Parsing as Abstract Interpretation of Grammar Semantics

Patrick COUSOT
Département d'informatique
École normale supérieure
45, rue d'Ulm
75230 Paris cedex 05, France
cousot@ens.fr
http://www.di.ens.fr/~cousot

Radhia COUSOT
Laboratoire d'informatique
CNRS & École polytechnique
91128 Palaiseau cedex
France
rcousot@lix.polytechnique.fr
http://lix.polytechnique.fr/~rcousot

Received 15 January 2001, accepted 16 August 2001

Abstract

Earley's parsing algorithm is shown to be an abstract interpretation of a refinement of the derivation semantics of context-free grammars.

1 Introduction

Abstract interpretation is a theory of the approximation of the mathematical structures involved in the formalization of the semantics of computer systems [6]. It offers a unifying point of view on *static program analysis* [4] (including *data flow analysis* [6, 8] and *typing* [2]) of specification and programming languages, *model-checking* [8], etc. Following this synthetic point of view, we show that Earley's parsing algorithm [9] can be formally designed by abstract interpretation of a refinement of the derivation semantics of context-free grammars.

2 Context-free Grammars, Derivations, Generated Language and Parsing

The set of *finite words* on an alphabet $\mathcal{A}$ is denoted $\mathcal{A}^\star$. This includes the empty word ϵ. A *language* on the alphabet $\mathcal{A}$ is a subset of $\mathcal{A}^\star$. A *context-free grammar* $\mathcal{G}$ is a quadruple $\langle \mathcal{N}, \mathcal{T}, \mathcal{P}, A\rangle$, where:

- $X, Y, \ldots \in \mathcal{N}$ is the finite set of *nonterminals*;
- the distinguished nonterminal $A \in \mathcal{N}$ is the *axiom*;
- $a, b, \ldots \in \mathcal{T}$, such that $\mathcal{T} \cap \mathcal{N} = \emptyset$, is the finite set of *terminals*;
- $\mathcal{V} \triangleq (\mathcal{N} \cup \mathcal{T}) \setminus \{A\}$ is the *vocabulary*;
- $\alpha, \beta, \ldots \in \mathcal{V}^\star$ is the set of finite words on the vocabulary $\mathcal{V}$;

- $\mathcal{P} \subseteq \mathcal{N} \times \mathcal{V}^\star$ is the finite set of *productions*, $\langle X, \alpha\rangle \in \mathcal{P}$ being written $X \xrightarrow{\mathcal{G}} \alpha$.

Observe that the axiom A cannot appear on the righthand side α of productions $\langle X, \alpha\rangle$. This restriction can be easily bypassed by introducing a new axiom A' such that $A' \xrightarrow{\mathcal{G}} A$.

The *semantics* of a grammar $\mathcal{G}$ can be defined as the *derivation relation* $\overset{\mathcal{G}}{\Longrightarrow}$ which is the least relation such that a nonterminal derives to the righthand side of any of its productions, as specified by the following axiom schema ($X \in \mathcal{N}$, $\alpha \in \mathcal{V}^\star$):

$$X \overset{\mathcal{G}}{\Longrightarrow} \alpha, \quad \text{whenever } X \xrightarrow{\mathcal{G}} \alpha \tag{1}$$

and a word derives to another word by replacement of a nonterminal by any one of its derivations, as specified by the following inference rule schema:

$$\frac{X \overset{\mathcal{G}}{\Longrightarrow} \alpha Y \gamma, \quad Y \overset{\mathcal{G}}{\Longrightarrow} \beta}{X \overset{\mathcal{G}}{\Longrightarrow} \alpha\beta\gamma}, \quad X, Y \in \mathcal{N}, \alpha, \beta, \gamma \in \mathcal{V}^\star\ . \tag{2}$$

The *leftmost derivation* $\overset{\mathcal{G}}{\Longrightarrow}_\ell$ is defined in the same way but for the nonterminal replacement which is restricted to the leftmost nonterminal:

$$X \overset{\mathcal{G}}{\Longrightarrow}_\ell \alpha, \qquad \text{whenever } X \xrightarrow{\mathcal{G}} \alpha \tag{3}$$

$$\frac{X \overset{\mathcal{G}}{\Longrightarrow}_\ell \alpha Y \gamma, \quad Y \overset{\mathcal{G}}{\Longrightarrow}_\ell \beta}{X \overset{\mathcal{G}}{\Longrightarrow}_\ell \alpha\beta\gamma}, \quad X, Y \in \mathcal{N},\ \alpha \in \mathcal{T}^\star,\ \beta, \gamma \in \mathcal{V}^\star \tag{4}$$

Similarly, the *leftmost derivation from the axiom* $\overset{\mathcal{G}}{\Longrightarrow}_{A,\ell}$ is the restriction of the leftmost derivation $\overset{\mathcal{G}}{\Longrightarrow}_\ell$ to nonterminals deriving from the grammar axiom:

$$A \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \alpha, \qquad \text{whenever } A \xrightarrow{\mathcal{G}} \alpha \tag{5}$$

$$\frac{X \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \alpha Y \gamma}{Y \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \beta}, \qquad X, Y \in \mathcal{N},\ \alpha \in \mathcal{T}^\star,\ \gamma \in \mathcal{V}^\star, Y \xrightarrow{\mathcal{G}} \beta \tag{6}$$

$$\frac{X \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \alpha Y \gamma, \quad Y \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \beta}{X \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \alpha\beta\gamma}, \quad X, Y \in \mathcal{N},\ \alpha \in \mathcal{T}^\star,\ \beta, \gamma \in \mathcal{V}^\star \tag{7}$$

The *language* $\mathcal{L}_\mathcal{G}$ *generated by a grammar* $\mathcal{G}$ is the set of terminal words deriving from the axiom A:

$$\mathcal{L}_\mathcal{G} \quad \triangleq \quad \{\alpha \in \mathcal{T}^\star \mid A \overset{\mathcal{G}}{\Longrightarrow} \alpha\}\ . \tag{8}$$

Equivalently, the language generated by a grammar can be defined using the leftmost derivation [12, Theorem 4.1.1]:

$$\mathcal{L}_\mathcal{G} \quad = \quad \{\alpha \in \mathcal{T}^\star \mid A \overset{\mathcal{G}}{\Longrightarrow}_\ell \alpha\}\ . \tag{9}$$

Equivalently, we can also use the leftmost derivation from the axiom:

Lemma 1

$$\mathcal{L}_{\mathcal{G}} \quad = \quad \{\alpha \in \mathcal{T}^\star \mid A \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \alpha\} \ . \tag{10}$$

Proof Obviously if we have proved $X \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \alpha$ we can prove $X \overset{\mathcal{G}}{\Longrightarrow}_{\ell} \alpha$ using (3) for either (5) or (6) and (4) for (7).

