25. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar. Then, M_G is a model of G (which will be called the well-founded model of G).

Proof. It suffices to demonstrate that for every rule $A \rightarrow l_1 \& \dots \& l_r$ in P and for every $w \in \Sigma^*$ it is $M_G(A)(w) \geq \widehat{M_G}(l_1 \& \dots \& l_r)(w)$. Let $v = \min\{\widehat{M_G}(l_1)(w), \dots, \widehat{M_G}(l_r)(w)\}$. Then, for every l_i it is $\widehat{M_G}(l_i)(w) \geq v$. Now, since from Theorem 24 it is $M_G = \Omega_G(M_G)$, for every l_i it is $(\Omega_G(M_G))(l_i)(w) \geq v$. This implies that there exists $k \geq 0$ such that for all $n \geq k$ and for every positive l_i , $([\Theta_G]_{M_G}^{\uparrow n})(l_i)(w) \geq v$. Applying Definition 14 we get that for every $n \geq k$, $([\Theta_G]_{M_G}^{\uparrow n+1})(A)(w) \geq v$, which implies that $\Omega_G(M_G)(A)(w) \geq v$. But then from Theorem 24 we get that $M_G(A)(w) \geq v$. Therefore, M_G is a model of G . \square

We now give an example that illustrates the well-founded construction as this has been defined above:

Example 26. Let G be the grammar given in Example 2. We will demonstrate that $M_G = M_2$, i.e., that in order to converge to the well-founded model of G we need exactly two iterations of Ω_G .

First, recall that $M_0 = \perp_F$ and $M_1 = \Omega_G(M_0) = [\Theta_G]_{\perp_F}^{\omega}$. Since C is defined by $C \rightarrow a|b$, we easily obtain that for every $n \geq 1$ it holds $[\Theta_G]_{\perp_F}^{\uparrow n}(C)(a) = [\Theta_G]_{\perp_F}^{\uparrow n}(C)(b) = 1$ and $[\Theta_G]_{\perp_F}^{\uparrow n}(C)(w) = 0$, for every $w \in \Sigma^* - \{a, b\}$. Moreover, for every $n \geq 1$ it holds

$$[\Theta_G]_{\perp_F}^{\uparrow n}(A)(w) = \begin{cases} 1, & w = u_1 a u_2 \text{ where } u_1, u_2 \in \{a, b\}^*, |u_1| = |u_2| < n \\ 0, & \text{otherwise} \end{cases}$$

This can be proved by an easy induction on n . For the basis case ($n = 1$), it is $[\Theta_G]_{\perp_F}^{\uparrow n}(A)(a) = 1$, which is derived from the rule $A \rightarrow a$. Suppose that the claim holds for n and consider a string $w = c_1 u_1 a u_2 c_2$, where $u_1, u_2 \in \{a, b\}^*$, $c_1, c_2 \in \{a, b\}$, $|c_1 u_1| = |c_2 u_2| < n + 1$. From the induction hypothesis it is $[\Theta_G]_{\perp_F}^{\uparrow n}(A)(u_1 a u_2) = 1$ and since it is also $[\Theta_G]_{\perp_F}^{\uparrow n}(C)(a) = [\Theta_G]_{\perp_F}^{\uparrow n}(C)(b) = 1$, from the rule $A \rightarrow CAC$ we obtain that $[\Theta_G]_{\perp_F}^{\uparrow n+1}(A)(w) = 1$.

Therefore:

$$M_1(A)(w) = \begin{cases} 1, & w = u_1 a u_2 \text{ where } u_1, u_2 \in \{a, b\}^*, |u_1| = |u_2| \\ 0, & \text{otherwise} \end{cases}$$

and similarly:

$$M_1(B)(w) = \begin{cases} 1, & w = u_1 b u_2 \text{ where } u_1, u_2 \in \{a, b\}^*, |u_1| = |u_2| \\ 0, & \text{otherwise} \end{cases}$$

Notice that the languages assigned by M_1 to the non-terminals A, B and C , which are defined by rules that are actually context-free, coincide with the languages that would be assigned to these symbols by the standard derivation-based semantics of context-free grammars.

On the other hand, the denotation of S remains completely undefined in M_1 : since in the unique rule defining S all conjuncts are negative, in order to compute the value of $[\Theta_G]_{\perp_F}^{\uparrow n}(S)(w)$ for any $n \geq 1$ and for any $w \in \Sigma^*$, we must use (according to Definition 14) the interpretation \perp_F in order to evaluate these conjuncts. Therefore, in all cases it is $[\Theta_G]_{\perp_F}^{\uparrow n}(S)(w) = \frac{1}{2}$, and therefore:

$$M_1(S)(w) = \frac{1}{2}$$

However, the situation regarding S changes when we proceed to compute M_2 : it is $M_2 = \Omega_G(M_1) = [\Theta_G]_{M_1}^{\omega}$, and now M_1 contains all the information we need regarding the non-terminals A, B and C . Consider any string $w = uu$, where $u \in \{a, b\}^*$. Since w has an even length, it is $M_1(A)(w) = M_1(B)(w) = 0$, which implies $\widehat{M}_1(\neg A)(w) = \widehat{M}_1(\neg B)(w) = 1$. Moreover, for every pair of odd length strings $v_1, v_2 \in \{a, b\}^*$ such that $v_1 v_2 = w$, the symbols in the middle of v_1 and v_2 are identical (as they are the i th and $(i + |w|)$ th symbols of w , for some i). Therefore, if $M_1(A)(v_1) = 1$, then $M_1(B)(v_2) = 0$, which implies that $\widehat{M}_1(AB)(w) = 0$, or equivalently $\widehat{M}_1(\neg AB)(w) = 1$. Similarly we obtain that $\widehat{M}_1(\neg BA)(w) = 1$. Thus, from the rule $S \rightarrow \neg(AB) \ \& \ \neg(BA) \ \& \ \neg A \ \& \ \neg B$ we derive $[\Theta_G]_{M_1}^{\uparrow n}(S)(w) = 1$, for every $n \geq 1$. On the other hand, for any string w that is not of the form uu , one of $\widehat{M}_1(\neg A)(w)$, $\widehat{M}_1(\neg B)(w)$, $\widehat{M}_1(\neg AB)(w)$, $\widehat{M}_1(\neg BA)(w)$ is 0, which implies that $[\Theta_G]_{M_1}^{\uparrow n}(S)(w) = 0$. In short,

$$[\Theta_G]_{M_1}^{\uparrow n}(S)(w) = \begin{cases} 1, & w \in \{uu \mid u \in \Sigma^*\} \\ 0, & \text{otherwise} \end{cases}$$

Moreover, it holds that $M_2(V) = M_1(V)$, for every $V \in \{A, B, C\}$, since the rules defining these symbols are negation-free. Additionally, $M_k = M_2$, for all $k \geq 2$. Therefore, $M_G = M_2$. Notice that the language produced by this grammar is two-valued.

At this point we examine a natural question that springs to mind after the introduction of the three-valued well-founded model. Since most of the current work in formal language theory is based on two-valued languages, it is reasonable to wonder whether the problem ‘‘Given a Boolean grammar G , is M_G two-valued?’’ is decidable. The following theorem demonstrates that this is not the case.

Theorem 27. *The following problem is undecidable: ‘‘Given a Boolean grammar $G = (\Sigma, N, P, S)$, decide whether for all $w \in \Sigma^*$, $M_G(S)(w) \in \{0, 1\}$ ’’.*

Proof. We present a reduction from the following well-known undecidable problem: ‘‘Given a context-free grammar over an alphabet Σ , decide whether the language defined by this grammar is Σ^* ’’. Let $G_1 = (\Sigma, N_1, P_1, S_1)$ be a context-free grammar. Consider the Boolean grammar $G = (\Sigma, N, P, S)$ where:

- $N = N_1 \cup \{S\}$, where $S \notin N_1$,
- $P = P_1 \cup \{S \rightarrow S_1, S \rightarrow \neg S\}$.

Suppose first that the language defined by G_1 is Σ^* . We easily obtain that $M_G(S_1)(w) = 1$ for every $w \in \Sigma^*$, since the well-founded semantics extends the standard semantics of context-free grammars. Since we also have the rule $S \rightarrow S_1$ in P , it will also be the case that $M_G(S)(w) = 1$ for every $w \in \Sigma^*$ (and therefore M_G is trivially two-valued).

For the other direction, suppose that for every $w \in \Sigma^*$, it is $M_G(S)(w) \in \{0, 1\}$. Since P contains the rule $S \rightarrow \neg S$ it cannot be $M_G(S)(w) = 0$ for any w . Therefore, for every w it holds that $M_G(S)(w) = 1$, which implies that $M_G(S_1)(w) = 1$. Since the well-founded semantics extends the standard semantics of context-free grammars, we get that w belongs to the language defined by G_1 . \square

Closing this section, we can now state the relationship between the well-founded semantics and the naturally reachable semantics of Boolean grammars. For the definition of the naturally reachable solution and the related terminology the reader is referred to [8] or the definition given in Section 2.

Theorem 28. *Suppose that a Boolean grammar G has a two-valued (i.e., with values 0 and 1) well-founded semantics. Then the naturally reachable solution for this grammar either coincides with the well-founded semantics or is undefined.*

Proof. We present an outline of the proof.

Let $\{X_1, \dots, X_n\}$ be the set of non-terminal symbols in G and assume that G has a two-valued well-founded model M_G . Then, $M_G(X_i)$ can be thought of as a two-valued language, i.e., as a set of strings. Moreover, let $X_1 = \phi_1(X_1, \dots, X_n), \dots, X_n = \phi_n(X_1, \dots, X_n)$ be the system of equations that corresponds to G and assume that it has a naturally reachable solution $L = (L_1, \dots, L_n)$.

Suppose for the sake of contradiction that $L \neq (M_G(X_1), \dots, M_G(X_n))$. Let w be a string of minimum length for which there exists an index j such that w belongs to exactly one of $M_G(X_j)$ and L_j . Consider the modulus M that consists of all the proper substrings of w . From the definition of w , we have $(L_1 \cap M, \dots, L_n \cap M) = (M_G(X_1) \cap M, \dots, M_G(X_n) \cap M)$ and $(L_1 \cap (M \cup \{w\}), \dots, L_n \cap (M \cup \{w\})) \neq (M_G(X_1) \cap (M \cup \{w\}), \dots, M_G(X_n) \cap (M \cup \{w\}))$.

Define $Q = \{X_i \mid M_G(X_i)(w) = 1\}$. For every $X_i \in Q$ there exist two integers n_i, m_i with the following properties: $M_{G, n_i}(X_i)(w) = 1$, $M_{G, n_i-1}(X_i)(w) = \frac{1}{2}$, $[\Theta_G]_{M_{G, n_i-1}}^{\uparrow m_i}(X_i)(w) = 1$ and $[\Theta_G]_{M_{G, n_i-1}}^{\uparrow m_i-1}(X_i)(w) \neq 1$. Intuitively, n_i and m_i indicate the point in the construction of M_G where the fact that $M_G(X_i)(w) = 1$ is obtained. For every $X_i, X_j \in Q$ we write $X_i \lesssim X_j$ if $n_i < n_j$ or $n_i = n_j$ and $m_i \leq m_j$. Informally, $X_i \lesssim X_j$ if $M_G(X_i)(w)$ takes the value 1 not later than $M_G(X_j)(w)$ does in the construction of M_G .

