Locally stratified Boolean grammars

Christos Nomikos\textsuperscript{a,}* \textsuperscript{a}, Panos Rondogiannis\textsuperscript{b}

\textsuperscript{a} Department of Computer Science, University of Ioannina, P.O. Box 1186, 45 110 Ioannina, Greece
\textsuperscript{b} Department of Informatics & Telecommunications, University of Athens, Panepistimiopolis, 157 84 Athens, Greece

\section*{Article info}

Article history:
Received 29 June 2007
Revised 16 February 2008
Available online 7 June 2008

\section*{Abstract}

We introduce \textit{locally stratified Boolean grammars}, a natural subclass of Boolean grammars with many desirable properties. Informally, if a grammar is locally stratified then the set of all pairs of the form (\textit{nonterminal, string}) of the grammar can be mapped to a (possibly infinite) set of strata so as that the following holds: if the membership of a string $w'$ in the language defined by nonterminal $A$ depends on the membership of string $w$ in the language defined by nonterminal $B$, then $(B, w')$ cannot belong to a stratum higher than the stratum of $(A, w)$; furthermore, if the above dependency is obtained through negation, $(B, w')$ must belong to a stratum lower than the stratum of $(A, w)$. We prove that local stratifiability can be tested in linear time with respect to the size of the given grammar. We then develop the semantics of locally stratified grammars and prove that it is independent of the choice of the stratification mapping. We argue that the class of locally stratified Boolean grammars appears at present to be the broadest subclass of Boolean grammars that can be given a classical semantics (i.e., without resorting to three-valued formal language theory).

© 2008 Elsevier Inc. All rights reserved.

\section*{1. Introduction}

Boolean grammars [10] is a recent extension of context-free grammars which allows conjunction and negation to appear in the right hand side of rules. This seemingly innocent extension of context-free grammars, unexpectedly created a number of interesting and non-trivial theoretical questions (which nowadays remain largely unanswered, see [11] for details). As an example of the power of the new formalism, it has been demonstrated that Boolean grammars can easily express certain languages that are notoriously non context-free [10]. Even the negation-free Boolean grammars (namely the conjunctive ones) appear to be very interesting from an expressibility point of view: it has recently been demonstrated in [4] (extending the results of [3]) that there is no recursive function that can bound the growth rate of unary conjunctive languages (when these are viewed as increasing sequences of natural numbers).

From the more practical side, it has been demonstrated in [10] that Boolean grammars can be parsed efficiently (actually, in time $O(n^3)$). This fact renders Boolean grammars a promising alternative to the traditional formalisms that are currently used for the syntax analysis phase of compiler construction. However, despite their syntactic simplicity, Boolean grammars have also been proved to be non-trivial from a semantic point of view. In particular, the use of negation makes it difficult to define a simple derivation-style semantics (such as the well-known one that is being widely used in the case of context-free grammars). For example, it is not immediately obvious whether a grammar of the form $S \rightarrow ¬ S$ has any meaning at all.

The study of the semantics of Boolean grammars was initiated by A. Okhotin in [10], in which two different approaches are proposed. Given a Boolean grammar $G$, the \textit{unique solution semantics} and the \textit{naturally reachable semantics} are based on
identifying a solution of a system of language equations that is associated to \( G \). Both of these semantics exhibit undesirable behavior in certain cases (see [5] for a detailed discussion on this issue). Moreover, as discussed in [10], given a Boolean grammar one cannot effectively determine whether the grammar complies to any of these two semantics. More recently, based on well-known ideas from logic programming [16], Koutouriotis et al. proposed the well-founded semantics of Boolean grammars [5]. This latter approach requires the study of three-valued languages, i.e., languages in which the membership of a string can be classified as true, false or unknown. The advantage of this approach is that it applies to all Boolean grammars, independently of their syntax. The generality of this approach is due to the use of a three-valued formal language theory as the underlying mathematical machinery.

Another direction of research which was initiated by M. Wrona [17] seeks to find subclasses of Boolean grammars that are well-behaved both semantically and from an application point of view. In other words, we are seeking subclasses that are as broad as possible, and which possess a classical semantics (i.e., semantics based on ordinary formal language theory). More specifically, in [17] the class of stratified Boolean grammars is defined and its properties are investigated. The notion of stratification has its roots in the area of logic programming (see for example [1]). Intuitively, a Boolean grammar is stratified if its nonterminals are not defined in a circular way that “passes through” negation.

In this paper, motivated again by ideas in logic programming [13], we introduce the locally stratified Boolean grammars which form a proper superset of the stratified ones. Informally, if a grammar is locally stratified then the set of all pairs of the form \((\text{nonterminal}, \text{string})\) of the grammar can be mapped to a (possibly infinite) set of strata so as that the following holds: if the membership of a string \( w \) in the language defined by nonterminal \( A \) depends on the membership of string \( w' \) in the language defined by nonterminal \( B \), then \( (B,w') \) cannot belong to a stratum higher than the stratum of \( (A,w) \); furthermore, if the above dependency is obtained through negation, \( (B,w') \) must belong to a stratum lower than the stratum of \( (A,w) \). We demonstrate that the property of local stratifiability can be tested in linear time with respect to the size of the grammar under consideration. This is a surprising fact because local stratifiability in logic programming is undecidable (more specifically \( \Pi_1 \)-complete [2]). We then develop the semantics of locally stratified Boolean grammars and demonstrate that it is independent of the choice of the stratification mapping. The independence proof is based on the well-founded construction of [5]. More specifically, we show that the locally stratified semantics of a Boolean grammar, constructed using an arbitrary proper partition into strata, coincides with the well-founded model of the grammar (which is unique by definition). It should be noted at this point that the locally stratified semantics proceeds in a different way than the well-founded semantics. In other words, the locally stratified construction appears to give an interesting alternative to the well-founded one. We demonstrate the potential of the new approach by computing the meaning of certain simple but interesting Boolean grammars, and show that such a computation usually involves some simple inductive arguments. The paper concludes with a discussion of open questions.

2. Boolean grammars

In [8] and [10] A. Okhotin introduced the classes of conjunctive and Boolean grammars, respectively:

**Definition 1** ([10]). A Boolean grammar is a quadruple \( G = (\Sigma, N, P, S) \), where \( \Sigma \) and \( N \) are disjoint finite nonempty sets of terminal and nonterminal symbols, respectively, \( P \) is a finite set of rules, each of the form

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

and \( S \in N \) is the start symbol of the grammar. We will call the \( \alpha_i \)'s positive conjuncts and the \( \neg \beta_j \)'s negative. A Boolean grammar is called conjunctive if all its rules contain only positive conjuncts.

We will often abbreviate a collection of rules \( A \rightarrow \phi_1, 1 \leq i \leq l \) defining the nonterminal \( A \) of a Boolean grammar, as \( A \rightarrow \phi_1 | \cdots | \phi_l \). It is obvious that conjunctive grammars form a subclass of Boolean grammars and it would therefore be possible to present many notions regarding these two classes in a unified way. However, since conjunctive grammars will play an important role in our subsequent development of the locally stratified semantics (see Section 4), in the rest of this section we will be discussing about the two classes more or less independently. For example, conjunctive grammars have a particularly simple and elegant derivational semantics, which resembles the well-known one for context-free grammars. It has been proven convenient for us to introduce and use this semantics when dealing with conjunctive grammars (instead of using the more heavy machinery required for general Boolean grammars).

The basic ideas behind these two classes of grammars can now be illustrated by the following examples:

**Example 2** ([9]). Consider the grammar:

\[
S \rightarrow SAh&Cb \mid b \\
A \rightarrow aA \mid \varepsilon \\
C \rightarrow bCa \mid aC \mid baa
\]
It can be seen (see [9] for further explanations) that this grammar defines the (non context-free) language \(\{ba^2b a^4b \cdots ba^{2n-2}ba^{2n}b \mid n \geq 0\}\).

**Example 3** ([10]). Let \(\Sigma = \{a,b\}\). We define:

\[
\begin{align*}
  S &\rightarrow \neg(AB) \& \neg(BA) \& \neg(A \& B) \\
  A &\rightarrow a \mid CAC \\
  B &\rightarrow b \mid CBC \\
  C &\rightarrow a \mid b
\end{align*}
\]

It can be seen that this grammar defines the language \(\{ww \mid w \in \{a,b\}^*\}\) (see [10] for details). It is known that this language is not context-free.

In the rest of this section we present the semantics of conjunctive and Boolean grammars. Our presentation for conjunctive grammars follows the one given originally in [8] (where one can also find a more detailed account on this kind of grammars).

**Definition 4.** Let \(G = (\Sigma, N, P, S)\) be a conjunctive grammar. Then:

1. Any symbol in \(\Sigma \cup N\) is a conjunctive formula.
2. If \(A\) and \(B\) are conjunctive formulae, then \(AB\) is a conjunctive formula.
3. If \(A_1, \ldots, A_n, n \geq 1\), are conjunctive formulae, then \((A_1 \& \cdots \& A_n)\) is a conjunctive formula.

Given a conjunctive grammar \(G = (\Sigma, N, P, S)\), we denote by \(V_G\) the set \(\Sigma \cup N \cup \{\text{"",","}\}\). The notion of derivability in conjunctive grammars is defined as follows:

**Definition 5.** Let \(G = (\Sigma, N, P, S)\) be a conjunctive grammar. Define the relation \(\xrightarrow{G}\) of immediate derivability on the set of conjunctive formulae, as follows:

- For all \(s_1, s_2 \in V_G^+\) and for all \(A \in N\), if \(s_1 As_2\) is a formula, then for all \(A \rightarrow \alpha_1 \& \cdots \& \alpha_n \in P\)
  \[s_1 As_2 \xrightarrow{G} s_1(\alpha_1 \& \cdots \& \alpha_n)s_2\]
- For all \(s_1, s_2 \in V_G^+\) and for all \(w \in \Sigma^*\), if \(s_1(w \& \cdots \& w)s_2\) is a formula, then
  \[s_1(w \& \cdots \& w)s_2 \xrightarrow{G} s_1ws_2\]

We denote by \(\xrightarrow{G}^*\) the reflexive transitive closure of \(\xrightarrow{G}\).

We now define the notion of language generated by a conjunctive grammar:

**Definition 6.** Let \(G = (\Sigma, N, P, S)\) be a conjunctive grammar. The language generated by \(A \in N\) is the set of all strings over \(\Sigma\) derivable from \(A\). The language generated by \(G\) is the language generated by the start symbol \(S\) of \(G\).

