Conservative Ambiguity Detection in Context-Free Grammars

Sylvain Schmitz

Laboratoire I3S, Université de Nice - Sophia Antipolis & CNRS, France

July 12, 2007

Ambiguity in CFGs

- natural
- an issue
  - in language acquisition [Cheung and Uzgalis, 1995]
  - in RNA analysis [Reeder et al., 2005]
  - in computer languages
  - need a conservative test: no false negatives
- undecidable problem [Cantor, 1962, Chomsky and Schützenberger, 1963]

Ambiguity in CFGs

- natural. . . in natural languages
- an issue
  - in language acquisition [Cheung and Uzgalis, 1995]
  - in RNA analysis [Reeder et al., 2005]
  - in computer languages
  - need a conservative test: no false negatives
- undecidable problem [Cantor, 1962, Chomsky and Schützenberger, 1963]

Ambiguity in CFGs

- natural. . . in natural languages
- an issue
  - in language acquisition [Cheung and Uzgalis, 1995]
  - in RNA analysis [Reeder et al., 2005]
  - in computer languages
  - need a conservative test: no false negatives
- undecidable problem [Cantor, 1962, Chomsky and Schützenberger, 1963]

Ambiguity in CFGs

- natural... in natural languages
- an issue
  - in language acquisition [Cheung and Uzgalis, 1995]
  - in RNA analysis [Reeder et al., 2005]
  - in computer languages
  - need a conservative test: no false negatives
- undecidable problem [Cantor, 1962, Chomsky and Schützenberger, 1963]

Ambiguity in CFGs

- natural... in natural languages
- an issue
  - in language acquisition [Cheung and Uzgalis, 1995]
  - in RNA analysis [Reeder et al., 2005]
  - in computer languages
  - need a conservative test: no false negatives
- undecidable problem [Cantor, 1962, Chomsky and Schützenberger, 1963]

An Ambiguity

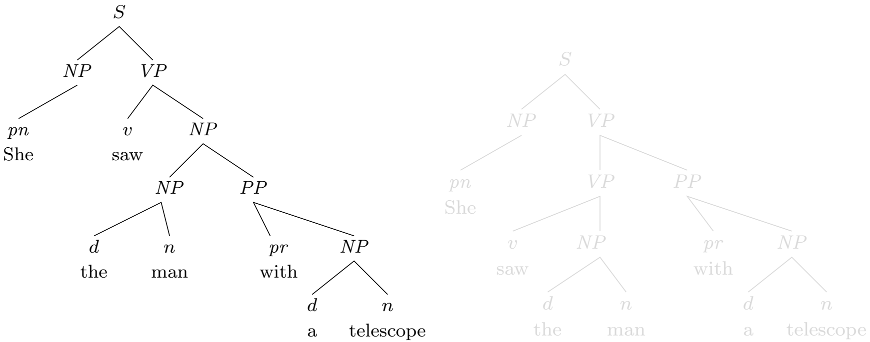
Example

"She saw the man with a telescope ."

An Ambiguity

Example

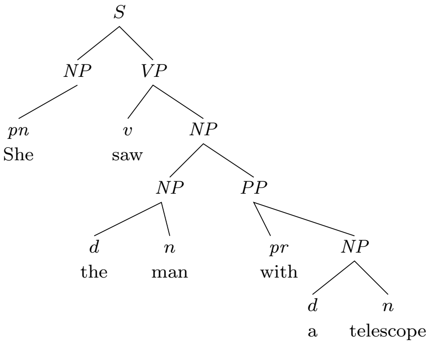
"She saw the man with a telescope ."



An Ambiguity

Example

"She saw the man with a telescope ."



Bracketed Grammars

$$\mathcal{G} = \langle N, T, P, S \rangle,\ V = N \cup T$$

$$\begin{aligned}
S &\xrightarrow{2} NP\ VP \\
NP &\xrightarrow{3} d\ n \\
NP &\xrightarrow{4} pn \\
NP &\xrightarrow{5} NP\ PP \\
VP &\xrightarrow{6} v\ NP \\
VP &\xrightarrow{7} VP\ PP \\
PP &\xrightarrow{8} pr\ NP
\end{aligned}$$

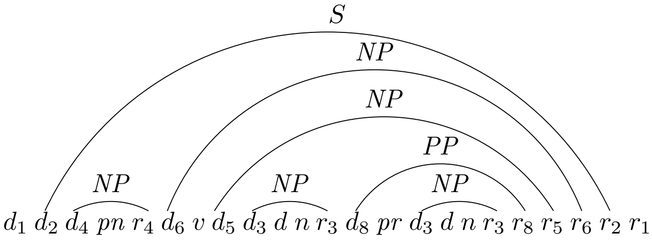
Bracketed Grammars

$$\mathcal{G}_b = \langle N, T_b, P_b, S \rangle,\ V_b = N \cup T_b$$

$$\begin{aligned}
S &\xrightarrow{2} d_2\ NP\ VP\ r_2 \\
NP &\xrightarrow{3} d_3\ d\ n\ r_3 \\
NP &\xrightarrow{4} d_4\ pn\ r_4 \\
NP &\xrightarrow{5} d_5\ NP\ PP\ r_5 \\
VP &\xrightarrow{6} d_6\ v\ NP\ r_6 \\
VP &\xrightarrow{7} d_7\ VP\ PP\ r_7 \\
PP &\xrightarrow{8} d_8\ pr\ NP\ r_8
\end{aligned}$$

The Approach

- a bracketed sentence = a derivation tree



- ambiguity = more than one tree with the same yield
- construct a FSA $\mathcal{A}$ such that $\mathcal{L}(\mathcal{G}_b) \subseteq \mathcal{L}(\mathcal{A})$, and look for bracketed sentences with the same yield

The Approach

- a bracketed sentence = a derivation tree
- ambiguity = more than one tree with the same yield
  Let h be the bracket-erasing homomorphism $V_b^* \to V^*$ with $h(X) = X$ for X in V and $h(d_i) = h(r_i) = \varepsilon$; the yield of w_b is $w = h(w_b)$.
- construct a FSA $\mathcal{A}$ such that $\mathcal{L}(\mathcal{G}_b) \subseteq \mathcal{L}(\mathcal{A})$, and look for bracketed sentences with the same yield

