Graph-Based Parsing Strategies for Categorial Grammars

Richard Moot
Utrecht institute of Linguistics-OTS
Trans 10
3512 JK Utrecht
Richard.Moot@let.uu.nl
Overview

Simple Tree Grammars

Graph Grammars

Extensions

Computational Aspects

Conclusions
**Simple Tree Grammars**

**Terminals**

- Zaphod
- Marvin
- android

**Nonterminals**

- $s$, $n$, $np$, ...

**Complex Lexical Entries**

Binary trees, with a nonterminal as root and at least one terminal and any number of nonterminals as leaves.

**Grammatical Expressions**

A tree where all leaves are terminals with a nonterminal $x$ as root is a grammatical expression of type $x$. 
Example: Lexical Trees

```
np    np    np    n    n    n
Zaphod  Marvin  Ford  ship  android  towel

np

the

n

harmless

n

paranoid

n

s

s

s

np    np    np
snores  annoys  stole

np    np    np

Graph-Based Parsing Strategies for Categorial Grammars

CLIN99
Example: ‘Ford annoys the paranoid android’
Graph Systems

New Connectors

Example: Lexical Graphs
Example: ‘Everyone annoys the paranoid android’
Graph Contractions

Example Revisited
Extensions

Unary Branches

Mode Information

Structural Rules
Example: Modified Lexicon

Example: ‘ship which Zaphod stole’
which ship stole Zaphod
which ship

Zaphod stole

\( n \)
which ship Zaphod stole

which ship Zaphod stole
Computational Aspects

Lexical Lookup

In the worst case, a grammar where each expression has at most $k$ lexical entries, produces $k^n$ lookups for a sentence of length $n$.

Connections

In the worst case, when there are $n$ occurrences of a nonterminal as both a leaf and a root, we have to consider $n!$ connections.

Graph Conversions

Without restrictions on the structural rules, converting the graph to a tree is undecidable in the worst case.
Conclusions

– Switching from trees to graphs allows us to give a natural account of phenomena like extraction and quantification.

– Grammar specific structural rules make our system flexible and extendable.

– The mode system, combined with the structural rules gives us a modular system.

– Generality has a (computational) price.

Goal

To identify subsystems with better computational properties.
Categorial Grammars

\[
\begin{align*}
\| x : A / _i B \| & = \quad \text{\underline{\| A \| ^-}} \\
\| x : B \setminus _i A \| & = \quad \text{\underline{\| B \| ^+}} \\
\| x : A \bullet _i B \| & = \\
\| A / _i B \|^- & = \quad \text{\underline{\| B \| ^+}} \\
\| A / _i B \|^+ & = \quad \text{\underline{\| A \| ^+}}
\end{align*}
\]