We now turn our attention to the more general case of Boolean grammars; in this case a more general construction is required. The well-founded semantics is based on the notion of three-valued formal languages. Our presentation follows the one in [5].

The central concept in the semantics of Boolean grammars is that of *interpretation*, a notion that has its origins in mathematical logic. 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 dictates 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. However, these difficulties arise because we are trying to find classical models of Boolean grammars, which are based on classical two-valued logic. It turns out that if 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 7. Let $\Sigma$ be a finite non-empty set of symbols. Then, a (three-valued) language over $\Sigma$ is a function from $\Sigma^*$ 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 $w \not\in L$ (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 $\Sigma \rightarrow \neg \Sigma$: its meaning is the language which assigns to every string the value $\frac{1}{2}$. In the rest of the paper we will treat the usual two-valued languages over $\Sigma$ either as subsets of $\Sigma^*$ or as functions from $\Sigma^*$ to the set $\{0,1\}$.

The following definition, which generalizes the familiar notion of concatenation of languages, is also needed:

Definition 8. Let $\Sigma$ be a finite set of symbols and let $L_1, \ldots, L_n$ be (three-valued) languages over $\Sigma$. We define the \textit{three-valued concatenation} of the languages $L_1, \ldots, L_n$ to be the language $L$ such that:

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

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

We can now define the notion of \textit{interpretation} of a given Boolean grammar:

Definition 9. An \textit{interpretation} $I$ of a Boolean grammar $G = (\Sigma, N, P, S)$ is a function $I : N \rightarrow \left(\Sigma^* \rightarrow \{0, \frac{1}{2}, 1\}\right)$.

An interpretation $I$ can be recursively extended to apply to more general expressions:

Definition 10. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar and let $I$ be an interpretation of $G$. Then $I$ can be extended to apply to expressions that appear as the right-hand sides of Boolean grammar rules, as follows:

- For every $w \in \Sigma^*$, it is $I(\epsilon)(w) = 1$ if $w = \epsilon$ and 0 otherwise.
- Let $a \in \Sigma$. Then, for every $w \in \Sigma^*$, it is $I(a)(w) = 1$ if $w = a$ and 0 otherwise.
- Let $\alpha = a_1 \cdots a_n$, $n \geq 2$, $a_i \in \Sigma \cup N$. Then, for every $w \in \Sigma^*$, it is $I(\alpha)(w) = (I(a_1) \circ \cdots \circ I(a_n))(w)$.
- Let $\alpha \in (\Sigma \cup N)^*$. Then, for every $w \in \Sigma^*$, it is $I(\neg \alpha)(w) = 1 - I(\alpha)(w)$.
- Let $l_1, \ldots, l_n$ be conjuncts. Then, for every $w \in \Sigma^*$, it is $I(l_1 \& \cdots \& l_n)(w) = \min(I(l_1)(w), \ldots, I(l_n)(w))$.

The notion of a model of a Boolean grammar can now be defined. Intuitively, a model is an interpretation that does not violate any rule of the grammar:

Definition 11. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar and $I$ an interpretation of $G$. Then, $I$ is a \textit{model} of $G$ if for every rule $A \rightarrow l_1 \& \cdots \& l_n$ in $P$ and for every $w \in \Sigma^*$, it is $I(A)(w) \leq I(l_1)(w) \& \cdots \& I(l_n)(w)$.

In the definition of the well-founded model, two orderings on interpretations play a crucial role (see [12] for the corresponding orderings regarding logic programs). Given two interpretations, the first ordering (usually called the \textit{standard ordering}) compares their \textit{degree of truth}:

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

Among the interpretations of a given Boolean grammar, there is one which is the least with respect to the $\leq$ ordering, namely the interpretation $\bot$ which for all $A$ and all $w$, $\bot(A)(w) = 0$.

The second ordering (usually called the Fitting ordering) compares the \textit{degree of information} of two interpretations. We first need to define the corresponding ordering for truth values:

Definition 13. Let $v_1, v_2$ be truth values in $\{0, \frac{1}{2}, 1\}$. We write $v_1 <_F v_2$ if $v_1 = \frac{1}{2}$ and $v_2 \in \{0, 1\}$. We write $v_1 \leq_F v_2$ if $v_1 <_F v_2$ or $v_1 = v_2$.

Using the relation $\leq_F$ we can now define $\leq_F$ as follows:

Definition 14. 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 $w \in \Sigma^*$, $I(A)(w) \leq_F J(A)(w)$. 
Among the interpretations of a given Boolean grammar, there is one which is the least with respect to the $\preceq_F$ ordering, namely the interpretation $\downarrow_F$, for which for all $A$ and all $w$, $\bot_F (A)(w) = \frac{1}{2}$.

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

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

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

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

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

The basic idea behind the well-founded semantics is that the intended model of the grammar is constructed in stages corresponding to the levels of negation used by the grammar. At each step of this process and for every nonterminal 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 nonterminal in a previous stage, cannot be altered by the next stages). At the end of all the stages, certain strings for certain nonterminals 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 $G$. Then, for any interpretation $J$ of $G$ we define the operator $\Theta_J : I \to I$ on the set $I$ of all 3-valued interpretations of $G$. This operator is analogous to the one used in the logic programming domain (see for example [12]).

**Definition 15.** Let $G = (\Sigma, N, P, S)$ be a Boolean grammar, let $I$ be the set of all three-valued interpretations of $G$ and let $J \in I$. The operator $\Theta_J : I \to I$ is defined as follows. For every $I \in I$, for all $A \in N$ and for all $w \in \Sigma^*$:

1. $\Theta_J(I)(A)(w) = 1$ if there exists a rule $A \to l_1 \& \cdots \& l_n$ in $P$ such that for all positive $l_i$ it is $I(l_i)(w) = 1$ and for all negative $l_i$ it is $I(l_i)(w) = 1$;
2. $\Theta_J(I)(A)(w) = 0$ if for every rule $A \to l_1 \& \cdots \& l_n$ in $P$, either there exists a positive $l_i$ such that $I(l_i)(w) = 0$ or there exists a negative $l_i$ such that $I(l_i)(w) = 0$;
3. $\Theta_J(I)(A)(w) = \frac{1}{2}$, otherwise.

An important fact regarding the operator $\Theta_J$ is that it is monotonic with respect to the $\preceq$ ordering of interpretations; this property ensures that $\Theta_J$ has a least fixed-point with respect to $\preceq$. These ideas are captured by the following definition and theorem:

**Definition 16.** Let $G = (\Sigma, N, P, S)$ be a Boolean grammar and let $J$ be an interpretation of $G$. Define:

$$\Theta_J^0 = \bot \\
\Theta_J^{n+1} = \Theta_J(\Theta_J^n) \\
\Omega(J) = \text{lub}_\preceq(\Theta_J^n \mid n \in \mathbb{N})$$

The following theorem can then be demonstrated [5]:

**Theorem 17.** Let $G = (\Sigma, N, P, S)$ be a Boolean grammar and let $J$ be an interpretation of $G$. Then, the operator $\Theta_J$ is monotonic with respect to the $\preceq$ ordering of interpretations. Moreover, the sequence $\{\Theta_J^n\}_{n \in \mathbb{N}}$ is increasing with respect to the ordering $\preceq$ and $\Omega(J)$ is the unique least (with respect to $\preceq$) fixed point of $\Theta_J$.

Given a grammar $G$, we can use the $\Omega$ operator to construct a sequence of interpretations whose least upper bound $M_G$ (with respect to $\preceq_F$) is a distinguished model of $G$.

**Definition 18.** Let $G = (\Sigma, N, P, S)$ be a Boolean grammar. Define:
Theorem 19. Let \( G = (\Sigma,N,P,S) \) be a Boolean grammar. Then, the operator \( \Omega \) is monotonic with respect to the \( \preceq_F \) ordering of interpretations. Moreover, the sequence \( \{M_n\}_{n \in \mathbb{N}} \) is increasing with respect to the ordering \( \preceq_F \) and \( M_G \) is the unique least (with respect to \( \preceq_F \)) fixed point of \( \Omega \).

The following theorem completes our presentation of the semantics of Boolean grammars by stating the fact that \( M_G \) satisfies all the rules of the Boolean grammar \( G \):

Theorem 20. 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 \( G \)).

In the following section we will see that the well-founded construction will play the central role in establishing that the locally stratified semantics is well-defined.

3. Locally stratified Boolean grammars

In this section, we define the class of locally stratified Boolean grammars. This class extends the stratified Boolean grammars, introduced by M. Wrona.

Definition 21 (\cite{17}). A Boolean grammar \( G = (\Sigma,N,P,S) \) is called stratified if there exists a function \( g : N \rightarrow \mathbb{N} \) such that for every rule
\[
C \rightarrow \alpha_1 \& \ldots \& \alpha_m \& \neg \beta_1 \& \ldots \& \neg \beta_n
\]
in \( P \) the following conditions hold:

- for every \( i, 1 \leq i \leq m \), and for every \( A \in N \) that appears in \( \alpha_i \), \( g(C) \geq g(A) \)
- for every \( j, 1 \leq j \leq n \), and for every \( B \in N \) that appears in \( \beta_j \), \( g(C) \geq g(B) \).

The class of stratified Boolean grammars is a proper subclass of Boolean grammars, but it appears to have an interest in its own right. For example, questions of the form "are there languages that can be defined by general Boolean grammars but not by stratified ones?" do not in general have obvious answers (and may trigger deeper investigations in the theory of these grammars).

There exist however many simple and intuitive Boolean grammars that fail to be stratified. To motivate the new and broader class, consider the following Boolean grammar with start symbol \( S = E \), that defines the (regular) set of strings of even length over the alphabet \( \Sigma = \{a\} \):

\[
\begin{align*}
E & \rightarrow \varepsilon \\
E & \rightarrow aO \\
O & \rightarrow \neg E
\end{align*}
\]

One can verify that the above grammar is not stratified. However, it can easily be seen that the grammar specifies the language we mentioned above. For example, the string \( aa \) belongs to the language corresponding to \( E \) because the string \( a \) belongs to the language corresponding to \( O \) (since it does not belong to the language corresponding to \( E \)).