The Approach

- a bracketed sentence = a derivation tree
- ambiguity = more than one tree with the same yield
- construct a FSA $\mathcal{A}$ such that $\mathcal{L}(\mathcal{G}_b) \subseteq \mathcal{L}(\mathcal{A})$, and look for bracketed sentences with the same yield

$$w_b \neq w'_b \in \mathcal{L}(\mathcal{A}), h(w_b) = h(w'_b)$$

The Approach

- a bracketed sentence = a derivation tree
- ambiguity = more than one tree with the same yield
- construct a FSA $\mathcal{A}$ such that $\mathcal{L}(\mathcal{G}_b) \subseteq \mathcal{L}(\mathcal{A})$, and look for bracketed sentences with the same yield

$$w_b \neq w'_b \in \mathcal{L}(\mathcal{A}), h(w_b) = h(w'_b)$$

$$d_1\ d_2\ d_4\ pn\ r_4\ d_6\ v\ d_5\ d_3\ d\ \ n\ r_3\ d_8\ pr\ d_3\ d\ \ n\ r_3\ r_8\ r_5\ r_6\ r_2\ r_1$$
$$d_1\ d_2\ d_4\ pn\ r_4\ d_7\ d_6\ v\ d_3\ d\ \ n\ r_3\ r_6\ d_8\ pr\ d_3\ d\ \ n\ r_3\ r_8\ r_7\ r_2\ r_1$$

The Approach

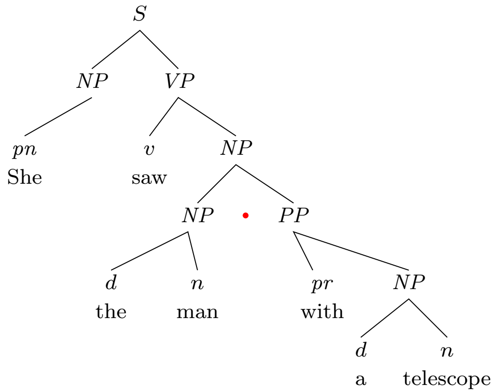
- a bracketed sentence = a derivation tree
- ambiguity = more than one tree with the same yield
- construct a FSA $\mathcal{A}$ such that $\mathcal{L}(\mathcal{G}_b) \subseteq \mathcal{L}(\mathcal{A})$, and look for bracketed sentences with the same yield

$$w_b \neq w'_b \in \mathcal{L}(\mathcal{A}), h(w_b) = h(w'_b)$$

$d_1\ d_2\ d_4\ pn\ r_4\ d_6\ v\ d_5\ d_3\ d\ n\ r_3\ d_8\ pr\ d_3\ d\ n\ r_3\ r_8\ r_5\ r_6\ r_2\ r_1$

$d_1\ d_2\ d_4\ pn\ r_4\ d_7\ d_6\ v\ d_3\ d\ n\ r_3\ r_6\ d_8\ pr\ d_3\ d\ n\ r_3\ r_8\ r_7\ r_2\ r_1$

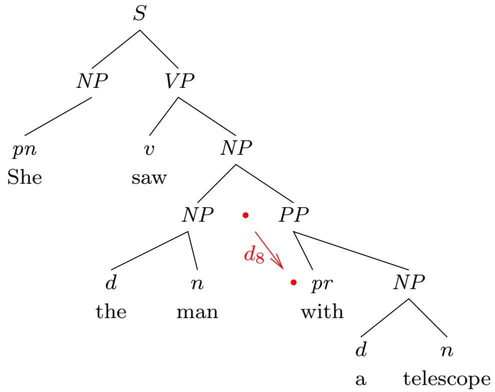
Positions



$d_1\ d_2\ d_4\ pn\ r_4\ d_6\ v\ d_5\ d_3\ d\ n\ r_3 \bullet d_8\ pr\ d_3\ d\ n\ r_3\ r_8\ r_5\ r_6\ r_2\ r_1$

Position Graph Γ

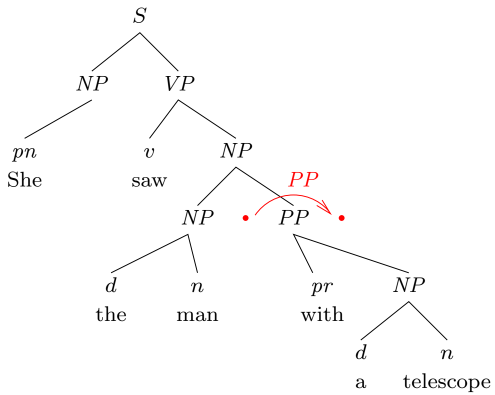
Left-to-right Walks in Trees



$d_1\ d_2\ d_4\ pn\ r_4\ d_6\ v\ d_5\ d_3\ d\ n\ r_3\ d_8\ \bullet\ pr\ d_3\ d\ n\ r_3\ r_8\ r_5\ r_6\ r_2\ r_1$

Position Graph Γ

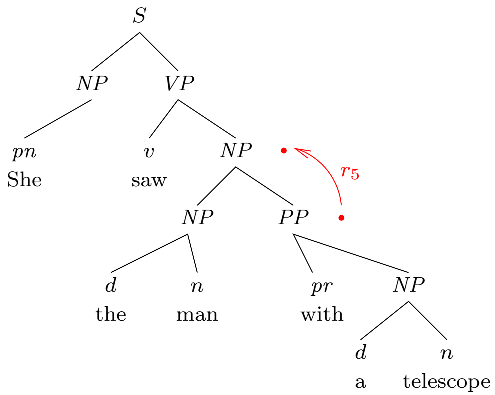
Left-to-right Walks in Trees



$d_1\ d_2\ d_4\ pn\ r_4\ d_6\ v\ d_5\ d_3\ d\ \ n\ r_3\ d_8\ pr\ d_3\ d\ \ n\ r_3\ r_8 \bullet\ r_5\ r_6\ r_2\ r_1$

Position Graph Γ

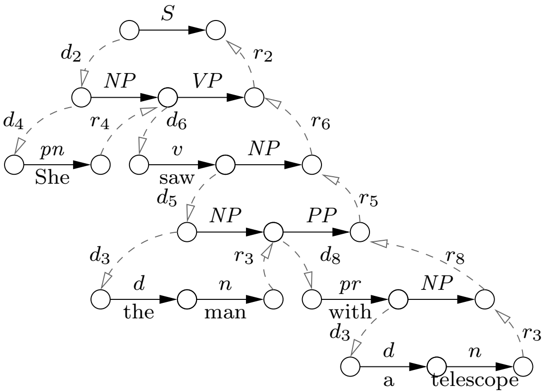
Left-to-right Walks in Trees



$d_1\ d_2\ d_4\ pn\ r_4\ d_6\ v\ d_5\ d_3\ d\ \ n\ r_3 d_8\ pr\ d_3\ d\ \ n\ r_3\ r_8\ r_5 \bullet\ r_6\ r_2\ r_1$