Consider now a sequence $X_{k_0}, X_{k_1}, \dots, X_{k_{|Q|-1}}$, such that $X_{k_j} \lesssim X_{k_{j+1}}$ for all j , in which every element of Q appears exactly once. We construct a specific sequence of vectors of the form $L^{(0)}, L^{(1)}, \dots, L^{(i)}, \dots, L^{(|Q|)}$, where $L^{(0)} = (L_1 \cap M, \dots, L_n \cap M) = (M_G(X_1) \cap M, \dots, M_G(X_n) \cap M)$ and $L^{(i+1)}$ is obtained from $L^{(i)}$ by substituting the (k_i) th component with $\phi_{k_i}(L^{(i)}) \cap (M \cup \{w\})$. It can be proved by induction that before the i th step the (k_i) th component of $L^{(i)}$ is $M_G(X_{k_i}) \cap M$ and that this step replaces it with $(M_G(X_{k_i}) \cap M) \cup \{w\} = M_G(X_{k_i}) \cap (M \cup \{w\})$. That is, the result of the i th step is the insertion of w in the (k_i) th component. The proof is based on two observations: the first is that all the information that was used to decide that $w \in M_G(X_{k_i})$, also appears in $L^{(i)}$, which implies that $w \in \phi_{k_i}(L^{(i)})$; the second is that the membership of every string in M in the (k_i) th component remains unchanged after the application of ϕ_{k_i} , since L is a solution of the system of equations.

Based on the above, it is easy to prove that the selected sequence converges to $(M_G(X_1) \cap (M \cup \{w\}), \dots, M_G(X_n) \cap (M \cup \{w\}))$. On the other hand, since L is a naturally reachable solution, the sequence converges to $(L_1 \cap (M \cup \{w\}), \dots, L_n \cap (M \cup \{w\}))$. Therefore, $(L_1 \cap (M \cup \{w\}), \dots, L_n \cap (M \cup \{w\})) = (M_G(X_1) \cap (M \cup \{w\}), \dots, M_G(X_n) \cap (M \cup \{w\}))$, which is a contradiction. \square

It is easy to see that if a Boolean grammar has a naturally reachable solution semantics, then it is possible that this semantics differs from the well-founded one. For example, in the four-rule grammar of Section 2 (the one given just after Definition 3), the well-founded semantics assigns the \perp_F interpretation to all the non-terminal symbols of the grammar. Notice that although the naturally reachable semantics for this grammar is defined, it appears to be counterintuitive.

5. Normal form

In this section, we demonstrate that every Boolean grammar can be converted into an equivalent one that belongs to a *binary normal form*. Based on this normal form, in Section 6 we derive an $\mathcal{O}(n^3)$ parsing algorithm for Boolean grammars. The binary normal form is defined as follows:

Definition 29. A Boolean grammar $G = (\Sigma, N \cup \{U, T\}, P, S)$ is said to be in binary normal form if P contains the rules $U \rightarrow \neg U$ and $T \rightarrow \neg \epsilon$, where U and T are two special symbols not in N , and every other rule in P is of the form:

$$\begin{aligned} A &\rightarrow B_1 C_1 \& \dots \& B_m C_m \& \neg D_1 E_1 \& \dots \& \neg D_n E_n \& TT[\&U] \quad (m, n \geq 0) \\ A &\rightarrow a[\&U] \\ S &\rightarrow \epsilon[\&U] \quad (\text{only if } S \text{ does not appear in right-hand sides of rules}) \end{aligned}$$

where $A, B_i, C_i, D_j, E_j \in N$, $a \in \Sigma$, and the brackets denote an optional part.

The main theorem of this section is the following:

Theorem 30. *Let $G = (\Sigma, N, P, S)$ be a Boolean grammar. Then there exists a grammar $G' = (\Sigma, N', P', S)$ in binary normal form such that $M_G(S) = M_{G'}(S)$.*

The proof of Theorem 30 is based on the definition of certain meaning-preserving grammar transformations. It can be easily checked that each transformation step can be effectively performed; in other words, the normal form of a given grammar G can be constructed from G in an algorithmic way.

The normal form we derive, generalizes the well-known Chomsky normal form for context-free grammars as-well-as the binary normal form for Boolean grammars introduced in [8]. Actually, certain of the steps we adopt, were initially proposed in [8], the main difference being that the binary normal form obtained there, always produces two-valued Boolean languages.

The steps of the proposed procedure, can be summarized as follows:

- The initial Boolean grammar is first brought into *pre-normal form*. This is just a simpler and more manageable form of the initial grammar.
- The grammar is then transformed into *direct form*. This means that if a non-terminal of the previous form of the grammar could produce a string of length one (possibly through the use of many rules), then this fact is recorded by using a single rule in the new grammar. The same happens even if the status of the string of length one was undefined in the previous grammar.
- The next step is to bring the grammar into an ϵ -free form, i.e., a form in which no non-terminal produces the string ϵ .
- The final step is to bring the grammar into a *binary normal form*, i.e., a form in which the “long” rules of the grammar contain conjuncts which consist of two non-terminals (with the possible exception of the non-terminal U , see Definition 29 above).

In the rest of this section, we will describe one-by-one the above transformation steps.

5.1. Pre-normal form

Consider a Boolean grammar $G = (\Sigma, N, P, S)$. Without loss of generality we may assume that S does not appear in the right-hand side of any rule (otherwise we can replace S with S' in every rule, and add a rule $S \rightarrow S'$). Initially, we bring the grammar into a form, which we call *pre-normal form*:

Definition 31. A Boolean grammar $G = (\Sigma, N, P, S)$ is said to be in pre-normal form if every rule in P is of the form:

$$\begin{array}{lll} A & \rightarrow & B_1 \& \dots \& B_m \& \neg C_1 \& \dots \& \neg C_n & (m + n \geq 1, B_i, C_j \in N \cup \{\epsilon\}) \\ A & \rightarrow & BC & (B, C \in N) \\ A & \rightarrow & a & (a \in \Sigma) \end{array}$$

In order to prove that for every Boolean grammar there exists an equivalent one in pre-normal form, we need the following lemma:

Lemma 32. *Let $G = (\Sigma, N, P, S)$, $G' = (\Sigma, N \cup \{B_\beta\}, P' \cup \{B_\beta \rightarrow \beta\}, S)$ be two Boolean grammars, such that:*

- $B_\beta \notin N$
- $\beta \in (\Sigma \cup N)^+$
- P is obtained from P' by replacing in every rule each occurrence of B_β with β .

Then, for every $A \in N$, $M_G(A) = M_{G'}(A)$.

Proof. It suffices to show that for every $A \in N$, $M_{G,n}(A) = M_{G',n}(A)$. In order to establish this fact we will use the following idea. Let J, J' be interpretations for G and G' , respectively, such that $J(A) = J'(A)$ for every $A \in N$ and $\widehat{J}(\beta) = J'(B_\beta)$. We will demonstrate that for every $A \in N$, $\Omega_G(J)(A) = \Omega_{G'}(J')(A)$. This result will then be used in the inductive proof of the fact that for all n , $M_{G,n}(A) = M_{G',n}(A)$.

We start by proving some useful facts. First notice that P' does not contain any rule that defines B_β , since $B_\beta \notin N$. Thus, the only rule in G' that defines B_β is $B_\beta \rightarrow \beta$, which from Definition 14 implies that for every $n > 0$, $[\Theta_{G'}]_{J'}^{\uparrow n}(B_\beta) = ([\Theta_{G'}]_{J'}^{\uparrow n-1})(\beta)$. Moreover, since the sequence $\{[\Theta_{G'}]_{J'}^{\uparrow n}\}_{n < \omega}$ is increasing with respect to \leq (Theorem 20) and $[\Theta_{G'}]_{J'}^{\uparrow 0} = \perp$, we obtain that for every $n \geq 0$ and for all $w \in \Sigma^*$ it holds $[\Theta_{G'}]_{J'}^{\uparrow n}(B_\beta)(w) \leq ([\Theta_{G'}]_{J'}^{\uparrow n})(\beta)(w)$.

In order to show that for all $A \in N$, $\Omega_G(J)(A) = \Omega_{G'}(J')(A)$, we prove by induction on n that for every $A \in N$ and for all $w \in \Sigma^*$ it is $[\Theta_{G'}]_{J'}^{\uparrow n}(A)(w) \leq [\Theta_G]_J^{\uparrow n}(A)(w) \leq [\Theta_{G'}]_{J'}^{\uparrow 2n}(A)(w)$.

The basis case is obvious since $[\Theta_{G'}]_{J'}^{\uparrow 0} = [\Theta_G]_J^{\uparrow 0} = \perp$. Assume the statement holds for n ; we demonstrate that $[\Theta_{G'}]_{J'}^{\uparrow n+1}(A)(w) \leq [\Theta_G]_J^{\uparrow n+1}(A)(w) \leq [\Theta_{G'}]_{J'}^{\uparrow 2n+2}(A)(w)$.

Define the intermediate interpretation I' :

$$I'(C) = \begin{cases} [\Theta_{G'}]_{J'}^{\uparrow 2n}(C), & C \in N \\ [\Theta_{G'}]_{J'}^{\uparrow 2n+1}(C), & C = B_\beta \end{cases}$$

Then, $[\Theta_{G'}]_{J'}^{\uparrow 2n} \leq I' \leq [\Theta_{G'}]_{J'}^{\uparrow 2n+1}$. Furthermore, $I'(B_\beta) = [\Theta_{G'}]_{J'}^{\uparrow 2n+1}(B_\beta) = [\Theta_{G'}]_{J'}^{\uparrow 2n}(\beta) = \widehat{I}(\beta)$.

Consider any rule $A \rightarrow \gamma_1 \& \dots \& \gamma_m \& \neg \delta_1 \& \dots \& \neg \delta_r$ in P . From the definition of G and G' , there is a rule $A \rightarrow \gamma'_1 \& \dots \& \gamma'_m \& \neg \delta'_1 \& \dots \& \neg \delta'_r$ in P' , such that each γ_i (or δ_j) has resulted by replacing every occurrence of B_β in γ'_i (respectively, δ'_j) with β .

Then, it is easy to see that $\widehat{I}(\gamma_i) = \widehat{I}'(\gamma'_i)$. From the induction hypothesis, using the fact that $[\Theta_{G'}]_{J'}^{\uparrow n}(B_\beta)(w) \leq ([\Theta_{G'}]_{J'}^{\uparrow n})(\beta)(w)$, we get: $([\Theta_{G'}]_{J'}^{\uparrow n})(\gamma'_i)(w) \leq ([\Theta_{G'}]_{J'}^{\uparrow n})(\gamma_i)(w) \leq ([\Theta_G]_{J'}^{\uparrow n})(\gamma_i)(w) \leq ([\Theta_{G'}]_{J'}^{\uparrow 2n})(\gamma_i)(w) \leq \widehat{I}(\gamma_i)(w) = \widehat{I}'(\gamma'_i)(w) \leq ([\Theta_{G'}]_{J'}^{\uparrow 2n+1})(\gamma'_i)(w)$. Furthermore, from the definition of J, J' we have that $\widehat{J}(\delta_j) = \widehat{J}'(\delta_j) = \widehat{J}'(\delta'_j)$.

The above facts imply that if there exists a rule $A \rightarrow \gamma_1 \& \dots \& \gamma_m \& \neg \delta_1 \& \dots \& \neg \delta_r$ in P such that $([\Theta_G]_{J'}^{\uparrow n})(\gamma_i)(w) = 1$ for every i and $\widehat{J}(\delta_j)(w) = 0$ for every j , then there exists a corresponding rule $A \rightarrow \gamma'_1 \& \dots \& \gamma'_m \& \neg \delta'_1 \& \dots \& \neg \delta'_r$ in P' such that $([\Theta_{G'}]_{J'}^{\uparrow 2n+1})(\gamma'_i)(w) = 1$ for every i and $\widehat{J}'(\delta'_j)(w) = 0$ for every j . Thus, if $[\Theta_G]_{J'}^{\uparrow n+1}(A)(w) = [\Theta_G]_{J'}([\Theta_G]_{J'}^{\uparrow n})(A)(w) = 1$, then $[\Theta_{G'}]_{J'}^{\uparrow 2n+2}(A)(w) = [\Theta_{G'}]_{J'}([\Theta_{G'}]_{J'}^{\uparrow 2n+1})(A)(w) = 1$. In the same way we get that, if $[\Theta_{G'}]_{J'}^{\uparrow 2n+2}(A)(w) = 0$, then $[\Theta_G]_{J'}^{\uparrow n+1}(A)(w) = 0$, which implies that if $[\Theta_G]_{J'}^{\uparrow n+1}(A)(w) = \frac{1}{2}$, then $[\Theta_{G'}]_{J'}^{\uparrow 2n+2}(A)(w) \geq \frac{1}{2}$. Therefore, in any case it holds $[\Theta_G]_{J'}^{\uparrow n+1}(A)(w) \leq [\Theta_{G'}]_{J'}^{\uparrow 2n+2}(A)(w)$.