We can now define a much broader class of Boolean grammars that covers cases such as the above one:

Definition 22. A Boolean grammar \( G = (\Sigma,N,P,S) \) is called locally stratified if there exists a function \( f : (N \times \Sigma^*) \rightarrow \mathbb{N} \) such that for every rule
\[
C \rightarrow \alpha_1 \& \ldots \& \alpha_m \& \neg \beta_1 \& \ldots \& \neg \beta_n
\]
in \( P \), the following conditions hold for every \( i, 1 \leq i \leq m \), and for every \( j, 1 \leq j \leq n \):

- Suppose that \( \alpha_i = \sigma_1 A_1 \sigma_2 A_2 \ldots \sigma_k A_k \sigma_{k+1} \), for \( k \geq 1 \), \( \sigma_p \in \Sigma^* \), \( A_p \in N \). Then for every \( w_1,w_2,\ldots,w_k \in \Sigma^* \) and for every \( p, 1 \leq p \leq k \), it holds \( f(C,\sigma_1 w_1 \sigma_2 w_2 \ldots \sigma_k w_k \sigma_{k+1}) \geq f(A_p,w_p) \).
- Suppose that \( \beta_j = \tau_1 B_1 \tau_2 B_2 \ldots \tau_q B_q \tau_{q+1} \), for \( q \geq 1 \), \( \tau_q \in \Sigma^* \), \( B_q \in N \). Then for every \( w_1,w_2,\ldots,w_q \in \Sigma^* \) and for every \( q, 1 \leq q \leq \ell \), it holds \( f(C,\tau_1 w_1 \tau_2 w_2 \ldots \tau_q w_q) > f(B_q,w_q) \).
As we have already mentioned, local stratification is a notion that was initially proposed in the area of logic programming [13]. There are however some crucial differences that make the study of local stratification in Boolean grammars even more interesting. First, local stratification of logic programs in many cases requires a transfinite number of strata (i.e., the use of countable ordinals beyond the natural numbers in the labeling of the strata). Second, the problem of detecting whether a logic program is locally stratified is in general unsolvable (see [2]) and one can only hope to find subclasses of logic programs in which the notion is decidable (see for example [14,7]). Surprisingly, it turns out that local stratifiability of Boolean grammars can be decided in polynomial time (actually, in linear time with respect to the size of the grammar). Finally, as it will be demonstrated in the rest of this section, local stratification in Boolean grammars can be completely characterized by the use of a special form of stratum-functions, called canonical stratum functions; such a notion does not seem to be applicable in the case of logic programs. It should be emphasized at this point that the idea of canonical stratum functions will lead in the next section to the locally stratified semantics which has significant differences in concept even from the well-founded construction of Boolean grammars [5].

**Definition 23.** Let \( G = (\Sigma, N, P, S) \) be a Boolean grammar that is locally stratified by a function \( f \). We say that \( f \) is a canonical stratum-function if

- for every \( w, w' \in \Sigma^* \) and for every \( A, B \in N \), if \( |w| > |w'| \) then \( f(A, w) > f(B, w') \).
- for every \( w, w' \in \Sigma^* \) and for every \( A \in N \), if \( |w| = |w'| \) then \( f(A, w) = f(A, w') \).

We can now demonstrate that local stratifiability of Boolean grammars is decidable (and actually, efficiently so). Before we state Theorem 25 that proves this fact, we need the following definition:

**Definition 24.** Let \( G = (\Sigma, N, P, S) \) be a Boolean grammar. The skeleton of \( G \) is the grammar \( G' = (\Sigma, N, P', S) \), where \( P' \) is obtained from \( P \) by removing from the right-hand side of each rule every conjunct that equals \( \epsilon \) or \( \neg \epsilon \), or contains terminal symbols and then removing all rules that end with an empty right-hand side.

**Theorem 25.** Let \( G = (\Sigma, N, P, S) \) be a Boolean grammar and let \( G' = (\Sigma, N, P', S) \) be its skeleton. Then, the following three conditions are equivalent:

1. \( G \) is locally stratified.
2. \( G' \) is stratified.
3. \( G \) is locally stratified by a canonical stratum function.

**Proof.** It is obvious that (3) \( \Rightarrow \) (1). In order to show that (1) \( \Rightarrow \) (2), suppose that \( G \) is locally stratified by \( f \). Define a function \( g : N \to \mathbb{N} \) such that \( g(A) = f(A, \epsilon) \). Let \( C \to \alpha_1 \& \cdots \& \alpha_m \& \neg \beta_1 \& \cdots \& \neg \beta_n \) be a rule in \( P \). Suppose that \( A \in N \) appears in some \( \alpha_i \). Since \( G' \) is the skeleton of \( G \), \( \alpha_i \) is of the form \( A_1 A_2 \cdots A_k \), \( k \geq 1 \) and \( A_p \in N \) for \( 1 \leq p \leq k \), and \( A = A_r \) for some \( r, 1 \leq r \leq k \).

Notice that, \( \alpha_i = \sigma_1 \sigma_2 A_2 \cdots \sigma_k A_k \sigma_{k+1} \), where \( \sigma_1 = \sigma_2 = \cdots = \sigma_k = \sigma_{k+1} = \epsilon \). Let \( w_1 = w_2 = \cdots = w_k = \epsilon \). From the definition of local stratification we get \( f(C, \epsilon) \geq f(A_r, \epsilon) \), which implies \( g(C) \geq g(A_r) = g(A) \). Similarly it can be proved that if \( B \in N \) appears in some \( \beta_j \), then \( g(C) > g(B) \). Consequently, \( G' \) is stratified by \( g \).

To show that (2) \( \Rightarrow \) (3), suppose that the skeleton \( G' \) is stratified by \( g \) and let \( s = 1 + \max\{i \in \mathbb{N} \mid \exists A \in N \) such that \( g(A) = i\} \). In other words \( s \) is an upper bound for the number of the non-empty strata according to \( g \). Define \( f : (N \times \Sigma^*) \to \mathbb{N} \) such that \( f(A, w) = s \cdot |w| + g(A) \). It is easy to see that \( f \) is a canonical stratum-function. Let \( C \to \alpha_1 \& \cdots \& \alpha_m \& \neg \beta_1 \& \cdots \& \neg \beta_n \) be a rule in \( P \). Consider an \( \alpha_i = \sigma_1 \sigma_2 A_2 \cdots \sigma_k A_k \sigma_{k+1} \) and an arbitrary sequence of strings \( w_1, w_2, \ldots, w_k \in \Sigma^* \). Let \( w = \sigma_1 w_1 \sigma_2 w_2, \ldots, \sigma_k w_k \sigma_{k+1} \). If \( |w| > |w_p| \), then \( f(C, w) > f(A_p, w_p) \) from the canonicity of \( f \). Otherwise (\( |w| = |w_p| \)) it holds \( \sigma_1 = \sigma_2 = \cdots = \sigma_k = \sigma_{k+1} = \epsilon \), i.e., \( \alpha_i \) \( \in N \). Therefore \( G' \) contains a rule with \( C \) in the left-hand side and \( \alpha_i \) in the right-hand side, which implies \( g(C) \geq g(A_p) \). Since \( |w| = |w_p| \), from the definition of \( f \) we get \( f(C, w) \geq f(A_p, w_p) \). The case for \( \beta_j \) is similar. Consequently, \( G \) is locally stratified by \( f \). □

**Corollary 26.** If a Boolean grammar \( G \) is stratified then it is locally stratified.

**Proof.** If \( G \) is stratified by \( f \) then its skeleton is also stratified by \( f \). □

The converse of Corollary 26 does not hold as the example in the beginning of this section as well as the following example demonstrate:

**Example 27.** Consider the Boolean grammar \( G = (\Sigma, N, P, S) \), where \( \Sigma = \{a\} \) and \( P \) contains the following rules:

\[ S \rightarrow A \& \neg aA \ | \ aB \& \neg aB \ | \ aC \& \neg aC \]

\[ A \rightarrow aBB \]
\[
\begin{align*}
B & \rightarrow \neg CC \\
C & \rightarrow \neg EE \\
E & \rightarrow \neg A
\end{align*}
\]

The above grammar defines the language \( L = \{a^{2n} \mid n \in \mathbb{N}\} \) (see [11]). This grammar is not stratified. However, it is locally stratified. In order to prove this claim, consider the skeleton \( G' \) of \( G \) which contains the following set of rules:

\[
\begin{align*}
S & \rightarrow A \mid \neg B \mid \neg C \\
B & \rightarrow \neg CC \\
C & \rightarrow \neg EE \\
E & \rightarrow \neg A
\end{align*}
\]

It is easy to see that \( G' \) is stratified by the function \( g \), such that \( g(A) = 0, g(E) = 1, g(C) = 2, g(B) = 3 \), and \( g(S) = 4 \). Thus \( G \) is locally stratified.

The above theorem shows that testing local stratifiability of a Boolean grammar \( G \) can be reduced to testing (ordinary) stratifiability of the skeleton of \( G \). The reduction requires time \( O(|G|) \), where \( |G| \) denotes the size of the representation of \( G \), and produces a grammar \( G' \) with \( |G'| \leq |G| \). Testing if \( G' \) is stratified requires time \( O(|G'|) \) [17], using simple graph algorithms. Consequently local stratifiability of a Boolean grammar \( G \) can be tested in time \( O(|G|) \). Notice that, as we have already mentioned, testing local stratifiability of logic programs is an unsolvable problem.

4. The locally stratified semantics

In this section, we define the semantics of locally stratified Boolean grammars. Assume that \( G = (\Sigma, N, P, S) \) is a Boolean grammar that is locally stratified by a canonical stratum function \( f \). The languages defined by the non-terminal symbols of \( G \), can be constructed in stages. During the \( i \)th stage, for every pair \((A, w)\) that belongs to the \( i \)th stratum, we decide whether \( w \) belongs to the language defined by \( A \). More specifically, assume that we have computed the meaning of \( G \), for all strata smaller than \( i \) that is, we have computed a two-valued interpretation \( I_i \) which has the following property: for all \( A \in N \) and all \( w \in \Sigma^* \), if \( f(A, w) < i \), then \( I_i(A)(w) = 1 \) if and only if \( w \) belongs to the language defined by \( A \) according to \( G \); otherwise, \( I_i(A)(w) = 0 \). The question we face now is how we can use this partial meaning \( I_i \) of \( G \) to compute a two-valued interpretation \( I_{i+1} \) that captures the meaning of \( G \) for all strata less than \( i + 1 \). Assume that \( M \) is the set of nonterminals that belong to stratum \( i \) and let \( n \) be the unique length of strings that participate in this stratum. We will now show how one can construct a conjunctive grammar \( G/(I_i, M, n) \), whose meaning will define the difference between \( I_{i+1} \) and \( I_i \). The intuition behind the details of the construction, will be explained just after the following definitions:

**Definition 28.** Let \( \Sigma \) be an alphabet. We denote by \( \Sigma^n \) the set \( \{w \in \Sigma^* \mid |w| = n\} \) and by \( \Sigma^{\leq n} \) the set \( \bigcup_{i=0}^{n} \Sigma^i \).

