Proof composition mechanisms and their geometric interpretation

(Somewhat *IDEOLOGICAL* talk)

Alessio Guglielmi

University of Bath

5 November 2013

This talk is available at `http://cs.bath.ac.uk/ag/t/PCMTGI.pdf`
Deep inference web site: `http://alessio.guglielmi.name/res/cos/`

Outline

Problem: Compressing proofs and proof (search) spaces

Solution: Two proof composition mechanisms beyond Gentzen

Open deduction: composition by connectives and inference

Open deduction and complexity: smaller proofs than in Gentzen, locality

Atomic flows: locality brings geometry

Cut elimination by experiments: Gentzen's structure is too rigid

Normalisation with atomic flows: geometry is enough to normalise

Composition by substitution: more geometry, more efficiency, more naturality

Compressing proofs

How can we make proofs smaller? Known mechanisms:

1. Re-use the same sub-proof: cut rule. Proof theory.
2. Re-use the same sub-proof: dagness[1], or cocontraction: $\mathsf{c}{\uparrow}\,\dfrac{A}{A \wedge A}$.
3. Substitution: $\mathsf{sub}\,\dfrac{A}{A\sigma}$. In Frege, equivalent to (4).
4. Tseitin extension: $p \leftrightarrow A$ (where p is a fresh atom). Optimal?
5. Higher orders (including 2nd order propositional).

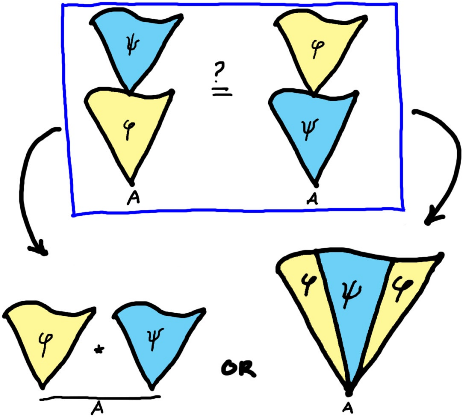
We will see that 1–4 (and a bit also 5) have a lot to do with proof composition: this is our main tool.

Our main objective is providing for small spaces of canonical proofs (= eliminating bureaucracy = getting good proof semantics).

[1] *Dagness is a terrible word. It is a combination of everything evil in this world. For example lies, smoke, gall, ugly teeth, darkness, greedy, venereal disease, war, death and so on. The word dagness can be compared to the devil but it is much worse. "My biggest fear is dagness".* The Urban Dictionary.

Compressing proof (search) spaces

How can we make proof (search) spaces smaller? This also has a lot to do with proof composition:



By allowing for more composed proofs we get:

- More proofs in the proof (search) space. This might be bad.
- Small (search) subspaces of canonical proofs. This is good.

How can we get better composition mechanisms?

Less is more. Let's make better use of what we have already.

Given two proofs $\phi : A \Rightarrow B$ and $\psi : C \Rightarrow D$:

1. For a logical connective $\star$ we have:

   $$\phi \star \psi : (A \star C) \Rightarrow (B \star D) \quad .$$

   Proofs are composed by the connectives of the formula language.

2. For an inference rule B/C we have:

   $$\phi/\psi : A \Rightarrow D \quad .$$

   Proofs are composed by inference rules.

3. For an atom a we have:

   $$\phi\{a \leftarrow \psi\} : A\{a \leftarrow C, \bar{a} \leftarrow \bar{D}\} \Rightarrow B\{a \leftarrow D, \bar{a} \leftarrow \bar{C}\} \quad .$$

   Proofs are composed by substitution.

Two new formalisms

Open deduction: composition by (1) connectives and (2) inference rules.

- It exists [7] and it can be taken as a definition for deep inference.
- It generalises[2] the sequent calculus by removing its restrictions.
- Sequents ≈ 'depth-1 inference without full inference composition'.
- Hypersequents ≈ 'depth-2 inference without full inference composition'.
- It compresses proofs by cut, dagness and depth itself (new, exponential speed-up).
- I'll show the main ideas and some results.

'Formalism B': open deduction + (3) substitution.

- It almost exists (work with Bruscoli, Gundersen and Parigot).
- It further compresses proofs by substitution (conjectured further superpolynomial speed-up, it is equivalent to Frege + substitution).
- I'll show some ideas and what I think we can get.

