Algorithmic Properties of CatLog3

Max Kanovich, Stepan G. Kuznetsov,
Stepan L. Kuznetsov, Andre Scedrov

CatLog

- CatLog is a categorial grammar parser (theorem-prover), being developed by Glyn Morrill and his team in UPC, Barcelona.
- In categorial grammars, words (lexemes) of the target language are associated with syntactic types, that is, formulae of a non-classical logic.
- $a_1 \dots a_n$ is accepted, if there exist such formulae $A_1, \dots, A_n$ that A_i is associated with a_i and the sequent $A_1, \dots, A_n \Rightarrow S$ is derivable.
- The calculus used in CatLog is an extension of the Lambek calculus (Lambek, 1958).

The Lambek Calculus with the Unit

$$\frac{}{A \Rightarrow A}\ id$$

$$\frac{\Gamma \Rightarrow B \quad \Delta_1, C, \Delta_2 \Rightarrow D}{\Delta_1, C \,/\, B, \Gamma, \Delta_2 \Rightarrow D}\ /L \qquad \frac{\Gamma, B \Rightarrow C}{\Gamma \Rightarrow C \,/\, B}\ /R$$

$$\frac{\Gamma \Rightarrow A \quad \Delta_1, C, \Delta_2 \Rightarrow D}{\Delta_1, \Gamma, A \setminus C, \Delta_2 \Rightarrow D}\ \setminus L \qquad \frac{A, \Gamma \Rightarrow C}{\Gamma \Rightarrow A \setminus C}\ \setminus R$$

$$\frac{\Delta_1, A, B, \Delta_2 \Rightarrow D}{\Delta_1, A \bullet B, \Delta_2 \Rightarrow D}\ \bullet L \qquad \frac{\Delta \Rightarrow A \quad \Gamma \Rightarrow B}{\Delta, \Gamma \Rightarrow A \bullet B}\ \bullet R$$

$$\frac{\Delta_1, \Delta_2 \Rightarrow A}{\Delta_1, \mathbf{I}, \Delta_2 \Rightarrow A}\ \mathbf{I}L \qquad \frac{}{\Lambda \Rightarrow \mathbf{I}}\ \mathbf{I}R$$

Parasitic Extraction and Subexponentials

"the paper that John signed [] without reading []"

- Two *gaps* (extraction sites); the one after *"reading"* is the *parasitic* one.
- Parasitic extraction is handled using a *subexponential* modality which allows *non-local contraction*:

$$\frac{\Gamma_1, !A, \Gamma_2, !A, \Gamma_3 \Rightarrow C}{\Gamma_1, !A, \Gamma_2, \Gamma_3 \Rightarrow C} \qquad \frac{\Gamma_1, !A, \Gamma_2, !A, \Gamma_3 \Rightarrow C}{\Gamma_1, \Gamma_2, !A, \Gamma_3 \Rightarrow C}$$

- Another important rule for ! is *dereliction:*

$$\frac{\Gamma_1, A, \Gamma_2 \Rightarrow C}{\Gamma_1, !A, \Gamma_2 \Rightarrow C}$$

- One can also impose permutation rules, but usually not weakening (which is linguistically inadequate).

Parasitic Extraction and Subexponentials

"the paper that John signed [] without reading []"

- In the lexicon, *"that"* receives a type with !, namely, $(CN \setminus CN) / (S / !N)$.
- The corresponding sequent is derived as follows:

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{N, (N \setminus S) / N, N, ((N \setminus S) \setminus (N \setminus S)) / (N \setminus S), (N \setminus S) / N, N \Rightarrow S}{N, (N \setminus S) / N, !N, ((N \setminus S) \setminus (N \setminus S)) / (N \setminus S), (N \setminus S) / N, !N \Rightarrow S}}{N, (N \setminus S) / N, ((N \setminus S) \setminus (N \setminus S)) / (N \setminus S), (N \setminus S) / N, !N \Rightarrow S}}{N, (N \setminus S) / N, ((N \setminus S) \setminus (N \setminus S)) / (N \setminus S), (N \setminus S) / N \Rightarrow S / !N \qquad CN, CN \setminus CN \Rightarrow CN}}{CN, (CN \setminus CN) / (S / !N), N, (N \setminus S) / N, ((N \setminus S) \setminus (N \setminus S)) / (N \setminus S), (N \setminus S) / N \Rightarrow CN \qquad N \Rightarrow N}}{N / CN, CN, (CN \setminus CN) / (S / !N), N, (N \setminus S) / N, ((N \setminus S) \setminus (N \setminus S)) / (N \setminus S), (N \setminus S) / N \Rightarrow N}
$$

- The sequent on top corresponds to

  "John signed the paper without reading the paper"

  and is derivable in the Lambek calculus.

