Department of Computer Science
University of Bath

PHD CONFIRMATION REPORT

BENJAMIN RALPH

A report submitted in partial fulfilment of the requirements for confirmation as a PhD student based on the University Quality Assurance Code of Practice for Research Degrees (QA7, Appendix 5).

This report presents a proposed direction of research for the continuation and eventual completion of my PhD. It is a course of study recommended by my own academic interests, the expertise of staff in the research group, and the formal requirements of a PhD in Computer Science.

Towards a Natural Theory of Quantification for Deep Inference

Much of my first year at Bath has been dedicated to immersing myself in what is, to me, largely a new field: the mathematical foundations of computation. From my undergraduate degree, I gained a perspective on logic very much influenced by the mathematical and philosophical tradition. Despite the obvious difficulties and misunderstandings that arose from this change of field, my academic past hopefully allows me to attack problems and design issues with a varied and broad logical arsenal. Many of the interests from my Master's Degree in Mathematics and Philosophy have carried over into my doctorate: a focus on classical logic (both propositional and first-order), a sensitivity to the relationship between syntax and semantics, and an appreciation and assessment of the motivations behind historical design choices in logic.

The past and future course of my study—developed throughout the year in frequent discussion with my supervisor and other members of the research group—reflects these interests, as well as the academic strengths of the group. In addition, there is much scope for reasonable modification of the aims and expectations of the research over the next two years, while preserving a consistent intellectual direction.

A natural grouping of my research—past, present and future—is into three areas. The first is largely material that has been known for a while by various people working in the area, but has not yet been formally presented. The second can perhaps be considered new work, although it is largely synthetic: bringing together a few strands already extant in recent literature to show an interesting result. The third area will likely represent the creative core of my PhD thesis, and consists of speculative, design-orientated work. As a totality, they represent a continuation of a research programme going back over 15 years, that of deep inference [4, 20].[1] More specifically, I hope to add to the many contributions that deep inference has provided to the structural proof theory of classical logic. As well as carrying out preliminary research in the well-furrowed field of classical propositional logic, I hope to enrich the deep-inference proof theory of predicate logic, which, *pace* the work of Kai Brünnler and Richard McKinley [6, 7, 8, 26], is comparatively unstudied.

Natural and confluent cut elimination for propositional logic. Ever since the proof of Gentzen's *Hauptsatz* for the sequent calculus [16], which proceeds by an induction on both the height and depth of a cut, it has been noticed that there are certain non-convergent critical pairs in the cut elimination procedure. This has long caused angst among proof theorists: the resulting lack of confluence denies classical proof theory many desirable properties that follow from the existence of unique normal forms: a denotational semantics, a Curry-Howard style correspondence and a confluent normalisation procedure for natural deduction [17, 30]. The two most prominent examples of such critical pairs in the classic cut elimination procedure arise from the following proofs [17, 18, 24]:

$$\text{cut}\,\dfrac{\text{weak}\,\dfrac{\begin{array}{c}\Pi_1\\ \Gamma\end{array}}{\Gamma, A} \qquad \text{weak}\,\dfrac{\begin{array}{c}\Pi_2\\ \Delta\end{array}}{\bar{A}, \Delta}}{\Gamma, \Delta} \qquad \text{and} \qquad \text{cut}\,\dfrac{\text{cont}\,\dfrac{\begin{array}{c}\Pi_1\\ \Gamma, A, A\end{array}}{\Gamma, A} \qquad \text{cont}\,\dfrac{\begin{array}{c}\Pi_2\\ \bar{A}, \bar{A}, \Delta\end{array}}{\bar{A}, \Delta}}{\Gamma, \Delta}$$

In each of these cases, the proof of the *Hauptsatz* relies on the fact that one can replace such proofs with another where the cut has either been eliminated, or the induction measure has decreased. For example:

$$\text{cut}\,\dfrac{\text{weak}\,\dfrac{\begin{array}{c}\Pi_1\\ \Gamma\end{array}}{\Gamma, A} \qquad \text{weak}\,\dfrac{\begin{array}{c}\Pi_2\\ \Delta\end{array}}{\bar{A}, \Delta}}{\Gamma, \Delta} \quad \Longrightarrow \quad \text{weak}\,\dfrac{\begin{array}{c}\Pi_1\\ \Gamma\end{array}}{\Gamma, \Delta} \;\Big/\; \text{weak}\,\dfrac{\begin{array}{c}\Pi_2\\ \Delta\end{array}}{\Gamma, \Delta}$$

However, as is shown by the example above, in these situations there seems to be no canonical way to proceed. Thus, it has become the received view that one needs an extra-logical strategy—Girard's *LC* [17], Parigot's $\lambda\mu$-calculus [29] and Coquand's game semantics [14] can be seen as examples of this phenomenon—to achieve confluent cut elimination.