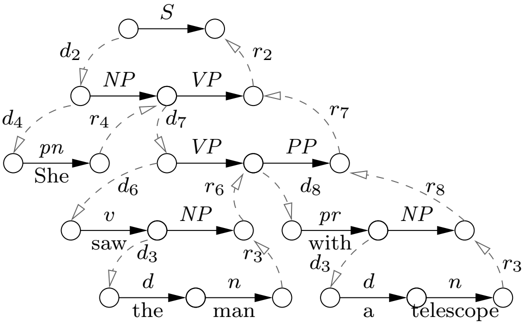
Position Graph Γ

Left-to-right Walks in Trees



Position Graph Γ

Left-to-right Walks in Trees



Position Automaton $\Gamma/\equiv$

Definition

$\Gamma/\equiv$ is the quotient of Γ by an equivalence relation $\equiv$ between positions.

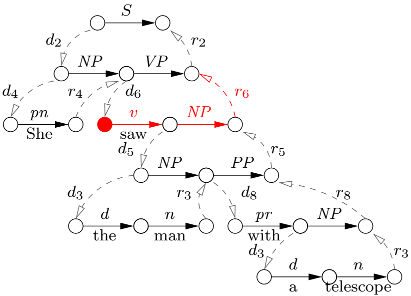
Lemma

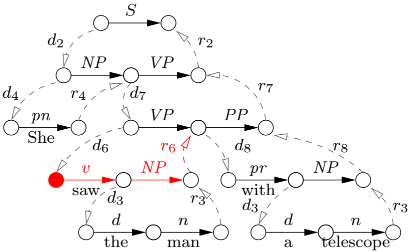
If $\nu \xrightarrow{\delta_b} \nu'$ *in* Γ, *then* $[\nu]_\equiv \delta_b \models^* [\nu']_\equiv$ *in* $\Gamma/\equiv$.

Theorem

$$\mathcal{L}(\mathcal{G}_b) \subseteq \mathcal{L}(\Gamma/\equiv) \cap T_b^*$$

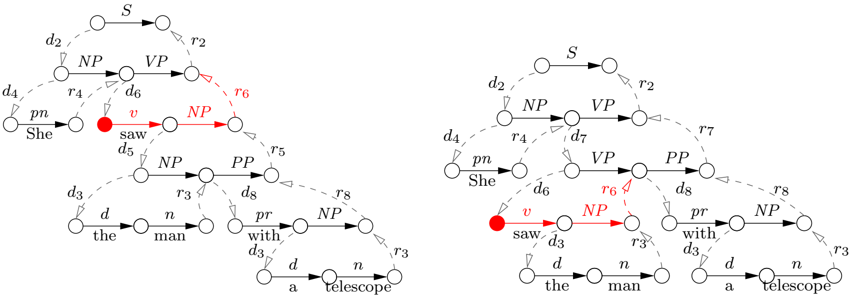
Example: item_0 Equivalence





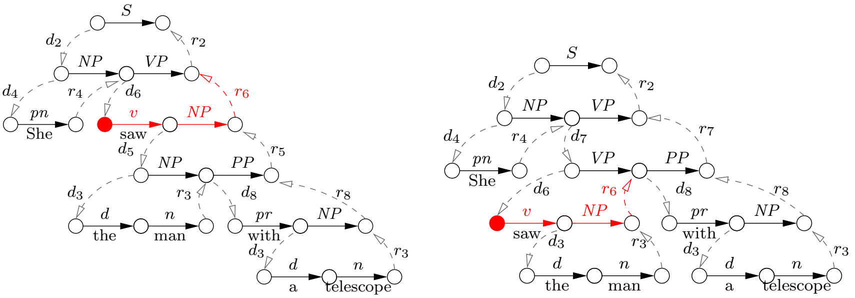
- This equivalence class: $[VP \overset{6}{\rightarrow} .v\ NP]$
- item_0 equivalence classes: LR(0) items
- Γ/item_0: nondeterministic LR(0) automaton

Example: item_0 Equivalence



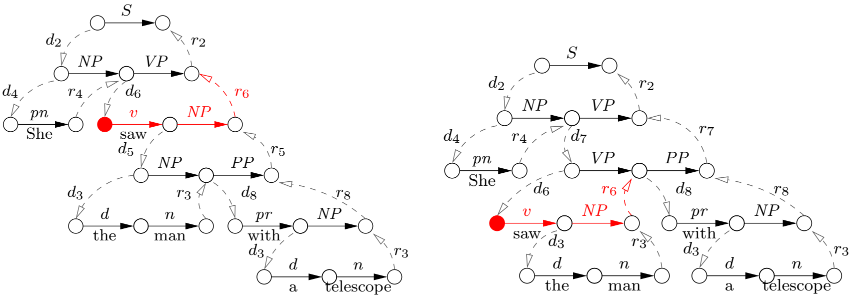
- This equivalence class: $[VP \xrightarrow{6} . v\ NP]$
- item_0 equivalence classes: LR(0) items
- Γ/item_0: nondeterministic LR(0) automaton

Example: item_0 Equivalence



- This equivalence class: $[VP \xrightarrow{6} \bullet v\ NP]$
- item_0 equivalence classes: LR(0) items
- Γ/item_0: nondeterministic LR(0) automaton

Example: item_0 Equivalence



- This equivalence class: $[VP \xrightarrow{6} . v\ NP]$
- item_0 equivalence classes: LR(0) items
- Γ/item_0: nondeterministic LR(0) automaton

Back to Ambiguity Detection

- Class of grammars: $\mathcal{G}$ is regular unambiguous for $\equiv$ of finite index, noted $\mathrm{RU}(\equiv)$, if there does not exist $w_b \neq w'_b$ in $\mathcal{L}(\Gamma/\equiv) \cap T_b^*$ with $h(w_b) = h(w'_b)$
- $\forall \equiv, \mathrm{RU}(\equiv) \subseteq \mathrm{UCFG}$
- $\mathrm{RU}(\mathrm{item}_0) \not\subseteq \mathrm{LR}(0)$
- regular approximations are too weak