The Workflow

General idea: contraction leads to undecidability.

Technical issues: undecidability proof becomes more involved for sophisticated versions of contraction which involve brackets.

System with contraction	**Undecidability proof**
Lambek calculus with full-power exponential (allows weakening, permutation, and contraction)	Lincoln et al. 1992
Lambek calculus with a *relevant* modality (permutation & contraction, no weakening) no brackets	K. K. S., FG 2016
The system with brackets of Morrill & Valentin 2015, Morrill 2017	K. K. S., FCT 2017
The new system with brackets of Morrill 2018–19 (resembling Morrill 2011)	**this talk**

Brackets

*"the paper that John signed and Pete ate a pie"

- This noun phrase is clearly ungrammatical.
- However it is generated by our grammar, since *"John signed the paper and Pete ate a pie"* is a correct sentence.
- In order to address this issue, Morrill (1992) and Moortgat (1996) introduce *brackets* which embrace *islands* not allowed to be penetrated by $!N$.
- *Strong* islands, like *and*-coordinated sentences, are double-bracketed; no penetration possible.
- Subject groups, *without*-clauses, ... are *weak* islands. These are single-bracketed and can be penetrated using the special form of contraction. This is used for parasitic extraction.

The Lambek Calculus with Brackets

- Now the antecedents are built using both comma (metasyntactic product) and brackets.
- Lambek rules: $\overline{A \Rightarrow A}$

$$\frac{\Gamma \Rightarrow B \quad \Xi(\Delta_1, C, \Delta_2) \Rightarrow D}{\Xi(\Delta_1, C \,/\, B, \Gamma, \Delta_2) \Rightarrow D} \;/L \qquad \frac{\Gamma, B \Rightarrow C}{\Gamma \Rightarrow C \,/\, B} \;/R \qquad \frac{}{\Lambda \Rightarrow \mathbf{I}} \;\mathbf{I}R$$

$$\frac{\Gamma \Rightarrow A \quad \Xi(\Delta_1, C, \Delta_2) \Rightarrow D}{\Xi(\Delta_1, \Gamma, A \setminus C, \Delta_2) \Rightarrow D} \;\setminus L \qquad \frac{A, \Gamma \Rightarrow C}{\Gamma \Rightarrow A \setminus C} \;\setminus R \qquad \frac{\Xi(\Delta_1, \Delta_2) \Rightarrow A}{\Xi(\Delta_1, \mathbf{I}, \Delta_2) \Rightarrow A} \;\mathbf{I}L$$

$$\frac{\Xi(\Delta_1, A, B, \Delta_2) \Rightarrow D}{\Xi(\Delta_1, A \bullet B, \Delta_2) \Rightarrow D} \;\bullet L \qquad \frac{\Delta \Rightarrow A \quad \Gamma \Rightarrow B}{\Delta, \Gamma \Rightarrow A \bullet B} \;\bullet R$$

- Rules operating brackets, using *bracket modalities*:

$$\frac{\Xi(\Delta_1, A, \Delta_2) \Rightarrow B}{\Xi(\Delta_1, [[]^{-1}A], \Delta_2) \Rightarrow B} \;[]^{-1}L \qquad \frac{[\Xi] \Rightarrow A}{\Xi \Rightarrow []^{-1}A} \;[]^{-1}R$$

$$\frac{\Xi(\Delta_1, [A], \Delta_2) \Rightarrow B}{\Xi(\Delta_1, \langle\rangle A, \Delta_2) \Rightarrow B} \;\langle\rangle L \qquad \frac{\Xi \Rightarrow A}{[\Xi] \Rightarrow \langle\rangle A} \;\langle\rangle R$$