**Definition 29.** Let \( G = (\Sigma, N, P, S) \) be a Boolean grammar, \( I \) be an interpretation, \( M \subseteq N \) be a set of non-terminal symbols, and \( n \geq 0 \) be an integer. Let \( R \) be the set of all conjuncts that appear in the right hand sides of the rules in \( P \) in which the left-hand side symbol is in \( M \). We denote by \( G/(I, M, n) \) the grammar \((\Sigma, N', P', S)\), such that:

- \( N' = N \cup \{D_l \mid l \in R\} \), where the \( D_l \)'s are new non-terminal symbols not belonging to \( N \).
- For every rule of the form \( C \rightarrow l_1 \& l_2 \& \ldots \& l_m \) in \( P \), such that \( C \in M \), \( P' \) contains the rule: \( C \rightarrow D_{l_1} \& D_{l_2} \& \ldots \& D_{l_m} \).
- For every conjunct \( l \in R \) and for every \( w \in (i(l) \cap \Sigma^0) \), \( P' \) contains the rule: \( D_l \rightarrow w \).
- If \( n > 0 \) then for every conjunct \( l = A_1 A_2 \cdots A_k \in R \cap N^+ \) and for every \( i, 1 \leq i \leq k \), \( P' \) contains the rule: \( D_l \rightarrow A_i \) if for every \( j \) with \( 1 \leq j \leq k, j \neq i \), it holds \( \epsilon \in I(A_j) \).
- If \( n = 0 \) then for every conjunct \( l = A_1 A_2 \cdots A_k \in R \cap N^+ \), \( P' \) contains the rule: \( D_l \rightarrow \ell' \), where \( \ell' = \alpha_1 \alpha_2 \cdots \alpha_k \), with \( \alpha_i = \epsilon \) if \( \epsilon \in I(A_i) \) and \( \alpha_i = A_i \) otherwise.

Assume now that we apply the above definition to a locally stratified Boolean grammar \( G \). Moreover, assume that the interpretation \( I \) in the above definition is equal to the interpretation \( I_i \). \( M \) is the set of non-terminal symbols in the \( i \)th stratum and \( n \) is the unique length of strings that belong to this \( i \)th stratum. Then, the grammar \( G/(I, M, n) \) contains all the necessary information so as to decide the membership of strings of length \( n \) in the languages of non-terminals that belong to \( M \). The membership of \( w \), with \( |w| = n \), under \( A \in M \) can be determined in the following alternative ways:

- either directly, using only information determined at previous strata (this is expressed by the third item in the above definition),
- or indirectly, using information determined in previous strata, combined with information obtained at the current stratum (this is expressed by the last two items in the above definition).
Based on the above definition we can now formally define the locally stratified semantics of Boolean grammars. The following definition uses the (derivational) semantics of conjunctive grammars that has been presented in Section 2.

**Definition 30.** Let \( G = (\Sigma, N, P, S) \) be a Boolean grammar stratified by a canonical stratum-function \( f \). Let \( n_i \) be the (unique) length of strings in the \( i \)-th stratum and \( N_i \) be the set of nonterminal symbols in this stratum. The locally stratified semantics of \( G \) is the interpretation \( L_G \) with \( L_G(A) = \bigcup_{i=0}^{n_i} I_i(A) \), where \( I_0 = \bot \) and \( I_{i+1}(A) = I_i(A) \cup \Delta_i(A) \), for every \( A \in N \), and \( \Delta_i(A) \) is the language generated by nonterminal \( A \) under the conjunctive grammar \( G_i = G/(I_i, N_i) \).

Certain comments are in order. As it becomes obvious from the above definitions, each stratum that is created by the locally stratified approach corresponds to strings that have the same length. This is not the case in the well-founded semantics, in which each implicit level in the construction may correspond to strings that have a variety of lengths. In other words, the two constructions proceed in different ways and this may prove to be an interesting fact (since each approach may turn out to have its own application domain).

The above ideas are illustrated by the following two examples (the first one is rather simple while the second one demonstrates all the ideas that are present in Definitions 29 and 30).

**Example 31.** Consider again the Boolean grammar \( G \) given by the rules:

\[
E \rightarrow \epsilon \\
E \rightarrow aO \\
O \rightarrow \neg E
\]

Obviously, \( G \) is locally stratified by the canonical stratum function: \( f(E,w) = 2 \cdot |w| \) and \( f(O,w) = 2 \cdot |w| + 1 \). In order to compute \( L_G \) we initially create the conjunctive grammar \( G_0 = G/(\bot, E, 0) \). The rules of \( G_0 \) are: \( E \rightarrow D_1, E \rightarrow D_6, \) and \( D_6 \rightarrow \epsilon \).

By computing the meaning of \( G_0 \) we get that \( I_1(E) = \{ \epsilon \} \). We next create the conjunctive grammar \( G_1 = G/(I_1, O, 0) \). The only rule in \( G_1 \) is: \( O \rightarrow D_\neg \). By computing the meaning of \( G_1 \) we get that \( I_2(O) = \emptyset \). More generally, it will be \( G_{2i} = G/(I_{2i}, E, i) \) with \( I_{2i+1}(E) = \{ \alpha^{2i} | 2i \leq i \} \), if \( i \) is even and \( I_{2i+1}(E) = \{ \alpha^{2i} | 2i < i \} \), if \( i \) is odd; moreover, \( G_{2i+1} = G/(I_{2i+1}, O, i) \) with \( I_{2i+2}(O) = \{ \alpha^{2i+1} | 2i + 1 < i \} \), if \( i \) is even and \( G_{2i+1} = \{ \alpha^{2i+1} | 2i + 1 \leq i \} \), if \( i \) is odd.

**Example 32.** Consider the Boolean grammar \( G = (\Sigma, N, P, S) \), where \( \Sigma = \{ a, b \} \) and \( P \) contains the following rules:

\[
S \rightarrow TB \& \neg C \\
T \rightarrow A \& \neg C \\
A \rightarrow a | aA \\
B \rightarrow \epsilon | bB \\
C \rightarrow \epsilon | \neg aC \& \neg bC
\]

We will demonstrate that the grammar is locally stratified and that it expresses the language \( \{ a^{2k+1} b^{2r} | k, r \in \mathbb{N} \} \).

The skeleton of the Boolean grammar \( G' \) that consists of the rules:

\[
S \rightarrow TB \& \neg C \\
T \rightarrow A \& \neg C
\]

which is stratified by \( g \), with \( g(A) = g(B) = g(C) = 0 \) and \( g(S) = g(T) = 1 \). Therefore, \( G \) is locally stratified by the canonical stratum-function \( f \) defined as follows: for every \( w \in \Sigma^* \), \( f(A,w) = f(B,w) = f(C,w) = 2 \cdot |w| \), and \( f(S,w) = f(T,w) = 2 \cdot |w| + 1 \).

We assign to the conjuncts that appear in the right-hand side of rules new non-terminal symbols as shown in the following table:

<table>
<thead>
<tr>
<th>Conjunct</th>
<th>( TB )</th>
<th>( \neg C )</th>
<th>( A )</th>
<th>( a )</th>
<th>( aA )</th>
<th>( \epsilon )</th>
<th>( bB )</th>
<th>( \neg aC )</th>
<th>( \neg bC )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-terminal symbol</td>
<td>( D_1 )</td>
<td>( D_2 )</td>
<td>( D_3 )</td>
<td>( D_4 )</td>
<td>( D_5 )</td>
<td>( D_6 )</td>
<td>( D_7 )</td>
<td>( D_8 )</td>
<td>( D_9 )</td>
</tr>
</tbody>
</table>

Then, every grammar \( G_i \) contains the set of rules \( P_e = \{ A \rightarrow D_4, A \rightarrow D_5, B \rightarrow D_6, B \rightarrow D_7, C \rightarrow D_6, C \rightarrow D_8 \& D_9 \} \) if \( i \) is an even number, and the set of rules \( P_o = \{ S \rightarrow D_1 \& D_2, T \rightarrow D_6 \& D_2 \} \) if \( i \) is an odd number. The construction of the sequence \( I_0, I_1, I_2, \ldots \) is illustrated below:
Induction hypothesis:

\[ P_0 = P_v \cup \{ D_6 \rightarrow \epsilon, D_8 \rightarrow \epsilon, D_9 \rightarrow \epsilon \}, \]
\[ \Delta_0(A) = \emptyset, \Delta_0(B) = \Delta_0(C) = \{ \epsilon \}, I_1(A) = \emptyset, I_1(B) = I_1(C) = \{ \epsilon \} \text{ and } I_1(S) = I_1(T) = \emptyset. \]

\[ P_1 = P_0 \cup \{ D_1 \rightarrow T, D_3 \rightarrow A \}, \]
\[ \Delta_1(S) = \Delta_1(T) = \emptyset \text{ and } I_2 = I_1. \]

\[ P_2 = P_1 \cup \{ D_4 \rightarrow a, D_7 \rightarrow b, D_b \rightarrow b, D_9 \rightarrow a \}, \]
\[ \Delta_2(A) = \{ a \}, \Delta_2(B) = \{ b \}, \Delta_2(C) = \emptyset, I_3(A) = \{ a \}, I_3(B) = \{ \epsilon, b \}, I_3(C) = \{ \epsilon \} \text{ and } I_3(S) = I_3(T) = \emptyset. \]

\[ P_3 = P_2 \cup \{ D_2 \rightarrow a, D_2 \rightarrow b, D_2 \rightarrow a, D_1 \rightarrow T, D_3 \rightarrow A \}, \]
\[ \Delta_3(S) = \Delta_3(T) = \{ a \}, I_4(A) = \{ a \}, I_4(B) = \{ \epsilon, b \}, I_4(C) = \{ \epsilon \} \text{ and } I_4(S) = I_4(T) = \{ a \}. \]

