Normalisation with Atomic Flows

Alessio Guglielmi

University of Bath and LORIA & INRIA Nancy-Grand Est

Joint work with Paola Bruscoli, Tom Gundersen and Michel Parigot
January 2010
This talk is available at http://cs.bath.ac.uk/ag/t/NAF.pdf
Outline

Deep Inference
  Propositional Logic and System SKS
  Examples

Goal of This Talk

The Big Picture

Atomic Flows
  Examples
  Flow Reductions

Normalisation
  Overview
  Cut Elimination: Experiments
  Streamlining: Generalised Cut Elimination
  Streamlining: Removal of Simple Edges
  Streamlining: The Path Breaker
  Quasipolynomial Cut Elimination

Conjectures

Conclusion
(Proof) System SKS
[Brünnler & Tiu(2001)]

- **Atomic** rules:

- **Linear** rules:

- Plus an ‘=’ linear rule (associativity, commutativity, units).
- Rules are applied anywhere inside formulae.
- Negation on atoms only.
- Cut is atomic.
- SKS is **complete** and implicationally complete for propositional logic.
Example 1

- In the calculus of structures (CoS):

- In ‘Formalism A’:

Top-down symmetry: so inference steps can be made atomic (the medial rule, m, is impossible in the sequent calculus).
Example 2

\[
\begin{align*}
\begin{array}{c}
\text{Example 2} \\
\text{▶ In CoS:} \\
\text{▶ In ‘Formalism A’:}
\end{array}
\end{align*}
\]

\[
\begin{align*}
\frac{t}{a \lor \bar{a}} \\
\frac{t}{a \lor \bar{a}} \\
\frac{[a \lor t] \land [t \lor \bar{a}]}{[a \lor t] \land [t \lor \bar{a}]} \\
\frac{[a \lor t] \land [\bar{a} \lor t]}{[a \lor t] \land [\bar{a} \lor t]} \\
\frac{((a \lor t) \land \bar{a}) \lor t}{((a \lor t) \land \bar{a}) \lor t} \\
\frac{(\bar{a} \land [a \lor t]) \lor t}{(\bar{a} \land [a \lor t]) \lor t} \\
\frac{[(a \land \bar{a}) \lor t] \lor t}{[(a \land \bar{a}) \lor t] \lor t} \\
\frac{(a \land \bar{a}) \lor t}{(a \land \bar{a}) \lor t} \\
\frac{f \lor t}{f \lor t} \\
\frac{t}{t}
\end{align*}
\]
Locality

- Deep inference allows locality,
- \( i.e., \) inference steps can be checked in constant time (so, inference steps are small).

Example, atomic cocontraction:

\[
\frac{a}{a \land a} \quad \frac{b}{b \land b} \quad m \quad \frac{[a \lor b] \land [a \lor b]}{a \land a}
\]

Note: the sequent calculus
- does not allow locality in contraction (counterexample in \([\text{Brünnler}(2004)]\)), and
- does not allow local reduction of cut into atomic form.
Goal of This Talk

To illustrate the slogans:

- **Deep inference** = locality (+ symmetry).
- **Locality** = linearity + atomicity.
- **Geometry** = syntax independence (elimination of bureaucracy).
- Locality → geometry → semantics of proofs (Lamarche *dixit*).

This is a path towards solving the problem of **proof identity**, *i.e.*, determining when two proofs are the same (Hilbert’s ‘24th problem’).

To show that:

- We can normalise in a somewhat **syntax-independent** way.
- Normalisation is a very **robust** phenomenon.
- Perhaps traditional proof theory is prejudiced on analyticity and complexity: analyticity is much cheaper than exponential!
What Do We Need to Solve the Proof Identity Problem?

A finer representation of proofs, achieving locality.

This yields:

- more proofs to choose representatives from, and especially
- bureaucracy-free proofs;
- nice geometric models [Guiraud(2006)];
- smaller proofs, but
- not as small as proof nets [Lamarche & Straßburger(2005)];
- more manipulation possibilities, viz., for normalisation (focus of this talk, and where we got surprises).
Elimination of Bureaucracy

- Propositional logic.
- **Proof system** $\approx$ proofs can be checked in polytime.
- Normalisation $=$ mainly, but not only!, cut elimination.
- Objective: **eliminate bureaucracy**, *i.e.*, find ‘something’ at the boundary.
Deep inference has as small proofs as the best proof systems and it has a normalisation theory and its analytic proof systems are more powerful than Gentzen ones and cut elimination is quasipolynomial (instead of exponential). (See [Jeřábek(2009), Bruscoli & Guglielmi(2009), Bruscoli et al.(2009)Bruscoli, Guglielmi, Gundersen, & Parigot]).
(Atomic) Flows

Below derivations, their (atomic) flows are shown.