Lambek Grammars with Brackets

- For bracketed calculi, the definition of acceptance of a word by the grammar should be modified.
- $a_1 \ldots a_n$ is *t-accepted* by the grammar, if there exists such Π that $\Pi \Rightarrow S$ is derivable and removing all brackets (but not bracket modalities) from Π yields $A_1, \ldots, A_n$, where A_i is associated with a_i.
- For example, *"John likes Mary and Pete likes Ann"* should be bracketed as follows: "[[[*John*] *likes Mary and* [*Pete*] *likes Ann*]]" before deriving.
- In CatLog, the bracketing is requested from the user. However, there exist bracket induction (guessing) algorithms for fragments of the CatLog calculus (Morrill et al., FG 2018).
- We shall discuss bracket induction in the end of the talk.

Contraction with Brackets

- In Morrill's systems, the !-formulae are kept in special areas called *stoups*. Stoups are multisets of formulae.
- This is an element of *focusing* used to facilitate proof search.
- Each bracketed domain has a stoup, as well as the whole antecedent.
- Morrill presents two versions of the rule set for !. Their general idea is that contraction erases one !-formula from a weak island, provided the same formula is in the outer area. However, the island has to be somehow modified, in order to prevent double usage.

Two Morrill's Systems

- The rules operating the stoup and ! on the left (dereliction and permutation) are the same in both systems:

$$\frac{\Xi(\zeta;\Gamma_1,A,\Gamma_2)\Rightarrow B}{\Xi(\zeta,A;\Gamma_1,\Gamma_2)\Rightarrow B}\ !P \qquad \frac{\Xi(\zeta,A;\Gamma_1,\Gamma_2)\Rightarrow B}{\Xi(\zeta;\Gamma_1,!A,\Gamma_2)\Rightarrow B}\ !L$$

- The older system (Morrill 2017 in *Linguistics and Philosophy*) uses the following versions of contraction and right rule for !:

$$\frac{\Xi(\zeta,A;\Gamma_1,[A;\Gamma_2],\Gamma_3)\Rightarrow B}{\Xi(\zeta,A;\Gamma_1,\Gamma_2,\Gamma_3)\Rightarrow B}\ !C \qquad \frac{\zeta;\Lambda\Rightarrow A}{\zeta;\Lambda\Rightarrow !A}\ !R$$

  Undecidability proved in our FCT 2017 paper.

- The new system (Morrill 2018 in *J. Lang. Model.* and 2019 in *J. Log. Lang. Inform.*, resembling Morrill 2011 book):

$$\frac{\Xi(\zeta,A;\Gamma_1,[\xi,A;\Gamma_2],\Gamma_3)\Rightarrow B}{\Xi(\zeta,A;\Gamma_1,[[\xi;\Gamma_2]],\Gamma_3)\Rightarrow B}\ !C \qquad \frac{\varnothing;!A\Rightarrow B}{\varnothing;!A\Rightarrow !B}\ !R$$

- The right rule is not needed in undecidability proofs. In linguistic applications, one needs $!A \Rightarrow !A$.

Lambek and Bracket Rules with Stoups

$$\frac{}{\varnothing; A \Rightarrow A}\ id$$

$$\frac{\zeta_1; \Gamma \Rightarrow B \quad \Xi(\zeta_2; \Delta_1, C, \Delta_2) \Rightarrow D}{\Xi(\zeta_1, \zeta_2; \Delta_1, C \,/\, B, \Gamma, \Delta_2) \Rightarrow D}\ /L \qquad \frac{\zeta; \Gamma, B \Rightarrow C}{\zeta; \Gamma \Rightarrow C \,/\, B}\ /R$$

$$\frac{\zeta_1; \Gamma \Rightarrow A \quad \Xi(\zeta_2; \Delta_1, C, \Delta_2) \Rightarrow D}{\Xi(\zeta_1, \zeta_2; \Delta_1, \Gamma, A \backslash C, \Delta_2) \Rightarrow D}\ \backslash L \qquad \frac{\zeta; A, \Gamma \Rightarrow C}{\zeta; \Gamma \Rightarrow A \backslash C}\ \backslash R$$