\[ P_4 = P_3 \cup \{ D_5 \rightarrow aa, D_7 \rightarrow bb, D_8 \rightarrow aa, D_8 \rightarrow ab, D_8 \rightarrow bb, D_9 \rightarrow aa, D_9 \rightarrow ab, D_9 \rightarrow ba, D_9 \rightarrow bb \}, \]
\[ \Delta_4(A) = \{ aa \}, \Delta_4(B) = \{ bb \}, \Delta_4(C) = \{ aa, ab, ba, bb \}, I_5(A) = \{ aa \}, I_5(B) = \{ \epsilon, b, bb \}, I_5(C) = \{ \epsilon, aa, ab, ba, bb \} \text{ and } I_5(S) = I_5(T) = \{ a \}. \]

\[ P_5 = P_4 \cup \{ D_1 \rightarrow ab, D_3 \rightarrow aa, D_1 \rightarrow T, D_3 \rightarrow A \}, \]
\[ \Delta_5(S) = \Delta_5(T) = \emptyset \text{ and } I_6 = I_5. \]

\[ P_6 = P_5 \cup \{ D_2 \rightarrow aaa, D_2 \rightarrow bbb, D_8 \rightarrow baa, D_8 \rightarrow bab, D_8 \rightarrow bba, D_8 \rightarrow bbb, D_9 \rightarrow aaa, D_9 \rightarrow ab, D_9 \rightarrow aba, D_9 \rightarrow abb \}, \]
\[ \Delta_6(A) = \{ aaa \}, \Delta_6(B) = \{ bbb \}, \Delta_6(C) = \emptyset, I_7(A) = \{ aa, aaa \}, I_7(B) = \{ \epsilon, bb, bbb \}, I_7(C) = \{ \epsilon, aa, ab, ba, bb \} \text{ and } I_7(S) = I_7(T) = \{ a \}. \]

\[ P_7 = P_6 \cup \{ D_1 \rightarrow abb, D_2 \rightarrow aaa, D_2 \rightarrow ab, D_2 \rightarrow aba, D_2 \rightarrow baa, D_2 \rightarrow abb, D_2 \rightarrow bba, D_2 \rightarrow bbb, D_3 \rightarrow aaa, D_1 \rightarrow T, D_3 \rightarrow A \}, \]
\[ \Delta_7(S) = \{ aaa, abb \}, \Delta_7(T) = \{ aa, aaa \}, I_8(A) = \{ aa, aaa \}, I_8(B) = \{ \epsilon, bb, bbb \}, I_8(C) = \{ \epsilon, aa, ab, ba, bb \}, I_8(S) = \{ aa, aaa, abb \} \text{ and } I_8(T) = \{ a, aaa \}. \]

It can be proved by induction that for every \( j \geq 0 \), it holds that:
\[ \Delta_{2j+2}(A) = \{ a^{j+1} \}, \Delta_{2j+1}(B) = \{ b^j \}, \Delta_{4j}(C) = \Sigma^{2j}, \Delta_{4j+3}(T) = \{ a^{2j+1} \} \text{ and } \Delta_{4j+3}(S) = \{ a^{2j+1} b^{2j-2k} | 0 \leq k \leq j \}. \]
Therefore, \( L_C(A) = \{ a^{k+1} | k \in \mathbb{N} \}, L_C(B) = \{ b^k | k \in \mathbb{N} \}, L_C(C) = \{ w \in \Sigma^{2k} | k \in \mathbb{N} \}, L_C(T) = \{ a^{2k+1} | k \in \mathbb{N} \} \) and \( L_C(S) = \{ a^{2k+1} b^{2r} | k, r \in \mathbb{N} \} \). Therefore, the locally stratified semantics of \( G \) coincides with its intuitive meaning. \( \square \)

5. Independence from the stratum function

It is not immediately obvious that the construction described in the previous section is well-defined. More specifically, at first sight it appears that the interpretation \( L_C \) depends on the stratum function that has been adopted. In this section, we demonstrate that \( L_C \) is independent of the stratum function; actually, \( L_C \) is identical to the well-founded model \( M_C \) of \( G \), despite the fact that the construction of \( L_C \) proceeds in a different way than \( M_C \).

First, we present the following lemma and theorem, which state that the well-founded model of locally stratified Boolean grammars is total.

**Lemma 33.** Let \( G = (\Sigma, N, P, S) \) be a Boolean grammar which is locally stratified by a canonical stratum function \( f \). Then, for every \( n \in \mathbb{N} \), for every \( C \in N \) and for every \( w \in \Sigma^* \), if \( f(C, w) < n \) then \( M_n(C)(w) \in \{ 0, 1 \} \).

**Proof.** We will prove our claim by induction on \( n \).

**Induction basis:** For \( n = 0 \) the statement holds trivially, since there is no \((C, w)\) such that \( f(C, w) < 0 \).

**Induction hypothesis:** Assume that for every \( C \in N \) and for every \( w \in \Sigma^* \), if \( f(C, w) < n \) then \( M_n(C)(w) \in \{ 0, 1 \} \).

**Induction step:** We show that for every \( C \in N \) and for every \( w \in \Sigma^* \), if \( f(C, w) < n + 1 \) then \( M_{n+1}(C)(w) \in \{ 0, 1 \} \). In order to prove this, it suffices to show that for every \( m \in \mathbb{N} \), for every \( C \in N \) and for every \( w \in \Sigma^* \), if \( f(C, w) < n + 1 \) then \( M_m(C)(w) \in \{ 0, 1 \} \). Then the desired result follows from Definition 18.

The proof uses an inner induction on \( m \). The basis case is immediate since we know that \( M_0(C) = \emptyset _{m} ^{M_0} \), that is \( \emptyset _{M_0} ^{M_0} (C)(w) = 0 \).

Assume now that the statement holds for \( m \) and suppose that \( f(C, w) < n + 1 \). It is obvious that if \( \emptyset _{M_{n+1}} ^{M_{n+1}} (C)(w) = 1 \), then \( \emptyset _{M_{n+1}} ^{M_{n+1}} (I_1 \& \ldots \& I_k)(w) = 0 \). So suppose that \( \emptyset _{M_{n+1}} ^{M_{n+1}} (C)(w) 
eq 1 \). Then for every rule of the form \( C \rightarrow I_1 \& \ldots \& I_k \) in \( G \), there exists a conjunct \( I_q \) that prevents the application of the first case in the definition of the \( \emptyset \) operator. There are two cases:
Case 1: $I_p$ is a positive conjunct of the form $I_p = \sigma_1 A_1 \ldots \sigma_r A_r \sigma_{r+1}$, where $r \geq 0$, $\sigma_i \in \Sigma^*$, $A_j \in N$, such that $\theta_{M_{\sigma_i}}^{I_p}(w) \neq 1$. Consider any choice of $w_1, \ldots, w_r \in \Sigma^*$ such that $\sigma_1 w_1 \ldots \sigma_r w_r \sigma_{r+1} = w$. Since $G$ is locally stratified by $f$, for every $j$ it is $f(A_j w_j) \leq f(C w) < n + 1$, and from the inner induction hypothesis we have $\theta_{M_0}^{I_p}(A_j(w_j)) \in \{0,1\}$. Therefore, $\theta_{M_0}^{I_p}(I_p(w)) \in \{0,1\}$, from which we obtain that $\theta_{M_{\sigma_i}}^{I_p}(I_p(w)) = 0$.

Case 2: $I_p$ is a negative conjunct of the form $I_p = \neg \sigma_1 A_1 \ldots \sigma_r A_r \sigma_{r+1}$, where $r \geq 0$, $\sigma_i \in \Sigma^*$, $A_j \in N$, such that $M_0(I_p(w)) \neq 1$. Consider any choice of $w_1, \ldots, w_r \in \Sigma^*$ such that $\sigma_1 w_1 \ldots \sigma_r w_r \sigma_{r+1} = w$. Since $G$ is locally stratified by $f$, for every $j$ it is $f(A_j w_j) < f(C w) < n + 1$. Thus, $f(A_j w_j) < n$ and from the outer induction hypothesis we have $M_0(A_j(w_j)) \in \{0,1\}$. Therefore, $M_0(I_p(w)) \in \{0,1\}$, from which we obtain that $M_0(I_p(w)) = 0$.

Therefore, if $\theta_{M_{\sigma_i}}^{I_p}(I_p(w)) \neq 1$, then for every rule of the form $C \rightarrow I_1 \& \cdots \& I_k$ in $G$, there exists either a positive conjunct $I_p$ such that $\theta_{M_0}^{I_p}(I_p(w)) = 0$ or a negative conjunct $I_p$ such that $M_0(I_p(w)) = 0$. Thus, from the second case in the definition of the $\theta$ operator we get $\theta_{M_{\sigma_i}}^{I_p}(I_p(w)) = 0$.

Therefore, if $f(C w) < n + 1$ then $M_{\sigma_i+1}(C(w)) \in \{0,1\}$. This completes the inductive proof. □

Theorem 34. Let $G = (\Sigma, N, P, S)$ be a locally stratified Boolean grammar. Then, for every $C \in N$ and for every $w \in \Sigma^*$, $M_C(C(w)) \in \{0,1\}$ (i.e., the well-founded model of a locally stratified Boolean grammar is total).

Proof. It follows immediately from the definition of $M_C$, using Lemma 33. □

The following theorem demonstrates that, independently from the particular stratum function used, the interpretation $L_C$ coincides with the well-founded model $M_C$. Therefore, the locally stratified semantics is well defined.

Theorem 35. Let $G = (\Sigma, N, P, S)$ be a Boolean grammar which is locally stratified by a canonical stratum function $f$. Then, for every $C \in N$ and for every $w \in \Sigma^*$, $L_C(C(w)) = M_C(C(w))$ and therefore $L_C$ is independent of the choice of the canonical stratum function.

Proof. We will prove by induction on $n$ that for every $n \in \mathbb{N}$, for every $C \in N$ and for every $w \in \Sigma^*$, if $f(C w) < n$ then $I_{n}(C(w)) = M_{n}(C(w))$. Then, the theorem follows from the definition of $L_C$ and $M_C$.

Induction basis: For $n = 0$ the statement holds trivially, since there is no $(C, w)$ such that $f(C, w) < 0$.

Induction hypothesis: Assume that for every $C \in N$ and for every $w \in \Sigma^*$, if $f(C w) < n$ then $I_{n}(C(w)) = M_{n}(C(w))$.