Non-confluence is also a feature of the most wide-ranging cut elimination procedure for deep inference proof systems, *splitting* [6, 19]. Due to the fact that a deep-inference proof is not tree-like (instead there is *top-down symmetry* [19]), the Gentzen method of cut elimination is not going to work in the same way. Instead, we use the fact that the cut rule can be reduced to an *atomic* cut rule, where the eigenformula must be atomic [12]. Non-convergent critical pairs arise when eliminating atomic cuts using the splitting lemma. The lemma guarantees that we can

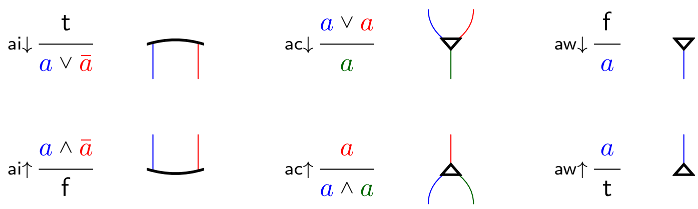
[1] In light of this, I have joined the research project *Efficient and Natural Proof Systems* (EPSRC Project EP/K018868/1).

"split" a proof Π with an atomic cut at the bottom into three:

$$\Pi = C\left\{ \begin{array}{c} \Big\| \text{ai}\!\uparrow \text{free} \\ \text{ai}\!\uparrow \dfrac{a \wedge \bar{a}}{\mathsf{f}} \end{array} \right\} \quad \Longrightarrow \quad \begin{array}{c} \Pi_1 \Big\| \text{ai}\!\uparrow \text{free} \\ [U \vee a] \end{array}, \quad \begin{array}{c} \Pi_2 \Big\| \text{ai}\!\uparrow \text{free} \\ [U \vee \bar{a}] \end{array} \quad \text{and} \quad \begin{array}{c} U \\ \Delta \,\|\, \text{ai}\!\uparrow \text{free} \\ C\{\mathsf{f}\} \end{array}$$

Now, we can create a proof of $\begin{array}{c} \Pi_3 \Big\| \text{ai}\!\uparrow \text{free} \\ [U \vee U] \end{array}$ from Π_1 and Π_2 and then use contraction to compose it with Δ, creating a cut-free proof of $C\{\mathsf{f}\}$. But, crucially, there are two ways to create Π_3: either by plugging Π_1 into Π_2 or vice versa. So the move to deep inference does not immediately solve our problems.

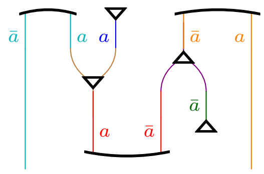
Nevertheless, we can make progress by taking inspiration from the fact that in deep inference formalisms, in particular *open deduction* [20], there are more natural ways to compose derivations than in Gentzen systems, one of the key features of the methodology. To be precise: given two derivations $\begin{array}{c} A \\ \Phi \| \\ B \end{array}, \begin{array}{c} C \\ \Psi \| \\ D \end{array}$, we can compose vertically, with an inference rule $\sigma \dfrac{B}{C}$, to form $\begin{array}{c} A \\ \| \\ D \end{array}$; or horizontally, with a binary logical relation $\star$, to form $\begin{array}{c} A \star C \\ \| \\ B \star D \end{array}$ (and similarly for n-ary logical relations and negation of derivations). This characteristic is exploited in the use of a geometric invariant of derivations, the *atomic flow*, a graph where the edges trace the atoms in a derivation and the nodes are structural rules [20, 21]:



Thus, only the structural information about the derivation is conserved; the logical and equality rules are not represented.