$$\frac{\Xi(\zeta; \Delta_1, A, B, \Delta_2) \Rightarrow D}{\Xi(\zeta; \Delta_1, A \bullet B, \Delta_2) \Rightarrow D}\ \bullet L \qquad \frac{\zeta_1; \Delta \Rightarrow A \quad \zeta_2; \Gamma \Rightarrow B}{\zeta_1, \zeta_2; \Delta, \Gamma \Rightarrow A \bullet B}\ \bullet R$$

$$\frac{\Xi(\zeta; \Delta_1, \Delta_2) \Rightarrow A}{\Xi(\zeta; \Delta_1, \mathbf{I}, \Delta_2) \Rightarrow A}\ \mathbf{I}L \qquad \frac{}{\varnothing; \Lambda \Rightarrow \mathbf{I}}\ \mathbf{I}R$$

$$\frac{\Xi(\zeta; \Delta_1, A, \Delta_2) \Rightarrow B}{\Xi(\zeta; \Delta_1, [\varnothing; []^{-1}A], \Delta_2) \Rightarrow B}\ []^{-1}L \qquad \frac{\varnothing; [\Xi] \Rightarrow A}{\Xi \Rightarrow []^{-1}A}\ []^{-1}R$$

$$\frac{\Xi(\zeta; \Delta_1, [\varnothing; A], \Delta_2) \Rightarrow B}{\Xi(\zeta; \Delta_1, \langle\rangle A, \Delta_2) \Rightarrow B}\ \langle\rangle L \qquad \frac{\Xi \Rightarrow A}{\varnothing; [\Xi] \Rightarrow \langle\rangle A}\ \langle\rangle R$$

Example of Derivation

$$N \Rightarrow N$$

$$[N] \Rightarrow \langle\rangle N \qquad S \Rightarrow S$$

$$\langle\rangle N \backslash S \Rightarrow \langle\rangle N \backslash S \qquad [N], \langle\rangle N \backslash S \Rightarrow S$$

$$N \Rightarrow N \qquad [N], \langle\rangle N \backslash S, (\langle\rangle N \backslash S) \backslash (\langle\rangle N \backslash S)) \Rightarrow S$$

$$[N], (\langle\rangle N \backslash S) / N, N, (\langle\rangle N \backslash S) \backslash (\langle\rangle N \backslash S)) \Rightarrow S$$

$$\langle\rangle N \backslash S \Rightarrow \langle\rangle N \backslash S \qquad [N], (\langle\rangle N \backslash S) / N, N, [[]^{-1}((\langle\rangle N \backslash S) \backslash (\langle\rangle N \backslash S))] \Rightarrow S$$

$$N \Rightarrow N \qquad [N], (\langle\rangle N \backslash S) / N, N, [([]^{-1}((\langle\rangle N \backslash S) \backslash (\langle\rangle N \backslash S))) / (\langle\rangle N \backslash S), \langle\rangle N \backslash S] \Rightarrow S$$

$$[N], (\langle\rangle N \backslash S) / N, N, [([]^{-1}((\langle\rangle N \backslash S) \backslash (\langle\rangle N \backslash S))) / (\langle\rangle N \backslash S), (\langle\rangle N \backslash S) / N, N] \Rightarrow S$$

$$[N], (\langle\rangle N \backslash S) / N, N, [N; ([]^{-1}((\langle\rangle N \backslash S) \backslash (\langle\rangle N \backslash S))) / (\langle\rangle N \backslash S), (\langle\rangle N \backslash S) / N] \Rightarrow S$$

$$N; [N], (\langle\rangle N \backslash S) / N, [N; ([]^{-1}((\langle\rangle N \backslash S) \backslash (\langle\rangle N \backslash S))) / (\langle\rangle N \backslash S), (\langle\rangle N \backslash S) / N] \Rightarrow S$$

$$N; [N], (\langle\rangle N \backslash S) / N, [[([]^{-1}((\langle\rangle N \backslash S) \backslash (\langle\rangle N \backslash S))) / (\langle\rangle N \backslash S), (\langle\rangle N \backslash S) / N]] \Rightarrow S$$