Induction step: We show that for every $C \in N$ and for every $w \in \Sigma^*$, if $f(C w) < n + 1$ then $I_{n+1}(C(w)) = M_{n+1}(C(w))$. Suppose that $f(C w) < n + 1$. From Lemma 33 it suffices to show that $I_{n+1}(C(w)) = 1$ if and only if $M_{n+1}(C(w)) = 1$. We prove the two directions of this statement:

For the left-to-right direction, suppose that $I_{n+1}(C(w)) = 1$. From Definition 30 it holds that for all $i, j > f(C w)$, $I_j(C(w)) = I_i(C(w))$. Therefore, if $f(C w) < n$, then $I_{n+1}(C(w)) = I_{n+1}(C(w)) = 1$. Then, from the induction hypothesis we obtain $M_{n+1}(C(w)) = 1$. Using the fact that $M_0 \preceq M_{n+1}$, we conclude that $M_{n+1}(C(w)) = 1$.

It remains to prove that for every $C \in N$ and for every $w \in \Sigma^*$, if $f(C w) = n$ and $I_{n+1}(C(w)) = 1$, then $M_{n+1}(C(w)) = 1$.

Define $S_n = \{(A u) \mid A \in N, u \in \sum^*, f(A u) < n\}$. If $(A u) \in S_n$, then $|u| \leq |w|$, since $f$ is a canonical stratum function. Therefore $S_n$ is a finite set. Suppose now that for some $(A u) \in S_n$, it is $M_0(A(u)) = 1$. Then it is also $M_{n+1}(A(u)) = 1$, which implies that there exists an integer $t_{A u}$ such that $\theta_{M_{t_{A u}}}^{I_{n+1}}(A(u)) = 1$. Let $t = \max(t_{A u} \mid (A u) \in S_n$ and $M_0(A(u)) = 1 \cup \{0\})$. Since $S_n$ is finite, $t$ is well defined. Notice that $t$ has the following property: if $(A u) \in S_n$ and $M_0(A(u)) = 1$, then $\theta_{M_0}^{I_{n+1}}(A(u)) = 1$; furthermore, from the monotonicity of $\theta$ with respect to $\leq$, it is also $\theta_{M_{t_{A u}}}^{I_{n+1}}(A(u)) = 1$, for every $i \geq 0$.

Now, if $f(C w) = n$ and $I_{n+1}(C(w)) = 1$, then from Definition 30 it holds that $\Delta_0(C(w)) = 1$; therefore, there exists a derivation of $w$ from $C$, which uses rules of the conjunctive grammar $G_0$. It is easy to check that this derivation has length at least 2. Hence, in order to complete the proof for the left to right direction, it suffices to show that for every $m \geq 2$, for every $C \in N$ and for every $w \in \Sigma^*$, if there is a derivation of $w$ from $C$ of length $m$, which uses rules of $G_0$, then $\theta_{M_{m-1}}^{I_{n+1}}(C(w)) = 1$.

Then, the desired result follows from Definition 18.

The proof uses an inner induction on $m$. For the basis case, consider a derivation of $w$ from $C$ of length 2. Then, this derivation uses two rules of the form $C \rightarrow D_1$ and $D_1 \rightarrow w$. From the construction of $G_0$, we get that $C \rightarrow I$ is a rule in $G$ and $I_0(I(w)) = 1$. We distinguish the following two cases:

Case 1: Suppose that $I$ is a positive conjunct of the form $I = \sigma_1 A_1 \ldots \sigma_r A_r \sigma_{r+1}$, where $r \geq 0$, $\sigma_i \in \Sigma^*$, $A_j \in N$. Then $I_0(I(w)) = 1$ implies that there exist $w_1, \ldots, w_r \in \Sigma^*$ such that $\sigma_1 w_1 \ldots \sigma_r w_r \sigma_{r+1} = w$ and $I_0(A_j(w_j)) = 1$ for all $j$. From the definition of $I_0$, $I_0(A_j(w_j)) = 1$ implies $f(A_j w_j) < n$. Using the outer induction hypothesis we conclude that $M_0(A_j(w_j)) = 1$ for all $j$. Therefore, $\theta_{M_0}^{I_{n+1}}(A_j(w_j)) = 1$, which implies $\theta_{M_0}^{I_{n+1}}(I(w)) = 1$.  


Case 2: Suppose that \( l \) is a negative conjunct of the form \( l = \neg \sigma_1 A_1 \ldots \neg \sigma_r A_r \sigma_{r+1}, \) where \( r \geq 0, \sigma_j \in \Sigma^*, A_j \in N. \) Consider any selection of strings \( w_1, \ldots, w_r \in \Sigma^* \) such that \( \sigma_1 w_1 \ldots \sigma_r w_r \sigma_{r+1} = w \) (if there is no such a selection of strings then \( M_n(l)(w) = 1 \)). Then \( l_n(l)(w) = 1 \) implies that \( l_n(A_p)(w) = 0 \) for some \( p. \) From the definition of local stratification \( f(A_p,w) < n. \) Using the outer induction hypothesis we conclude that \( M_n(A_p)(w_p) = 0. \) Thus, \( M_n(l)(w) = 1. \)

In both cases, since \( C \to l \) is a rule in \( G, \) from the first case of the definition of \( \Theta \) we get \( \Theta^{m+1}_n(C)(w) = 1. \) This completes the proof of the basis case of the inner induction.

Assume now the statement holds for all derivations of length at most \( m \). Consider a derivation of \( w \) from \( C \) of length \( m + 1 \) and let \( C \to D_{i_1} \& \cdots \& D_{i_k}, k \geq 1 \) be the first rule of \( G_n \) used in this derivation. Then for every \( i, 1 \leq i \leq k, \) there exists a derivation of length at most \( m \) from \( D_{i_1} \) to \( w. \)

If this derivation uses a single rule \( D_{i_k} \to w, \) then using the same arguments as in the base case we get that \( \Theta^{m+1}_n(D_{i_k})(w) = 1, \) if \( l_i \) is positive and \( M_n(l_i)(w) = 1 \) if \( l_i \) is negative.

Suppose that the derivation from \( D_{i_1} \) to \( w \) has length greater than \( 1. \) Then we have to distinguish two cases, depending on the length of \( w: \)

Case 1: Suppose that \( |w| \geq 0. \) Then the derivation consists of an application of a rule \( D_{i_1} \to E \) for some \( E \in N, \) followed by a derivation of \( w \) from \( E \) of length at most \( m - 1. \) From the inner induction hypothesis we get \( \Theta^{m+1}_n(E)(w) = 1. \) Now, since \( D_{i_1} \to E \) is a rule in \( G_n, \) from the construction of \( G_n \) we get that \( l_i \) is a positive conjunct of the form \( A_1 \alpha_2 \ldots A_r, \) where \( A_1 \in N, E = A_r, \) for some \( A_r \) with \( 1 \leq p \leq r \) and \( l_n(A_p)(w) = 1 \) for every \( j \neq p. \) From \( E = A_p, \) we obtain that \( \Theta^{m+1}_n(A_p)(w) = 1. \) Furthermore, since \( |e| < |w|, \) from the canonicity of \( f \) we have that \( f(A_j,e) < n. \) Using the outer induction hypothesis we obtain that for every \( j \neq p \) it is \( M_n(A_j)(e) = 1, \) which implies \( \Theta^{m+1}_n(A_j)(e) = 1. \) Combining all the above we get \( \Theta^{m+1}_n(D_{i_1})(w) = 1. \)

Case 2: Suppose that \( |w| = 0 \) (i.e., \( w = e. \)) Then the derivation consists of an application of a rule of the form \( D_{i_1} \to \alpha_1 \ldots \alpha_s, \) where \( \alpha_s \in G \cup e \cup \{ \} \cup \Sigma \), followed by a derivation of \( e \) from \( \alpha_1 \ldots \alpha_s \) of length at most \( m - 1. \) From the construction of \( G_n \) we get that \( l_i \) is a positive conjunct of the form \( A_1 \alpha_2 \ldots A_r \in \Gamma \cap N^*, \) with \( \alpha_s \in \Gamma \) if \( l_n(A_p)(w) = 1 \) and \( \alpha_s = \alpha_i \) otherwise. Consider any \( j \) with \( 1 \leq j \leq r. \) Suppose first that \( l_n(A_j)(e) = 1. \) This implies that \( f(A_j,e) \leq n. \) Using the outer induction hypothesis we obtain that \( M_n(A_j)(e) = 1, \) which implies \( \Theta^{m+1}_n(A_j)(e) = 1. \) Suppose now that \( l_n(A_j)(e) = 0. \) Then, \( \alpha_j = \alpha_i \) and from the derivation of \( e \) from \( \alpha_1 \ldots \alpha_r \) we can obtain a derivation of \( e \) from \( A_j \) of length at most \( m - 1. \) From the inner induction hypothesis we get \( \Theta^{m+1}_n(A_j)(e) = 1. \) Combining all the above we get \( \Theta^{m+1}_n(D_{i_1})(w) = 1. \)

Therefore, for every \( i, 1 \leq i \leq k, \) either \( l_i \) is positive and \( \Theta^{m+1}_n(D_{i_k})(w) = 1, \) or \( l_i \) is negative and \( M_n(l_i)(w) = 1. \) Furthermore, since \( C \to D_{i_1} \& \cdots \& D_{i_k} \) is a rule in \( G_n, \) \( G \) contains the rule \( C \to l_1 \& \cdots \& l_k. \) From the definition of \( \Theta \) we get \( \Theta^{m+1}_n(C)(w) = 1. \) This completes the inner induction step and the right-to-left direction of the outer induction step.