In order to prove that $[\Theta_{G'}]_{J'}^{\uparrow n+1}(A)(w) \leq [\Theta_G]_{J'}^{\uparrow n+1}(A)(w)$, we consider analogous cases as above (using the fact that $([\Theta_G]_{J'}^{\uparrow n})(\gamma'_i)(w) \leq ([\Theta_G]_{J'}^{\uparrow n})(\gamma_i)(w)$ and $\widehat{J}(\delta_j) = \widehat{J}'(\delta'_j)$).

Therefore, for all $n \geq 0$, $[\Theta_{G'}]_{J'}^{\uparrow n}(A)(w) \leq [\Theta_G]_{J'}^{\uparrow n}(A)(w) \leq [\Theta_{G'}]_{J'}^{\uparrow 2n}(A)(w)$.

From the definition of Ω_G , the above two inequalities imply that $\Omega_G(J)(A) = \Omega_{G'}(J')(A)$, for every $A \in N$. This implies that, $\Omega_G(\widehat{J})(\beta) = \Omega_{G'}(\widehat{J}')(\beta)$. Since we have shown that $[\Theta_{G'}]_{J'}^{\uparrow 2n+1}(B_\beta) = ([\Theta_{G'}]_{J'}^{\uparrow 2n})(\beta)$, we have $\Omega_{G'}(J')(B_\beta) = \Omega_{G'}(\widehat{J}')(\beta)$. Combining the last two equalities we have $\Omega_{G'}(J')(B_\beta) = \Omega_G(\widehat{J})(\beta)$.

Using the above facts and an easy induction on n , we can prove that $M_{G,n}(A) = M_{G',n}(A)$ from which the lemma follows. \square

Lemma 33. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar. Then, there exists a Boolean grammar $G' = (\Sigma, N', P', S)$ in pre-normal form, such that $M_G(S) = M_{G'}(S)$.

Proof. The Boolean grammar G' is constructed from G , using a transformation that consists of three steps. In the first step, terminal symbols are eliminated from rules containing Boolean connectives or concatenation. This is obtained by adding a new rule $A_a \rightarrow a$, for every terminal symbol $a \in \Sigma$, where A_a is a new non-terminal symbol, and then replacing every occurrence of a in the rules of the above kind by A_a .

The second step of the transformation eliminates concatenation from the rules of the new grammar containing conjunction. In order to do this, for every $\beta \in N^*$ with $|\beta| \geq 2$, such that at least one of the literals β or $\neg\beta$ appears in the body of some rule that contains conjunction, we add a new rule $B_\beta \rightarrow \beta$, where B_β is a new non-terminal symbol. Then, we replace every occurrence of literal β (or $\neg\beta$) by B_β (respectively, $\neg B_\beta$) in any rule with conjunction.

Finally, the third step of the transformation eliminates long concatenations. More specifically, while there exists a rule $A \rightarrow B_1 B_2 B_3 \dots B_k$ with $k \geq 3$ in the current grammar, we pick a new non-terminal D and replace this rule by the rules $A \rightarrow DB_3 \dots B_k$ and $D \rightarrow B_1 B_2$.

It is easy to see that the third step, after finitely many iterations, produces a Boolean grammar G' in pre-normal form. Moreover, from Lemma 32 (which is applied several times for each step of the transformation) it follows that $M_G(S) = M_{G'}(S)$. \square

5.2. Direct form

Based on the pre-normal form derived in the previous section, we now construct the *direct form* of the grammar: if a string of length one can be produced by a non-terminal in the previous form of the grammar, then a rule expressing directly this fact is inserted into the grammar.

Definition 34. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar in pre-normal form. Then, the *direct form* of G , denoted by G_δ , is the Boolean grammar $G_\delta = (\Sigma, N \cup \{U\}, P \cup R, S)$, where $U \notin N$ is a special non-terminal symbol that represents the set

in which all strings have the value $\frac{1}{2}$ and $R = \{U \rightarrow \neg U\} \cup \{A \rightarrow a \mid a \in \Sigma, A \in N \text{ and } M_G(A)(a) = 1\} \cup \{A \rightarrow a \& U \mid a \in \Sigma, A \in N \text{ and } M_G(A)(a) = \frac{1}{2}\}$.

The proof of the following lemma (as-well-as of Lemma 37 later on) are quite straightforward but rather tedious since they require lengthy inductions and the analysis of different cases that are quite similar in their treatment. The proof given below adapts and uses a well-known technique from the theory of programming languages (see for example [12] [pp. 209]): in order to show that two grammars, say G_1 and G_2 , are equivalent, it suffices to show that the well-founded model of each grammar is a fixed-point of the Ω operator of the other grammar. In other words, it suffices to show that $M_{G_1} = \Omega_{G_2}(M_{G_1})$ and $M_{G_2} = \Omega_{G_1}(M_{G_2})$. Then, since we know that the least fixed-point of Ω_{G_1} is M_{G_1} and the least fixed-point of Ω_{G_2} is M_{G_2} , we get that $M_{G_1} \leq_F M_{G_2}$ and $M_{G_2} \leq_F M_{G_1}$, which implies that $M_{G_1} = M_{G_2}$. The proof of the following lemma illustrates in a more precise way this technique:

Lemma 35. *Let $G = (\Sigma, N, P, S)$ be a Boolean grammar in pre-normal form, and let $G_\delta = (\Sigma, N \cup \{U\}, P \cup R, S)$ be its direct form. Then, for every $C \in N$ and for every $w \in \Sigma^*$, $M_G(C)(w) = M_{G_\delta}(C)(w)$.*

Proof. We demonstrate that an appropriate extension of M_G to $N \cup \{U\}$, which we denote by M_G^* , is a fixed-point of Ω_{G_δ} . Similarly, we argue that the restriction of M_{G_δ} to N , which we denote by $M_{G_\delta}^-$, is a fixed-point of Ω_G . The result then follows easily.

Define M_G^* so that $M_G^*(C) = M_G(C)$ for every $C \in N$ and $M_G^*(U)(w) = \frac{1}{2}$ for every $w \in \Sigma^*$. We claim that $M_G^* = \Omega_{G_\delta}(M_G^*)$. It suffices to show that for all $A \in N \cup \{U\}$ and all $w \in \Sigma^*$, it is $M_G^*(A)(w) = \Omega_{G_\delta}(M_G^*)(A)(w)$.

Suppose first that $A = U$. Since $M_G^*(U)(w) = \frac{1}{2}$ and the only rule in $P \cup R$ that defines U is $U \rightarrow \neg U$, it follows that $[\Theta_G]_{M_G^*}^{\uparrow n}(U)(w) = \frac{1}{2}$, for every w and for every $n \geq 1$. Therefore $\Omega_{G_\delta}(M_G^*)(U)(w) = \frac{1}{2} = M_G^*(U)(w)$.

Consider now the remaining case, namely $A \in N$. We know that $M_G^*(A)(w) = M_G(A)(w) = \Omega_G(M_G)(A)(w)$ (from the definition of M_G^* and from the fact that M_G is a fixed-point of Ω_G). Therefore, it suffices to prove that for every $A \in N$ and for every $w \in \Sigma^*$, $\Omega_G(M_G)(A)(w) = \Omega_{G_\delta}(M_G^*)(A)(w)$. In order to prove this, we will first show that there exists some integer constant k such that for every n , $[\Theta_G]_{M_G}^{\uparrow n}(A)(w) \leq [\Theta_{G_\delta}]_{M_G^*}^{\uparrow n}(A)(w) \leq [\Theta_G]_{M_G}^{\uparrow n+k}(A)(w)$. We select k as follows: consider any $B \in N$ and $a \in \Sigma$. Since $M_G(B)(a) = \Omega_G(M_G)(B)(a)$ from the definition of the Ω_G operator there exists a least integer $k_{B,a}$ such that $M_G(B)(a) = [\Theta_G]_{M_G}^{\uparrow k_{B,a}}(B)(a)$. We now define $k = \max\{k_{B,a} \mid B \in N, a \in \Sigma\}$.

We now prove by induction on n that for every $n \geq 0$, for every $A \in N$ and for every $w \in \Sigma^*$, $[\Theta_G]_{M_G}^{\uparrow n}(A)(w) \leq [\Theta_{G_\delta}]_{M_G^*}^{\uparrow n}(A)(w) \leq [\Theta_G]_{M_G}^{\uparrow n+k}(A)(w)$. The basis case is obvious, since $[\Theta_G]_{M_G}^{\uparrow 0}(A)(w) = [\Theta_{G_\delta}]_{M_G^*}^{\uparrow 0}(A)(w) = 0$. Assume the statement holds for n ; we first demonstrate that $[\Theta_{G_\delta}]_{M_G^*}^{\uparrow n+1}(A)(w) \leq [\Theta_G]_{M_G}^{\uparrow n+k+1}(A)(w)$.

Suppose first that $[\Theta_{G_\delta}]_{M_G^*}^{\uparrow n+1}(A)(w) = 1$. If this value is obtained using a rule in P , then using the induction hypothesis and the relationship between M_G and M_G^* , we get that $[\Theta_G]_{M_G}^{\uparrow n+k+1}(A)(w) = 1$. On the other hand, if this value is obtained using a rule $A \rightarrow a$ in R , then $w = a$ and from the construction of G_δ it is $M_G(A)(a) = 1$. From the definition of k we have $[\Theta_G]_{M_G}^{\uparrow n+k+1}(A)(w) = 1$.

Next, suppose that $[\Theta_{G_\delta}]_{M_G^*}^{\uparrow n+1}(A)(w) = \frac{1}{2}$. Suppose for the sake of contradiction that $[\Theta_G]_{M_G}^{\uparrow n+k+1}(A)(w) = 0$. If $w = \alpha \in \Sigma$, then from the definition of k , it is $M_G(A)(\alpha) = 0$. Therefore, all the rules in R with head A are of the form $A \rightarrow b$, with $b \neq \alpha$. Using the induction hypothesis and the relationship between M_G and M_G^* , we easily obtain that $[\Theta_{G_\delta}]_{M_G^*}^{\uparrow n+1}(A)(w) = 0$, which is a contradiction. Therefore, $[\Theta_G]_{M_G}^{\uparrow n+k+1}(A)(w) \geq \frac{1}{2}$.

Finally, the case in which $[\Theta_{G_\delta}]_{M_G^*}^{\uparrow n+1}(A)(w) = 0$ is trivial. Therefore, in any case it is $[\Theta_{G_\delta}]_{M_G^*}^{\uparrow n+1}(A)(w) \leq [\Theta_G]_{M_G}^{\uparrow n+k+1}(A)(w)$.

Now, in order to prove that $[\Theta_G]_{M_G}^{\uparrow n+1}(A)(w) \leq [\Theta_{G_\delta}]_{M_G^*}^{\uparrow n+1}(A)(w)$, we observe that the set of rules of G that define A , is a subset of the corresponding set of rules of G_δ . This implies, using the induction hypothesis and the relationship between M_G and M_G^* , that if $[\Theta_G]_{M_G}^{\uparrow n+1}(A)(w) = 1$ then $[\Theta_{G_\delta}]_{M_G^*}^{\uparrow n+1}(A)(w) = 1$ and if $[\Theta_{G_\delta}]_{M_G^*}^{\uparrow n+1}(A)(w) = 0$ then $[\Theta_G]_{M_G}^{\uparrow n+1}(A)(w) = 0$ from which our claim follows immediately.