[2] It does not extend it. 'Extension' is another terrible word much worse than the devil, Agata.

Open-deduction system SKS

- Atomic rules:

$$\begin{array}{ccc}
\mathsf{ai}{\downarrow}\,\dfrac{\mathsf{t}}{a \vee \bar{a}} & \mathsf{aw}{\downarrow}\,\dfrac{\mathsf{f}}{a} & \mathsf{ac}{\downarrow}\,\dfrac{a \vee a}{a} \\
\textit{identity} & \textit{weakening} & \textit{contraction} \\
\mathsf{ai}{\uparrow}\,\dfrac{a \wedge \bar{a}}{\mathsf{f}} & \mathsf{aw}{\uparrow}\,\dfrac{a}{\mathsf{t}} & \mathsf{ac}{\uparrow}\,\dfrac{a}{a \wedge a} \\
\textit{cut} & \textit{coweakening} & \textit{cocontraction}
\end{array}$$

- Linear rules:

$$\begin{array}{cc}
\mathsf{s}\,\dfrac{A \wedge [B \vee C]}{(A \wedge B) \vee C} & \mathsf{m}\,\dfrac{(A \wedge B) \vee (C \wedge D)}{[A \vee C] \wedge [B \vee D]} \\
\textit{switch} & \textit{medial}
\end{array}$$

- Plus an ‘=’ linear rule (associativity, commutativity, units).
- Negation on atoms only.

The cut is atomic.

SKS is complete for propositional logic. See [2].

Examples of open deduction

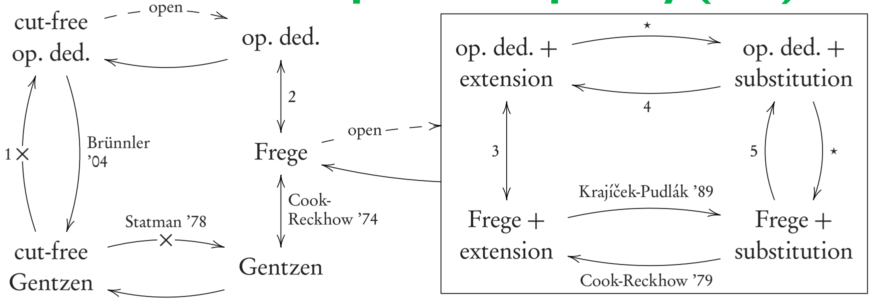
- $$\mathsf{m}\,\dfrac{\dfrac{a}{a\wedge a}\vee\dfrac{b}{b\wedge b}}{[a\vee b]\wedge[a\vee b]}\quad\wedge\quad\dfrac{a}{a\wedge a}$$

- $$\mathsf{s}\,\dfrac{\mathsf{m}\,\dfrac{\dfrac{\mathsf{t}}{a\vee\bar{a}}}{[a\vee\mathsf{t}]\wedge[\mathsf{t}\vee\bar{a}]}}{\left[\mathsf{s}\,\dfrac{[a\vee\mathsf{t}]\wedge\bar{a}}{\dfrac{a\wedge\bar{a}}{\mathsf{f}}\vee\mathsf{t}}\quad\vee\quad\mathsf{t}\right]}$$

Proofs are composed by the same operators as formulae (horizontally) and by inference rules (vertically).

Top-down symmetry: so inference steps can be made atomic (the medial rule, m, is illegal in Gentzen).

Open deduction and proof complexity (size)



⟶ = 'polynomially simulates'.

Open deduction:

- in the cut-free case, thanks to deep inference, has an exponential speed-up over the cut-free sequent calculus (e.g., over Statman tautologies)—1, see [3];
- has as small proofs as the best formalisms—2, 3, 4, 5, see [3];
- thanks to dagness, has quasipolynomial cut elimination (instead of exponential) [4, 10].

Open deduction and proof search complexity

Unconstrained bottom-up formula-driven proof search has horrendous complexity due to deep inference, because every connective can make the search tree branch.