$$CN \Rightarrow CN \qquad CN \Rightarrow CN$$

$$CN, CN \backslash CN \Rightarrow CN$$

$$[N], (\langle\rangle N \backslash S) / N, [[([]^{-1}((\langle\rangle N \backslash S) \backslash (\langle\rangle N \backslash S))) / (\langle\rangle N \backslash S), (\langle\rangle N \backslash S) / N]], !N \Rightarrow S \qquad CN, [[]^{-1}(CN \backslash CN)] \Rightarrow CN$$

$$[N], (\langle\rangle N \backslash S) / N, [[([]^{-1}((\langle\rangle N \backslash S) \backslash (\langle\rangle N \backslash S))) / (\langle\rangle N \backslash S), (\langle\rangle N \backslash S) / N]] \Rightarrow S / !N \qquad CN, [[[]^{-1}[]^{-1}(CN \backslash CN)]] \Rightarrow CN$$

$$CN, [[([]^{-1}[]^{-1}(CN \backslash CN)) / (S / !N), [N], (\langle\rangle N \backslash S) / N, [[([]^{-1}((\langle\rangle N \backslash S) \backslash (\langle\rangle N \backslash S))) / (\langle\rangle N \backslash S), (\langle\rangle N \backslash S) / N]]\ ]] \Rightarrow CN \qquad N \Rightarrow N$$

$$N / CN, CN, [[([]^{-1}[]^{-1}(CN \backslash CN)) / (S / !N), [N], (\langle\rangle N \backslash S) / N, [[([]^{-1}((\langle\rangle N \backslash S) \backslash (\langle\rangle N \backslash S))) / (\langle\rangle N \backslash S), (\langle\rangle N \backslash S) / N]]\ ]] \Rightarrow N$$

the paper [[that John signed [[without reading]]]]

Motivation for the New System

Following Morrill (2018), consider the following example:

**"the man who likes"*

Using the old contraction rule, one can analyse *"likes"* as a dependent clause with two gaps, where the parasitic one is the subject ("man who likes himself"):

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{N \Rightarrow N \qquad \dfrac{\dfrac{N \Rightarrow N}{[N] \Rightarrow \langle\rangle N} \quad S \Rightarrow S}{[N], \langle\rangle N \setminus S \Rightarrow S}}{[N], (\langle\rangle N \setminus S) / N, N \Rightarrow S}}{[N; \Lambda], (\langle\rangle N \setminus S) / N, N \Rightarrow S}}{N; [N; \Lambda], (\langle\rangle N \setminus S) / N \Rightarrow S}}{N; (\langle\rangle N \setminus S) / N \Rightarrow S}}{(\langle\rangle N \setminus S) / N, !N \Rightarrow S}}{(\langle\rangle N \setminus S) / N \Rightarrow S / !N}
$$

In this derivation, $!C$ instantiates an *empty* subject island.
With the new $!C$, this island should be explicitly declared (as a strong one) in the bracketing, and we can disallow empty ones.

Motivation for the New System

Following Morrill (2018), consider the following example:

**"the man who likes"*

Using the old contraction rule, one can analyse *"likes"* as a dependent clause with two gaps, where the parasitic one is the subject ("man who likes himself"):

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{N \Rightarrow N \qquad \dfrac{\dfrac{N \Rightarrow N}{[N] \Rightarrow \langle\rangle N} \quad S \Rightarrow S}{[N], \langle\rangle N \setminus S \Rightarrow S}}{[N], (\langle\rangle N \setminus S) \,/\, N, N \Rightarrow S}}{[N; \Lambda], (\langle\rangle N \setminus S) \,/\, N, N \Rightarrow S}}{N; [N; \Lambda], (\langle\rangle N \setminus S) \,/\, N \Rightarrow S}}{N; (\langle\rangle N \setminus S) \,/\, N \Rightarrow S}}{(\langle\rangle N \setminus S) \,/\, N, !N \Rightarrow S}}{(\langle\rangle N \setminus S) \,/\, N \Rightarrow S \,/\, !N}
$$

In this derivation, $!C$ instantiates an *empty* subject island.
With the new $!C$, this island should be explicitly declared (as a strong one) in the bracketing, and we can disallow empty ones.