Reciprocally, we prove that if $X = A$ or $\exists \eta \in \mathcal{T}^\star, \xi \in \mathcal{V}^\star : A \overset{\mathcal{G}}{\Longrightarrow}_{\ell} \eta X \xi$ and $X \overset{\mathcal{G}}{\Longrightarrow}_{\ell} \delta$ then $X \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \delta$.

The proof is on the length of the proof of $X \overset{\mathcal{G}}{\Longrightarrow}_{\ell} \delta$ by the formal system (3) – (4).

- If we have proved $X \overset{\mathcal{G}}{\Longrightarrow}_{\ell} \delta$ by (5) then $X \overset{\mathcal{G}}{\longrightarrow} \delta$ and there are two subcases:
  - If $X = A$ then $X \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \delta$ follows from (5);
  - Otherwise, there exist $\eta \in \mathcal{T}^\star$ and $\xi \in \mathcal{V}^\star$ such that $A \overset{\mathcal{G}}{\Longrightarrow}_{\ell} \eta X \xi$. So, by induction, we can prove that $A \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \eta X \xi$ whence $X \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \delta$ by (6).
- Otherwise, we have proved $X \overset{\mathcal{G}}{\Longrightarrow}_{\ell} \delta$ by (4) so we have $\delta = \alpha\beta\gamma$, $\alpha \in \mathcal{T}^\star$ and we made subproofs for $X \overset{\mathcal{G}}{\Longrightarrow}_{\ell} \alpha Y \gamma$ and $Y \overset{\mathcal{G}}{\Longrightarrow}_{\ell} \beta$. There are now two subcases:
  - If $X = A$ then by induction $X \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \alpha Y \gamma$ that is $A \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \alpha Y \gamma$ with $\alpha \in \mathcal{T}^\star$ so that again by induction $Y \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \beta$. By (7), we conclude that $X \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \alpha\beta\gamma$ that is $X \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \delta$;
  - Otherwise, there exist $\eta \in \mathcal{T}^\star$ and $\xi \in \mathcal{V}^\star$ such that $A \overset{\mathcal{G}}{\Longrightarrow}_{\ell} \eta X \xi$. By $X \overset{\mathcal{G}}{\Longrightarrow}_{\ell} \alpha Y \gamma$ and (4), it follows that $A \overset{\mathcal{G}}{\Longrightarrow}_{\ell} \eta \alpha Y \gamma \xi$ with $\eta\alpha \in \mathcal{T}^\star$. Hence we can apply the induction hypothesis and therefore prove that $Y \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \beta$. By (7), we conclude that $X \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \alpha\beta\gamma$, whence $X \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \delta$.

We conclude that $A \overset{\mathcal{G}}{\Longrightarrow}_{\ell} \alpha$ if and only if $A \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \alpha$ so that (9) implies (10). □

Parsing of a given terminal word $\omega \in \mathcal{T}^\star$ for a given grammar $\mathcal{G}$ consists in deciding whether this word ω belongs to the language generated by the grammar $\mathcal{G}$: $\omega \in \mathcal{L}_{\mathcal{G}}$.

3 Fixpoint Semantics of Formal Systems

It is well-known that formal systems specify a least fixpoint [1, 7]. The axioms and rule schemata of a formal system are interpreted as rule instances $\Phi \triangleq \left\{ \dfrac{P_i}{c_i} \;\middle|\; i \in \Delta \right\}$ on a given *universe* $\mathcal{U}$ where for all $i \in \Delta$, $P \subseteq \mathcal{U}$ is the *premise* (which is the empty set $\emptyset$ for axiom instances) and $c_i \in \mathcal{U}$ is the *conclusion* of the *rule instance* $\dfrac{P_i}{c_i}$. The subset of the universe $\mathcal{U}$ specified by the formal system Φ is defined as its semantics $[\![\Phi]\!] \triangleq \mathrm{lfp}^{\subseteq} F_\Phi$ where the *consequence operator*:

$$F_\Phi(X) \quad \triangleq \quad \{c_i \mid i \in \Delta \wedge P_i \subseteq X\} \tag{11}$$

is the set of valid consequences of the hypothesis X. The consequence operator F_Φ on $\wp(\mathcal{U})$ is $\subseteq$-monotonic so that the least fixpoint $\mathrm{lfp}^{\subseteq} F_\Phi$ does exist [13]. The fixpoint semantics is equivalent to the more traditional one based on *formal proofs* [1].

For example the formal system (5) – (7) defines the leftmost derivation from the grammar axiom as:

$$\begin{aligned}
\overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \quad &= \quad \mathrm{lfp}^{\subseteq} \mathcal{D}^{\mathcal{G}}_{A,\ell}\ , \qquad (12)\\
\mathcal{D}^{\mathcal{G}}_{A,\ell}(R) \quad &\triangleq \quad \{\langle A, \alpha\rangle \mid A \overset{\mathcal{G}}{\longrightarrow} \alpha\}\\
&\quad \cup \{\langle Y, \beta\rangle \mid \langle X, \alpha Y \gamma\rangle \in R \wedge \alpha \in \mathcal{T}^\star \wedge Y \overset{\mathcal{G}}{\longrightarrow} \beta\}\\
&\quad \cup \{\langle X, \alpha\beta\gamma\rangle \mid \langle X, \alpha Y \gamma\rangle \in R \wedge \alpha \in \mathcal{T}^\star \wedge \langle Y, \beta\rangle \in R\}\ .
\end{aligned}$$

4 Earley's Parsing Algorithm

4.1 Earley's Items

Given a terminal word $\omega \in \mathcal{T}^\star$, $\omega = \omega_1 \ldots \omega_n$, $n \geq 0$ (which is ϵ when $n = 0$), *Earley's parsing algorithm* [9, 11] involves *Earley's items* which are quintuples written:

$$\langle X \to \alpha{\bullet}\beta, i, j\rangle$$

where $X \overset{\mathcal{G}}{\longrightarrow} \alpha\beta$ is a production of the given grammar $\mathcal{G}$ and $0 \leq i \leq j \leq n$. A *valid Earley's item* is an assertion or judgement stating that $\alpha \overset{\mathcal{G}}{\Longrightarrow} \omega_{i+1} \ldots \omega_j$ (that is $\alpha \overset{\mathcal{G}}{\Longrightarrow} \epsilon$ when $i = j$). Valid Earley's items are derived left to right and top-down starting from the grammar axiom. The set $\mathcal{I}^{\mathrm{E}}_{\mathcal{G},\omega}$ of valid Earley's items for the gammar $\mathcal{G}$ and input word ω is specified by the formal system (13) – (16) below.