- Only **structural** information is retained in flows.
- Logical information is **lost**.
- Flow size is **polynomially related** to derivation size.
Flow Reductions: (Co)Weakening (1)

Consider these flow reductions:

\[
\begin{align*}
\text{aw} \downarrow - \text{ac} \downarrow : & \quad 1 \rightarrow 1, 2 \\
\text{aw} \downarrow - \text{ai} \uparrow : & \quad 1 \rightarrow 1 \\
\text{aw} \downarrow - \text{aw} \uparrow : & \quad \rightarrow \\
\text{ac} \uparrow - \text{aw} \uparrow : & \quad 1, 2 \rightarrow 1, 2 \\
\text{ai} \downarrow - \text{aw} \uparrow : & \quad 1 \rightarrow 1 \\
\text{ac} \downarrow - \text{aw} \uparrow : & \quad 1, 2 \rightarrow 1, 2
\end{align*}
\]

Each of them corresponds to a correct derivation reduction.
Flow Reductions: (Co)Weakening (2)

For example, \( \text{ai}\downarrow\text{aw}\uparrow : \quad \begin{array}{c}
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
\_ \\
Relation With Interaction Combinators?

Lots of coincidences, but also differences: no apparent logical meaning for two ‘contractions’:

\[
\gamma \delta \quad \Rightarrow \quad \delta \delta \quad \gamma \varepsilon \quad \Rightarrow \quad \varepsilon \varepsilon \quad \delta \varepsilon \quad \Rightarrow \quad \varepsilon \varepsilon
\]
Flow Reductions: (Co)Contraction

Consider these flow reductions:

- $c \downarrow - i \uparrow$: 
  \[ \begin{array}{c}
  \text{(a)} \\
  1 \quad 2 \\
  \quad \quad \\
  3 \\
  \end{array} \quad \rightarrow \quad \begin{array}{c}
  \text{(b)} \\
  1 \\
  \quad \quad \\
  2 \quad 3 \\
  \end{array} \quad \begin{array}{c}
  \text{(c)} \\
  1 \quad 2 \\
  \quad \quad \\
  3 \\
  \end{array} \]

- $i \downarrow - c \uparrow$: 
  \[ \begin{array}{c}
  \text{(a)} \\
  1 \\
  2 \quad 3 \\
  \end{array} \quad \rightarrow \quad \begin{array}{c}
  \text{(b)} \\
  1 \\
  \quad \quad \\
  2 \\
  \end{array} \quad \begin{array}{c}
  \text{(c)} \\
  1 \\
  \quad \quad \\
  2 \\
  3 \\
  \end{array} \]

- $c \downarrow - c \uparrow$: 
  \[ \begin{array}{c}
  \text{(a)} \\
  1 \quad 2 \\
  \quad \quad \\
  3 \quad 4 \\
  \end{array} \quad \rightarrow \quad \begin{array}{c}
  \text{(b)} \\
  3 \quad 4 \\
  \quad \quad \\
  1 \quad 2 \\
  \end{array} \]

- They conserve the **number and length of paths**.
- Note that they can blow up a derivation **exponentially**.
- It’s a good thing: cocontraction is a **new compression mechanism** (sharing?).
- Open problem: **does cocontraction provide exponential compression?** Conjecture: yes.
Normalisation Overview

- None of these methods existed before atomic flows, none of them requires permutations or other syntactic devices.
- Quasipolynomial procedures are surprising.

(1) [Guglielmi & Gundersen(2008)]; (2) LICS 2010 submission; (3) [Bruscoli et al.(2009)Bruscoli, Guglielmi, Gundersen, & Parigot].
Cut Elimination (on Proofs) by ‘Experiments’

We do:

Simple, exponential cut elimination; proof generates $2^n$ experiments.
Generalising the Cut-Free Form

- Normalised proof:

- Normalised derivation:

- The symmetric form is called streamlined.
- Cut elimination is a corollary of streamlining.
Removal of a ‘Simple Edge’

Remove identity and cut:

We can do so on simple edges, like 1 above.
The procedure requires a strategy, not to loop.
The chunks to be copied can be small.
Open: computational interpretation?