However:

1. Das proved that in the presence of distributivity, a depth 2 proof system polynomially simulates any unbounded depth proof system [5]. This means that a very moderate increase of nondeterminism buys exponentially smaller proofs.
2. Focusing techniques should be facilitated by the more liberal proof composition.
3. In particular it should be possible to confine the search inside small sub-spaces of canonical proofs.
4. *THE SEQUENT CALCULUS WAS DESIGNED TO MAKE PROOF SEARCH FINITE, NOT NECESSARILY TO MAKE IT EFFICIENT.*
5. *SUGGESTION*: To exploit (1) design sort of a hypersequent calculus (so, depth 2 and no more) with full inference composition (= top-down symmetry). Start from Statman tautologies.

Locality

Deep inference allows for locality,

i.e.,

inference steps can be checked in constant time
(so, they are small).

E.g., atomic cocontraction:

$$\mathsf{m}\,\frac{\dfrac{a}{a \wedge a} \vee \dfrac{b}{b \wedge b}}{[a \vee b] \wedge [a \vee b]} \quad \wedge \quad \frac{a}{a \wedge a}$$

Thanks to locality Gundersen, Heijltjes and Parigot obtained a typed λ-calculus that achieves fully lazy sharing [9].

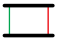
In Gentzen:

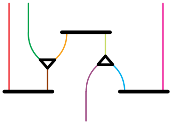
- no locality for (co)contraction (counterexample in [1]),
- no local reduction of cut into atomic form.

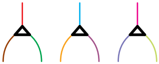
THIS IS BECAUSE AT THE TIME OF GENTZEN COMPLEXITY AND SEMANTICS WERE NOT A CONCERN: GENTZEN ONLY WANTED A WELL-ORDER.

Atomic flows: locality brings geometry

$$\mathsf{s}\,\dfrac{\mathsf{m}\,\dfrac{\dfrac{\mathsf{t}}{a\vee\bar a}}{[a\vee\mathsf{t}]\wedge[\mathsf{t}\vee\bar a]}}{\left[\mathsf{s}\,\dfrac{[a\vee\mathsf{t}]\wedge\bar a}{\dfrac{a\wedge\bar a}{\mathsf{f}}\vee\mathsf{t}}\vee\mathsf{t}\right]}=\dfrac{\left(\mathsf{s}\,\dfrac{a\wedge\left[\bar a\vee\dfrac{\mathsf{t}}{\bar a\vee a}\right]}{\dfrac{a\wedge\dfrac{\bar a\vee\bar a}{\bar a}}{\mathsf{f}}\vee\dfrac{a}{a\wedge a}}\wedge\bar a\right)}{a\wedge\dfrac{a\wedge\bar a}{\mathsf{f}}}\qquad\mathsf{m}\,\dfrac{\dfrac{a}{a\wedge a}\vee\dfrac{b}{b\wedge b}}{[a\vee b]\wedge[a\vee b]}\wedge\dfrac{a}{a\wedge a}$$



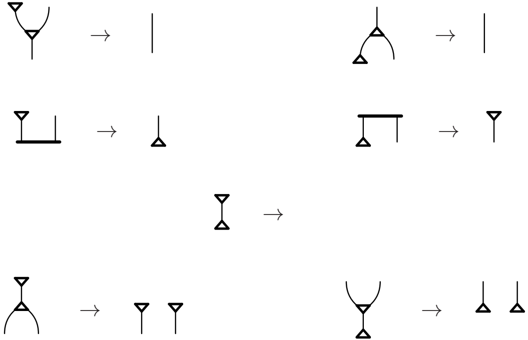




Below the proofs, their (atomic) flows [6] are shown:

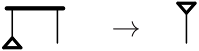
- only structural information is retained in flows;
- logical information is lost;
- flow size is polynomially related to derivation size;
- composition of proofs naturally correspond to composition of flows.

Flow reductions: (co)weakening (1)



Each flow reduction corresponds to a correct proof reduction.

