FORMALISM A

Alessio Guglielmi (TU Dresden)
23.4.2004

My current view of formalisms for deep inference is that we can isolate three of them: one is the (now well developed) calculus of structures, and two others are possible, A and B (I currently have no better name for them). In A there's a more general notion of derivation than in CoS, in B there's this, plus a more general notion of inference rule.

In moving from CoS to A and B one progressively abstracts away from unnecessary details, like having to choose an arbitrary order for permutable inference rules. B is the formalism with less bureaucracy, A is intermediate. On the other hand, CoS is easily definable and its derivations are easily written on paper; instead, I currently have no viable syntax for B, and the one for A is a bit challenging (perhaps I just have to develop some good macros). It is likely that B corresponds to some notion of proof net.

The three formalisms are of course mutually compatible, meaning that every property observed in one can be observed in the other by reasonable transformations on derivations. What changes is how convenient it is to observe and define the property of interest.

In this note I will define A, the formal definitions are at the end on this document. (I posted all what exists about B to the Frogs mailing list on 9.2.04.)

What is Deep Inference?

So far, we avoided giving a general definition for CoS, because we wanted to gain experience first. I'm very glad we did so, because the definition I propose in the attachment is rather different than (although compatible to) what we did so far.

The first crucial decision is about equations. Using equations, and how much to use them, is always debatable. Unfortunately, there is tension between

no equations = good for implementing

and

as many equations as semantics ask for = good for doing proof theory .

The only reasonable choice here is not to choose, and just providing a place where equations can be put under control.

That said, in the definition of derivation that I propose, there are more equations than ever, although in practical cases these can be reduced to one simple associativity law. Nonetheless, the definition is more delicate than the one we use for CoS.

The basic idea is to treat derivations the same way we treat structures, meaning that derivations can be composed by the same operators: *this* is deep inference, it's the pure essence of it.

Permutations in CoS

Arguably, there is one weak spot in CoS: the abundance of permutations. The problem stems from the way we use inference rules: we can only chain them in a sequence, and this means that at each step we have to copy the context. We can fix this, although we all know that it is, morally, just a `cosmetic´ problem.

Example: In CoS we can write the derivations

```
S[(a,b),(a,b)]          S[(a,b),(a,b)]
--------------          --------------
S([a,a],[b,b])          S([a,a],[b,b])
--------------          --------------
S([a,a], b   )          S( a    ,[b,b])
--------------   and    -------------- .
S( a    , b   )         S( a    , b   )
```

With the new notion we can write

```
  {  [(a,b),(a,b)]  }
  {  -------------  }
  {        |        }
  { --------------- }
S{ ( [a,a] [b,b] ) } ,
  { ( ----- ----- ) }
  { (   |  ,  |   ) }
  { (   -     -   ) }
  { (   a     b   ) }
```

where the two contractions live in parallel. Proofs are not sequences anymore, they become (generalised) series-parallel orders.

Shallow Rules?

Given what I said above, one natural question is to ask what can we do when we need the occasional shallow rule. Well, it's actually possible to squeeze shallow rules into the definitions I provide, although they quite go against the philosophy of the whole thing.

In that case, resorting to plain CoS might be a good idea, since in the rare circumstances when one deals with shallow rules, the good properties brought forward by formalism A are usually not the focus of attention.

Web Site

http://alessio.guglielmi.name/res/cos.

9 February 2004

Deep Inference - Formalism 'A'

Alessio Guglielmi
Technische Universität Dresden
Fakultät Informatik - 01062 Dresden - Germany
Alessio.Guglielmi@Inf.TU-Dresden.DE

1 Definitions

Formulae are freely built from units and atoms by modalities and binary connectives. We do not worry about negation at this stage. We use the Polish notation for operators, to save on parentheses.

1.1 Definition We have the following mutually disjoint sets of symbols:

1 a set $\mathscr{U}$ of *units*, denoted by u;

2 a set $\mathscr{A}$ of *atoms*, denoted by α;

3 a set $\mathscr{M}$ of *modalities*, denoted by μ;

4 a set $\mathscr{C}$ of *binary connectives*, denoted by γ;

each of these sets may be finite (and perhaps empty) or infinite; the set of *formulae* $\mathscr{F}(\mathscr{U},\mathscr{A},\mathscr{M},\mathscr{C})$, denoted simply by $\mathscr{F}$ when no confusion is possible, is defined as

$$\mathscr{F} ::= \mathscr{U} \mid \mathscr{A} \mid \mathscr{M}\mathscr{F} \mid \mathscr{C}\mathscr{F}\mathscr{F} \quad ;$$

formulae are denoted by F and G.