4.2 Rule-Based Specification of Earley's Parsing Algorithm

The *initialization axioms* are instances of the following schema (for all productions $A \overset{\mathcal{G}}{\longrightarrow} \gamma$ of the grammar axiom A):

$$\langle A \to {\bullet}\gamma, 0, 0\rangle\ . \tag{13}$$

The *derivation rules* are instances of the following schema (for all productions $X \overset{\mathcal{G}}{\longrightarrow} \alpha Y \beta$ and $Y \overset{\mathcal{G}}{\longrightarrow} \gamma$ of the grammar $\mathcal{G}$ and $0 \leq i \leq j \leq n$):

$$\frac{\langle X \to \alpha{\bullet}Y\beta, i, j\rangle}{\langle Y \to {\bullet}\gamma, j, j\rangle}\ . \tag{14}$$

The *reduction rule schema* is (for all productions $X \overset{\mathcal{G}}{\longrightarrow} \alpha Y \beta$ and $Y \overset{\mathcal{G}}{\longrightarrow} \gamma$ of the grammar $\mathcal{G}$ and $0 \leq k \leq i \leq j \leq n$):

$$\frac{\langle X \to \alpha{\bullet}Y\beta, k, i\rangle, \quad \langle Y \to \gamma{\bullet}, i, j\rangle}{\langle X \to \alpha Y{\bullet}\beta, k, j\rangle}\ . \tag{15}$$

The *advance rule schema* is (for all productions $X \xrightarrow{\mathcal{G}} \alpha a \beta$ of the grammar $\mathcal{G}$ and $0 \leq i < j \leq n$ such that $a = \omega_j$):

$$\frac{\langle X \to \alpha{\bullet}\omega_j\beta, i, j-1\rangle}{\langle X \to \alpha\omega_j{\bullet}\beta, i, j\rangle} \ . \tag{16}$$

The parsing succeeds, that is $\omega \in \mathcal{L}_{\mathcal{G}}$, if and only if one can derive a final Earley's item of the form $\langle A \to \gamma{\bullet}, 0, n\rangle$ where A is the grammar axiom.

4.3 Fixpoint Specification of Earley's Parsing Algorithm

The derivation of the set $\mathcal{I}^{\mathrm{E}}_{\mathcal{G},\omega}$ of valid Earley's items by the formal system (13) – (16) consists in computing the least fixpoint:

$$\begin{aligned}
\mathcal{I}^{\mathrm{E}}_{\mathcal{G},\omega} &\triangleq \operatorname{lfp}^{\subseteq} \mathcal{F}^{\mathrm{E}}_{\mathcal{G},\omega} \ , & (17)\\
\mathcal{F}^{\mathrm{E}}_{\mathcal{G},\omega}(I) &\triangleq \{\langle A \to {\bullet}\gamma, 0, 0\rangle \mid A \xrightarrow{\mathcal{G}} \gamma\}\\
&\quad \cup \{\langle Y \to {\bullet}\gamma, j, j\rangle \mid \langle X \to \alpha{\bullet}Y\beta, i, j\rangle \in I\}\\
&\quad \cup \{\langle X \to \alpha Y{\bullet}\beta, k, j\rangle \mid \langle X \to \alpha{\bullet}Y\beta, k, i\rangle \in I \wedge \langle Y \to \gamma{\bullet}, i, j\rangle \in I\}\\
&\quad \cup \{\langle X \to \alpha\omega_j{\bullet}\beta, i, j\rangle \mid \langle X \to \alpha{\bullet}\omega_j\beta, i, j-1\rangle \in I\} \ .
\end{aligned}$$

The Earley's parsing algorithm [9] terminates by checking that a final item is valid, so that the correctness of the original algorithm and its variants can be specified as:

$$\omega \in \mathcal{L}_{\mathcal{G}} \iff \langle A \to \gamma{\bullet}, 0, n\rangle \in \mathcal{I}^{\mathrm{E}}_{\mathcal{G},\omega} \ . \tag{18}$$

5 Elements of Abstract Interpretation

5.1 The Abstraction

The approximation or abstraction of a semantics is specified by a *Galois connection* [6] that is a pair of maps $\boldsymbol{\alpha} \in L \mapsto M$ and $\boldsymbol{\gamma} \in M \mapsto L$ between posets $\langle L, \leq\rangle$ and $\langle M, \sqsubseteq\rangle$ satisfying $\forall x \in L : \forall y \in M : \boldsymbol{\alpha}(x) \sqsubseteq y \iff x \leq \boldsymbol{\gamma}(y)$ which is written $\langle L, \leq\rangle \underset{\boldsymbol{\alpha}}{\overset{\boldsymbol{\gamma}}{\leftrightarrows}} \langle M, \sqsubseteq\rangle$.

An equivalent definition is $\boldsymbol{\alpha} \in L \mapsto M$ and $\boldsymbol{\gamma} \in M \mapsto L$ are monotonic, $\boldsymbol{\alpha} \circ \boldsymbol{\gamma} \,\dot{\sqsubseteq}\, \mathbf{1}_M$ and $\mathbf{1}_L \,\dot{\leq}\, \boldsymbol{\gamma} \circ \boldsymbol{\alpha}$ where $f \,\dot{\preceq}\, g$ is the pointwise extension of $\preceq$ that is $\forall x \in L : f(x) \preceq g(x)$ and $\mathbf{1}_S$ is the identity map $\forall x \in S : \mathbf{1}_S(x) = x$ on the set S.

We will use the fact that if $\langle L, \leq\rangle$ is a complete lattice and $\boldsymbol{\alpha}$ preserves least upper bounds then it has a unique adjoint $\boldsymbol{\gamma}$ such that $\langle L, \leq\rangle \underset{\boldsymbol{\alpha}}{\overset{\boldsymbol{\gamma}}{\leftrightarrows}} \langle M, \sqsubseteq\rangle$, which is $\boldsymbol{\gamma}(y) = \bigvee\{x \mid \alpha(x) \sqsubseteq y\}$.