From the definition of the Ω operator, we get $\Omega_G(M_G)(A)(w) = \Omega_{G_\delta}(M_G^*)(A)(w)$. Thus, we have proved that $M_G^* = \Omega_{G_\delta}(M_G^*)$. Since $M_{G_\delta}^-$ is the least fixed-point of Ω_{G_δ} with respect to \leq_F , this implies that $M_{G_\delta}^- \leq_F M_G^*$.

Now, let $M_{G_\delta}^-$ be the restriction of M_{G_δ} to N . In order to use a similar technique as above, we need to show that for every $A \in N$ and for every $a \in \Sigma$, $[\Theta_G]_{M_{G_\delta}^-}^{\uparrow k}(A)(a) = M_G(A)(a)$.

We first show that $M_G(A)(a) = M_{G_\delta}^-(A)(a)$. From $M_{G_\delta}^- \leq_F M_G^*$, it follows that $M_{G_\delta}^-(A)(a) \leq_F M_G(A)(a)$. Furthermore, $M_G(A)(a) \leq_F M_{G_\delta}^-(A)(a)$ follows from the following two facts:

- if $M_G(A)(a) = 1$, then R contains the rule $A \rightarrow a$ and
- if $M_G(A)(a) = 0$, then the rules in R that define A are of the form $A \rightarrow b$ with $b \neq a$.

Therefore, for every $A \in N$ and for every $a \in \Sigma$, $M_G(A)(a) = M_{G_\delta}^-(A)(a)$, which by a simple induction on n , gives that $[\Theta_G]_{M_G}^{\uparrow n}(A)(a) = [\Theta_G]_{M_{G_\delta}^-}^{\uparrow n}(A)(a)$. In particular $[\Theta_G]_{M_G}^{\uparrow k}(A)(a) = [\Theta_G]_{M_{G_\delta}^-}^{\uparrow k}(A)(a) = M_G(A)(a)$.

Now it is easy to prove that for every n , for every $A \in N$ and for every $w \in \Sigma^*$, $[\Theta_G]_{M_G}^{\uparrow n}(A)(w) \leq [\Theta_{G_\delta}]_{M_{G_\delta}^-}^{\uparrow n}(A)(w) \leq [\Theta_G]_{M_{G_\delta}^-}^{\uparrow n+k}(A)(w)$. This implies that $\Omega_G(M_{G_\delta}^-)(A)(w) = \Omega_{G_\delta}(M_{G_\delta}^-)(A)(w) = M_{G_\delta}^-(A)(w) = M_G(A)(w)$, that is $M_{G_\delta}^-$ is a fixed-point of Ω_G . Since M_G is the least fixed-point of Ω_G with respect to \leq_F , we obtain that $M_G \leq_F M_{G_\delta}^-$. Combining with $M_{G_\delta} \leq_F M_G^*$, we get that for every $A \in N$ and for every $w \in \Sigma^*$, it is $M_G(A)(w) = M_{G_\delta}(A)(w)$. \square

5.3. ϵ -Free form

The direct form of the grammar can now be transformed into the ϵ -free form, i.e., a form in which no non-terminal produces the string ϵ .

Definition 36. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar in pre-normal form and let $G_\delta = (\Sigma, N \cup \{U\}, P_\delta, S)$ be its direct form. The ϵ -free version of G , denoted by G_ϵ , is the Boolean grammar $(\Sigma, N \cup \{U\}, P_\epsilon, S)$ where P_ϵ is obtained as follows:

1. For every rule of the form $A \rightarrow B_1 \& \dots \& B_m \& \neg C_1 \& \dots \& \neg C_n$, ($m + n \geq 1, B_i, C_j \in N \cup \{\epsilon\}$) in P_δ
 - If $B_i = \epsilon$ for some i , then the rule is ignored in the construction of P_ϵ .
 - Otherwise, if $C_i = \epsilon$ for some i , then the rule is included in P_ϵ as it is.
 - Otherwise, P_ϵ contains the rule $A \rightarrow B_1 \& \dots \& B_m \& \neg C_1 \& \dots \& \neg C_n \& \neg \epsilon$.
2. For every rule of the form $A \rightarrow BC$ ($B, C \in N$) in P_δ
 - P_ϵ contains the rule $A \rightarrow BC \& \neg \epsilon$.
 - If $M_G(B)(\epsilon) = 1$ (respectively, $M_G(C)(\epsilon) = 1$), then P_ϵ contains the rule $A \rightarrow C \& \neg \epsilon$ (respectively, the rule $A \rightarrow B \& \neg \epsilon$).
 - If $M_G(B)(\epsilon) = \frac{1}{2}$ (respectively, $M_G(C)(\epsilon) = \frac{1}{2}$), then P_ϵ contains the rule $A \rightarrow C \& U \& \neg \epsilon$ (respectively, the rule $A \rightarrow B \& U \& \neg \epsilon$).
3. All the other rules in P_δ (i.e., the rules of the form $U \rightarrow \neg U, A \rightarrow a$, and $A \rightarrow a \& U$, where $a \in \Sigma$) are retained in P_ϵ .

Lemma 37. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar in pre-normal form, let $G_\delta = (\Sigma, N \cup \{U\}, P_\delta, S)$ be its direct form, and let $G_\epsilon = (\Sigma, N \cup \{U\}, P_\epsilon, S)$ be its ϵ -free version. Then, for every $A \in N$ and for every $w \in \Sigma^+$, $M_G(A)(w) = M_{G_\epsilon}(A)(w)$.

Proof. We demonstrate that a slightly modified version of M_{G_δ} , which we denote by $M_{G_\delta}^*$, is a fixed-point of Ω_{G_ϵ} . Similarly, we argue that a slightly modified version $M_{G_\epsilon}^+$ of M_{G_ϵ} is a fixed-point of Ω_{G_δ} . The result then follows easily.

We start by defining the interpretation $M_{G_\delta}^*$:

$$M_{G_\delta}^*(A)(w) = \begin{cases} M_{G_\delta}(A)(w), & w \neq \epsilon \\ 0, & \text{otherwise} \end{cases}$$

We claim that $M_{G_\delta}^* = \Omega_{G_\epsilon}(M_{G_\delta}^*)$. It suffices to show that for all $A \in N$ and all $w \in \Sigma^*$, it is $M_{G_\delta}^*(A)(w) = \Omega_{G_\epsilon}(M_{G_\delta}^*)(A)(w)$. We distinguish two cases. The first case is for $w = \epsilon$. Since every rule that defines A in G_ϵ has a conjunct that is either $\neg \epsilon$ or a terminal symbol, by an easy induction on n , we obtain that $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n}(A)(\epsilon) = 0$ for every $A \in N$. Therefore, $\Omega_{G_\epsilon}(M_{G_\delta}^*)(A)(\epsilon) = 0 = M_{G_\delta}^*(A)(\epsilon)$.

Consider now the second case, namely $w \neq \epsilon$. We know that $M_{G_\delta}^*(A)(w) = M_{G_\delta}(A)(w) = \Omega_{G_\delta}(M_{G_\delta})(A)(w)$ (from the definition of $M_{G_\delta}^*$ and from the fact that M_{G_δ} is a fixed-point of Ω_{G_δ}). Thus, it suffices to prove that $\Omega_{G_\delta}(M_{G_\delta})(A)(w) = \Omega_{G_\epsilon}(M_{G_\delta}^*)(A)(w)$. In order to prove this it suffices to prove that there exists some constant k such that for every n , $[\Theta_{G_\delta}]_{M_{G_\delta}}^{\uparrow n}(A)(w) \leq [\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n}(A)(w) \leq [\Theta_{G_\delta}]_{M_{G_\delta}}^{\uparrow n+k}(A)(w)$. We select k as follows: consider any symbol $B \in N$. Since $M_{G_\delta}(B)(\epsilon) = \Omega_{G_\delta}(M_{G_\delta})(B)(\epsilon)$ from the definition of Ω_{G_δ} , there exists a least integer k_B such that $M_{G_\delta}(B)(\epsilon) = [\Theta_{G_\delta}]_{M_{G_\delta}}^{\uparrow k_B}(B)(\epsilon)$. We define $k = \max\{k_B | B \in N\}$.

We will prove by induction on n that for every $n \geq 0$, for every $A \in N$ and for all $w \in \Sigma^+$ it is: $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n}(A)(w) \leq [\Theta_{G_\delta}]_{M_{G_\delta}}^{\uparrow n+k}(A)(w)$.

The basis case is obvious, since $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow 0}(A)(w) = 0$. Assume the statement holds for n ; we demonstrate that $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n+1}(A)(w) \leq [\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k+1}(A)(w)$. We distinguish three cases:

Case 1: $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n+1}(A)(w) = 1$. We examine in P_ϵ the rule types that may have forced the value of $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n+1}(A)(w)$ to become equal to 1 (notice that we need to consider only rules that do not have the conjunct U in their bodies):

- $A \rightarrow a$. This implies that $w = a$. Moreover, this rule also appears in P_δ . Therefore, $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k+1}(A)(w) = 1$.
- $A \rightarrow B_1 \& \dots \& B_m \& \neg C_1 \& \dots \& \neg C_r \& \neg \epsilon$, which also appears in P_δ (possibly without the $\neg \epsilon$ at the end). From Definition 14, we have that for all B_i it is $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n}(B_i)(w) = 1$ and for all C_j it is $\widehat{M_{G_\delta}^*}(\neg C_j)(w) = 1$. From the induction hypothesis, we have $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k}(B_i)(w) = 1$ for all B_i , and from the definition of $M_{G_\delta}^*$ we have $\widehat{M_{G_\delta}^*}(\neg C_j)(w) = 1$ for all C_j . Since P_δ also contains this rule, we have $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k+1}(A)(w) = 1$.
- $A \rightarrow BC \& \neg \epsilon$. Then there exist $w_1, w_2 \in \Sigma^*$ such that $w_1 w_2 = w$ and also $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n}(B)(w_1) = 1$ and $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n}(C)(w_2) = 1$. Since $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n}(B)(\epsilon) = [\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n}(C)(\epsilon) = 0$, we have that $w_1 \neq \epsilon$ and $w_2 \neq \epsilon$. From the induction hypothesis we have $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k}(B)(w_1) = 1$ and $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k}(C)(w_2) = 1$. Moreover, P_δ contains the rule $A \rightarrow BC$, which implies that $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k+1}(A)(w) = 1$.
- $A \rightarrow B \& \neg \epsilon$, where P_δ contains the rule $A \rightarrow BC$ (or the rule $A \rightarrow CB$) for some C such that $M_{G_\delta}(C)(\epsilon) = 1$. Then, it is $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n}(B)(w) = 1$. From the induction hypothesis we have $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k}(B)(w) = 1$. Furthermore, from the selection of k we have that $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k}(C)(\epsilon) = 1$. Consequently, $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k+1}(A)(w) = 1$.

Case 2: $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n+1}(A)(w) = \frac{1}{2}$. We will show that $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k+1}(A)(w) \geq \frac{1}{2}$, or equivalently that $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k+1}(A)(w) \neq 0$.