• The following definition is currently not used in the rest of the paper.

1.2 Definition The set of *formula contexts* $\mathscr{F}^{\mathsf{c}}$ is defined as

$$\mathscr{F}^{\mathsf{c}} ::= \{\ \} \mid \mathscr{M}\mathscr{F}^{\mathsf{c}} \mid \mathscr{C}\mathscr{F}^{\mathsf{c}}\mathscr{F} \mid \mathscr{C}\mathscr{F}\mathscr{F}^{\mathsf{c}} \quad ,$$

where $\{\ \}$ is called a *hole*; formula contexts are denoted by $F\{\ \}$. We write $F\{G\}$ for the formula obtained from the formula context $F\{\ \}$ by filling the hole with the formula G.

1.3 Definition Suppose we are given a set of formulae $\mathscr{F}(\mathscr{U},\mathscr{A},\mathscr{M},\mathscr{C})$ and a decidable equivalence relation $=_{\mathsf{s}}$ on it such that

$$\text{if} \quad F =_{\mathsf{s}} F' \quad \text{then} \quad \mu F =_{\mathsf{s}} \mu F' \quad , \quad \gamma FG =_{\mathsf{s}} \gamma F'G \quad \text{and} \quad \gamma GF =_{\mathsf{s}} \gamma GF' \quad ,$$

for all formulae F, F', G and for any $\mu \in \mathscr{M}$ and $\gamma \in \mathscr{C}$. We say that the equivalence class $[F]_{=_{\mathsf{s}}}$ is a *structure* and we denote the set of structures by $\mathscr{S}(\mathscr{U},\mathscr{A},\mathscr{M},\mathscr{C},=_{\mathsf{s}})$, or by $\mathscr{S}$ when no confusion is possible; structures are denoted by S. Given structures S and S' and any formulae $F \in S$ and $F' \in S'$, we may sometimes denote

$$[\gamma FF']_{=_{\mathsf{s}}} \quad \text{by} \quad [\gamma SS']_{=_{\mathsf{s}}} \quad ;$$

the independence of the structure from the choice of F and F' is guaranteed by the condition above on $=_{\mathsf{s}}$.

We build derivations out of elementary derivations, which correspond to inference rule instances in the standard sequent calculus terminology.

1.4 Definition An *elementary prederivation* of kind ρ for $\mathscr{F}(\mathscr{U},\mathscr{A},\mathscr{M},\mathscr{C})$ is an expression of the form

$$\rho\,\frac{F}{G}\quad ,$$

where formulae F and G are the *premise* and the *conclusion* of the elementary prederivation. We denote by ρ the set of all prederivations of kind ρ and we require that such set be decidable; usually, an inference rule scheme (again denoted by ρ) is all we need for deciding whether an elementary derivation belongs to a certain kind. The set of all given elementary prederivations is denoted by $\mathscr{E}^{\mathsf{P}}$, and it is the union of all the given kinds.

1.5 Definition Given $\mathscr{S}(\mathscr{U},\mathscr{A},\mathscr{M},\mathscr{C},=_{\mathsf{s}})$ and a set of elementary prederivations $\mathscr{E}^{\mathsf{P}}$ for it, we define the set of *elementary derivations* for $\mathscr{S}(\mathscr{U},\mathscr{A},\mathscr{M},\mathscr{C},=_{\mathsf{s}})$ as

$$\mathscr{E}=\left\{\,\rho\,\frac{[F]_{=_{\mathsf{s}}}}{[G]_{=_{\mathsf{s}}}}\;\middle|\;\rho\,\frac{F}{G}\in\mathscr{E}^{\mathsf{P}}\right\}\quad .$$

Given $\rho\,\dfrac{[F]_{=_{\mathsf{s}}}}{[G]_{=_{\mathsf{s}}}}$, the structures $[F]_{=_{\mathsf{s}}}$ and $[G]_{=_{\mathsf{s}}}$ are its *premise* and *conclusion*, respectively.

We build derivations the same way we build structures, but in addition we have a composition operator, which corresponds to plugging them together based on their premises and conclusion.