5.2 The Abstract Interpretation of the Semantics

If $\langle L, \leq\rangle$ is a complete lattice and $f \in L \mapsto L$ is a monotone map on L, then it has a least fixpoint $\operatorname{lfp}^{\leq} f$ [13] which is interpreted as a *concrete semantics*. The monotone map $g \in M \mapsto M$ on M is a said to be a *locally complete abstraction* of f if and only if $\boldsymbol{\alpha} \circ f = g \circ \boldsymbol{\alpha}$ (see [6, 7.1.0.4 (3)]). This implies *fixpoint* *completeness* in that the *abstract semantics* $\text{lfp}^{\leq} g = \boldsymbol{\alpha}(\text{lfp}^{\leq} f)$ is the precise or exact abstraction of the concrete semantics $\text{lfp}^{\leq} f$ by the abstraction function $\boldsymbol{\alpha}$:

Lemma 2 If $\langle L, \leq\rangle$ is a complete lattice, $\langle L, \leq\rangle \underset{\boldsymbol{\alpha}}{\overset{\boldsymbol{\gamma}}{\leftrightarrows}} \langle M, \sqsubseteq\rangle$, $f \in L \mapsto L$ and $g \in M \mapsto M$ are monotone maps and $\boldsymbol{\alpha} \circ f = g \circ \boldsymbol{\alpha}$ then $\boldsymbol{\alpha}(\text{lfp}^{\leq} f) = \text{lfp}^{\sqsubseteq} g$.

Proof $\boldsymbol{\alpha} \circ f \circ \boldsymbol{\gamma} = g \circ \boldsymbol{\alpha} \circ \boldsymbol{\gamma} \mathrel{\dot{\sqsubseteq}} g$ by monotony and $\boldsymbol{\alpha}(\text{lfp}^{\leq} f) = \text{lfp}^{\sqsubseteq} g$ by [6, 7.1.0.4 (3)]. □

Numerous examples of locally complete abstractions of the derivation semantics of context-free grammars are given in [3]. In this paper, we show that parsing is another one.

6 Concrete Grammar Item Semantics

Our task is now to show that Earley's parsing algorithm (17) is an abstract interpretation of the grammar semantics. We consider a refinement of the left-most derivation from the axiom semantics (12) in order to take into account the possible contexts of derivations.

6.1 Grammar Items

The grammar semantics defines *grammar items* which are quintuples written:

$$[\lambda, X \to \alpha \bullet \beta, \gamma] \ ,$$

where $\lambda, \gamma \in \mathcal{T}^\star$ and $X \xrightarrow{\mathcal{G}} \alpha\beta$. The interpretation of a valid grammar item is that there exists $\eta \in \mathcal{V}^\star$ such that $A \overset{\mathcal{G}}{\Longrightarrow} \lambda X \eta$, $X \xrightarrow{\mathcal{G}} \alpha\beta$ and $\alpha \overset{\mathcal{G}}{\Longrightarrow} \gamma$.

The set $\mathcal{I}_{\mathcal{G}}$ of *valid grammar items* is defined by the formal system (19) – (22) below.

6.2 Rule-Based Specification of the Grammar Item Semantics

The *initialization axiom schema* is (for all productions $A \xrightarrow{\mathcal{G}} \beta$ of the grammar axiom A):

$$[\epsilon, A \to \bullet\beta, \epsilon] \ . \tag{19}$$

The *derivation rule schema* is (for all productions $X \xrightarrow{\mathcal{G}} \alpha Y \beta$ and $Y \xrightarrow{\mathcal{G}} \delta$ of the grammar $\mathcal{G}$):

$$\frac{[\lambda, X \to \alpha \bullet Y \beta, \gamma]}{[\lambda\gamma, Y \to \bullet\delta, \epsilon]} \ . \tag{20}$$

The *reduction rule schema* is (for all productions $X \xrightarrow{\mathcal{G}} \alpha Y \beta$ and $Y \xrightarrow{\mathcal{G}} \gamma$ of the grammar $\mathcal{G}$):

$$\frac{[\lambda, X \to \alpha \bullet Y \beta, \gamma], \quad [\lambda\gamma, Y \to \delta\bullet, \xi]}{[\lambda, X \to \alpha Y \bullet \beta, \gamma\xi]} \ . \tag{21}$$

The *advance rule schema* is (for all productions $X \xrightarrow{\mathcal{G}} \alpha a \beta$ of the grammar $\mathcal{G}$):

$$\frac{[\lambda, X \to \alpha \bullet a\beta, \gamma]}{[\lambda, X \to \alpha a \bullet \beta, \gamma a]} \ . \tag{22}$$

The derivation context from the axiom is always empty since the axiom never appears in the righthand side of production:

Lemma 3 If $[\lambda, A \to \alpha \bullet \beta, \gamma] \in \mathcal{I}_{\mathcal{G}}$ then $\lambda = \epsilon$.

Proof We proceed by induction on the length of the proof that $[\lambda, A \to \alpha \bullet \beta, \gamma] \in \mathcal{I}_{\mathcal{G}}$ using (19) – (22).

This is obvious for the basis by (19). For the induction step, we cannot conclude the proof with (20) because there would be a grammar production of the form $\langle X, \alpha A \alpha \rangle$. So the proof ends with the use of either (21) or (22) and in both cases $\lambda = \epsilon$ follows by induction. $\square$

6.3 Fixpoint Specification of the Grammar Item Semantics

In fixpoint form, the grammar item semantics is:

$$\begin{aligned}
\mathcal{I}_{\mathcal{G}} &= \operatorname{lfp}^{\subseteq} \mathcal{F}_{\mathcal{G}} \ , \\
\mathcal{F}_{\mathcal{G}}(I) &\triangleq \{[\epsilon, A \to \bullet\beta, \epsilon] \mid A \xrightarrow{\mathcal{G}} \beta\} \\
&\quad \cup \{[\lambda\gamma, Y \to \bullet\delta, \epsilon] \mid [\lambda, X \to \alpha \bullet Y\beta, \gamma] \in I \wedge Y \xrightarrow{\mathcal{G}} \delta\} \\
&\quad \cup \{[\lambda, X \to \alpha Y \bullet \beta, \gamma\xi] \mid [\lambda, X \to \alpha \bullet Y\beta, \gamma] \in I \wedge [\lambda\gamma, Y \to \delta\bullet, \xi] \in I\} \\
&\quad \cup \{[\lambda, X \to \alpha a \bullet \beta, \gamma a] \mid [\lambda, X \to \alpha \bullet a\beta, \gamma] \in I\} \ .
\end{aligned} \tag{23}$$

7 The Leftmost Derivation from the Axiom is a Complete Abstraction of the Grammar Item Semantics

7.1 The Abstraction

We consider the elementwise abstraction:

$$\boldsymbol{\alpha}^{\ell}(I) \triangleq \{\langle X, \gamma\beta \rangle \mid \exists \lambda \in \mathcal{T}^{\star} : [\lambda, X \to \alpha \bullet \beta, \gamma] \in I\} \ . \tag{24}$$