Suppose for the sake of contradiction that $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k+1}(A)(w) = 0$. We examine the rules that define A in P_ϵ . Each of them has one of the following types:

- $A \rightarrow a[\&U]$. But then, this rule also exists in P_δ . Since $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k+1}(A)(w) = 0$, we have that $a \neq w$, which implies $([\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n})(a)(w) = 0$.
- $A \rightarrow B_1 \& \dots \& B_m \& \neg C_1 \& \dots \& \neg C_r \& \neg \epsilon$, which also appears in P_δ (possibly without the $\neg \epsilon$ at the end). From Definition 14, $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k+1}(A)(w) = 0$ implies that either there exists some B_i such that $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k}(B_i)(w) = 0$, or there exists some C_j such that $\widehat{M_{G_\delta}^*}(\neg C_j)(w) = 0$. From the induction hypothesis and the definition of $M_{G_\delta}^*$ we have that either there exists some B_i such that $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n}(B_i)(w) = 0$, or there exists some C_j such that $\widehat{M_{G_\delta}^*}(\neg C_j)(w) = 0$.
- $A \rightarrow BC \& \neg \epsilon$. But then, P_δ contains the rule $A \rightarrow BC$. Thus, the fact that $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k+1}(A)(w) = 0$ implies that for every w_1, w_2 such that $w_1 w_2 = w$ we have that either $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k}(B)(w_1) = 0$ or $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k}(C)(w_2) = 0$. But then, by the induction hypothesis, together with the fact that $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n}(B)(\epsilon) = [\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n}(C)(\epsilon) = 0$, we have that for every w_1, w_2 such that $w_1 w_2 = w$ it will be either $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n}(B)(w_1) = 0$ or $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n}(C)(w_2) = 0$. This implies that $([\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n})(BC)(w) = 0$.
- $A \rightarrow B[\&U] \& \neg \epsilon$. But then, the rule $A \rightarrow BC$ (or the rule $A \rightarrow CB$) belongs to P_δ for some C such that $M_{G_\delta}(C)(\epsilon) \geq \frac{1}{2}$. From the selection of k we have that $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k}(C)(\epsilon) = M_{G_\delta}(C)(\epsilon) \geq \frac{1}{2}$. Now, since $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k+1}(A)(w) = 0$ it must be the case that $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k}(B)(w) = 0$. From the induction hypothesis, this implies that $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n}(B)(w) = 0$.

Therefore, for each rule that defines A in P_ϵ , there either exists a positive l_i such that $([\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n})(l_i)(w) = 0$ or a negative l_i such that $\widehat{M_{G_\delta}^*}(l_i)(w) = 0$. From Definition 14, this implies that $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n+1}(A)(w) = 0$, which is a contradiction. Therefore, $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k+1}(A)(w) \geq \frac{1}{2}$.

Case 3: $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n+1}(A)(w) = 0$. In this case our claim obviously holds.

Thus, we have proved that $[\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n}(A)(w) \leq [\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k}(A)(w)$. By using a similar inductive proof, we can show that for every n , $[\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n}(A)(w) \leq [\Theta_{G_\epsilon}]_{M_{G_\delta}^*}^{\uparrow n}(A)(w)$.

All the above lead us to the conclusion that $M_{G_\delta}^* = \Omega_{G_\epsilon}(M_{G_\delta}^*)$. Since M_{G_ϵ} is the least fixed-point of Ω_{G_ϵ} with respect to \leq_F , this implies that $M_{G_\epsilon} \leq_F M_{G_\delta}^*$.

Now, it remains to show that a slightly modified version of M_{G_ϵ} is a fixed-point of Ω_{G_δ} . More specifically, define:

$$M_{G_\epsilon}^+(A)(w) = \begin{cases} M_{G_\epsilon}(A)(w), & w \neq \epsilon \\ M_{G_\delta}(A)(w), & w = \epsilon \end{cases}$$

We claim that $M_{G_\epsilon}^+ = \Omega_{G_\delta}(M_{G_\epsilon}^+)$. It suffices to show that for all $A \in N$ and all $w \in \Sigma^*$, it is $M_{G_\epsilon}^+(A)(w) = \Omega_{G_\delta}(M_{G_\epsilon}^+)(A)(w)$.

For $w = \epsilon$, using the fact that $M_{G_\epsilon}^+(A)(\epsilon) = M_{G_\delta}(A)(\epsilon)$, we can prove by an easy induction on n that $[\Theta_{G_\delta}]_{M_{G_\epsilon}^+}^{\uparrow n}(A)(\epsilon) = [\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n}(A)(\epsilon)$ for every $A \in N$. Therefore, $\Omega_{G_\delta}(M_{G_\epsilon}^+)(A)(\epsilon) = \Omega_{G_\delta}(M_{G_\delta}^*)(A)(\epsilon) = M_{G_\delta}(A)(\epsilon) = M_{G_\epsilon}^+(A)(\epsilon)$.

For $w \neq \epsilon$, it suffices to prove that $\Omega_{G_\delta}(M_{G_\epsilon}^+)(A)(w) = \Omega_{G_\epsilon}(M_{G_\epsilon})(A)(w)$. In order to prove this it suffices to prove that for every n , $[\Theta_{G_\delta}]_{M_{G_\epsilon}^+}^{\uparrow n}(A)(w) \leq [\Theta_{G_\epsilon}]_{M_{G_\epsilon}}^{\uparrow n}(A)(w) \leq [\Theta_{G_\delta}]_{M_{G_\delta}^*}^{\uparrow n+k}(A)(w)$. This can be proven in an analogous way as above.

Now, since M_{G_δ} is the least fixed-point of Ω_{G_δ} with respect to \leq_F , we obtain that $M_{G_\delta} \leq_F M_{G_\epsilon}^+$. Combining with $M_{G_\epsilon} \leq_F M_{G_\delta}^*$, we get that for every $A \in N$ and for every $w \in \Sigma^+$, $M_{G_\delta}(A)(w) = M_{G_\epsilon}(A)(w)$.

The lemma then follows from Lemma 35. \square

5.4. The final step: binary normal form

In order to obtain a grammar in binary normal form, we need to eliminate rules of the form $A \rightarrow B_1 \& \dots \& B_m \& \neg C_1 \& \dots \& \neg C_n \& \neg \epsilon$. In order to do this we need to somehow pre-compute the effect of such rules. Notice now that the membership of a string w , where $|w| \geq 2$, in $M_G(A)$ depends only on the membership of w in each of $\widehat{M}_G(BC)$, for all BC that appear in the right-hand sides of rules. We can express this dependency directly by a set of rules. In order to do this we treat each BC that appears in the right-hand side of a rule as a Boolean variable (see also [8]).

We start by giving a definition that will play an important role in our subsequent development:

Definition 38. Let G be a Boolean grammar in pre-normal form and let $G_\epsilon = (\Sigma, N \cup \{U\}, P, S)$ be the ϵ -free version of G . Let $X = \{BC \mid A \rightarrow BC \& \neg \epsilon \in P\}$ and let V be a function from X to $\{0, \frac{1}{2}, 1\}$. Then, the extension of G_ϵ with respect to V is the grammar $G_\epsilon^V = (\Sigma, N' \cup \{U\}, P', S)$, which is defined as follows:

- $N' = N \cup \{Q_0, Q_1, Q_{\frac{1}{2}}\}$, where each Q_i represents the language in which all strings have value i .
- P' contains the rules $Q_1 \rightarrow \neg \epsilon$, $Q_1 \rightarrow \epsilon$ and $Q_{\frac{1}{2}} \rightarrow \neg Q_{\frac{1}{2}}$.
- Every rule $A \rightarrow BC \& \neg \epsilon$ in P is replaced in P' by the rule $A \rightarrow Q_{V(BC)} \& \neg \epsilon$.
- All the other rules in P are retained in P' .

Intuitively, in the above definition the non-terminals Q_0 , Q_1 and $Q_{\frac{1}{2}}$ correspond, respectively, to the constant languages \emptyset , Σ^* and the language in which all strings get the value $\frac{1}{2}$. Moreover, G_ϵ^V is a grammar in which every BC has been replaced by a non-terminal that corresponds to one of these constant languages.

It is therefore straightforward to see that given any $w_1, w_2 \in \Sigma^*$ with $|w_1| \geq 2$ and $|w_2| \geq 2$, and any $A \in N$, it holds that $M_{G_\epsilon^V}(A)(w_1) = M_{G_\epsilon^V}(A)(w_2)$. In other words, for every language generated by a non-terminal symbol in G_ϵ^V , one of the following is true:

- All the strings in Σ^* of length at least 2 are included in the language.
- The membership of all strings in Σ^* of length at least 2 in the language is undefined.
- All the strings in Σ^* of length at least 2 are excluded from the language.

This leads to our next definition:

Definition 39. Let G be a Boolean grammar in pre-normal form and let $G_\epsilon = (\Sigma, N \cup \{U\}, P, S)$ be the ϵ -free version of G . Let $X = \{BC \mid A \rightarrow BC \& \neg \epsilon \in P\}$ and let V be a function from X to $\{0, \frac{1}{2}, 1\}$. Then, the extension of V to non-terminal symbols in N is denoted by \widehat{V} and is defined as follows: $\widehat{V}(A) = M_{G_\epsilon^V}(A)(w)$, for any $w \in \Sigma^*$ with $|w| \geq 2$.

As we mentioned in the beginning of this subsection, for every string w with length at least 2, the value in $M_G(A)(w)$ can be computed from the values $\widehat{M}_G(BC)(w)$ for all BC that appear in the right-hand sides of rules. This is the intuition behind the following technical lemma that will be used in the proof of correctness of our final transformation step:

Lemma 40. Let G be a Boolean grammar in pre-normal form and let $G_\epsilon = (\Sigma, N \cup \{U\}, P, S)$ be the ϵ -free version of G . Let $X = \{BC|A \rightarrow BC \& \neg \epsilon \in P\}$ and let $w \in \Sigma^*$ with $|w| \geq 2$. Define the function V from X to $\{0, \frac{1}{2}, 1\}$ such that for all $BC \in X$ it is $V(BC) = \widehat{M_{G_\epsilon}}(BC)(w)$. Then, for all $A \in N$, $M_{G_\epsilon}(A)(w) = \widehat{V}(A)$.

Proof. It suffices to show that for all $A \in N$, $M_{G_\epsilon}(A)(w) = M_{G_\epsilon^V}(A)(w)$. This fact can be proved in two steps, namely that $M_{G_\epsilon^V}(A)(w) \leq_F M_{G_\epsilon}(A)(w)$ and $M_{G_\epsilon}(A)(w) \leq_F M_{G_\epsilon^V}(A)(w)$. We demonstrate the first direction; the second one is similar and omitted.

We therefore prove that $M_{G_\epsilon^V}(A)(w) \leq_F M_{G_\epsilon}(A)(w)$. Suppose for the sake of contradiction that there exists some $A \in N$ such that $M_{G_\epsilon^V}(A)(w) \not\leq_F M_{G_\epsilon}(A)(w)$. Then, there must exist a minimum index $k > 0$ such that there exists $A \in N$ with the following property:

$$M_{G_\epsilon^V, k}(A)(w) \in \{0, 1\} \quad \text{and} \quad M_{G_\epsilon^V, k}(A)(w) \neq M_{G_\epsilon}(A)(w)$$

Define the following sets:

$$\begin{aligned} S_1 &= \{A \in N \mid M_{G_\epsilon^V, k}(A)(w) = 1 \neq M_{G_\epsilon}(A)(w)\} \\ S_0 &= \{A \in N \mid M_{G_\epsilon^V, k}(A)(w) = 0 \neq M_{G_\epsilon}(A)(w)\} \end{aligned}$$

We distinguish the following two cases:

Case 1: $S_1 \neq \emptyset$. Then, for every $A \in S_1$ define $r(A)$ to be the index that satisfies the following property:

$$\left[\Theta_{G_\epsilon^V} \right]_{M_{G_\epsilon^V, k-1}}^{\uparrow r(A)}(A)(w) = 1 \quad \text{and} \quad \left[\Theta_{G_\epsilon^V} \right]_{M_{G_\epsilon^V, k-1}}^{\uparrow r(A)-1}(A)(w) \neq 1$$

Since $\left[\Theta_{G_\epsilon^V} \right]_{M_{G_\epsilon^V, k-1}}^{\uparrow 0}(A)(w) = 0$ and $M_{G_\epsilon^V, k}(A)(w) = 1$, $r(A)$ is well-defined. Choose $A \in S_1$ such that $r(A)$ is minimum. We distinguish the following two subcases:

Subcase 1.1: There exists some rule $A \rightarrow Q_1 \& \neg \epsilon$ in grammar G_ϵ^V . From the definition of G_ϵ^V this implies that there exists a rule of the form $A \rightarrow BC \& \neg \epsilon$ in grammar G_ϵ such that $V(BC) = 1$. From the definition of V we get that $\widehat{M_{G_\epsilon}}(BC)(w) = 1$. This implies that there must exist $m, j > 0$ such that $\left[\Theta_{G_\epsilon} \right]_{M_{G_\epsilon, m-1}}^{\uparrow j}(BC)(w) = 1$ which implies that $\left[\Theta_{G_\epsilon} \right]_{M_{G_\epsilon, m-1}}^{\uparrow j+1}(A)(w) = 1$ and therefore $M_{G_\epsilon}(A)(w) = 1$ (contradiction from our assumption that $A \in S_1$).