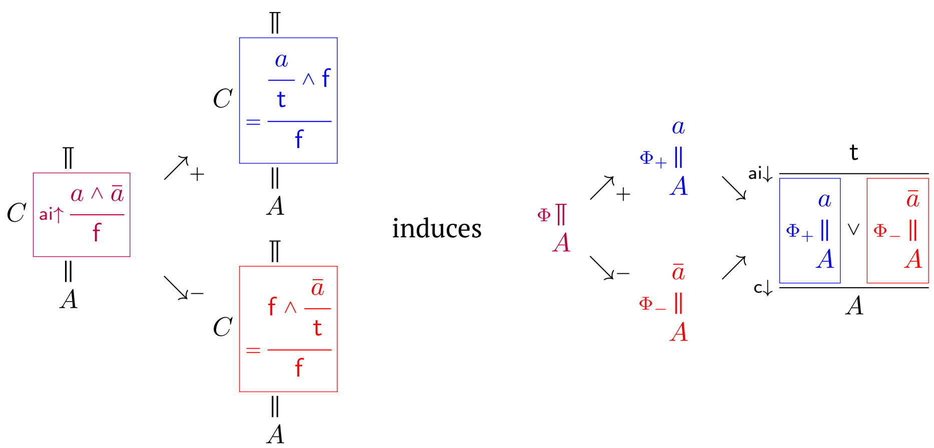
Atomic flow composition naturally corresponds to derivation composition in open deduction, and reasoning about various aspects of a derivation can be carried out at the level of the atomic flow: the most significant work is in normalisation [20, 21] and complexity [15]. Furthermore, the atomic flow gives us a simple, almost trivial, new way at looking at identity between atoms, and occurrence: by using the connected components of the atomic flow as the basis for an equivalence relation on atoms. This notion of occurrence is truly proof-theoretic, rather than an adapted formula-theoretic concept.

$$= \dfrac{\dfrac{\dfrac{\mathsf{t}}{\bar a \vee a} \vee \dfrac{\mathsf{f}}{a}}{\bar a \vee \dfrac{a \vee a}{a}} \wedge \dfrac{\dfrac{\mathsf{t}}{\bar a}}{\bar a \wedge \dfrac{\bar a}{\mathsf{t}} \vee a}}{\dfrac{[\bar a \vee a] \wedge [\bar a \vee a]}{\bar a \vee \dfrac{a \wedge \bar a}{\mathsf{f}} \vee a}}$$

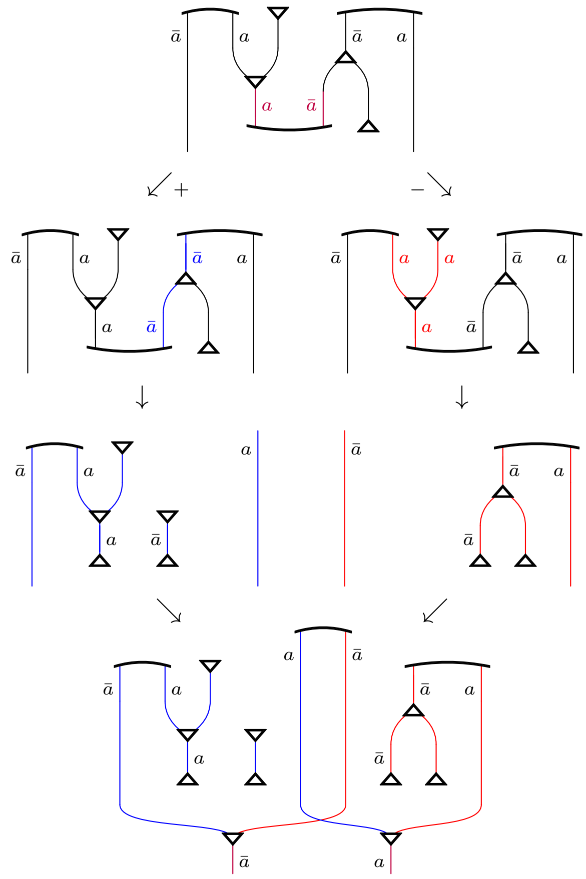


With this more geometric and compositional mindset, it is possible to circumvent the problems discussed above and construct a natural and confluent cut elimination procedure for classical propositional logic, which has been called the "experiments" method. It involves a simple and semantically motivated transformation of a proof into a number of derivations (exponential in the number of connected components of the flow), which are then disjunctively composed.

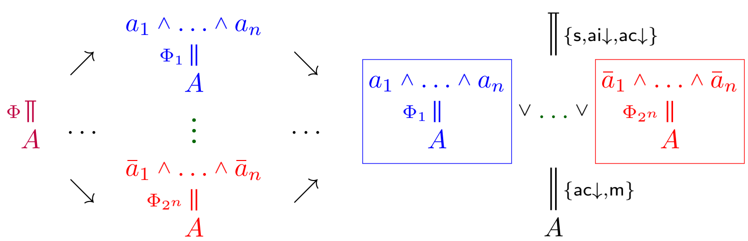
The intuition stems from the simple idea that each atom is either true or false. By tracing up the atoms up from a cut using the atomic flow, we create two derivations: one from the positive atom in the true case, and one from the negative atom in the false case. Unlike in the sequent calculus, we can disjunctively compose these derivations, adding an identity at the top and a contraction at the bottom to recover a proof:



To better see the dynamics of the procedure, it is sensible to look at the level of the atomic flow, rather than the derivation. Below is an example:



The above procedure is only sufficient for when there are cuts in one connected component of the proof. Thus, if there are $n > 1$ connected components containing cuts, we create a "truth table" of derivations from the original proof, by repeating the one cut procedure for each connected component. This creates 2^n "experiments", which, when disjunctively composed, have a tautology on top and copies of the conclusion at the bottom. Again, we can easily recover a proof, with a "cap" proving the tautology at the top, and contractions at the bottom.



Not only is this a simple procedure, but it is confluent, modulo equivalences such as associativity and commutativity of conjunction and disjunction—all critical pairs end up converging since we can disjunctively compose proofs.

Extending to predicate logic. With the approach and techniques of the previous section in mind, one natural progression is from propositional to first-order predicate logic. As noted before, the deep-inference proof theory of predicate logic is one of the less studied areas in the field and, in particular, there is little or no exploration (to my knowledge) since the shift of focus from the calculus of structures formalism to the less "bureaucratic" open deduction [20]. Thus, work in this area serves the dual purpose of updating previous research to the current formalism, as well as hopefully contributing new ideas.

An obvious starting point for any exploration of predicate logic proof theory is Herbrand's Theorem [22], especially when looking to develop an approach from propositional logic. Perhaps the standard modern reference to the theorem is a formulation and proof due to Samuel Buss [13], which proves equivalence between first-order provability and the existence of a "Herbrand Proof". Thus, an interesting development that recommends further study is McKinley's discovery of an error in Buss's proof [27], easily fixed by moving to a deep inference system. The error occurs in the reduction of contraction to quantifier-free formulae, which, as contraction reduction is so central to deep inference [12], suggests an important link between deep inference and this presentation of Herbrand's Theorem. In fact, the necessary contraction reduction has been shown to be impossible in the sequent calculus [5], meaning that deep inference is not only helpful for a "Buss-style" Herbrand's theorem, but essential.

For these reasons, my work is focusing on a factorisation of proofs (which I call a *Herbrand Stratification*), due to Brünnler [6], that recovers a "Buss-Style" Herbrand's Theorem for a minimal deep inference proof system for first-order predicate logic, KSgr. This system adds four rules (as well as some syntactic equalities), two to address quantification generally, and two to specifically deal with contraction of formulas containing quantifiers:

$$\text{Quantification rules:}\qquad \mathsf{n}{\downarrow}\,\frac{A[\tau/x]}{\exists x A}\qquad\qquad \mathsf{gr}{\downarrow}\,\frac{Q\{P\{R\}\}}{P\{Q\{R\}\}}$$

$$\text{Contraction:}\qquad \mathsf{qc}{\downarrow}\,\frac{\exists x A \vee \exists x A}{\exists x A}\qquad\qquad \mathsf{m}_2{\downarrow}\,\frac{\forall x A \vee \forall x B}{\forall x [A \vee B]}$$

where Q is a series of quantifiers, and no variable in P is bound by any quantifier in Q.

The Herbrand stratification procedure below, much like the experiments method, is simple, semantically natural (in that it is intimately related to Herbrand's Theorem), and confluent modulo equivalences such as associativity and commutativity of binary connectives, and variable renaming.

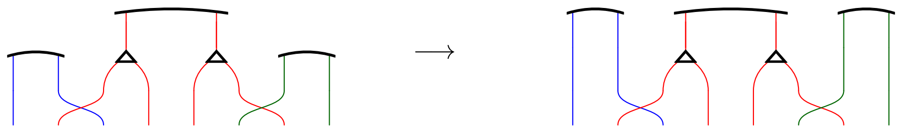
$$\begin{array}{l} \prod \mathsf{KSgr} \\ A \end{array} \xrightarrow{1} \begin{array}{l} \prod \mathsf{KS}\cup\{\mathsf{n}{\downarrow},\mathsf{gr}{\downarrow},\mathsf{m}_2{\downarrow},\mathsf{ai}{\uparrow}\} \\ A' \\ \Vert\, \mathsf{qc}{\downarrow} \\ A \end{array} \xrightarrow{2} \begin{array}{l} \prod \mathsf{KS}\cup\{\mathsf{n}{\downarrow},\mathsf{ai}{\uparrow}\} \\ Q\{B\} \\ \Vert\, \mathsf{gr}{\downarrow} \\ A' \\ \Vert\, \mathsf{qc}{\downarrow} \\ A \end{array} \xrightarrow{3} \begin{array}{l} \prod \mathsf{KS}\cup\{\mathsf{ai}{\uparrow}\} \\ \forall \vec{x}\, W(B) \\ \Vert\, \mathsf{n}{\downarrow} \\ Q\{B\} \\ \Vert\, \mathsf{gr}{\downarrow} \\ A' \\ \Vert\, \mathsf{qc}{\downarrow} \\ A \end{array}$$