Definition 5.1. We define the reduction $\to_{\text{se}}$ (where se stands for simple edge) as follows, for every atomic flow $A$: 

$\begin{array}{c} 
\epsilon_1 \cdots \epsilon_h \quad 2 \\
\epsilon'_1 \cdots \epsilon'_k \quad 3 \\
\end{array} \to_{\text{se}} 
\begin{array}{c} 
\hat{\epsilon}_1 \cdots \hat{\epsilon}_h \\
\hat{\epsilon}'_1 \cdots \hat{\epsilon}'_k \\
\end{array}$

where $h, k \geq 0$, edges have been renamed with $\hat{\text{and}} \, \tilde{\text{ accents}}$, flows $\tilde{A}$ and $\hat{A}$ are both isomorphic to $A$, and edges $\hat{\epsilon}_2$ and $\tilde{\epsilon}_3$ are identified.

A simple inspection of the definition of $\to_{\text{se}}$ suffices to prove the following statement, about $\to_{\text{se}}$ not introducing any $ai$-cycles.

Proposition 5.2. If atomic flow $B$ is cycle-free and $B \to_{\text{se}} C$, then $C$ is cycle-free.

Theorem 5.3. Reduction $\to_{\text{se}}$ is sound.

Proof. Let $\Phi$ be a derivation with flow $B$, such that $B \to_{\text{se}} C$. We show that there exists a derivation $\Psi$ with flow $C$ and with the same premiss and conclusion as $\Phi$. In the following, we refer to the figure in Definition 5.1. We assume that $\Phi$ has premiss $\xi\{t\cdot\}$ and conclusion $\zeta\{f\cdot\}$, where the evidenced and labelled $t\cdot$ and $f\cdot$ can be traced to the interaction and cointeraction vertices eliminated by $\to_{\text{se}}$, respectively (this can always be done by using switches and unit equations). Intuitively, we can think of $t\cdot$ and $f\cdot$ as mapping to special ‘unit edges’, which can be substituted just like normal edges. So, we assume that $\Phi$ is $\xi\{t\cdot\} \Phi_1 \parallel \parallel \xi'\{t\cdot\} \downarrow \xi'\{\bar{a}_2 \lor a_1\} \Phi_2 \parallel \parallel \zeta'\{\bar{a}_3 \land a_1\} \uparrow \zeta'\{f\cdot\} \Phi_3 \parallel \parallel \zeta\{f\cdot\}$. We can do so on simple edges, like 1 above.

The procedure requires a strategy, not to loop.

The chunks to be copied can be small.

Open: computational interpretation?
Composition of Simple Edge Removal

Figure 5: Example of a two-step $\rightarrow_{bc}$ (or $\rightarrow_{ex}$) reduction.

where $h, k \geq 0$, and let $B' = \tilde{\epsilon}_1 \cdots \tilde{\epsilon}_h \tilde{A} \tilde{\epsilon}_1' \cdots \tilde{\epsilon}_k'$ and $B'' = \hat{\epsilon}_1 \cdots \hat{\epsilon}_h \hat{A} \hat{\epsilon}_1' \cdots \hat{\epsilon}_k'$, where the correspondence of edges has been indicated by adding accents to their labels. We have that:

• if $1$ is an edge belonging to an $a_i$-cycle, $B' \rightarrow_{bc} D'$ and $B'' \rightarrow_{bc} D''$ then $B \rightarrow_{bc} C$;

• if $1$ is an extremal simple edge, $B' \rightarrow_{ex} D'$ and $B'' \rightarrow_{ex} D''$ then $B \rightarrow_{ex} C$.

Example 5.7. Consider the atomic flow to the left in Figure 5. Assuming that the two evidenced simple edges both belong to $a_i$-cycles and that the box $A$ stands for a cycle-free flow, then the atomic flow on the right is the result of a $\rightarrow_{bc}$ reduction. Similarly, if the two evidenced simple edges are extremal simple edges, and the box stands for a flow that contains no simple edges, then the atomic flow on the right is the result of a $\rightarrow_{ex}$ reduction.

Notice that the flow in Figure 5 represents the 'external' shape of any flow after eliminating any two simple edges. Eliminating more simple edges would follow the same pattern.

Remark 5.8. It is possible to generalise the construction in Figure 5 to any number $n$ of simple edges: for any $n$, there is an atomic flow of the same nature as the one at the right.
How to Obtain a Simple Edge?