For the right-to-left direction, suppose that \( M_{n+1}(C)(w) = 1. \) If \( f(C,w) < n, \) then from Lemma 33 we have \( M_n(C)(w) \notin \{0,1\}. \) Using the fact that \( M_n \preceq M_{n+1}, \) we conclude that \( M_n(C)(w) = 1. \) From the induction hypothesis we obtain that \( l_n(C)(w) = 1, \) which implies that \( l_n(C)(w) = 1. \)

It remains to prove that for every \( C \in N \) and for every \( w \in \Sigma^*, \) if \( f(C,w) = n \) and \( M_{n+1}(C)(w) = 1 \) then \( l_n(C)(w) = 1. \) In order to prove this, it suffices to show that for every \( C \in N \) and for every \( w \in \Sigma^*, \) if \( f(C,w) = n \) and \( \Theta^{m+1}_n(C)(w) = 1, \) then there exists a derivation from \( C \) to \( w \) that uses rules of \( G_n. \) Then the desired result follows from Definition 18 and 29. The proof is by an inner induction on \( m. \) The basis case is trivial since \( \Theta^{0}_n = \bot. \)

Assume that the statement holds for \( m \) and suppose that \( f(C,w) = n \) and \( \Theta^{m+1}_n(C)(w) = \Theta_n(M_n,\Theta^{m+1}_n(C)(w) = 1. \) Then there exists a rule \( C \to l_1 \& \cdots \& l_k \) in \( G \) such that for all positive \( l_i \) it is \( \Theta^{m+1}_n(l_i)(w) = 1 \) and for all negative \( l_i \) it is \( M_n(l_i)(w) = 1. \) For each \( l_i \) we have to distinguish the following two cases:

Case 1: Suppose that \( l_i \) is a positive conjunct of the form \( l_i = \sigma_1 A_1 \ldots \sigma_r A_r \sigma_{r+1}, \) where \( r \geq 0, \sigma_j \in \Sigma^*, A_j \in N, \) and \( \Theta^{m+1}_n(l_i)(w) = 1. \) Then there exist \( w_1, \ldots, w_r \in \Sigma^* \) such that \( \sigma_1 w_1 \ldots \sigma_r w_r \sigma_{r+1} = w \) and \( \Theta^{m+1}_n(A_j)(w_j) = 1 \) for all \( j. \) Since \( G \) is locally stratified by \( f, \) it is \( f(A_j,w_j) \leq f(C,w) = n \) for all \( j. \) Moreover, if for some \( j \) it is \( f(A_j,w_j) < n, \) then from Lemma 33, using the fact that \( M_n \preceq M_{n+1}, \) we get \( M_n(A_j)(w_j) = 1 \) and from the induction hypothesis we obtain that \( l_n(A_j)(w) = 1. \)

Subcase 1.1: Suppose that for every \( j, f(A_j,w_j) < n. \) Then \( l_n(A_j)(w_j) = 1 \) for all \( j, \) which implies that \( l_n(C)(w) = 1. \) Consequently, \( G_n \) contains the rule of the form \( D_{i_k} \to w, \) which implies that there exists a derivation of length 1 from \( D_{i_k} \) to \( w. \)

Subcase 1.2: Suppose that there exists some \( p \) such that \( f(A_p,w_p) = n \) and \( |w| > 0. \) Since \( f \) is a canonical stratification function, it must be \( w_p = w \) and \( w_j = e \) for \( j \neq p. \) Furthermore, since \( |e| < |w| \), we obtain that for \( j \neq p \) it is \( f(A_j,e) < n. \) Therefore, for \( j \neq p \) it is \( l_n(A_j)(e) = 1. \) Thus, \( G_n \) contains the rule \( D_{i_k} \to A_p. \) Since \( \Theta^{m+1}_n(A_p)(w_p) = 1, \) from the inner induction hypothesis, there
exists a derivation from $A_p$ to $w$ that uses rules of $G_n$. Using the rule $D_j \rightarrow A_p$, this can be extended to a derivation from $D_i$ to $w$.

Subcase 1.3: Suppose that there exists some $p$ such that $f(A_p, w_p) = n$ and $w = \epsilon$. Then, it is $w_j = \epsilon$ for all $j$ and there exists a set of indices $\{p_1, \ldots, p_r\} \subseteq \{1, \ldots, r\}$ such that $f(A_j, \epsilon) = n$ for $j \in \{p_1, \ldots, p_r\}$.

For every $j \in \{p_1, \ldots, p_r\}$, from the definition of $I_n$ we have $I_n(A_j, \epsilon) = 0$. For every $1 \leq j \leq r$ such that $j \notin \{p_1, \ldots, p_r\}$ we have $f(A_j, \epsilon) < n$, which implies that $I_n(A_j, \epsilon) = 1$. Therefore, $G_n$ contains the rule $D_i \rightarrow A_{p_1} \ldots A_{p_r}$. Consider any $j \in \{p_1, \ldots, p_r\}$. Since $\epsilon \in \{A_j\}(\epsilon) = 1$, from the inner induction hypothesis, there exists a derivation from $A_j$ to $\epsilon$ using rules of $G_n$. By applying the rule $D_j \rightarrow A_{p_1} \ldots A_{p_r}$, and then combining the above derivations, we obtain a derivation from $D_i$ to $\epsilon$.

Case 2: Suppose that $I_n$ is a negative conjunct of the form $I_n = \neg \sigma_1 A_1 \ldots \sigma_k A_k \mathcal{S}_{\gamma} + 1$, where $r \geq 0$, $\sigma_i \in \Sigma^*$, $A_j \in N$ and $M_n(I_n)(w) = 1$. Then, for any selection of $w_1, \ldots, w_k \in \Sigma^*$, such that $\sigma_1 w_1 \ldots \sigma_k w_k \mathcal{S}_{\gamma} + 1 = w$, there exists some $p$, such that $M_n(A_p)(w) = 0$. Since $G$ is locally stratified by $f$, we have $f(A_p, w_p) < f(C, w) = n$. From the induction hypothesis we obtain that $I_n(A_p)(w) = 0$. Therefore, $I_n(I_n)(w) = 1$. Consequently, $G_n$ contains a rule of the form $D_i \rightarrow w$, which implies that there exists a derivation of length 1 from $D_i$ to $w$.

We have shown that for every $i$ there exists a derivation from $D_i$ to $w$ using rules of $G_n$. Furthermore, since $G$ contains the rule $C \rightarrow I_1 \& \cdots \& I_k$, $G_n$ contains the rule $C \rightarrow D_1 \& \cdots \& D_k$. The latter rule can be combined with the above derivations, to obtain a derivation from $C$ to $w$. This completes the inner induction step and the right-to-left direction of the outer induction step. $\square$

Notice that, since $L_G = M_G$, from the results in [5] we conclude that $L_G$ is the least fixed point of the operator $\omega$ associated with the grammar $G$.

6. An application of the locally stratified semantics

In this section, we demonstrate that the locally stratified semantics can be used in order to compute the meaning of interesting Boolean grammars, using relatively simple inductive arguments. In particular, we will apply the locally stratified semantics to a grammar given in [11] (see also our earlier Example 27), and we will demonstrate that the meaning of this grammar is the language $L = \{a^{2^n} \mid n \geq 0\}$.

Consider the Boolean grammar $G = (\Sigma, N, P, S)$, where $\Sigma = \{a\}$ and $P$ contains the following rules:

\[
\begin{align*}
S & \rightarrow \& \neg A \mid aB \& \neg B \mid aC \& \neg C \\
A & \rightarrow aB \\
B & \rightarrow \neg C \\
C & \rightarrow \neg E \\
E & \rightarrow \neg A
\end{align*}
\]

Grammar $G$ is locally stratified by the canonical stratum-function $f$ with $f(A, w) = 5 \cdot |w|$, $f(E, w) = 5 \cdot |w| + 1$, $f(C, w) = 5 \cdot |w| + 2$, $f(B, w) = 5 \cdot |w| + 3$ and $f(S, w) = 5 \cdot |w| + 4$. We show that $L_G(S) = L$,

Notice that, since $L_G = M_G$, from the results in [5] we conclude that $L_G$ is the least fixed point of the operator $\omega$ associated with the grammar $G$.

6. An application of the locally stratified semantics

In this section, we demonstrate that the locally stratified semantics can be used in order to compute the meaning of interesting Boolean grammars, using relatively simple inductive arguments. In particular, we will apply the locally stratified semantics to a grammar given in [11] (see also our earlier Example 27), and we will demonstrate that the meaning of this grammar is the language $L = \{a^{2^n} \mid n \geq 0\}$.

Consider the Boolean grammar $G = (\Sigma, N, P, S)$, where $\Sigma = \{a\}$ and $P$ contains the following rules:

\[
\begin{align*}
S & \rightarrow \& \neg A \mid aB \& \neg B \mid aC \& \neg C \\
A & \rightarrow aB \\
B & \rightarrow \neg C \\
C & \rightarrow \neg E \\
E & \rightarrow \neg A
\end{align*}
\]

Grammar $G$ is locally stratified by the canonical stratum-function $f$ with $f(A, w) = 5 \cdot |w|$, $f(E, w) = 5 \cdot |w| + 1$, $f(C, w) = 5 \cdot |w| + 2$, $f(B, w) = 5 \cdot |w| + 3$ and $f(S, w) = 5 \cdot |w| + 4$. We show that $L_G(S) = L$,

Notice that, since $L_G = M_G$, from the results in [5] we conclude that $L_G$ is the least fixed point of the operator $\omega$ associated with the grammar $G$.

6. An application of the locally stratified semantics

In this section, we demonstrate that the locally stratified semantics can be used in order to compute the meaning of interesting Boolean grammars, using relatively simple inductive arguments. In particular, we will apply the locally stratified semantics to a grammar given in [11] (see also our earlier Example 27), and we will demonstrate that the meaning of this grammar is the language $L = \{a^{2^n} \mid n \geq 0\}$.

Consider the Boolean grammar $G = (\Sigma, N, P, S)$, where $\Sigma = \{a\}$ and $P$ contains the following rules:

\[
\begin{align*}
S & \rightarrow \& \neg A \mid aB \& \neg B \mid aC \& \neg C \\
A & \rightarrow aB \\
B & \rightarrow \neg C \\
C & \rightarrow \neg E \\
E & \rightarrow \neg A
\end{align*}
\]

Grammar $G$ is locally stratified by the canonical stratum-function $f$ with $f(A, w) = 5 \cdot |w|$, $f(E, w) = 5 \cdot |w| + 1$, $f(C, w) = 5 \cdot |w| + 2$, $f(B, w) = 5 \cdot |w| + 3$ and $f(S, w) = 5 \cdot |w| + 4$. We show that $L_G(S) = L$,

Notice that, since $L_G = M_G$, from the results in [5] we conclude that $L_G$ is the least fixed point of the operator $\omega$ associated with the grammar $G$.

6. An application of the locally stratified semantics

In this section, we demonstrate that the locally stratified semantics can be used in order to compute the meaning of interesting Boolean grammars, using relatively simple inductive arguments. In particular, we will apply the locally stratified semantics to a grammar given in [11] (see also our earlier Example 27), and we will demonstrate that the meaning of this grammar is the language $L = \{a^{2^n} \mid n \geq 0\}$.