$\boldsymbol{\alpha}^{\ell}$ is a complete $\cup$-morphism so it is the lower adjoint of a Galois connection:

$$\langle \wp(\mathcal{T}^{\star} \times \mathcal{N} \times \mathcal{V}^{\star} \times \mathcal{V}^{\star} \times \mathcal{T}^{\star}), \subseteq \rangle \ \underset{\boldsymbol{\alpha}^{\ell}}{\overset{\boldsymbol{\gamma}^{\ell}}{\leftrightarrows}} \ \langle \wp(\mathcal{N} \times \mathcal{V}^{\star}), \subseteq \rangle \ . \tag{25}$$

7.2 The Abstract Interpretation of the Semantics

The leftmost derivation from the axiom semantics is a complete abstract interpretation of the grammar item semantics:

Lemma 4

$$\overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \ = \ \boldsymbol{\alpha}^{\ell}(\mathcal{I}_{\mathcal{G}}) \ . \tag{26}$$

Proof

$$
\begin{array}{ll}
& \boldsymbol{\alpha}^\ell \circ \mathcal{F}_\mathcal{G}(I) \\
= & \wr\text{by definition (24) of } \boldsymbol{\alpha}^\ell \text{ and (23) of } \mathcal{F}_\mathcal{G}\wr \\
& \{\langle A, \beta\rangle \mid A \xrightarrow{\mathcal{G}} \beta\} \\
& \cup \{\langle Y, \delta\rangle \mid \exists \lambda \in \mathcal{T}^\star, \alpha, \beta \in \mathcal{V}^\star : [\lambda, X \to \alpha \centerdot Y\beta, \gamma] \in I \wedge Y \xrightarrow{\mathcal{G}} \delta\} \\
& \cup \{\langle X, \gamma\xi\beta\rangle \mid \exists \lambda \in \mathcal{T}^\star : [\lambda, X \to \alpha \centerdot Y\beta, \gamma] \in I \wedge [\lambda\gamma, Y \to \delta \centerdot, \xi] \in I\} \\
& \cup \{\langle X, \gamma a\beta\rangle \mid \exists \lambda \in \mathcal{T}^\star : [\lambda, X \to \alpha \centerdot a\beta, \gamma] \in I\} \; . \\
= & \wr\text{by definition (24) of } \boldsymbol{\alpha}^\ell \text{ so that } \exists \lambda \in \mathcal{T}^\star : [\lambda, X \to \alpha \centerdot \beta, \gamma] \in I \text{ if and} \\
& \quad \text{only if } \langle X, \gamma\beta\rangle \in \boldsymbol{\alpha}^\ell(I) \text{ and } \gamma \in \mathcal{T}^\star\wr \\
& \{\langle A, \beta\rangle \mid A \xrightarrow{\mathcal{G}} \beta\} \\
& \cup \{\langle Y, \delta\rangle \mid \langle X, \gamma Y\beta\rangle \in \boldsymbol{\alpha}^\ell(I) \wedge \gamma \in \mathcal{T}^\star \wedge Y \xrightarrow{\mathcal{G}} \delta\} \\
& \cup \{\langle X, \gamma\xi\beta\rangle \mid \langle X, \gamma Y\beta\rangle \in \boldsymbol{\alpha}^\ell(I) \wedge \gamma \in \mathcal{T}^\star \wedge \langle Y, \xi\rangle \in \boldsymbol{\alpha}^\ell(I)\} \\
& \cup \{\langle X, \gamma a\beta\rangle \mid \langle X, \gamma a\beta\rangle \in \boldsymbol{\alpha}^\ell(I) \wedge \gamma \in \mathcal{T}^\star\} \; . \\
= & \mathcal{F}^\sharp_\mathcal{G} \circ \boldsymbol{\alpha}^\ell(I),
\end{array}
$$

by defining:

$$
\begin{array}{rcl}
\mathcal{F}^\sharp_\mathcal{G}(R) & \triangleq & \{\langle A, \alpha\rangle \mid A \xrightarrow{\mathcal{G}} \alpha\} \\
& & \cup \{\langle Y, \beta\rangle \mid \langle X, \alpha Y\gamma\rangle \in R \wedge \alpha \in \mathcal{T}^\star \wedge Y \xrightarrow{\mathcal{G}} \beta\} \\
& & \cup \{\langle X, \alpha\beta\gamma\rangle \mid \langle X, \alpha Y\gamma\rangle \in R \wedge \alpha \in \mathcal{T}^\star \wedge \langle Y, \beta\rangle \in R\} \\
& & \cup \{\langle X, \alpha a\beta\rangle \mid \langle X, \alpha a\beta\rangle \in R \wedge \alpha \in \mathcal{T}^\star\} \; .
\end{array} \tag{27}
$$

By lemma 2, we conclude that $\boldsymbol{\alpha}^\ell(\mathrm{lfp}^{\subseteq} \mathcal{F}_\mathcal{G}) = \mathrm{lfp}^{\subseteq} \mathcal{F}^\sharp_\mathcal{G}$. Since $\mathcal{F}^\sharp_\mathcal{G}(R) = \mathcal{D}^\mathcal{G}_{A,\ell}(R) \cup \{\langle X, \alpha a\beta\rangle \mid \langle X, \alpha a\beta\rangle \in R \wedge \alpha \in \mathcal{T}^\star\}$ and the last term $\{\langle X, \alpha a\beta\rangle \mid \langle X, \alpha a\beta\rangle \in R \wedge \alpha \in \mathcal{T}^\star\}$ of $\mathcal{F}^\sharp_\mathcal{G}(R)$ in (27) adds no new element to the transfinite iterates [5] of $\mathrm{lfp}^{\subseteq} \mathcal{F}^\sharp_\mathcal{G}(R)$, we have $\mathrm{lfp}^{\subseteq} \mathcal{F}^\sharp_\mathcal{G}(R) = \mathrm{lfp}^{\subseteq} \mathcal{D}^\mathcal{G}_{A,\ell}$ proving $\boldsymbol{\alpha}^\ell(\mathrm{lfp}^{\subseteq} \mathcal{F}_\mathcal{G}) = \mathrm{lfp}^{\subseteq} \mathcal{D}^\mathcal{G}_{A,\ell}$ whence (26) by (12) and (23). □

8 Item Semantics Based Specification of the Language Generated by a Grammar

It directly follows from (26) that the language $\mathcal{L}_\mathcal{G}$ generated by a grammar $\mathcal{G}$ traditionally defined by (8) can be equivalently defined using the grammar item semantics $\mathcal{I}_\mathcal{G}$:

Corollary 5

$$
\mathcal{L}_\mathcal{G} \quad = \quad \{\gamma \in \mathcal{T}^\star \mid \exists \lambda \in \mathcal{T}^\star : [\lambda, A \to \alpha \centerdot, \gamma] \in \mathcal{I}_\mathcal{G}\} \; . \tag{28}
$$