Back to Ambiguity Detection

- Class of grammars: $\mathcal{G}$ is regular unambiguous for $\equiv$ of finite index, noted RU($\equiv$), if there does not exist $w_b \neq w'_b$ in $\mathcal{L}(\Gamma/\equiv) \cap T_b^*$ with $h(w_b) = h(w'_b)$
- $\forall \equiv$, RU($\equiv$) $\subseteq$ UCFG
- RU(item_0) $\not\subseteq$ LR(0)
- regular approximations are too weak

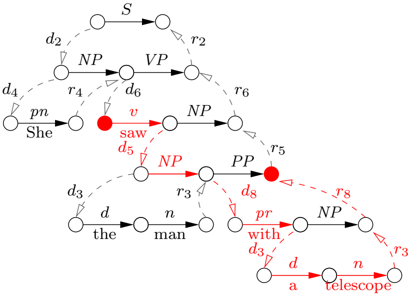
Back to Ambiguity Detection

- Class of grammars: $\mathcal{G}$ is regular unambiguous for $\equiv$ of finite index, noted RU($\equiv$), if there does not exist $w_b \neq w'_b$ in $\mathcal{L}(\Gamma/\equiv) \cap T_b^*$ with $h(w_b) = h(w'_b)$
- $\forall \equiv$, RU($\equiv$) $\subseteq$ UCFG
- RU(item_0) $\not\subseteq$ LR(0)
- regular approximations are too weak

Back to Ambiguity Detection

- Class of grammars: $\mathcal{G}$ is regular unambiguous for $\equiv$ of finite index, noted $\mathrm{RU}(\equiv)$, if there does not exist $w_b \neq w'_b$ in $\mathcal{L}(\Gamma/\equiv) \cap T_b^*$ with $h(w_b) = h(w'_b)$
- $\forall \equiv$, $\mathrm{RU}(\equiv) \subseteq \mathrm{UCFG}$
- $\mathrm{RU}(\mathrm{item}_0) \not\subseteq \mathrm{LR}(0)$
- regular approximations are too weak

Using Nonterminal Transitions

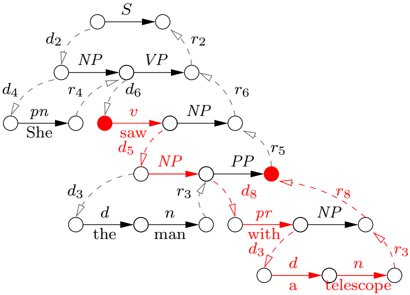


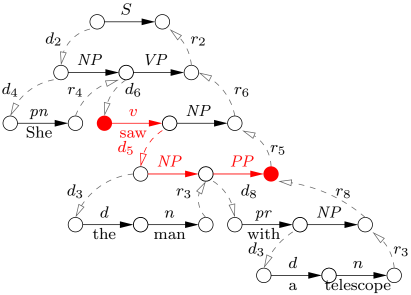
$$v\ d_5\ NP\ PP \Rightarrow_b^* v\ d_5\ NP\ d_8\ pr\ d_3\ d\ \ n\ r_3\ r_8$$

Lemma

If $\nu \xrightarrow{\delta_b} \nu'$ *in* Γ *and* $\gamma_b \Rightarrow_b^* \delta_b$ *in* $\mathcal{G}_b$*, then* $\nu \xrightarrow{\gamma_b} \nu'$ *in* Γ.

Using Nonterminal Transitions





$$v\ d_5\ NP\ PP \Rightarrow_b^* v\ d_5\ NP\ d_8\ pr\ d_3\ d\ \ n\ r_3\ r_8$$

Lemma

If $\nu \xrightarrow{\delta_b} \nu'$ *in* Γ *and* $\gamma_b \Rightarrow_b^* \delta_b$ *in* $\mathcal{G}_b$*, then* $\nu \xrightarrow{\gamma_b} \nu'$ *in* Γ*.*

Using Nonterminal Transitions

Lemma

If $\nu \xrightarrow{\delta_b} \nu'$ *in* Γ *and* $\gamma_b \Rightarrow_b^* \delta_b$ *in* $\mathcal{G}_b$*, then* $\nu \xrightarrow{\gamma_b} \nu'$ *in* Γ.

Corollary

If $[\nu]_\equiv \delta_b \models^* [\nu']_\equiv$ *in* $\Gamma/\!\equiv$ *and* $\gamma_b \Rightarrow_b^* \delta_b$ *in* $\mathcal{G}_b$*, then* $[\nu]_\equiv \gamma_b \models^* [\nu']_\equiv$ *in* $\Gamma/\!\equiv$.

Common Prefixes with Conflicts

$$d_1\ d_2\ d_4\ \ pn\ \ r_4\ d_6\ \ v\ \ d_5\ d_3\ \ d\ \ n\ \ r_3\ d_8\ \ pr\ \ d_3\ \ d\ \ n\ \ r_3\ r_8\ r_5\ r_6\ r_2\ r_1$$
$$d_1\ d_2\ d_4\ \ pn\ \ r_4\ d_7\ d_6\ \ v\ \ d_3\ \ d\ \ n\ \ r_3\ r_6\ d_8\ \ pr\ \ d_3\ \ d\ \ n\ \ r_3\ r_8\ r_7\ r_2\ r_1$$

Common Prefixes with Conflicts

$d_1\ d_2\ d_4\ \ pn\ \ r_4\ d_6\ \ v\ \ d_5\ d_3\ \ d\ \ n\ \ r_3\ d_8\ \ pr\ \ d_3\ \ d\ \ n\ \ r_3\ r_8\ r_5\ r_6\ r_2\ r_1$

$d_1\ d_2\ d_4\ \ pn\ \ r_4\ d_7\ d_6\ \ v\ \ d_3\ \ d\ \ n\ \ r_3\ r_6\ d_8\ \ pr\ \ d_3\ \ d\ \ n\ \ r_3\ r_8\ r_7\ r_2\ r_1$

Common Prefixes with Conflicts

NP

$$d_1 \; d_2 \; d_4 \; pn \; r_4 \; d_6 \; v \; d_5 \; d_3 \; d \; n \; r_3 \; d_8 \; pr \; d_3 \; d \; n \; r_3 \; r_8 \; r_5 \; r_6 \; r_2 \; r_1$$

$$d_1 \; d_2 \; d_4 \; pn \; r_4 \; d_7 \; d_6 \; v \; d_3 \; d \; n \; r_3 \; r_6 \; d_8 \; pr \; d_3 \; d \; n \; r_3 \; r_8 \; r_7 \; r_2 \; r_1$$