Subcase 1.2: There exists a rule $A \rightarrow B_1 \& \dots \& B_b \& \neg C_1 \& \dots \& \neg C_c \& \neg \epsilon$ in grammar G_ϵ^V such that for all $1 \leq i \leq b$, $\left[\Theta_{G_\epsilon^V} \right]_{M_{G_\epsilon^V, k-1}}^{\uparrow r(A)-1}(B_i)(w) = 1$ and for all $1 \leq j \leq c$, $M_{G_\epsilon^V, k-1}(C_j)(w) = 0$. Then, $M_{G_\epsilon^V, k}(B_i)(w) = 1$ and from the minimality of k and of $r(A)$ we have $M_{G_\epsilon}(B_i)(w) = 1$, for all i . Also, $M_{G_\epsilon^V}(C_j)(w) = 0$ and from the minimality of k we have that $M_{G_\epsilon}(C_j)(w) = 0$, for all j . But since M_{G_ϵ} is a model of G_ϵ , this implies that $M_{G_\epsilon}(A)(w) = 1$ (contradiction).

Case 2: $S_1 = \emptyset$, which implies that $S_0 \neq \emptyset$. Then, for every $A \in S_0$, consider the set of rules $\{R_1^A, \dots, R_{n_A}^A\}$ in G_ϵ^V with head A . For every such rule R_i^A there exists a conjunct l_i^A such that one of the following is true:

- $l_i^A = Q_0$, or
- $l_i^A = B$ and $M_{G_\epsilon^V, k-1}(B)(w) = 0$, or
- $l_i^A = \neg C$ and $M_{G_\epsilon^V, k-1}(C)(w) = 1$, or
- $l_i^A \in S_0$, or
- $l_i^A \in \Sigma$.

In the first of the above cases, there exists at least one rule of the form $A \rightarrow BC \& \neg \epsilon$ in G_ϵ such that $V(BC) = 0$. From the definition of V we get that for every such rule it is $\widehat{M_{G_\epsilon}}(BC)(w) = 0$. This implies that there exists a least integer $m_{BC}^A > 0$ such that $(\widehat{M_{G_\epsilon, m_{BC}^A}})(BC)(w) = 0$. Define $m_i^A = \max\{m_{BC}^A \mid A \rightarrow BC \& \neg \epsilon \in P, \widehat{M_{G_\epsilon}}(BC)(w) = 0\}$. In the second case, using the minimality of k we get that $M_{G_\epsilon}(B)(w) = 0$, which implies that there exists a least integer $m_i^A > 0$ such that $M_{G_\epsilon, m_i^A}(B)(w) = 0$. In the third case, using the minimality of k we get that $M_{G_\epsilon}(C)(w) = 1$, which implies that there exists a least integer $m_i^A > 0$ such that $M_{G_\epsilon, m_i^A}(C)(w) = 1$. Finally, in the last two cases, let us take $m_i^A = 0$. Now, define $m = \max\{m_i^A \mid A \in S_0, 1 \leq i \leq n_A\} + 1$.

We will demonstrate that for every $A \in S_0$, it is $M_{G_\epsilon, m}(A)(w) = 0$, which will immediately lead us to the contradiction that $M_{G_\epsilon}(A)(w) = 0$. Consider an arbitrary $A \in S_0$. Then for every rule R defining A in G_ϵ , there exists a corresponding rule R_i^A in G_ϵ^V ; moreover R contains a literal l_R that corresponds to l_i^A . More specifically, if R is contained in G_ϵ^V , i.e., it is $R = R_i^A$, then

$l_R = l_i^A$; otherwise R is of the form $A \rightarrow BC \& \neg \epsilon$ and $l_R = BC$. We claim that in the latter case $\widehat{M}_{G_\epsilon}(BC)(w) = 0$. In order to prove this claim, suppose (for the sake of contradiction) that $\widehat{M}_{G_\epsilon}(BC)(w) \neq 0$. Then, either $A \rightarrow Q_1$ or $A \rightarrow Q_{\frac{1}{2}}$ is a rule in G_ϵ^V which implies that $M_{G_\epsilon^V}(A)(w) \neq 0$ (contradiction). Therefore, in this case the corresponding rule of R in G_ϵ^V is $A \rightarrow Q_0$.

We now show by induction that for every $n \geq 0$ and for every $A \in S_0$ it is: $[\Theta_{G_\epsilon}]_{M_{G_\epsilon, m-1}}^{\uparrow n}(A)(w) = 0$. The basis case is trivial. Assume the result holds for n ; we demonstrate it for $n+1$. Consider any rule R in G_ϵ with head A . If $l_R \in S_0$, then from the induction hypothesis it is $[\Theta_{G_\epsilon}]_{M_{G_\epsilon, m-1}}^{\uparrow n}(l_R)(w) = 0$. If $l_R \in (N - S_0) \cup \Sigma$ or $l_R = BC$ (where $B, C \in N$), then from the definition of m it holds $\widehat{M}_{G_\epsilon, m}(l_R)(w) = 0$, which implies that $([\Theta_{G_\epsilon}]_{M_{G_\epsilon, m-1}}^{\uparrow n})(l_R)(w) = 0$. Finally, if $l_R = \neg C$ (where $C \in N$), then from the definition of m it holds $\widehat{M}_{G_\epsilon, m-1}(l_R)(w) = 0$. Therefore, $[\Theta_{G_\epsilon}]_{M_{G_\epsilon, m-1}}^{\uparrow n+1}(A)(w) = 0$, which completes the inductive step. Therefore, $M_{G_\epsilon, m}(A)(w) = 0$ which implies that $M_{G_\epsilon}(A)(w) = 0$ (contradicting our assumption that $A \in S_0$). \square

Given a non-empty set X , the functions from X to $\{0, \frac{1}{2}, 1\}$ can be ordered by the degree of information they contain (assuming that the value $\frac{1}{2}$ contains no information). The minimal and maximal functions with respect to this ordering will play an important role in the construction of the binary normal form of a given grammar G .

Definition 41. Let X be a non-empty set and let V, W be functions from X to $\{0, \frac{1}{2}, 1\}$. We denote by V_i the set $\{x \in X \mid V(x) = i\}$. We write $V \sqsubseteq_F W$ if $V_0 \subseteq W_0$ and $V_1 \subseteq W_1$.

The following lemma states that the extensions of functions of the above form to non-terminal symbols, respects the above ordering. The proof of the lemma is straightforward:

Lemma 42. Let G be a Boolean grammar in pre-normal form and let $G_\epsilon = (\Sigma, N, P, S)$ be its ϵ -free version. Moreover, let $X = \{BC \mid A \rightarrow BC \& \neg \epsilon \in P\}$ and let V, W be functions from X to $\{0, \frac{1}{2}, 1\}$ such that $V \sqsubseteq_F W$. Then, $\widehat{V}(A) \leq_F \widehat{W}(A)$ for every $A \in N$.

Using all the above, we can now define the transformation that brings a Boolean grammar into normal form:

Definition 43. Let G be a Boolean grammar in pre-normal form and let $G_\epsilon = (\Sigma, N \cup \{U\}, P, S)$ be the ϵ -free version of G . Let $X = \{BC \mid A \rightarrow BC \& \neg \epsilon \in P\}$ and let \mathcal{V} be the set of all functions from X to $\{0, \frac{1}{2}, 1\}$. The normal form $G_n = (\Sigma, N \cup \{U, T\}, P', S)$ of G is the grammar obtained from G_ϵ as follows:

- P' contains all the rules in P of the form $A \rightarrow a$ and $A \rightarrow a \& U$, where $a \in \Sigma$, the rule $U \rightarrow \neg U$ in P and the rule $T \rightarrow \neg \epsilon$, where $T \notin N$ is a special symbol which represents the set in which all non-empty strings have value 1.
- For every $A \in N$ let $\mathcal{T}_A = \{V \in \mathcal{V} \mid \widehat{V}(A) = 1\}$. For every minimal (with respect to \sqsubseteq_F) element V of \mathcal{T}_A , P' contains the rule:

$$A \rightarrow x_1 \& \dots \& x_k \& \neg y_1 \& \dots \& \neg y_m \& TT$$

where $\{x_1, \dots, x_k\} = V_1$ and $\{y_1, \dots, y_m\} = V_0$.

- For every $A \in N$ let $\mathcal{U}_A = \{V \in \mathcal{V} \mid \widehat{V}(A) = \frac{1}{2}\}$. For every maximal (with respect to \sqsubseteq_F) element V of \mathcal{U}_A , P' contains the rule:

$$A \rightarrow x_1 \& \dots \& x_k \& \neg y_1 \& \dots \& \neg y_m \& z_1 \& \neg z_1 \& \dots \& z_r \& \neg z_r \& TT \& U$$

where $\{x_1, \dots, x_k\} = V_1$, $\{y_1, \dots, y_m\} = V_0$ and $\{z_1, \dots, z_r\} = V_{\frac{1}{2}}$.

Notice that in the former case we consider only minimal elements, because if $V' \sqsubseteq_F V$ and $\widehat{V'}(A) = 1$ then $\widehat{V}(A) = 1$. Similarly in the latter case we consider only maximal elements, because if $V' \sqsubseteq_F V$ and $\widehat{V}(A) = \frac{1}{2}$ then $\widehat{V'}(A) = \frac{1}{2}$. These ideas are formalized by the proof of the following lemma.

Lemma 44. Let G be a Boolean grammar in pre-normal form, let $G_\epsilon = (\Sigma, N \cup \{U\}, P, S)$ be its ϵ -free form and let $G_n = (\Sigma, N \cup \{U, T\}, P', S)$ be its binary normal form. Then, for every $A \in N$ and for every $w \in \Sigma^+$, $M_G(A)(w) = M_{G_n}(A)(w)$.

Proof. Let $X = \{BC \mid A \rightarrow BC \& \neg \epsilon \in P\}$. We prove by induction on the length of w that for every $A \in N$ and for every $w \in \Sigma^*$, $M_{G_\epsilon}(A)(w) = M_{G_n}(A)(w)$. Then, the lemma follows from Lemma 37. For $w = \epsilon$, it holds $M_{G_\epsilon}(A)(w) = M_{G_n}(A)(w) = 0$, since every rule of G_ϵ with head in N contains a conjunct that is either a terminal symbol in Σ or $\neg \epsilon$ and every rule of G_n with head in N contains a conjunct that is either a terminal symbol in Σ or T .

Moreover, if $|w| = 1$ the statement follows easily due to the fact that in G_ϵ all the information regarding strings of length 1 is produced by simple rules (i.e., rules that have been introduced during the construction of the direct form of G); moreover, G_n contains these same rules regarding strings of length 1 while all its other rules concern strings of length 2 or more, since

they contain the conjunct TT . Assume now that the statement holds for all w of length less than or equal to n for some $n \geq 1$; we demonstrate the case for $n + 1$. In particular, we show that for every $v \in \{1, \frac{1}{2}\}$, $M_{G_\epsilon}(A)(w) = v$ if and only if $M_{G_n}(A)(w) = v$.

Consider a $w \in \Sigma^*$ with $|w| \geq 2$ and define function V as follows: $V(BC) = M_{G_\epsilon}(BC)(w)$, for all $BC \in X$. From Lemma 40, $\widehat{V}(A) = M_{G_\epsilon}(A)(w)$.