Motivation for the New System

Following Morrill (2018), consider the following example:

**"the man who likes"*

Using the old contraction rule, one can analyse *"likes"* as a dependent clause with two gaps, where the parasitic one is the subject ("man who likes himself"):

$$
\begin{prooftree}
\AxiomC{$N \Rightarrow N$}
\AxiomC{$N \Rightarrow N$}
\UnaryInfC{$[N] \Rightarrow \langle\rangle N$}
\AxiomC{$S \Rightarrow S$}
\BinaryInfC{$[N], \langle\rangle N \setminus S \Rightarrow S$}
\BinaryInfC{$[N], (\langle\rangle N \setminus S) / N, N \Rightarrow S$}
\UnaryInfC{$[N; \Lambda], (\langle\rangle N \setminus S) / N, N \Rightarrow S$}
\UnaryInfC{$N; [N; \Lambda], (\langle\rangle N \setminus S) / N \Rightarrow S$}
\UnaryInfC{$N; (\langle\rangle N \setminus S) / N \Rightarrow S$}
\UnaryInfC{$(\langle\rangle N \setminus S) / N, !N \Rightarrow S$}
\UnaryInfC{$(\langle\rangle N \setminus S) / N \Rightarrow S / !N$}
\end{prooftree}
$$

In this derivation, $!C$ instantiates an *empty* subject island.
With the new $!C$, this island should be explicitly declared (as a strong one) in the bracketing, and we can disallow empty ones.

Undecidability Proof Sketch

- Encoding semi-Thue systems.
- Each rewriting rule $x_1 \ldots x_m \to y_1 \ldots y_k$ is encoded by $A_i = (x_1 \bullet \ldots \bullet x_m) \,/\, (y_1 \bullet \ldots \bullet y_k)$.
- $Z_i = []^{-1}(!A_i \bullet \langle\rangle\langle\rangle \mathbf{I})$.
- **Theorem.** The sequent

$$\mathbf{I} \,/\, !Z_1, \ldots, \mathbf{I} \,/\, !Z_n, \mathbf{I} \,/ \langle\rangle\langle\rangle \mathbf{I}, !Z_1, \ldots, !Z_n; [[\Lambda]], a_1, \ldots, a_\ell \Rightarrow s$$

is derivable iff $a_1 \ldots a_\ell$ is derivable from s in the semi-Thue system.

Undecidability Proof Sketch

- **Theorem.** The sequent

$$\mathbf{I} \,/\, !Z_1, \ldots, \mathbf{I} \,/\, !Z_n, \mathbf{I} \,/\, \langle\rangle\langle\rangle\mathbf{I}, !Z_1, \ldots, !Z_n; [[\Lambda]], a_1, \ldots, a_\ell \Rightarrow s$$

  is derivable iff $a_1 \ldots a_\ell$ is derivable from s in the semi-Thue system.
- The $\mathbf{I} \,/ \ldots$ formulae are used to handle the base ($s \to s$) case.
- For the rewriting step, one uses contraction and puts $!Z_i$ into the island:

$$[[\Lambda]] \rightsquigarrow [![]^{-1}(!A_i \bullet \langle\rangle\langle\rangle\mathbf{I}); \Lambda]$$

  Next, $[]^{-1}$ removes the brackets, the extra $!$ over A_i puts it into the right place, and $\langle\rangle\langle\rangle\mathbf{I}$ restores the island.
- The backward translation is done by forgetting brackets and mapping onto the Lambek calculus with a full-power subexponential: this yields a sequent equivalent to $!A_1, \ldots, !A_n, a_1, \ldots, a_\ell \Rightarrow s$ (Lincoln et al. 1992).

Positive Results

- The calculi discussed above are actually used in CatLog—so how could they be undecidable?
- Linguistically interesting sequents fall into decidable fragments.
- For the older system, such a fragment is guarded by the *bracket non-negative condition* (BNNC): in any negative !-formula there should be no positive occurrences of $[]^{-1}$ and no negative occurrences of $\langle\rangle$.
- Derivability problem for sequents obeying BNNC is decidable (Morrill & Valentin, 2015), belonging to NP (K. K. S., FCT 2017).