NP

Common Prefixes with Conflicts

NP NP

$$d_1\ d_2\ d_4\ pn\ r_4\ d_6\ v\ d_5\ d_3\ d\ n\ r_3\ d_8\ pr\ d_3\ d\ n\ r_3\ r_8\ r_5\ r_6\ r_2\ r_1$$

$$d_1\ d_2\ d_4\ pn\ r_4\ d_7\ d_6\ v\ d_3\ d\ n\ r_3\ r_6\ d_8\ pr\ d_3\ d\ n\ r_3\ r_8\ r_7\ r_2\ r_1$$

NP NP

Common Prefixes with Conflicts

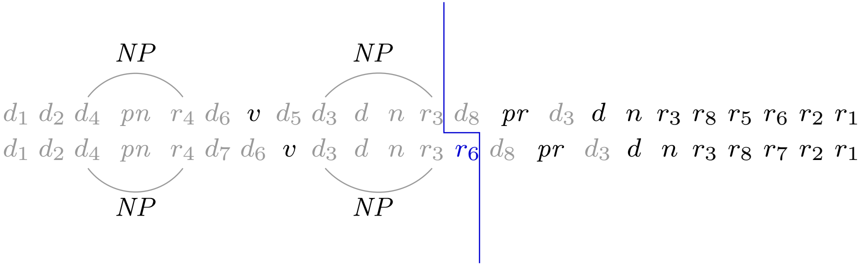
NP NP

$d_1\ d_2\ d_4\ pn\ r_4\ d_6\ v\ d_5\ d_3\ d\ n\ r_3\ d_8\ pr\ d_3\ d\ n\ r_3\ r_8\ r_5\ r_6\ r_2\ r_1$

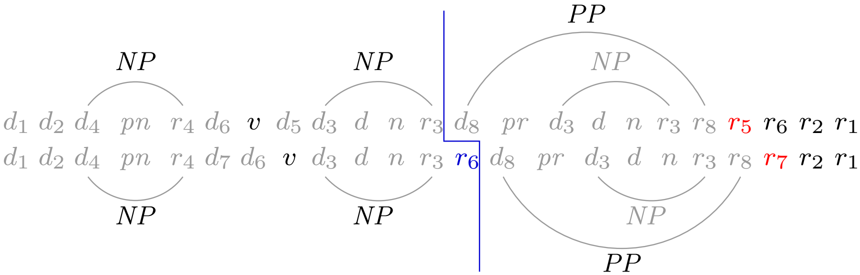
$d_1\ d_2\ d_4\ pn\ r_4\ d_7\ d_6\ v\ d_3\ d\ n\ r_3\ r_6\ d_8\ pr\ d_3\ d\ n\ r_3\ r_8\ r_7\ r_2\ r_1$

NP NP

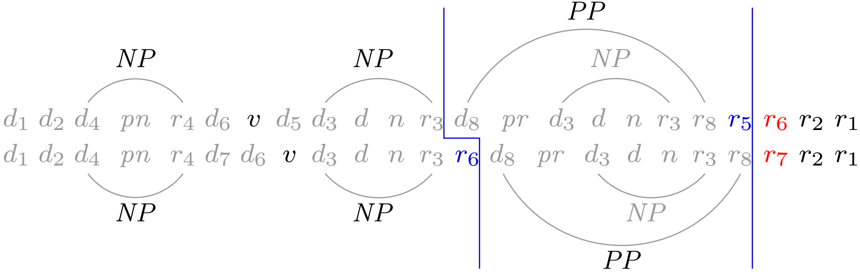
Common Prefixes with Conflicts



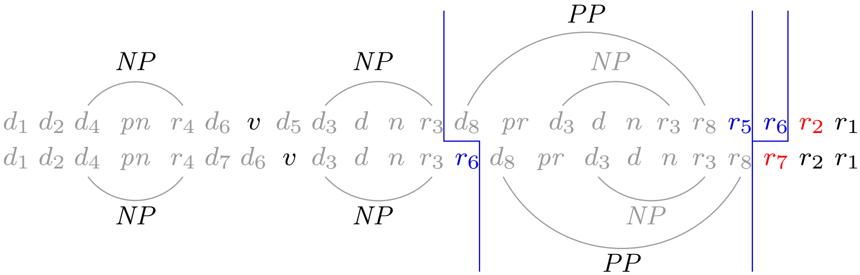
Common Prefixes with Conflicts



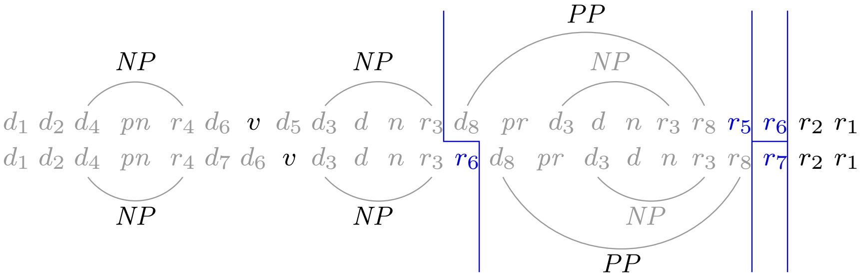
Common Prefixes with Conflicts



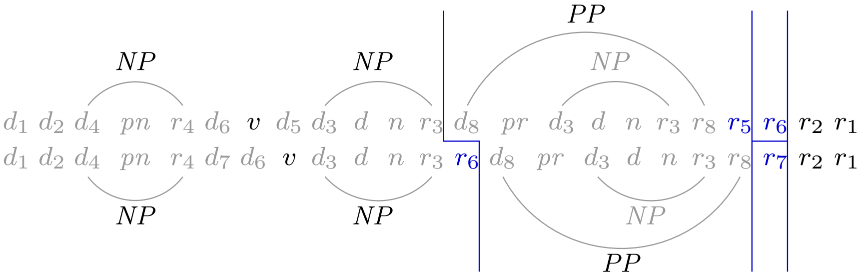
Common Prefixes with Conflicts



Common Prefixes with Conflicts



Common Prefixes with Conflicts



Two bracketed reduced sentential forms $\delta_b \neq \delta'_b$ with $h(\delta_b) = h(\delta'_b)$:

$$d_1\ d_2\ NP\ d_6\ v\ d_5\ NP\ PP\ r_5\ r_6\ r_2\ r_1$$

$$d_1\ d_2\ NP\ d_7\ d_6\ v\ NP\ r_6\ PP\ r_7\ r_2\ r_1$$

Mutual Accessibility Relations

- between pairs of states of $\Gamma/\equiv$, (q_1, q_2)
- synchronized left-to-right walks from an initial pair (q_s, q_s)

$$d_1\ d_2\ NP\ d_6\ v\ d_5\ NP\ PP\ r_5\ r_6\ r_2\ r_1$$
$$d_1\ d_2\ NP\ d_7\ d_6\ v\ NP\ r_6\ PP\ r_7\ r_2\ r_1$$