Proof We have:

$$
\begin{array}{ll}
& \langle X, \delta\rangle \in \boldsymbol{\alpha}^\ell(\mathcal{I}_\mathcal{G}) \wedge \delta \in \mathcal{T}^\star \\
\Longleftrightarrow & \wr\text{by definition (24) of } \boldsymbol{\alpha}^\ell\wr
\end{array}
$$

$\exists \lambda, \beta, \gamma \in \mathcal{T}^\star : \delta = \gamma\beta \wedge [\lambda, X \to \alpha\bullet\beta, \gamma] \in \mathcal{I}_\mathcal{G}$

$\Longrightarrow$ $\wr$since $[\lambda, X \to \alpha\bullet\beta, \gamma] \in \mathcal{I}_\mathcal{G} \wedge \beta \in \mathcal{T}^\star$ implies $[\lambda, X \to \alpha\beta\bullet, \gamma\beta] \in \mathcal{I}_\mathcal{G}$ by (22) and induction on the length of $\beta$$\wr$

$\exists \lambda, \beta \in \mathcal{T}^\star : [\lambda, X \to \alpha\beta\bullet, \delta] \in \mathcal{I}_\mathcal{G}$

$\Longrightarrow$ $\wr$by definition (24) of $\boldsymbol{\alpha}^\ell$$\wr$

$\langle X, \delta \rangle \in \boldsymbol{\alpha}^\ell(\mathcal{I}_\mathcal{G})$,

proving that for $\delta \in \mathcal{T}^\star$ we have the equivalence:

$$\langle X, \delta \rangle \in \boldsymbol{\alpha}^\ell(\mathcal{I}_\mathcal{G}) \Longleftrightarrow \exists \lambda \in \mathcal{T}^\star : [\lambda, X \to \alpha\bullet, \delta] \in \mathcal{I}_\mathcal{G} \ . \tag{29}$$

We conclude that the language generated by the grammar $\mathcal{G}$ is:

$$\begin{aligned}
& \mathcal{L}_\mathcal{G} \\
= \ & \{\delta \in \mathcal{T}^\star \mid A \overset{\mathcal{G}}{\Longrightarrow}_{A,\ell} \delta\} && \wr\text{by (10)}\wr \\
= \ & \{\delta \in \mathcal{T}^\star \mid \langle A, \delta \rangle \in \boldsymbol{\alpha}^\ell(\mathcal{I}_\mathcal{G})\} && \wr\text{by (26)}\wr \\
= \ & \{\delta \in \mathcal{T}^\star \mid \exists \lambda \in \mathcal{T}^\star : [\lambda, X \to \alpha\bullet, \delta] \in \mathcal{I}_\mathcal{G}\} && \wr\text{by (29)}\wr \ . \quad \square
\end{aligned}$$

9 Earley Parsing Algorithm is a Complete Abstraction of the Grammar Item Semantics

The Earley's parsing algorithm (17) derives the only grammar items which are valid for the given input word $\omega = \omega_1 \ldots \omega_n$, $n \geq 0$ to be analyzed.

9.1 The Abstraction

This is a forgetful abstraction disregarding all information provided by the grammar item semantics, but for the input word:

$$\begin{aligned}
\boldsymbol{\alpha}^{\mathrm{E}}_\omega(I) \ \ \triangleq \ \ & \{\langle X \to \alpha\bullet\beta, i, j\rangle \mid 0 \leq i \leq j \leq n \wedge \\
& \qquad [\omega_1 \ldots \omega_i, X \to \alpha\bullet\beta, \omega_{i+1} \ldots \omega_j] \in I\} \ .
\end{aligned} \tag{30}$$

$\boldsymbol{\alpha}^{\mathrm{E}}_\omega$ is a complete $\cup$-morphism so it is the lower adjoint of a Galois connection:

$$\langle \wp(\mathcal{T}^\star \times \mathcal{N} \times \mathcal{V}^\star \times \mathcal{V}^\star \times \mathcal{T}^\star), \subseteq\rangle \underset{\boldsymbol{\alpha}^{\mathrm{E}}_\omega}{\overset{\boldsymbol{\gamma}^{\mathrm{E}}_\omega}{\leftrightarrows}} \langle \wp(\mathcal{N} \times \mathcal{V}^\star \times \mathcal{V}^\star \times \mathbb{N} \times \mathbb{N}), \subseteq\rangle \ . \tag{31}$$

9.2 The Abstract Interpretation of the Semantics

By abstraction of the fixpoint definition (23) of the grammar item semantics with $\boldsymbol{\alpha}^{\mathrm{E}}_\omega$, we get the fixpoint characterization (17) of the Earley's valid item semantics:

Theorem 6

$$\mathcal{I}^{\mathrm{E}}_{\mathcal{G},\omega} \ = \ \boldsymbol{\alpha}^{\mathrm{E}}_\omega(\mathcal{I}_\mathcal{G}) \ . \tag{32}$$

Proof We must prove that $\mathcal{I}^{\mathrm{E}}_{\mathcal{G},\omega} = \mathrm{lfp}^{\subseteq} \mathcal{F}^{\mathrm{E}}_{\mathcal{G},\omega} = \boldsymbol{\alpha}^{\mathrm{E}}_{\omega}(\mathrm{lfp}^{\subseteq} \mathcal{F}_{\mathcal{G}}) = \boldsymbol{\alpha}^{\mathrm{E}}_{\omega}(\mathcal{I}_{\mathcal{G}})$ which, by lemma 2, immediately follows from $\boldsymbol{\alpha}^{\mathrm{E}}_{\omega} \circ \mathcal{F}_{\mathcal{G}} = \mathcal{F}^{\mathrm{E}}_{\mathcal{G},\omega} \circ \boldsymbol{\alpha}^{\mathrm{E}}_{\omega}$. Because $\boldsymbol{\alpha}^{\mathrm{E}}_{\omega}$ is a complete $\cup$-morphism, it is sufficient to do prove that term by term. We have:

- $\boldsymbol{\alpha}^{\mathrm{E}}_{\omega}(\{[\epsilon, A \to \bullet\beta, \epsilon] \mid A \xrightarrow{\mathcal{G}} \beta\})$

$=$ $\wr$Definition (30) of $\boldsymbol{\alpha}^{\mathrm{E}}_{\omega}\int$

$\{\langle X \to \alpha\bullet\beta, i, j\rangle \mid 0 \le i \le j \le n \wedge [\omega_1 \dots \omega_i, X \to \alpha\bullet\beta, \omega_{i+1} \dots \omega_j] = [\epsilon, A \to \bullet\beta, \epsilon] \wedge A \xrightarrow{\mathcal{G}} \beta\}$

$=$ $\wr\omega_1 \dots \omega_i = \epsilon$ so $i = 0$, $X = A$, $\alpha\bullet\beta = \bullet\beta$ so $\alpha = \epsilon$, $\omega_{i+1} \dots \omega_j = \omega_1 \dots \omega_j = \epsilon$ so $j = 0\int$

$\{\langle A \to \bullet\gamma, 0, 0\rangle \mid A \xrightarrow{\mathcal{G}} \gamma\}$.