We first prove that $M_{G_\epsilon}(A)(w) = 1$ if and only if $M_{G_n}(A)(w) = 1$. We examine the two directions of the statement. For the left-to-right direction, assume that $M_{G_\epsilon}(A)(w) = 1$. Then, it is also $\widehat{V}(A) = 1$. Consider a minimal function V' with respect to \sqsubseteq_F , such that $\widehat{V}'(A) = 1$ and $V' \sqsubseteq_F V$. By construction, in G_n there exists a rule of the form: $A \rightarrow x_1 \& \dots \& x_k \& \neg y_1 \& \dots \& \neg y_m \& TT$ where $\{x_1, \dots, x_k\} = V'_1 \subseteq V_1$ and $\{y_1, \dots, y_m\} = V'_0 \subseteq V_0$. Therefore, for all $1 \leq i \leq k$, $\widehat{M}_{G_\epsilon}(x_i)(w) = 1$ and for all $1 \leq j \leq m$, $\widehat{M}_{G_\epsilon}(y_j)(w) = 0$. From the induction hypothesis and the fact that for all $D \in N$ it holds that $M_{G_\epsilon}(D)(\epsilon) = M_{G_n}(D)(\epsilon) = 0$, we get that for all $1 \leq i \leq k$, $\widehat{M}_{G_n}(x_i)(w) = 1$ and for all $1 \leq j \leq m$, $\widehat{M}_{G_n}(y_j)(w) = 0$. But this implies that $M_{G_n}(A)(w) = 1$ (since M_{G_n} is a model of G_n).

In order to prove the right-to-left direction of the statement, assume that $M_{G_n}(A)(w) = 1$. This implies that in G_n there exists a rule of the form $A \rightarrow x_1 \& \dots \& x_k \& \neg y_1 \& \dots \& \neg y_m \& TT$ such that for all $1 \leq i \leq k$, $\widehat{M}_{G_n}(x_i)(w) = 1$, and for all $1 \leq j \leq m$, $\widehat{M}_{G_n}(y_j)(w) = 0$. From the induction hypothesis and the fact that $M_{G_\epsilon}(D)(\epsilon) = M_{G_n}(D)(\epsilon) = 0$ for every $D \in N$, we get that for all $1 \leq i \leq k$, $\widehat{M}_{G_\epsilon}(x_i)(w) = \widehat{V}(x_i) = 1$ and for all $1 \leq j \leq m$, $\widehat{M}_{G_\epsilon}(y_j)(w) = \widehat{V}(y_j) = 0$. Notice now that the existence of the rule $A \rightarrow x_1 \& \dots \& x_k \& \neg y_1 \& \dots \& \neg y_m \& TT$ in G_n implies that there exists a function V' such that for all $1 \leq i \leq k$ and for all $1 \leq j \leq m$, $V'(x_i) = 1$, $V'(y_j) = 0$, and for every $z \in X$ with $z \neq x_i$ and $z \neq y_j$, $V'(z) = \frac{1}{2}$; additionally, $\widehat{V}'(A) = 1$. From the first three properties of V' , we get that $V' \sqsubseteq_F V$. Using Lemma 42, we obtain that $\widehat{V}(A) = 1$, which implies that $M_{G_\epsilon}(A)(w) = 1$.

We now prove that $M_{G_\epsilon}(A)(w) = \frac{1}{2}$ if and only if $M_{G_n}(A)(w) = \frac{1}{2}$. For the left-to-right direction, assume that $M_{G_\epsilon}(A)(w) = \frac{1}{2}$. Then, it is also $\widehat{V}(A) = \frac{1}{2}$. Consider a maximal function V' with respect to \sqsubseteq_F , such that $\widehat{V}'(A) = \frac{1}{2}$ and $V \sqsubseteq_F V'$. By construction, in G_n there exists a rule:

$$A \rightarrow x_1 \& \dots \& x_k \& \neg y_1 \& \dots \& \neg y_m \& z_1 \& \neg z_1 \& \dots \& z_r \& \neg z_r \& TT \& U$$

such that $\{x_1, \dots, x_k\} = V'_1$, $\{y_1, \dots, y_m\} = V'_0$ and $\{z_1, \dots, z_r\} = V'_1$. Since $V \sqsubseteq_F V'$ we have that $V_1 \subseteq \{x_1, \dots, x_k\} \subseteq V_1 \cup V_{\frac{1}{2}}$, $V_0 \subseteq \{y_1, \dots, y_m\} \subseteq V_0 \cup V_{\frac{1}{2}}$ and $\{z_1, \dots, z_r\} \subseteq V_{\frac{1}{2}}$. This means that for all $1 \leq i \leq k$, $\widehat{M}_{G_\epsilon}(x_i)(w) \geq \frac{1}{2}$, for all $1 \leq j \leq m$, $\widehat{M}_{G_\epsilon}(y_j)(w) \leq \frac{1}{2}$ and for all $1 \leq l \leq r$, $\widehat{M}_{G_\epsilon}(z_l)(w) = \frac{1}{2}$. From the induction hypothesis and the fact that for all $D \in N$ it holds that $M_{G_\epsilon}(D)(\epsilon) = M_{G_n}(D)(\epsilon) = 0$, we get that for all $1 \leq i \leq k$, $\widehat{M}_{G_n}(x_i)(w) \geq \frac{1}{2}$, for all $1 \leq j \leq m$, $\widehat{M}_{G_n}(y_j)(w) \leq \frac{1}{2}$ and for all $1 \leq l \leq r$, $\widehat{M}_{G_n}(z_l)(w) = \frac{1}{2}$. Since M_{G_n} is a model of G_n , we obtain that $M_{G_n}(A)(w) \geq \frac{1}{2}$. Notice now that it cannot be $M_{G_n}(A)(w) = 1$: we have shown that this would imply $M_{G_\epsilon}(A)(w) = 1$, which is a contradiction. Therefore, $M_{G_n}(A)(w) = \frac{1}{2}$.

Now, consider the right-to-left direction of the statement, i.e., assume that $M_{G_n}(A)(w) = \frac{1}{2}$. We have to distinguish the following two cases:

Case 1: There exists in G_n a rule of the form:

$$A \rightarrow x_1 \& \dots \& x_k \& \neg y_1 \& \dots \& \neg y_m \& z_1 \& \neg z_1 \& \dots \& z_r \& \neg z_r \& TT \& U$$

such that for all $1 \leq i \leq k$, $\widehat{M}_{G_n}(x_i)(w) \geq \frac{1}{2}$, for all $1 \leq j \leq m$, $\widehat{M}_{G_n}(y_j)(w) \leq \frac{1}{2}$ and for all $1 \leq l \leq r$, $\widehat{M}_{G_n}(z_l)(w) = \frac{1}{2}$. From the induction hypothesis and the fact that $M_{G_\epsilon}(D)(\epsilon) = M_{G_n}(D)(\epsilon) = 0$ for every $D \in N$, we get that for all $1 \leq i \leq k$ it is $\widehat{M}_{G_\epsilon}(x_i)(w) = \widehat{V}(x_i) \geq \frac{1}{2}$, for all $1 \leq j \leq m$ it is $\widehat{M}_{G_\epsilon}(y_j)(w) = \widehat{V}(y_j) \leq \frac{1}{2}$ and for all $1 \leq l \leq r$ it is $\widehat{M}_{G_\epsilon}(z_l)(w) = \widehat{V}(z_l) = \frac{1}{2}$. Notice now that the existence of the above rule for A in G_n implies that there exists a function V' such that for all $1 \leq i \leq k$ it is $V'(x_i) = 1$, for all $1 \leq j \leq m$ it is $V'(y_j) = 0$, for all $1 \leq l \leq r$ it is $V'(z_l) = \frac{1}{2}$, and additionally, $\widehat{V}'(A) = \frac{1}{2}$. From the first three properties of V' , we get that $V \sqsubseteq_F V'$. Using Lemma 42, we obtain that $\widehat{V}(A) = \frac{1}{2}$ and therefore that $M_{G_\epsilon}(A)(w) = \frac{1}{2}$.

Case 2: There exists in G_n a rule of the form:

$$A \rightarrow x_1 \& \dots \& x_k \& \neg y_1 \& \dots \& \neg y_m \& TT$$

such that for all $1 \leq i \leq k$ it is $\widehat{M}_{G_n}(x_i)(w) \geq \frac{1}{2}$, for all $1 \leq j \leq m$ it is $\widehat{M}_{G_n}(y_j)(w) \leq \frac{1}{2}$ and there exists either some i , $1 \leq i \leq k$ such that $\widehat{M}_{G_n}(x_i)(w) = \frac{1}{2}$ or some j , $1 \leq j \leq m$ such that $\widehat{M}_{G_n}(y_j)(w) = \frac{1}{2}$. From the induction hypothesis and the fact that $M_{G_\epsilon}(D)(\epsilon) = M_{G_n}(D)(\epsilon) = 0$ for every $D \in N$, we get that for all $1 \leq i \leq k$ it is $\widehat{M}_{G_\epsilon}(x_i)(w) = \widehat{V}(x_i) \geq \frac{1}{2}$, for all $1 \leq j \leq m$ it is $\widehat{M}_{G_\epsilon}(y_j)(w) = \widehat{V}(y_j) \leq \frac{1}{2}$ and there exists either some i such that $\widehat{M}_{G_\epsilon}(x_i)(w) = \widehat{V}(x_i) = \frac{1}{2}$ or some j such that $\widehat{M}_{G_\epsilon}(y_j)(w) = \widehat{V}(y_j) = \frac{1}{2}$.

Notice now that the existence of rule $A \rightarrow x_1 \& \dots \& x_k \& \neg y_1 \& \dots \& \neg y_m \& TT$ in G_n implies that there exists a function V' such that $V'_1 = \{x_1, \dots, x_k\}$, $V'_0 = \{y_1, \dots, y_m\}$ and additionally, V' is a minimal function with respect to \sqsubseteq_F with the property $\widehat{V}'(A) = 1$. Now, define V^- so that $V_1^- = V_1 \cap V'_1$ and $V_0^- = V_0 \cap V'_0$. Also define V^+ so that $V_1^+ = V_1 \cup V'_1$ and $V_0^+ = V_0 \cup V'_0$. Using the properties of V and V' it is easy to check that $V_1^+ \cap V_0^+ = \emptyset$, that is, V^+ is well-defined. Obviously,

$V^- \sqsubseteq_F V' \sqsubseteq_F V^+$. Thus, from Lemma 42, we obtain that $\widehat{V}^+(A) = 1$. Furthermore, $V^- \neq V'$, since from the definition of V there exists either some i , $1 \leq i \leq k$, such that $V(x_i) = \frac{1}{2}$ or some j , $1 \leq j \leq m$, such that $V(y_j) = \frac{1}{2}$. From the minimality property of V we get that $\widehat{V}^-(A) \neq 1$. Thus, from Lemma 42 we obtain that $\widehat{V}^-(A) = \frac{1}{2}$. Moreover, $V^- \sqsubseteq_F V \sqsubseteq_F V^+$, which implies that $\widehat{V}(A) = M_{G_\epsilon}(A)(w) \in \{\frac{1}{2}, 1\}$. However, it cannot be $M_{G_\epsilon}(A)(w) = 1$, since we have shown that this would imply $M_{G_n}(A)(w) = 1$, which is a contradiction. Therefore, $M_{G_\epsilon}(A)(w) = \frac{1}{2}$. \square

Given the above lemmas, a simple step remains in order to reach the statement of Theorem 30: if in the original grammar G it is $M_G(S)(\epsilon) \neq 0$, then a rule of the form $S \rightarrow \epsilon$ or $S \rightarrow \epsilon \& U$ is added to the grammar that has resulted after the processing implied by all the above lemmas. The resulting grammar is then in binary normal form and defines the same language as the initial one.