Mutual Accessibility Relations

- between pairs of states of $\Gamma/\equiv$, (q_1, q_2)
- synchronized left-to-right walks from an initial pair (q_s, q_s)

$$d_1\ d_2\ NP\ d_6\ v\ d_5\ NP\ PP\ r_5\ r_6\ r_2\ r_1$$
$$d_1\ d_2\ NP\ d_7\ d_6\ v\ NP\ r_6\ PP\ r_7\ r_2\ r_1$$

Mutual Accessibility Relations

- between pairs of states of $\Gamma/\equiv$, (q_1, q_2)
- synchronized left-to-right walks from an initial pair (q_s, q_s)

$$d_1\ d_2\ NP\ d_6\ v\ d_5\ NP\ PP\ r_5\ r_6\ r_2\ r_1$$
$$d_1\ d_2\ NP\ d_7\ d_6\ v\ NP\ r_6\ PP\ r_7\ r_2\ r_1$$

Epsilon

mae: read a d_i symbol from one state

Mutual Accessibility Relations

- between pairs of states of $\Gamma/\equiv$, (q_1, q_2)
- synchronized left-to-right walks from an initial pair (q_s, q_s)

$$d_1\ d_2\ NP\ d_6\ v\ d_5\ NP\ PP\ r_5\ r_6\ r_2\ r_1$$
$$d_1\ d_2\ NP\ d_7\ d_6\ v\ NP\ r_6\ PP\ r_7\ r_2\ r_1$$

Epsilon

mae: read a d_i symbol from one state

Mutual Accessibility Relations

- between pairs of states of $\Gamma/\!\equiv$, (q_1, q_2)
- synchronized left-to-right walks from an initial pair (q_s, q_s)

$$d_1\ d_2\ NP\ d_6\ v\ d_5\ NP\ PP\ r_5\ r_6\ r_2\ r_1$$
$$d_1\ d_2\ NP\ d_7\ d_6\ v\ NP\ r_6\ PP\ r_7\ r_2\ r_1$$

Epsilon

mae: read a d_i symbol from one state

Mutual Accessibility Relations

- between pairs of states of $\Gamma/\equiv$, (q_1, q_2)
- synchronized left-to-right walks from an initial pair (q_s, q_s)

$$d_1\ d_2\ NP\ d_6\ v\ d_5\ NP\ PP\ r_5\ r_6\ r_2\ r_1$$

$$d_1\ d_2\ NP\ d_7\ d_6\ v\ NP\ r_6\ PP\ r_7\ r_2\ r_1$$

Epsilon

mae: read a d_i symbol from one state

Mutual Accessibility Relations

- between pairs of states of $\Gamma/\equiv$, (q_1, q_2)
- synchronized left-to-right walks from an initial pair (q_s, q_s)

$$d_1\ d_2\ NP\ d_6\ v\ d_5\ NP\ PP\ r_5\ r_6\ r_2\ r_1$$

$$d_1\ d_2\ NP\ d_7\ d_6\ v\ NP\ r_6\ PP\ r_7\ r_2\ r_1$$

Shift

mas: read the same symbol of V from both states

Mutual Accessibility Relations

- between pairs of states of $\Gamma/\equiv$, (q_1, q_2)
- synchronized left-to-right walks from an initial pair (q_s, q_s)

$$d_1\ d_2\ NP\ d_6\ v\ d_5\ NP\ PP\ r_5\ r_6\ r_2\ r_1$$
$$d_1\ d_2\ NP\ d_7\ d_6\ v\ NP\ r_6\ PP\ r_7\ r_2\ r_1$$

Epsilon

mae$^+$: read a d_i symbol from one state

Mutual Accessibility Relations

- between pairs of states of $\Gamma/\equiv$, (q_1, q_2)
- synchronized left-to-right walks from an initial pair (q_s, q_s)

$$d_1\ d_2\ NP\ d_6\ v\ d_5\ NP\ PP\ r_5\ r_6\ r_2\ r_1$$
$$d_1\ d_2\ NP\ d_7\ d_6\ v\ NP\ r_6\ PP\ r_7\ r_2\ r_1$$

Shift

mas: read the same symbol of V from both states

Mutual Accessibility Relations

- between pairs of states of $\Gamma/\equiv$, (q_1, q_2)
- synchronized left-to-right walks from an initial pair (q_s, q_s)

$$d_1\ d_2\ NP\ d_6\ v\ d_5\ NP\ PP\ r_5\ r_6\ r_2\ r_1$$

$$d_1\ d_2\ NP\ d_7\ d_6\ v\ NP\ r_6\ PP\ r_7\ r_2\ r_1$$

Epsilon

mae: read a d_i symbol from one state

Mutual Accessibility Relations

- between pairs of states of $\Gamma/\equiv$, (q_1, q_2)
- synchronized left-to-right walks from an initial pair (q_s, q_s)

$$d_1\ d_2\ NP\ d_6\ v\ d_5\ NP\ PP\ r_5\ r_6\ r_2\ r_1$$

$$d_1\ d_2\ NP\ d_7\ d_6\ v\ NP\ r_6\ PP\ r_7\ r_2\ r_1$$

Shift

mas: read the same symbol of V from both states

Mutual Accessibility Relations

- between pairs of states of $\Gamma/\equiv$, (q_1, q_2)
- synchronized left-to-right walks from an initial pair (q_s, q_s)

$$d_1\ d_2\ NP\ d_6\ v\ d_5\ NP\ PP\ r_5\ r_6\ r_2\ r_1$$
$$d_1\ d_2\ NP\ d_7\ d_6\ v\ NP\ r_6\ PP\ r_7\ r_2\ r_1$$

Conflict

mac: read a r_i symbol from one state, if the other state can read a symbol of T or r_j with $j \neq i$ after a sequence of d_k symbols

Mutual Accessibility Relations

- between pairs of states of $\Gamma/\equiv$, (q_1, q_2)
- synchronized left-to-right walks from an initial pair (q_s, q_s)

$$d_1\ d_2\ NP\ d_6\ v\ d_5\ NP\ PP\ r_5\ r_6\ r_2\ r_1$$
$$d_1\ d_2\ NP\ d_7\ d_6\ v\ NP\ r_6\ PP\ r_7\ r_2\ r_1$$