- $\boldsymbol{\alpha}^{\mathrm{E}}_{\omega}(\{[\lambda\alpha, Y \to \bullet\delta, \epsilon] \mid [\lambda, X \to \alpha\bullet Y\beta, \gamma] \in I \wedge Y \xrightarrow{\mathcal{G}} \delta\})$

$=$ $\wr$Definition (30) of $\boldsymbol{\alpha}^{\mathrm{E}}_{\omega}\int$

$\{\langle X \to \alpha\bullet\beta, i, j\rangle \mid 0 \le i \le j \le n \wedge [\omega_1 \dots \omega_i, X \to \alpha\bullet\beta, \omega_{i+1} \dots \omega_j] \in \{[\lambda'\gamma, Y \to \bullet\delta, \epsilon] \mid [\lambda', X \to \alpha'\bullet Y\beta', \gamma] \in I \wedge Y \xrightarrow{\mathcal{G}} \delta\}\}$

$=$ $\wr\lambda'\gamma = \omega_1 \dots \omega_i$ so $\exists k \in [0, i] : \lambda' = \omega_1 \dots \omega_k \wedge \gamma = \omega_{k+1} \dots \omega_i$, $X = Y$, $\alpha\bullet\beta = \bullet\delta$ so $\alpha = \epsilon$ and $\beta = \delta$, $\omega_{i+1} \dots \omega_j = \epsilon$ so $i = j\int$

$\{\langle Y \to \bullet\delta, j, j\rangle \mid 0 \le k \le j \le n \wedge [\omega_1 \dots \omega_k, X \to \alpha'\bullet Y\beta', \omega_{k+1} \dots \omega_j] \in I \wedge Y \xrightarrow{\mathcal{G}} \delta\}$

$=$ $\wr\alpha\bullet\beta = \alpha'\bullet Y\beta'$ if and only if $\alpha = \alpha'$ and $\beta = Y\beta'$, renaming k as $i\int$

$\{\langle Y \to \bullet\delta, j, j\rangle \mid \langle X \to \alpha\bullet Y\beta, i, j\rangle \in \{\langle X \to \alpha'\bullet\beta', i, j\rangle \mid 0 \le i \le j \le n \wedge [\omega_1 \dots \omega_i, X \to \alpha'\bullet\beta', \omega_{i+1} \dots \omega_j] \in I \wedge Y \xrightarrow{\mathcal{G}} \delta\}\}$

$=$ $\wr$Definition (30) of $\boldsymbol{\alpha}^{\mathrm{E}}_{\omega}\int$

$\{\langle Y \to \bullet\delta, j, j\rangle \mid \langle X \to \alpha\bullet Y\beta, i, j\rangle \in \boldsymbol{\alpha}^{\mathrm{E}}_{\omega}(I) \wedge Y \xrightarrow{\mathcal{G}} \delta\}$.

- $\boldsymbol{\alpha}^{\mathrm{E}}_{\omega}(\{[\lambda, X \to \alpha Y\bullet\beta, \gamma\xi] \mid [\lambda, X \to \alpha\bullet Y\beta, \gamma] \in I \wedge [\lambda\gamma, Y \to \delta\bullet, \xi] \in I\})$

$=$ $\wr$Definition (30) of $\boldsymbol{\alpha}^{\mathrm{E}}_{\omega}\int$

$\{\langle X \to \alpha\bullet\beta, i, j\rangle \mid 0 \le i \le j \le n \wedge [\omega_1 \dots \omega_i, X \to \alpha\bullet\beta, \omega_{i+1} \dots \omega_j] \in \{[\lambda, X \to \alpha' Y\bullet\beta', \gamma\xi] \mid [\lambda, X \to \alpha'\bullet Y\beta', \gamma] \in I \wedge [\lambda\gamma, Y \to \delta\bullet, \xi] \in I\}\}$

$=$ $\wr\lambda = \omega_1 \dots \omega_i$, $\alpha = \alpha' Y$, $\beta = \beta'$, $\gamma\xi = \omega_{i+1} \dots \omega_j$ so $\exists k \in [i, j] : \gamma = \omega_{i+1} \dots \omega_k \wedge \xi = \omega_{k+1} \dots \omega_j\int$

$\{\langle X \to \alpha' Y\bullet\beta', i, j\rangle \mid 0 \le i \le k \le j \le n \wedge [\omega_1 \dots \omega_i, X \to \alpha'\bullet Y\beta', \omega_{i+1} \dots \omega_k] \in I \wedge [\omega_1 \dots \omega_i\omega_{i+1} \dots \omega_k, Y \to \delta\bullet, \omega_{k+1} \dots \omega_j] \in I\}$

$=$ $\wr$def. (30) of $\boldsymbol{\alpha}^{\mathrm{E}}_{\omega}\int$

$\{\langle X \to \alpha Y\bullet\beta, k, j\rangle \mid \langle X \to \alpha\bullet Y\beta, k, i\rangle \in \boldsymbol{\alpha}^{\mathrm{E}}_{\omega}(I) \wedge \langle Y \to \gamma\bullet, i, j\rangle \in \boldsymbol{\alpha}^{\mathrm{E}}_{\omega}(I)\}$.

- $\boldsymbol{\alpha}^{\mathrm{E}}_{\omega}(\{[\lambda, X \to \alpha a{\boldsymbol{\cdot}}\beta, \gamma a] \mid [\lambda, X \to \alpha a{\boldsymbol{\cdot}}\beta, \gamma a] \in I\})$

$=$ $\wr$Definition (30) of $\boldsymbol{\alpha}^{\mathrm{E}}_{\omega}\int$

$\{\langle X \to \alpha{\boldsymbol{\cdot}}\beta, i, j\rangle \mid 0 \le i \le j \le n \wedge [\omega_1 \ldots \omega_i, X \to \alpha{\boldsymbol{\cdot}}\beta, \omega_{i+1} \ldots \omega_j] \in \{[\lambda, X \to \alpha' a{\boldsymbol{\cdot}}\beta', \gamma a] \mid [\lambda, X \to \alpha'{\boldsymbol{\cdot}}a\beta', \gamma] \in I\}\}$