Flow reductions: (co)weakening (2)

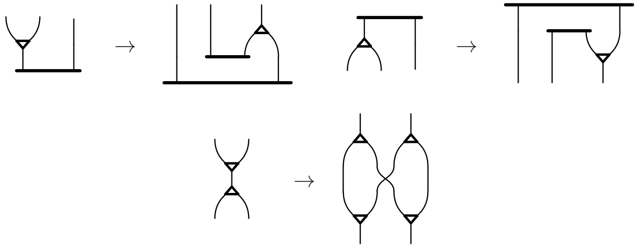
E.g.,  specifies that

$$\begin{array}{c} \Pi'' \Big\| \\ \xi\left\{\dfrac{\mathsf{t}}{a^{\epsilon}\vee\bar{a}}\right\} \\ \Phi \Big\| \\ \zeta\left\{\dfrac{a^{\epsilon}}{\mathsf{t}}\right\} \\ \Psi \Big\| \\ \alpha \end{array} \quad \text{becomes} \quad \begin{array}{c} \Pi'' \Big\| \\ \xi\left[\mathsf{t}\vee\dfrac{\mathsf{f}}{\bar{a}}\right] \\ \Phi\{a^{\epsilon}/\mathsf{t}\} \Big\| \\ \zeta\{\mathsf{t}\} \\ \Psi \Big\| \\ \alpha \end{array}$$

We can operate on flow reductions instead than on derivations:

- much easier,
- we get natural, syntax-independent induction measures.

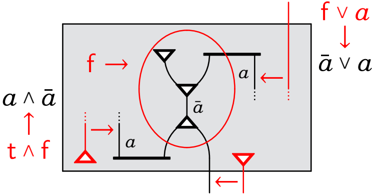
Flow reductions: (co)contraction



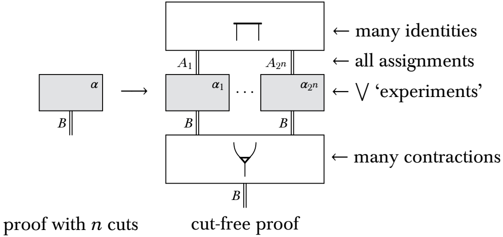
- These reductions conserve the number and length of paths.
- Open problem: does cocontraction yield superpolynomial compression?

Cut elimination by ‘experiments’

Experiment over a proof:



We do:



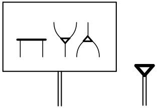
- Simple, exponential cut elimination;
- 2^n experiments, where n is the number of atoms;
- fairly syntax independent method.

The secret of success is in the proof composition mechanism.

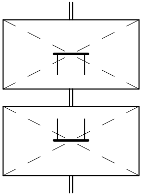
WHY IS THIS IMPOSSIBLE IN THE SEQUENT CALCULUS?

Generalising the cut-free form

- Normalised proof:



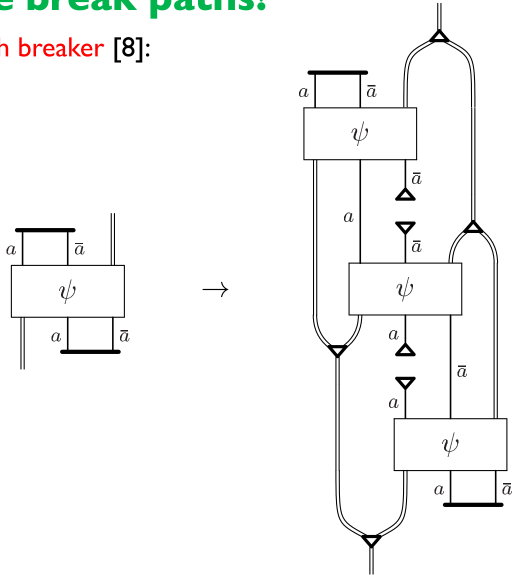
- Normalised derivation:



- The symmetric form is called streamlined.
- Cut elimination is a corollary of streamlining.
- We just need to break the paths between identities and cuts, and (co)weakenings do the rest.

How do we break paths?

With the path breaker [8]:



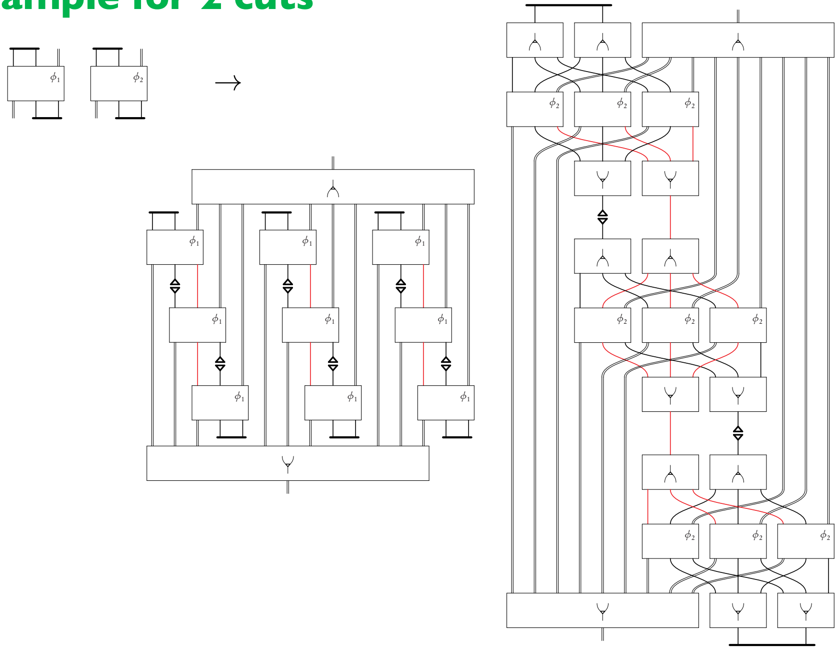
Even if there is a path between identity and cut on the left, there is none on the right.

We can do the same on derivations, of course

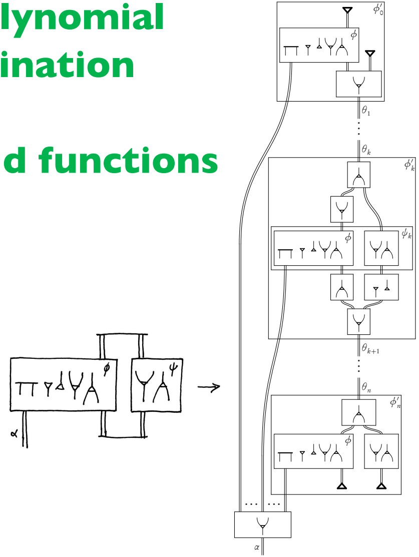
$$\begin{array}{c} A \\ \hline [a \vee \bar{a}] \wedge A \\ {\scriptstyle \Psi}\Vert \\ B \vee (a \wedge \bar{a}) \\ \hline B \end{array} \qquad \longrightarrow \qquad \begin{array}{c} A \\ \Vert{\scriptstyle \{\mathsf{c}\uparrow,\mathsf{ai}\downarrow,=\}} \\ (([a \vee \bar{a}] \wedge A) \wedge A) \wedge A \\ {\scriptstyle (\Psi \wedge A) \wedge A}\Vert \\ ([B \vee (a \wedge \bar{a})] \wedge A) \wedge A \\ {\scriptstyle \Phi_a \wedge A}\Vert \\ [B \vee ([a \vee \bar{a}] \wedge A)] \wedge A \\ {\scriptstyle [B \vee \Psi] \wedge A}\Vert \\ B \vee ([B \vee (a \wedge \bar{a})] \wedge A) \\ {\scriptstyle B \vee \Phi_{\bar{a}}}\Vert \\ B \vee [B \vee ([a \vee \bar{a}] \wedge A)] \\ {\scriptstyle B \vee [B \vee \Psi]}\Vert \\ B \vee [B \vee [B \vee (a \wedge \bar{a})]] \\ \Vert{\scriptstyle \{\mathsf{c}\downarrow,\mathsf{ai}\uparrow,=\}} \\ B \end{array}$$

- We can compose this as many times as there are paths between identities and cut.
- We obtain a family of normalisers that only depends on *n*.
- The construction is exponential.
- Finding something like this is unthinkable without flows.

Example for 2 cuts

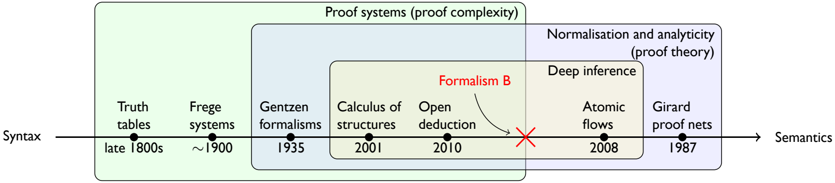


Quasipolynomial cut elimination by threshold functions



- Only $n + 1$ copies of the proof are stitched together.
- Note local cocontraction (= better sharing, not available in Gentzen).