Shift

mas: read the same symbol of V from both states

Mutual Accessibility Relations

- between pairs of states of $\Gamma/\!\equiv$, (q_1, q_2)
- synchronized left-to-right walks from an initial pair (q_s, q_s)

$$d_1\ d_2\ NP\ d_6\ v\ d_5\ NP\ PP\ r_5\ r_6\ r_2\ r_1$$
$$d_1\ d_2\ NP\ d_7\ d_6\ v\ NP\ r_6\ PP\ r_7\ r_2\ r_1$$

Conflict

mac$^+$: read a r_i symbol from one state, if the other state can read a symbol of T or r_j with $j \neq i$ after a sequence of d_k symbols

Mutual Accessibility Relations

- between pairs of states of $\Gamma/\!\equiv$, (q_1, q_2)
- synchronized left-to-right walks from an initial pair (q_s, q_s)

$$d_1\ d_2\ NP\ d_6\ v\ d_5\ NP\ PP\ r_5\ r_6\ r_2\ r_1$$
$$d_1\ d_2\ NP\ d_7\ d_6\ v\ NP\ r_6\ PP\ r_7\ r_2\ r_1$$

Reduce

mar$^+$: read the same r_i from both states, if no mae was used since the last mac or mar

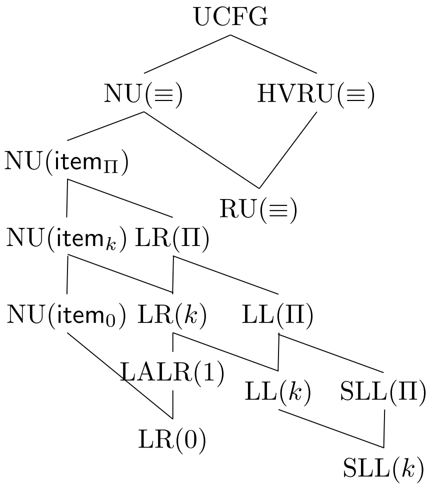
Noncanonical Unambiguity

- ma=mas ∪ mae ∪ mac ∪ mar
- $\mathcal{G}$ is noncanonically unambiguous, noted NU(≡), if there does not exist a relation (q_s, q_s) ma* (q_f, q_f) that uses mac at some step.
- Computation in $\mathcal{O}(|\Gamma/\equiv|^2)$ in space.

Comparisons

- Regular Unambiguity RU($\equiv$)
- Bounded-length detection schemes
- LR(k) and LR-Regular (LR(Π))
- Horizontal and vertical ambiguity (HVRU($\equiv$))

Comparisons



Bounded-length detection

[Gorn, 1963, Cheung and Uzgalis, 1995, Schröer, 2001, Jampana, 2005]

- generate sentences up to some length and find ambiguities
- not conservative
- define an equivalence relation prefix_m
- if $w_b \neq w'_b \in \mathcal{L}(\Gamma/\text{prefix}_m) \cap T_b^*$, $w = w'$ and $|w| \leq m$, then w_b and w'_b are in $\mathcal{L}(\mathcal{G}_b)$
- generation needs to construct the sentence a^{2^n+1} to find $\mathcal{G}_4^n$ ambiguous, but $\mathcal{G}_4^n \notin \text{NU}(\text{item}_0)$

$$S \to A \mid B_n a,\ A \to Aaa \mid a,\ B_1 \to aa,\ B_2 \to B_1 B_1,\ \ldots,\ B_n \to B_{n-1} B_{n-1} \qquad (\mathcal{G}_4^n)$$

Bounded-length detection

[Gorn, 1963, Cheung and Uzgalis, 1995, Schröer, 2001, Jampana, 2005]

- generate sentences up to some length and find ambiguities
- not conservative
- define an equivalence relation prefix_m
- if $w_b \neq w'_b \in \mathcal{L}(\Gamma/\text{prefix}_m) \cap T_b^*$, $w = w'$ and $|w| \leq m$, then w_b and w'_b are in $\mathcal{L}(\mathcal{G}_b)$
- generation needs to construct the sentence a^{2^n+1} to find $\mathcal{G}_4^n$ ambiguous, but $\mathcal{G}_4^n \notin \text{NU}(\text{item}_0)$

$$S \rightarrow A \mid B_n a,\ A \rightarrow Aaa \mid a,\ B_1 \rightarrow aa,\ B_2 \rightarrow B_1 B_1,\ \ldots,\ B_n \rightarrow B_{n-1} B_{n-1} \quad (\mathcal{G}_4^n)$$

Bounded-length detection

[Gorn, 1963, Cheung and Uzgalis, 1995, Schröer, 2001, Jampana, 2005]

- generate sentences up to some length and find ambiguities
- not conservative
- define an equivalence relation prefix_m
- if $w_b \neq w'_b \in \mathcal{L}(\Gamma/\text{prefix}_m) \cap T_b^*$, $w = w'$ and $|w| \leq m$, then w_b and w'_b are in $\mathcal{L}(\mathcal{G}_b)$
- generation needs to construct the sentence a^{2^n+1} to find $\mathcal{G}_4^n$ ambiguous, but $\mathcal{G}_4^n \notin \text{NU}(\text{item}_0)$

$$S \to A \mid B_n a,\ A \to Aaa \mid a,\ B_1 \to aa,\ B_2 \to B_1 B_1,\ \ldots,\ B_n \to B_{n-1} B_{n-1}$$

$(\mathcal{G}_4^n)$

Bounded-length detection

[Gorn, 1963, Cheung and Uzgalis, 1995, Schröer, 2001, Jampana, 2005]

- generate sentences up to some length and find ambiguities
- not conservative
- define an equivalence relation prefix_m
- if $w_b \neq w'_b \in \mathcal{L}(\Gamma/\text{prefix}_m) \cap T_b^*$, $w = w'$ and $|w| \leq m$, then w_b and w'_b are in $\mathcal{L}(\mathcal{G}_b)$
- generation needs to construct the sentence a^{2^n+1} to find $\mathcal{G}_4^n$ ambiguous, but $\mathcal{G}_4^n \notin \text{NU}(\text{item}_0)$

$$S \rightarrow A \mid B_n a,\ A \rightarrow Aaa \mid a,\ B_1 \rightarrow aa,\ B_2 \rightarrow B_1 B_1,\ \ldots,\ B_n \rightarrow B_{n-1} B_{n-1} \qquad (\mathcal{G}_4^n)$$