1.6 Definition We define *prederivations* and their *premises* and *conclusions* as follows; prederivations are denoted by Ψ, the premise and conclusion of Ψ are denoted by $\mathsf{p}\,\Psi$ and $\mathsf{c}\,\Psi$. Given a binary operator $\star$, called *composition*, that does not appear anywhere else, the set of *prederivations* $\mathscr{D}^{\mathsf{P}}$ for $\mathscr{S}(\mathscr{U},\mathscr{A},\mathscr{M},\mathscr{C},=_{\mathsf{s}})$ is defined as the smallest set such that:

1 Every structure is a prederivation: $\mathscr{S}\subseteq\mathscr{D}^{\mathsf{P}}$; for a structure S it holds $\mathsf{p}\,S=\mathsf{c}\,S=S$.

2 Every elementary derivation is a prederivation: $\mathscr{E}\subseteq\mathscr{D}^{\mathsf{P}}$; premise and conclusion are those defined for the elementary derivation.

3 Given two prederivations Ψ and Ψ' such that $\mathsf{p}\,\Psi=S$, $\mathsf{c}\,\Psi=\mathsf{p}\,\Psi'$ and $\mathsf{c}\,\Psi'=S'$, then

$$\star\Psi\Psi'$$

is a prederivation whose premise is S and conclusion is S'.

4 For all $\mu\in\mathscr{M}$ and for any prederivation Ψ,

$$\mu\Psi$$

is a prederivation whose premise and conclusion are

$$[\mu\;\;\mathsf{p}\,\Psi]_{=_{\mathsf{s}}}\qquad\text{and}\qquad[\mu\;\;\mathsf{c}\,\Psi]_{=_{\mathsf{s}}}\quad .$$

5 For any $\gamma\in\mathscr{C}$ and for all prederivations Ψ and Ψ',

$$\gamma\Psi\Psi'$$

is a prederivation whose premise and conclusion are

$$[\gamma\;\;\mathsf{p}\,\Psi\;\;\mathsf{p}\,\Psi']_{=_{\mathsf{s}}}\qquad\text{and}\qquad[\gamma\;\;\mathsf{c}\,\Psi\;\;\mathsf{c}\,\Psi']_{=_{\mathsf{s}}}\quad .$$

We now introduce an equivalence on prederivations which respects structures, respects composition of prederivations and is associative on composition.

1.7 Definition Suppose we have a set of prederivations $\mathscr{D}^{\mathsf{P}}$ for $\mathscr{S}(\mathscr{U},\mathscr{A},\mathscr{M},\mathscr{C},=_{\mathsf{S}})$ and we have a decidable equality relation $=_{\mathsf{D}}$ on $\mathscr{D}^{\mathsf{P}}$ such that, for all prederivations Ψ, Ψ_1 and Ψ_2 and for any $\mu \in \mathscr{M}$ and $\gamma \in \mathscr{C}$:

$$\begin{array}{llll}
\text{if} & \Psi \in \mathscr{S} & \text{then} & [\Psi]_{=_{\mathsf{D}}} = [\Psi]_{=_{\mathsf{S}}} \quad ; \\
\text{if} & \Psi_1 =_{\mathsf{D}} \Psi_2 & \text{then} & \star\,\Psi_1\Psi =_{\mathsf{D}} \star\Psi_2\Psi \quad \text{and} \quad \star\,\Psi\Psi_1 =_{\mathsf{D}} \star\Psi\Psi_2 \quad , \\
& & & \mu\Psi_1 =_{\mathsf{D}} \mu\Psi_2 \quad , \quad \gamma\Psi_1\Psi =_{\mathsf{D}} \gamma\Psi_2\Psi \quad \text{and} \quad \gamma\Psi\Psi_1 =_{\mathsf{D}} \gamma\Psi\Psi_2 \quad ; \\
\multicolumn{4}{l}{\star\,\star\Psi\Psi'\,\Psi'' =_{\mathsf{D}} \star\,\Psi\,\,\star\Psi'\Psi'' \quad .}
\end{array}$$

A *derivation* is an equivalence class $[\Psi]_{=_{\mathsf{D}}}$; we denote by $\mathscr{D}(\mathscr{U},\mathscr{A},\mathscr{M},\mathscr{C},=_{\mathsf{D}})$ the set of derivations, or we just use $\mathscr{D}$ when no confusion is possible; derivations are denoted by Δ.