6. Parsing under the well-founded semantics

We next present an algorithm that computes the truth value of the membership of an input string $w \neq \epsilon$ in the language defined by a grammar G , which is assumed to be in binary normal form. The algorithm computes the value of $M_G(A)(u)$ for every non-terminal symbol A and every substring u of w in a bottom-up manner. It uses two matrices M and Q to keep the appropriate intermediate values that are needed for the computation. Suppose that the input string is $w = a_1 \cdots a_n$. Then $M[A, i, j]$ keeps the value $M_G(A)(a_i \cdots a_j)$ and $Q[B, C, i, j]$ keeps the value $M_G(BC)(a_i \cdots a_j)$. By convention $\min_{i=1}^0 v_i = 1$.

Algorithm for parsing under $G = (\Sigma, N, P, S)$

Input: string $w = a_1 \cdots a_n \in \Sigma^+$

Initialization step:

```
for i := 1 to n do begin
  for every A ∈ N do
    if there exists a rule A → a_i then M[A, i, i] := 1
    else if there exists a rule A → a_i & U then M[A, i, i] := 1/2
    else M[A, i, i] := 0
end
```

Main loop:

```
for d := 2 to n do
  for i := 1 to n - d + 1 do begin
    j := i + d - 1
    for every B, C ∈ N such that BC appears in the right-hand side of a rule do
      Q[B, C, i, j] := max_{ℓ=i}^{j-1} min{M[B, i, ℓ], M[C, ℓ + 1, j]}
    for every A ∈ N do M[A, i, j] := 0
    for every rule A → B_1 C_1 & ... & B_m C_m & ¬D_1 E_1 & ... & ¬D_r E_r & TT & U do begin
      v := min{1/2, min_{p=1}^m Q[B_p, C_p, i, j], min_{q=1}^r (1 - Q[D_q, E_q, i, j])}
      if v > M[A, i, j] then M[A, i, j] := v
    end
    for every rule A → B_1 C_1 & ... & B_m C_m & ¬D_1 E_1 & ... & ¬D_r E_r & TT do begin
      v := min{min_{p=1}^m Q[B_p, C_p, i, j], min_{q=1}^r (1 - Q[D_q, E_q, i, j])}
      if v > M[A, i, j] then M[A, i, j] := v
    end
  end
end
return M[S, 1, n]
```

The correctness of the above algorithm is established by the following theorem:

Theorem 45. *Let $G = (\Sigma, N, P, S)$ be a fixed Boolean grammar. Then, for every string $w = a_1 \cdots a_n \in \Sigma^+$, the above algorithm computes the correct value $M_G(A)(w)$, in time $\mathcal{O}(n^3)$.*

Proof. In order to verify the correctness of the algorithm, we will prove that after the termination of the main loop, for every $A \in N$ and for every i, j , with $1 \leq i \leq j \leq n$, $M[A, i, j] = M_G(A)(a_i \cdots a_j)$. Observe that, for every i, j , if $i = j$ then the value $M[A, i, j]$ is determined in the initialization step and does not change in the main loop; if $i < j$ then the value $M[A, i, j]$ is determined in the iteration of the main loop in which $d = j - i + 1$ and does not change in the next iterations.

We will prove that $M[A, i, j] = M_G(A)(a_i \cdots a_j)$, by induction on the length k of $a_i \cdots a_j$. For the basis case, suppose that $k = 1$, that is, $i = j$.

We first show that $M[A, i, i] = 1$ if and only if $M_G(A)(a_i) = 1$. Suppose that $M[A, i, i] = 1$. Then there exists a rule $A \rightarrow a_i$ in P , which immediately implies that $M_G(A)(a_i) = 1$.

Conversely, suppose that $M_G(A)(a_i) = 1$. This value cannot be obtained by a rule containing the conjunct TT , and therefore it is obtained by a rule $A \rightarrow a_i$. But in this case the algorithm sets $M[A, i, i] = 1$ in its initialization step.

It remains to show that $M[A, i, i] = \frac{1}{2}$ if and only if $M_G(A)(a_i) = \frac{1}{2}$. Suppose that $M[A, i, i] = \frac{1}{2}$. Then there exists a rule $A \rightarrow a_i \& U$ in P . This implies that $M_G(A)(a_i) \neq 0$. Also, it cannot be $M_G(A)(a_i) = 1$, since in this case we would have $M[A, i, i] = 1$. Therefore, $M_G(A)(a_i) = \frac{1}{2}$.

Conversely, suppose that $M_G(A)(a_i) = \frac{1}{2}$. Obviously P does not contain the rule $A \rightarrow a_i$. We claim that P contains the rule $A \rightarrow a_i \& U$. Suppose, for the sake of contradiction, that our claim is not true. Then every rule in P with head A , contains in its body either conjunct TT or some conjunct $b \in \Sigma$ with $b \neq a_i$. This implies that $M_G(A)(a_i) = 0$ (contradiction). Therefore, P contains the rule $A \rightarrow a_i \& U$ and the algorithm sets $M[A, i, i] = \frac{1}{2}$ in its initialization step.

Suppose now that $M[A, i, j] = M_G(A)(a_i \cdots a_j)$ holds for every $A \in N$ and for all i, j with $j - i + 1 \leq k$ (i.e., for all substrings of w of length at most k).

Consider a substring $a_i \cdots a_j$ of w of length $k + 1$ (i.e., $j - i + 1 = k + 1$). The value of $M[A, i, j]$ is determined in the iteration of the main loop in which $d = k + 1$. Furthermore, at this point the values of $M[B, i, \ell]$ and $M[C, \ell + 1, j]$ have already been computed, for every $B, C \in N$ and for every ℓ such that $i \leq \ell < j$ (since $\ell - i + 1 \leq k$ and $j - (\ell + 1) + 1 \leq k$).

From the induction hypothesis $M[B, i, \ell] = M_G(B)(a_i \cdots a_\ell)$ and $M[C, \ell + 1, j] = M_G(C)(a_{\ell+1} \cdots a_j)$. This implies (using also the fact that $M_G(B)(\epsilon) = M_G(C)(\epsilon) = 0$) that $Q[B, C, i, j] = \widehat{M}_G(BC)(a_i \cdots a_j)$.

Now it is easy to prove that $M[A, i, j] = 1$ if and only if $M_G(A)(a_i \cdots a_j) = 1$ and $M[A, i, j] = \frac{1}{2}$ if and only if $M_G(A)(a_i \cdots a_j) = \frac{1}{2}$. We give a detailed proof only for the one direction of the first argument. The remaining parts of the proof are very similar.

Suppose that $M[A, i, j] = 1$. Then there exists a rule

$$A \rightarrow B_1 C_1 \& \dots \& B_m C_m \& \neg D_1 E_1 \& \dots \& \neg D_r E_r \& TT$$

in P such that $Q[B_p, C_p, i, j] = 1$, for $1 \leq p \leq m$ and $Q[D_q, E_q, i, j] = 0$, for $1 \leq q \leq r$. This implies that $\widehat{M}_G(B_p C_p)(a_i \cdots a_j) = 1$, for $1 \leq p \leq m$ and $\widehat{M}_G(\neg D_q E_q)(a_i \cdots a_j) = 1$, for $1 \leq q \leq r$. Since M_G is a model of G , we have $M_G(A)(a_i \cdots a_j) = 1$.

Therefore, for every $A \in N$, and for every i, j with $1 \leq i \leq j \leq n$ it is $M[A, i, j] = M_G(A)(a_i \cdots a_j)$. In particular $M[S, 1, n] = M_G(S)(a_1 \cdots a_n)$, that is, the algorithm is correct.

We now show that the above algorithm runs in time $\mathcal{O}(n^3)$. The initialization step performs n iterations, each requiring time which is independent of the input, and depends only on the grammar. Therefore the initialization step requires time $\mathcal{O}(n)$.

The main loop is a nested-loop that performs $\mathcal{O}(n^2)$ iterations. In each iteration the computation of $Q[B, C, i, j]$ requires time $\mathcal{O}(n)$, while all the remaining tasks require time which is independent of the input. Therefore, the main loop requires time $\mathcal{O}(n^3)$, which dominates the running time of the algorithm. \square

7. Conclusions

We have presented a novel semantics for Boolean grammars which has been inspired by techniques that have been developed in the logic programming domain. Under this new semantics every Boolean grammar has a distinguished (three-valued) model that satisfies its rules. Moreover, we have shown that this language is the least fixed-point of an appropriate operator that is associated with the grammar. Finally, we have demonstrated that every Boolean grammar can be transformed into an equivalent one in a binary normal form. For grammars in this normal form, we have derived an $\mathcal{O}(n^3)$ parsing algorithm.

We believe that the well-founded semantics will prove to be a useful tool for the further development of the theory of Boolean grammars. In particular, two of the authors have already used the well-founded approach in order to prove that the locally stratified construction is well-defined (see [5] for details). Also, it is expected that the well-founded semantics and its corresponding parsing algorithm can form the basis of general implementations of Boolean grammars. On the more theoretical side, the formal machinery behind the well-founded semantics can help to the further development of many-valued formal language theory (see for example [2]).

It should be noted that it is possible that the well-founded model M_G of a grammar G could also be obtained following slightly different constructions. For logic programs one such construction that is based on an infinite-valued logic, has recently been proposed in [11]. Adapting the technique of [11] to Boolean grammars would most probably require the introduction of infinite-valued formal languages. This is probably an interesting venue for further research.

Closing, we would like to express our strong belief that a further investigation of the connections between formal language theory and the theory of logic programming will prove to be very rewarding.

Acknowledgments

We would like to thank the anonymous reviewers for their detailed and insightful comments.

References

- [1] K. Apt, R. Bol, Logic programming and negation: a survey, *Journal of Logic Programming*, 19, 20 (1994) 9–71.
- [2] Z. Esik, W. Kuich, Boolean fuzzy sets, *International Journal of Foundations of Computer Science* 18 (6) (2007) 1197–1207.
- [3] H.B. Enderton, *A Mathematical Introduction to Logic*, Academic Press, 1972.
- [4] V. Kountouriotis, Ch. Nomikos, P. Rondogiannis, Well-founded semantics for Boolean grammars, in: *Tenth International Conference on Developments in Language Theory (DLT)*, 2006, pp. 203–214.
- [5] Ch. Nomikos, P. Rondogiannis, Locally stratified Boolean grammars, in: *First International Conference on Language and Automata Theory and Applications (LATA)*, 2007, pp. 437–447.
- [6] Ch. Nomikos, P. Rondogiannis, Locally stratified Boolean grammars, *Information and Computation* 206 (9–10) (2008) 1219–1233.
- [7] A. Okhotin, Conjunctive grammars, *Journal of Automata, Languages and Combinatorics* 6 (4) (2001) 519–535.
- [8] A. Okhotin, Boolean grammars, *Information and Computation* 194 (1) (2004) 19–48.
- [9] H. Przymusinska, T. Przymusinski, Semantic issues in deductive databases and logic programs, in: R. Banerji (Ed.), *Formal Techniques in Artificial Intelligence: A Source-Book*, North Holland, 1990, pp. 321–367.
- [10] T.C. Przymusinski, Every logic program has a natural stratification and an iterated fixed-point model, in: *Proceedings of the Eighth Symposium on Principles of Database Systems ACM SIGACT–SIGMOD*, 1989, pp. 11–21.
- [11] P. Rondogiannis, W.W. Wadge, Minimum model semantics for logic programs with negation-as-failure, *ACM Transactions on Computational Logic* 6 (2) (2005) 441–467.
- [12] J.E. Stoy, *Denotational Semantics: the Scott–Strachey Approach to Programming Language Theory*, The MIT Press, 1977.
- [13] A. van Gelder, K.A. Ross, J.S. Schlipf, The well-founded semantics for general logic programs, *Journal of the ACM* 38 (3) (1991) 620–650.
- [14] M. Wrona, Stratified Boolean grammars, in: *International Symposium on the Mathematical Foundations of Computer Science (MFCS)*, 2005, pp. 801–812.