Consider the Boolean grammar $G = (\Sigma, N, P, S)$, where $\Sigma = \{a\}$ and $P$ contains the following rules:

\[
\begin{align*}
S & \rightarrow \& \neg A \mid aB \& \neg B \mid aC \& \neg C \\
A & \rightarrow aB \\
B & \rightarrow \neg C \\
C & \rightarrow \neg E \\
E & \rightarrow \neg A
\end{align*}
\]

Grammar $G$ is locally stratified by the canonical stratum-function $f$ with $f(A, w) = 5 \cdot |w|$, $f(E, w) = 5 \cdot |w| + 1$, $f(C, w) = 5 \cdot |w| + 2$, $f(B, w) = 5 \cdot |w| + 3$ and $f(S, w) = 5 \cdot |w| + 4$. We show that $L_G(S) = L$,

Notice that, since $L_G = M_G$, from the results in [5] we conclude that $L_G$ is the least fixed point of the operator $\omega$ associated with the grammar $G$.
Case 2: $i \mod 5 = 1$. From the induction hypothesis $I_i(A) = I_0 \cap S^{\leq 0}$, $I_i(E) = I_1 \cap S^{\leq 1}$, $I_i(C) = I_2 \cap S^{\leq 2}$, $I_i(B) = I_3 \cap S^{\leq 3}$, $I_i(S) = I_4 \cap S^{\leq 4}$. We will show that $I_{i+1}(A) = I_0 \cap S^{\leq 0}$, $I_{i+1}(E) = I_1 \cap S^{\leq 1}$, $I_{i+1}(C) = I_2 \cap S^{\leq 2}$, $I_{i+1}(B) = I_3 \cap S^{\leq 3}$, $I_{i+1}(S) = I_4 \cap S^{\leq 4}$. Stratum $i$ contains a unique pair $(E, a^n)$, $N_i = \{E\}$ and $P_i$ contains the rule $E \rightarrow D_{-\mathcal{A}}$. Suppose that $a^n \notin I_1$. Then, $a^n \notin L_0$. From the induction hypothesis $a^n \notin I_1(A)$, which implies that the rule $D_{-\mathcal{A}} \rightarrow a^n$ is in $P_i$. Consequently, $\Delta_i(E) = (a^n)$ and $I_{i+1}(E) = I_1 \cup (a^n) = I_1 \cap S^{\leq 1}$. Suppose that $a^n \notin L_1$. Then, $a^n \notin L_0$. From the induction hypothesis $a^n \notin I_1(A)$. Therefore, $a^n \notin I_1(\neg A)$, which implies that the rule $D_{-\mathcal{A}} \rightarrow a^n$ is not in $P_i$. Consequently, $\Delta_i(E) = \emptyset$ and $I_{i+1}(E) = I_1 \cup \emptyset = L_1 \cap S^{\leq 1}$. Moreover, $I_{i+1}(V) = I_1(V)$, for every $V \in \{A, C, B, S\}$.

Case 3: $i \mod 5 = 2$. From the induction hypothesis $I_i(A) = I_2 \cap S^{\leq 2}$, $I_i(E) = I_3 \cap S^{\leq 3}$, $I_i(C) = I_4 \cap S^{\leq 4}$, $I_i(B) = I_0 \cap S^{\leq 0}$, $I_i(S) = I_1 \cap S^{\leq 1}$. We show that $I_{i+1}(A) = I_2 \cap S^{\leq 2}$, $I_{i+1}(E) = I_3 \cap S^{\leq 3}$, $I_{i+1}(C) = I_4 \cap S^{\leq 4}$, $I_{i+1}(B) = I_0 \cap S^{\leq 0}$, $I_{i+1}(S) = I_1 \cap S^{\leq 1}$. Stratum $i$ contains a unique pair $(C, a^n)$, $N_i = \{C\}$ and $P_i$ contains the rule $C \rightarrow D_{-\mathcal{E}}$. Suppose that $a^n \notin L_2$. Then, $a^n \notin L_1$. From the induction hypothesis $a^n \notin I_1(A)$. Therefore, $a^n \notin I_1(\neg A)$, which implies that the rule $D_{-\mathcal{E}} \rightarrow a^n$ is not in $P_i$. Therefore, $\Delta_i(C) = \emptyset$ and $I_{i+1}(C) = I_i(C) \cup \emptyset = L_2 \cap S^{\leq 2}$. Moreover, $I_{i+1}(V) = I_1(V)$, for every $V \in \{A, E, B, S\}$.

Case 4: $i \mod 5 = 3$. This case is similar to Case 3.

Case 5: $i \mod 5 = 4$. From the induction hypothesis $I_i(A) = I_0 \cap S^{\leq 0}$, $I_i(E) = I_1 \cap S^{\leq 1}$, $I_i(C) = I_2 \cap S^{\leq 2}$, $I_i(B) = I_3 \cap S^{\leq 3}$, $I_i(S) = I_4 \cap S^{\leq 4}$. We will show that $I_{i+1}(A) = I_0 \cap S^{\leq 0}$, $I_{i+1}(E) = I_1 \cap S^{\leq 1}$, $I_{i+1}(C) = I_2 \cap S^{\leq 2}$, $I_{i+1}(B) = I_3 \cap S^{\leq 3}$, $I_{i+1}(S) = I_4 \cap S^{\leq 4}$. Stratum $i$ contains a unique pair $(S, a^n)$, $N_i = \{S\}$ and $P_i$ contains the rules $S \rightarrow D_{-\mathcal{A}} \& D_{-\mathcal{E}}$, $S \rightarrow D_{-\mathcal{B}} \& D_{-\mathcal{C}}$, $S \rightarrow D_{-\mathcal{A}} \& D_{-\mathcal{C}}$. Suppose that $a^n \notin L_0$. Then, $a^n \notin L_0 \cap \{(a \circ 0) L_0\}$, which implies $a^n \notin L_0$ and $a^n \notin L_0$. From the induction hypothesis $a^n \notin I_1(\neg A)$, which implies that the rules $D_{-\mathcal{A}} \rightarrow a^n$ and $D_{-\mathcal{A}} \rightarrow a^n$ are in $P_i$. Consequently, $\Delta_i(S) = \emptyset$ and $I_{i+1}(S) = \{a^n\} \cup \emptyset = L_0 \cap S^{\leq 0}$. The cases for the remaining values of $n \mod 3$ are similar.

7. Conclusions and future work

We have defined the class of locally stratified Boolean grammars and have demonstrated that they possess a well-defined semantics. The class of locally stratified Boolean grammars is syntactically broader than that of the stratified ones (see Corollary 26). These new grammars can also be considered as “well-behaved” and useful for applications, since their well-founded semantics is total. There exist however certain questions that appear at present to have no obvious solutions. We believe that getting a further insight on the following issues, will lead us to a better understanding of Boolean grammars in general.

Beyond local stratification: There exist certain natural and useful Boolean grammars that fail to be locally stratified. Consider the following modified version of our motivating example in Section 3:

$$
\begin{align*}
E & \rightarrow \epsilon \\
E & \rightarrow A \ O \\
O & \rightarrow \neg E \\
A & \rightarrow a
\end{align*}
$$

Despite its obvious equivalence to the initial grammar, the above grammar is not locally stratified since its skeleton is not stratified. This same phenomenon occurs in a slightly different form in the area of logic programming. For example, consider the following (locally stratified) logic program:

$$
\begin{align*}
p & \leftarrow q(b) \\
q(a) & \leftarrow \neg p
\end{align*}
$$

This program can be written in an equivalent way as:

$$
\begin{align*}
p & \leftarrow equal(X, b), q(X) \\
q(a) & \leftarrow \neg p, equal(X, X).
\end{align*}
$$

Despite the fact that the two programs above are equivalent from a semantic point of view, the second program is not locally stratified. The reasons that lead to the above problem are not hard to detect. Local stratification is a purely syntactical notion, while the above phenomenon requires a more in-depth (namely semantical) inspection of the grammar (or logic program).
Therefore, given a Boolean grammar, if we would like to use some more powerful notion than local stratification, we would have to perform some kind of semantic analysis regarding the grammar under consideration. For example, consider again the grammar given in the beginning of this section. The decision procedure we introduced in Section 3 fails for this program because we do not know whether the nonterminal \( A \) can produce the empty string or not. In this particular example this can be checked by an easy inspection of the rules of the grammar. However, in other more complicated cases this would require a more thorough inspection of the rules of the grammar. It is possible that based on semantic information one could define more general and effective tests, but it is not obvious how far this approach can take us.

Of course, it is possible that the extensions to the notion of local stratification just discussed, are not really essential. It is conceivable that one could prove that every two-valued Boolean language can be expressed by a locally stratified Boolean grammar. Such a result would give a normal form theorem for two-valued Boolean grammars. Normal form theorems are very useful in formal language theory since they make the study of classes of languages much easier. In our case this would mean that if one wanted to study a property of “two-valued Boolean grammars” it would be sufficient to study the (easily and syntactically definable) class of locally stratified Boolean grammars. This discussion leads us to the next and important class of problems regarding Boolean grammars.

Expressibility of locally stratified Boolean grammars: One of the most important issues in the study of conjunctive and Boolean grammars, is the characterization of their expressibility. At present, our knowledge on this issue is very restricted (the best that has been obtained so far is presented in [11] and also in [3] and [4]). The introduction of the various semantics for Boolean grammars creates a new set of equally interesting and challenging open problems:

- Do there exist Boolean two-valued languages that are expressible under the well-founded semantics but not under the locally stratified semantics? We conjecture that the answer to this question in negative.
- Do there exist Boolean languages that are expressible under the locally stratified semantics but not under the stratified semantics? We believe that the answer to this question is positive but at present we are unable to conjecture a language that could separate the two classes.
- Is the class of stratified languages broader than the class of conjunctive languages? We believe that the answer to this question is positive and a language that could separate the two classes is probably \( \{ww \mid w \in \{a,b\}^* \} \). As it can easily be seen, if the answer to this question proves to be positive then this would imply that the conjunctive languages are not closed under complement, and vice-versa.

Notice that a positive answer to any one of the above questions would immediately solve a very important open problem in the theory of Boolean languages, namely their separation from conjunctive languages. Additionally, answering any one of the above questions would give further insights on the power of negation in the context of formal language theory.

References