‘Formalism B’: allowing substitution



Proof systems (proof complexity)

Normalisation and analyticity (proof theory)

Deep inference

Formalism B

Syntax

Truth tables
late 1800s

Frege systems
∼1900

Gentzen formalisms
1935

Calculus of structures
2001

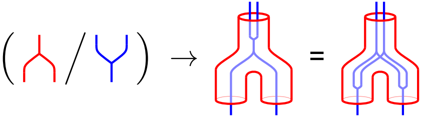
Open deduction
2010

Atomic flows
2008

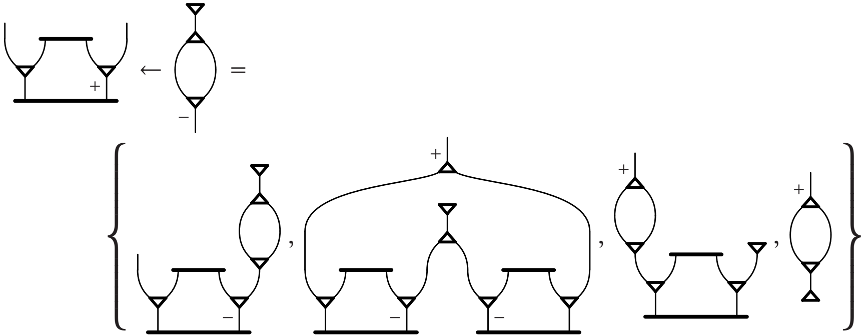
Girard proof nets
1987

Semantics

Achieving the power of Frege + substitution (possibly optimal proof system) by incorporating substitution, guided by the geometry of flows:



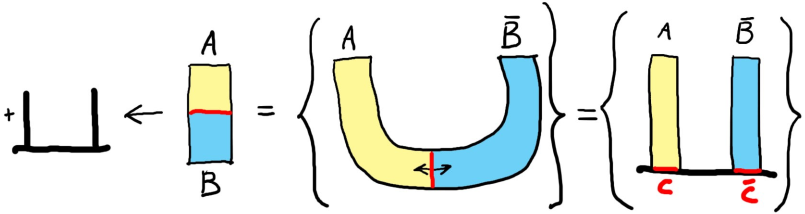
Example of flow substitution



The flows represent proofs. The bigger the set on the right, the more bureaucracy is captured by the substitution, the smaller the set of canonical proofs is.

Note the variety of shapes, all of which are equivalent. This is far more flexible than permutation of rules and similar Gentzen mechanisms.

Gundersen's substitution trick



Substituting a proof ϕ inside an identity or cut stands for a set of proofs with as many elements as ways to break ϕ.

This iterated mechanism alone generates one canonical form for an exponentially big class of proofs.

Lifting flow substitutions to proofs

Consider the following two synchronal open deduction derivations:

$$\phi = {\scriptstyle =}\,\dfrac{{\scriptstyle \mathsf{i}\downarrow}\dfrac{\mathsf{t}}{{\scriptstyle \mathsf{c}\uparrow}\dfrac{a}{a \wedge a} \vee \bar{a}} \vee \bar{a}}{(a \wedge a) \vee {\scriptstyle \mathsf{c}\downarrow}\dfrac{\bar{a} \vee \bar{a}}{{\scriptstyle \mathsf{w}\uparrow}\dfrac{\bar{a}}{\mathsf{t}}}} \qquad \text{and} \qquad \psi = \dfrac{b \vee \dfrac{\mathsf{f}}{b}}{b} \quad .$$

We want to define a denotation for the formal substitution $\phi \,|\, a \leftarrow \psi$. One element in the set of denotations of $\phi \,|\, a \leftarrow \psi$ is

$$ {\scriptstyle =}\,\dfrac{{\scriptstyle \mathsf{i}\downarrow}\dfrac{\mathsf{t}}{{\scriptstyle \mathsf{c}\uparrow}\dfrac{b \vee \mathsf{f}}{\left[b \vee \dfrac{\mathsf{f}}{b}\right] \wedge [b \vee \mathsf{f}]} \vee (\bar{b} \wedge \mathsf{t})} \vee \dfrac{\bar{b}}{\bar{b} \wedge \bar{b}}}{\left(\dfrac{b \vee b}{b} \wedge \dfrac{b \vee \dfrac{\mathsf{f}}{b}}{b}\right) \vee {\scriptstyle \mathsf{c}\downarrow}\dfrac{(\bar{b} \wedge \mathsf{t}) \vee \left(\bar{b} \wedge \dfrac{\bar{b}}{\mathsf{t}}\right)}{{\scriptstyle \mathsf{w}\uparrow}\dfrac{\bar{b} \wedge \mathsf{t}}{\mathsf{t}}}} \quad .$$