- By moving away (co)contractions by way of their reductions:

  \[
  \begin{align*}
  &c↓-i↑: \quad \begin{array}{c}
  1 \quad 2 \\
  \downarrow \\
  3
  \end{array} \quad \rightarrow \quad \begin{array}{c}
  1 \\
  \quad 2 \\
  \quad 3
  \end{array} \\
  &i↓-c↑: \quad \begin{array}{c}
  1 \\
  \quad 2 \\
  \quad 3
  \end{array} \quad \rightarrow \quad \begin{array}{c}
  1 \\
  \quad 2 \\
  \quad 3
  \end{array}
  \end{align*}
  \]

  \[
  \begin{align*}
  &c↓-c↑: \quad \begin{array}{c}
  1 \quad 2 \\
  \quad 3 \\
  \quad 4
  \end{array} \quad \rightarrow \quad \begin{array}{c}
  1 \\
  \quad 2 \\
  \quad 3 \\
  \quad 4
  \end{array}
  \end{align*}
  \]

- But beware of loops:

  \[
  \begin{align*}
  &+ \quad + \quad \rightarrow_c \quad + \quad \rightarrow_c \quad + \quad \rightarrow_c \quad \cdots
  \end{align*}
  \]

- This and more is in [Guglielmi & Gundersen(2008)].
How Do We Break Paths Without ‘Preprocessing’?

With the path breaker (Lutz Straßburger contributed here):

Even if there is a path between identity and cut on the left, there is none on the right.
We Can Do This on Derivations, of Course

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.

Note: finding something like this is unthinkable without flows.
Example for $n = 2$

Example 4.20. Given a derivation $\Phi$ where the atoms $a$ and $b$ occur, such that the atomic flow associated with $\Phi$ is $\phi_1 \phi_2 \psi$, where $\phi_1$ is the atomic flow associated with $a$, $\phi_2$ is the atomic flow associated with $b$ and $a$ and $b$ are the only non-weakly-streamlined atoms in $\Phi$, then the atomic flow associated with $\text{Norm}_2(a, b, \text{Core}(\Phi))$ is $\phi_1 \phi_1 \phi_1 \phi_1 \phi_1 \phi_1 \phi_1 \phi_1$. 

NORMALISATION CONTROL IN DEEP INFERENCE VIA ATOMIC FLOWS II
Quasipolynomial Cut Elimination by Threshold Functions

Only $n + 1$ copies of the proof are stitched together. It’s complicated, but note local cocontraction (= better sharing, not available in Gentzen).
Handwaving Explanation of Threshold Functions

- $\theta_i = \text{there are at least } i \text{ atoms that are true (out of given } n)$. 
- For example, for $n = 2$, we have $\theta_1 = a \lor b$ and $\theta_2 = a \land b$. 
- Each $\theta_i$ can be kind of projected into each atom to provide its pseudocomplement, for example the pseudocomplement of $a$ in $\theta_1$ is $b$. 
- The atom and the pseudocomplement fit into the scheme of the previous slide, and you can get, for example, $\theta_2$ from $\theta_1$. 
- Stitch derivations together until you get $\theta_{n+1} = f$. 
- The complexity is dominated by the complexity of the $\theta$'s, which is $n^{\O\left(\log n\right)}$. 

The difficulty is in defining the $\theta$'s and in finding proofs that stitch them together (this theory comes from circuit complexity and it had been applied to the monotone sequent calculus, which is weaker than propositional logic).
Conjecture 1

We can normalise in polynomial time, because:

▶ polynomial threshold function representations exist;
▶ deep inference is flexible.
We think that (*) might make for a proof system (see also recent work by Straßburger).

This means that there should exist a polynomial algorithm to check the correctness of (*).

If this is true, we have an excellent bureaucracy-free formalism.

Note: if such a thing existed for proof nets, then \( \text{coNP} = \text{NP} \).
Conclusion

- Normalisation does not depend on logical rules.
- It only depends on structural information, *i.e.*, geometry.
- Normalisation is extremely robust.
- Deep inference’s locality is key.
- Complexity-wise, deep inference is as powerful as the best formalisms,
- and more powerful if analiticity is requested.
- Deep inference is the continuation of Girard politics with other means.

In my opinion, much of the future of structural proof theory is in ‘geometric methods’.

This talk is available at [http://cs.bath.ac.uk/ag/t/NAF.pdf](http://cs.bath.ac.uk/ag/t/NAF.pdf)
*Deep Inference and Symmetry in Classical Proofs.*
Berlin: Logos Verlag.
http://www.iam.unibe.ch/~kai/Papers/phd.pdf.

A local system for classical logic.
Springer-Verlag.

On the proof complexity of deep inference.
*ACM Transactions on Computational Logic, 10*(2), 1–34.

Quasipolynomial normalisation in deep inference via atomic flows and threshold formulae.

Normalisation control in deep inference via atomic flows.

The three dimensions of proofs.

Proof complexity of the cut-free calculus of structures.

Naming proofs in classical propositional logic.
In P. Urzyczyn (Ed.) *Typed Lambda Calculi and Applications*, vol. 3461 of *Lecture Notes in Computer Science*, (pp. 246–261).
Springer-Verlag.