$=$ $\wr\lambda = \omega_1 \ldots \omega_i$, $\alpha = \alpha' a$, $\beta = \beta'$ and $\gamma a = \omega_{i+1} \ldots \omega_j$ so $\gamma = \omega_{i+1} \ldots \omega_{j-1}$ and $a = \omega_j\int$

$\{\langle X \to \alpha' \omega_j{\boldsymbol{\cdot}}\beta, i, j\rangle \mid 0 \le i \le j \le n \wedge [\omega_1 \ldots \omega_i, X \to \alpha'{\boldsymbol{\cdot}}\omega_j\beta', \omega_{i+1} \ldots \omega_{j-1}] \in I\}$

$=$ $\wr$def. (30) of $\boldsymbol{\alpha}^{\mathrm{E}}_{\omega}\int$

$\{\langle X \to \alpha\omega_j{\boldsymbol{\cdot}}\beta, i, j\rangle \mid \langle X \to \alpha{\boldsymbol{\cdot}}\omega_j\beta, i, j-1\rangle \in \boldsymbol{\alpha}^{\mathrm{E}}_{\omega}(I)\}$. □

10 Correctness of Earley's Parsing Algorithm

Earley's parsing algorithm approximates the grammar items for the given terminal input word. This word is in the language generated by the grammar only if is recognized by a grammar item for the axiom, so:

Corollary 7 The Earley's parsing algorithm is correct in that (18) holds.

Proof

$\langle A \to \gamma{\boldsymbol{\cdot}}, 0, n\rangle \in \mathcal{I}^{\mathrm{E}}_{\mathcal{G},\omega}$

$\Longleftrightarrow$ $\wr$by (32)$\int$

$\langle A \to \gamma{\boldsymbol{\cdot}}, 0, n\rangle \in \boldsymbol{\alpha}^{\mathrm{E}}_{\omega}(\mathcal{I}_{\mathcal{G}})$

$\Longleftrightarrow$ $\wr$by definition (30) of $\boldsymbol{\alpha}^{\mathrm{E}}_{\omega}\int$

$\langle A \to \gamma{\boldsymbol{\cdot}}, 0, n\rangle \in \{\langle X \to \alpha{\boldsymbol{\cdot}}\beta, i, j\rangle \mid 0 \le i \le j \le n \wedge [\omega_1 \ldots \omega_i, X \to \alpha{\boldsymbol{\cdot}}\beta, \omega_{i+1} \ldots \omega_j] \in \mathcal{I}_{\mathcal{G}}\}$

$\Longleftrightarrow$ $\wr X = A$, $\gamma{\boldsymbol{\cdot}} = \alpha{\boldsymbol{\cdot}}\beta$ so $\gamma = \alpha$ and $\beta = \epsilon$, $i = 0$, $j = n$ so $\omega_1 \ldots \omega_i = \epsilon$ and $\omega_{i+1} \ldots \omega_j = \omega\int$

$[\epsilon, A \to \alpha{\boldsymbol{\cdot}}, \omega] \in \mathcal{I}_{\mathcal{G}}$

$\Longleftrightarrow$ $\wr$choosing $\lambda = \epsilon$ and lemma 3$\int$

$\exists \lambda \in \mathcal{T}^\star : [\lambda, A \to \alpha{\boldsymbol{\cdot}}, \omega] \in \mathcal{I}_{\mathcal{G}}$

$\Longleftrightarrow$ $\wr$by (28)$\int$

$\omega \in \mathcal{L}_{\mathcal{G}}$. □

11 Conclusion

We have shown that Earley's parsing algorithm [9] is an abstract interpretation of a refinement of the derivation semantics of grammars.

Other parsing algorithms may certainly be formally derived in a similar way using a more refined item semantics with nonterminal left and right contexts. A compile-time/static analysis of the grammar (item semantics) is used for top-down left-to-right generation of sets of grammar items abstracted e.g. as states.

The same way a preliminary analysis of the grammar approximates terminal derivations from the right contexts by a lookahead. This preliminary static grammar analysis is used to ensure that the the bottom-up recognition with left context is deterministic. This point of view remains to be applied, e.g. to LR-parsing [10].

References

[1] P. Aczel. An introduction to inductive definitions. In J. Barwise, editor, *Handbook of Mathematical Logic*, volume 90 of *Studies in Logic and the Foundations of Mathematics*, pages 739–782. Elsevier Science Publishers B.V., 1977.

[2] P. Cousot. Types as abstract interpretations, invited paper. In *Conference Record of the Twentyfourth Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 316–331. ACM Press, 1997.

[3] P. Cousot and R. Cousot. Abstract interpretation of algebraic polynomial systems. In M. Johnson, editor, *Proceedings of the Sixth International Conference on Algebraic Methodology and Software Technology, AMAST '97*, Sydney, Australia, Lecture Notes in Computer Science 1349, pages 138–154. Springer-Verlag, 1997.

[4] P. Cousot and R. Cousot. Abstract interpretation: a unified lattice model for static analysis of programs by construction or approximation of fixpoints. In *Conference Record of the Fourth Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 238–252. ACM Press, 1977.

[5] P. Cousot and R. Cousot. Constructive versions of Tarski's fixed point theorems. *Pacific Journal of Mathematics*, 82(1):43–57, 1979.

[6] P. Cousot and R. Cousot. Systematic design of program analysis frameworks. In *Conference Record of the Sixth Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 269–282. ACM Press, 1979.

[7] P. Cousot and R. Cousot. Compositional and inductive semantic definitions in fixpoint, equational, constraint, closure-condition, rule-based and game-theoretic form, invited paper. In P. Wolper, editor, *Proceedings of the Seventh International Conference on Computer Aided Verification, CAV '95*, Liège, Belgium, Lecture Notes in Computer Science 939, pages 293–308. Springer-Verlag, 1995.

[8] P. Cousot and R. Cousot. Temporal abstract interpretation. In *Conference Record of the Twentyseventh Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 12–25. ACM Press, 2000.

[9] J. Earley. An efficient context-free parsing algorithm. *Communications of the Association for Computing Machinary*, 13(2):94–102, 1970.

[10] D.E. Knuth. On the translation of languages from left to right. *Information and Control*, 8:607–639, 1965.

[11] K. Sikkel and A. Nijholt. Parsing of context-free languages. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages, Vol. 2, Linear Modeling: Background and Application*, pages 61–100. Springer-Verlag, 1997.

[12] T.A. Sudkamp. *Languages and Machines : An Introduction to the Theory of Computer Science*. Addison-Wesley Pub. Co., Reading, Massachusetts, United States, second edition, November 1996.

[13] A. Tarski. A lattice theoretical fixpoint theorem and its applications. *Pacific Journal of Mathematics*, 5:285–310, 1955.