Conclusions

- We are interested in proof composition (so in the first and second order propositional proof theory).
- Composition in Gentzen is rigid (it was designed for consistency proofs, not much else).
- Deep inference composition is free and yields local proof systems.
- Locality = linearity + atomicity, so we are doing an extreme form of linear logic.
- Because of locality we obtain a sort of geometric control over proofs.
- So we obtain an efficient and natural formalism for proofs, where more proof theory can be done with lower complexity.
- We are obtaining interesting notions of proof semantics.

This talk is available at `http://cs.bath.ac.uk/ag/t/PCMTGI.pdf`
Deep inference web site: `http://alessio.guglielmi.name/res/cos/`

References

[1] K. Brünnler.
Deep Inference and Symmetry in Classical Proofs.
Logos Verlag, Berlin, 2004.
`http://www.iam.unibe.ch/~kai/Papers/phd.pdf`.

[2] K. Brünnler and A. F. Tiu.
A local system for classical logic.
In R. Nieuwenhuis and A. Voronkov, editors, *Logic for Programming, Artificial Intelligence, and Reasoning (LPAR)*, volume 2250 of *Lecture Notes in Computer Science*, pages 347–361. Springer-Verlag, 2001.
`http://www.iam.unibe.ch/~kai/Papers/lcl-lpar.pdf`.

[3] P. Bruscoli and A. Guglielmi.
On the proof complexity of deep inference.
ACM Transactions on Computational Logic, 10(2):14:1–34, 2009.
`http://cs.bath.ac.uk/ag/p/PrComplDI.pdf`.

[4] P. Bruscoli, A. Guglielmi, T. Gundersen, and M. Parigot.
A quasipolynomial cut-elimination procedure in deep inference via atomic flows and threshold formulae.
In E. M. Clarke and A. Voronkov, editors, *Logic for Programming, Artificial Intelligence, and Reasoning (LPAR-16)*, volume 6355 of *Lecture Notes in Computer Science*, pages 136–153. Springer-Verlag, 2010.
`http://cs.bath.ac.uk/ag/p/QPNDI.pdf`.

[5] A. Das.
On the proof complexity of cut-free bounded deep inference.
In K. Brünnler and G. Metcalfe, editors, *Tableaux 2011*, volume 6793 of *Lecture Notes in Artificial Intelligence*, pages 134–148. Springer-Verlag, 2011.
`http://www.anupamdas.com/items/PrCompII/ProofComplexityBoundedDI.pdf`.

[6] A. Guglielmi and T. Gundersen.
Normalisation control in deep inference via atomic flows.
Logical Methods in Computer Science, 4(1):9:1–36, 2008.
`http://www.lmcs-online.org/ojs/viewarticle.php?id=341`.

[7] A. Guglielmi, T. Gundersen, and M. Parigot.
A proof calculus which reduces syntactic bureaucracy.
In C. Lynch, editor, *21st International Conference on Rewriting Techniques and Applications*, volume 6 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 135–150. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2010.
`http://drops.dagstuhl.de/opus/volltexte/2010/2649`.

[8] A. Guglielmi, T. Gundersen, and L. Straßburger.
Breaking paths in atomic flows for classical logic.
In J.-P. Jouannaud, editor, *25th Annual IEEE Symposium on Logic in Computer Science (LICS)*, pages 284–293. IEEE, 2010.
`http://www.lix.polytechnique.fr/~lutz/papers/AFII.pdf`.

[9] T. Gundersen, W. Heijltjes, and M. Parigot.
Atomic lambda calculus: A typed lambda-calculus with explicit sharing.
In *28th Annual IEEE Symposium on Logic in Computer Science (LICS)*. IEEE, 2013.
In press.

[10] E. Jeřábek.
Proof complexity of the cut-free calculus of structures.
Journal of Logic and Computation, 19(2):323–339, 2009.
`http://www.math.cas.cz/~jerabek/papers/cos.pdf`.