LR(k) and LR-Regular

[Knuth, 1965, Hunt III et al., 1975, Čulik and Cohen, 1973, Heilbrunner, 1983]

- conservative tests
- define item_Π equivalence relation
- mac corresponds to an LR(0) conflict
- subsequent ma^* implies that lookaheads match
- $\text{LR}(\Pi) \subset \text{NU}(\text{item}_\Pi)$
- need a $\text{LR}(2^n)$ test to prove $\mathcal{G}_3^n$ unambiguous, but $\mathcal{G}_3^n \in \text{NU}(\text{item}_0)$

$$S \rightarrow A \,|\, B_n, \;\; A \rightarrow Aaa \,|\, a, \;\; B_1 \rightarrow aa, \;\; B_2 \rightarrow B_1 B_1, \;\; \ldots, \;\; B_n \rightarrow B_{n-1} B_{n-1} \qquad (\mathcal{G}_3^n)$$

LR(k) and LR-Regular

[Knuth, 1965, Hunt III et al., 1975, Čulik and Cohen, 1973, Heilbrunner, 1983]

- conservative tests
- define item_Π equivalence relation
- mac corresponds to an LR(0) conflict
- subsequent ma* implies that lookaheads match
- $\text{LR}(\Pi) \subset \text{NU}(\text{item}_\Pi)$
- need a LR(2^n) test to prove $\mathcal{G}_3^n$ unambiguous, but $\mathcal{G}_3^n \in \text{NU}(\text{item}_0)$

$$S \rightarrow A \mid B_n, \;\; A \rightarrow Aaa \mid a, \;\; B_1 \rightarrow aa, \;\; B_2 \rightarrow B_1 B_1, \;\; \dots, \;\; B_n \rightarrow B_{n-1} B_{n-1} \qquad (\mathcal{G}_3^n)$$

LR(k) and LR-Regular

[Knuth, 1965, Hunt III et al., 1975, Čulik and Cohen, 1973, Heilbrunner, 1983]

- conservative tests
- define item_Π equivalence relation
- mac corresponds to an LR(0) conflict
- subsequent ma^* implies that lookaheads match
- $\text{LR}(\Pi) \subset \text{NU}(\text{item}_\Pi)$
- need a $\text{LR}(2^n)$ test to prove $\mathcal{G}_3^n$ unambiguous, but $\mathcal{G}_3^n \in \text{NU}(\text{item}_0)$

$$S \rightarrow A \mid B_n, \;\; A \rightarrow Aaa \mid a, \;\; B_1 \rightarrow aa, \;\; B_2 \rightarrow B_1 B_1, \;\; \ldots, \;\; B_n \rightarrow B_{n-1} B_{n-1} \qquad (\mathcal{G}_3^n)$$

LR(k) and LR-Regular

[Knuth, 1965, Hunt III et al., 1975, Čulik and Cohen, 1973, Heilbrunner, 1983]

- conservative tests
- define item$_\Pi$ equivalence relation
- mac corresponds to an LR(0) conflict
- subsequent ma* implies that lookaheads match
- LR(Π) $\subset$ NU(item$_\Pi$)
- need a LR(2^n) test to prove $\mathcal{G}_3^n$ unambiguous, but $\mathcal{G}_3^n \in$ NU(item$_0$)

$$S \to A \mid B_n,\ A \to Aaa \mid a,\ B_1 \to aa,\ B_2 \to B_1 B_1,\ \ldots,\ B_n \to B_{n-1} B_{n-1} \quad (\mathcal{G}_3^n)$$

LR(k) and LR-Regular

[Knuth, 1965, Hunt III et al., 1975, Čulik and Cohen, 1973, Heilbrunner, 1983]

- conservative tests
- define item_Π equivalence relation
- mac corresponds to an LR(0) conflict
- subsequent ma^* implies that lookaheads match
- $\text{LR}(\Pi) \subset \text{NU}(\text{item}_\Pi)$
- need a LR(2^n) test to prove $\mathcal{G}_3^n$ unambiguous, but $\mathcal{G}_3^n \in \text{NU}(\text{item}_0)$

$$S \to A \mid B_n,\ A \to Aaa \mid a,\ B_1 \to aa,\ B_2 \to B_1 B_1,\ \ldots,\ B_n \to B_{n-1} B_{n-1} \quad (\mathcal{G}_3^n)$$

LR(k) and LR-Regular

[Knuth, 1965, Hunt III et al., 1975, Čulik and Cohen, 1973, Heilbrunner, 1983]

- conservative tests
- define item_Π equivalence relation
- mac corresponds to an LR(0) conflict
- subsequent ma^* implies that lookaheads match
- $\text{LR}(\Pi) \subset \text{NU}(\text{item}_\Pi)$
- need a $\text{LR}(2^n)$ test to prove $\mathcal{G}_3^n$ unambiguous, but $\mathcal{G}_3^n \in \text{NU}(\text{item}_0)$

$$S \to A \mid B_n,\ \ A \to Aaa \mid a,\ \ B_1 \to aa,\ \ B_2 \to B_1 B_1,\ \ \ldots,\ \ B_n \to B_{n-1} B_{n-1} \qquad (\mathcal{G}_3^n)$$

Conclusions

- ambiguity detection
  - conservative
  - provably better than several other techniques
  - also experimentally better
- position automata
  - abstract left-to-right walks in CFGs
  - lattice of equivalence relations
  - equivalence relations too constraining for other uses?

Conclusions

- ambiguity detection
  - conservative
  - provably better than several other techniques
  - also experimentally better
- position automata
  - abstract left-to-right walks in CFGs
  - lattice of equivalence relations
  - equivalence relations too constraining for other uses?

H. J. S. Basten. Ambiguity detection methods for context-free grammars. Master's thesis, Centrum voor Wiskunde en Informatica, Universiteit van Amsterdam, 2007.

D. G. Cantor. On the ambiguity problem of Backus systems. J. ACM, 9(4):477–479, 1962. ISSN 0004-5411. doi: 10.1145/321138.321145.

B. S. N. Cheung and R. C. Uzgalis. Ambiguity in context-free grammars. In SAC'95, pages 272–276. ACM Press, 1995. ISBN 0-89791-658-1. doi: 10.1145/315891.315991.

N. Chomsky and M. P. Schützenberger. The algebraic theory of context-free languages. In P. Braffort and D. Hirshberg, editors, Computer Programming and Formal Systems, Studies in Logic, pages 118–161