Positive Results

- For the new system, we introduce a dual *bracket non-positive condition* (BNPC): in any negative !-formula there should be no *negative* occurrences of $[]^{-1}$ and no *positive* occurrences of $\langle\rangle$.
- The reason for such a dualisation is as follows. In Morrill's older system, $!C$ removes a pair of brackets (when viewed from top to bottom). In the new system, it replaces one pair of brackets with two, i.e., adds a pair of brackets.
- Under the BNPC, Morrill's new system is decidable and belongs to NP.
- Interestingly enough, one does not need to invent sophisticated proof-search algorithms: once the BNPC is imposed, the standard non-deterministic proof search would terminate in polynomial time.

Decidability Proof Idea

- The only rule which could potentially make proof search infinite, is $!C$ (contraction).
- However, we can estimate the number of contractions using brackets.
- For the new calculus, we have

$$\#[] = \#B^{+} - \#B^{-} + \#!C,$$

  where $\#[]$ is the number of bracket pairs in the goal sequent; $\#B^{+}$ is the number of $[]^{-1}L$ and $\langle\rangle R$ applications; $\#B^{-}$ is the number of $[]^{-1}R$ and $\langle\rangle L$; $\#!C$ is the number of $!C$.
- This gives $\#!C \leq \#B^{-} + \#[]$, and the latter is bounded by the number of bracket pairs and bracket modalities ($[]^{-1}$, $\langle\rangle$) in the goal sequent.
- Thus, we get the necessary complexity bound and quickly achieve NP decidability.

Additive Connectives

- The systems in question can be extended by *additive* operations, $\wedge$ (additive conjuction) and $\vee$ (additive disjunction).
- Inferences rules for them are as follows:

$$\frac{\Xi(A) \Rightarrow D}{\Xi(A \wedge B) \Rightarrow D} \quad \frac{\Xi(B) \Rightarrow D}{\Xi(A \wedge B) \Rightarrow D} \wedge L \qquad \frac{\Gamma \Rightarrow A \quad \Gamma \Rightarrow B}{\Gamma \Rightarrow A \wedge B} \wedge R$$

$$\frac{\Xi(A) \Rightarrow D \quad \Xi(B) \Rightarrow D}{\Xi(A \vee B) \Rightarrow D} \vee L \qquad \frac{\Gamma \Rightarrow A}{\Gamma \Rightarrow A \vee B} \quad \frac{\Gamma \Rightarrow B}{\Gamma \Rightarrow A \vee B} \vee R$$

Additive Connectives

- The systems with additive operations are conservative extensions of the *multiplicative-additive Lambek calculus* (MALC).
- MALC is PSPACE-complete.
- The lower bound is due to Kanovich (1994), and the upper one is based on the fact that MALC proofs' depth can be bounded polynomially.
- For Morrill's systems extended with additive operations, under bracket restrictions (BNNC for the older one and BNPC for the new one), we also obtain the PSPACE upper bound.

Bracket Induction

- Finally, let us discuss more complicated algorithmic problems than derivability.
- These problems arise in practical parsing using categorial grammars.
- First, a word of the language can have several syntactic types, so before proving the sequent the algorithm should determine which types to use.
- This is not an issue, because our algorithms are already non-deterministic.
- A more serious issue is as follows. In order for our algorithms to work, the sequent should be properly bracketed before starting proof search.
- In real life, brackets should also be guessed, or *induced.*

Bracket Induction

- Formally, the *bracket induction problem* is formulated as follows: given a sequent of the form $A_1, \ldots, A_n \Rightarrow B$ (without brackets and stoups), determine whether there is a way to put brackets on its left-hand side, so that the resulting sequent would be derivable in the given calculus.
- For Morrill's older system, bracket induction is decidable for sequents obeying the BNNC.
- The reason is in the following estimation on the maximal number of brackets in the goal:

$$\#[] = \#B^+ - \#B^- - \#!C \leq \#B^+ \leq n.$$

Bracket Induction

- In contrast, for the newer system bracket induction is undecidable even under the BNPC.
- The undecidability proof here involves creating an unbounded number of empty strong islands, $[[\Lambda]]$, at the stage of guessing brackets.
- Each island allows one application of $!C$, thus, the number of contraction also becomes unbounded.
- This allows more or less standard undecidability encodings.
- Thus, for the newer system bracket induction is essentially harder than derivability check.

![Thank you]