The basic idea is to draw the predicate logic rules down to the bottom of the proof, isolating the propositional component at the top. This allows the use of techniques and methods from propositional logic, such as the atomic flow, and, crucially, the experiments method.

The work up to this point, mainly a synthesis of previous results and approaches, was presented at the PCC 2015 Oslo workshop in May.[2] An extended version is currently being written up for hopeful publication. But this research also serves as a springboard for the next topic.

A new syntax for quantification. There are certain attractive proof theoretic properties that characterise deep inference: locality [8], quasipolynomial-time normalisation [3] and low complexity [2], for example. One property especially pertinent to this research is a certain sort of homotopic flexibility in derivations, captured by the atomic flow. It reflects the fact that, as long as certain geometric relations are preserved, inference rules can be moved around with relative freedom. For example, it is simple to move identities to the top of a proof - a procedure important to the experiments method:

$$\cfrac{\cfrac{\mathsf{t}}{\cfrac{a}{\cfrac{\mathsf{t}}{b\vee\bar{b}}\wedge\cfrac{a}{a\wedge a}}\vee\cfrac{\bar{a}}{\cfrac{\bar{a}}{\bar{a}\wedge\bar{a}}\wedge\cfrac{\mathsf{t}}{b\vee\bar{b}}}}}{(b\wedge a)\vee(\bar{b}\wedge a)\vee(\bar{a}\wedge b)\vee(\bar{a}\wedge\bar{b})} \quad\longrightarrow\quad \cfrac{\cfrac{\cfrac{\mathsf{t}}{b\vee\bar{b}}\wedge\cfrac{\mathsf{t}}{a\vee\bar{a}}\wedge\cfrac{\mathsf{t}}{b\vee\bar{b}}}{\left([b\vee\bar{b}]\wedge\cfrac{a}{a\wedge a}\right)\vee\left(\cfrac{\bar{a}}{\bar{a}\wedge\bar{a}}\wedge[b\vee\bar{b}]\right)}\,\mathsf{s}^2}{(b\wedge a)\vee(\bar{b}\wedge a)\vee(\bar{a}\wedge b)\vee(\bar{a}\wedge\bar{b})}$$



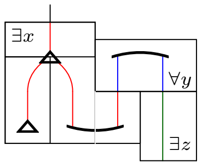
However, the traditional syntax for expressing quantification poses instant problems to achieving these desirable properties for a first-order proof system—this is why, in the previous section, it is convenient simply to move the predicate logic component of a proof out the way before starting the cut elimination process. In fact, derivations in SKSgr have certain similarities to an extension of the sequent calculus using some deep-inference principles, *nested sequents* [9], which allow for nesting of "boxed" sequents. Nested sequent systems are most commonly used for modal logics [11, 25], but have also been adapted for predicate logic [10]. However, such systems seem to preclude any notion of locality as strong as that we obtain for propositional logic due to the reasons spelled out in [8]—for example the fact that in the n↓ rule, A is an unbounded formula that needs to be inspected for occurrences of x (looking at derivations from the bottom-up, proof-search perspective).

So the question is framed: can we design a calculus for first-order logic and a proof system that possesses these attractive properties listed above? Furthermore, can we design this syntax such that it is without unnecessary bureaucracy, the elimination of which is a key motivation for much of structural proof theory? Encouraged by the results of the first two sections, I believe that deep inference is an appropriate proof theoretic tool to achieving these ends, and that a formalism can be developed which employs a similar cut elimination procedure to those described above, one which is semantically natural and confluent. Furthermore, I believe that an augmented version of the atomic flow can be developed for a first-order calculus, providing an important invariant that allows for a highly useful perspective on deep-inference derivations.

[2]The abstract and slides for the presentation can be found online at http://people.bath.ac.uk/bdr25/papers

Thus, a reasonable starting point is to look at the predicate logic system described in the previous section, KSgr, and seeing where improvements can be made. The example below, of a (rather trivial) derivation in SKSgr (i.e. KSgr plus all dual rules), exhibits some of the problems and syntactic inadequacies that must be overcome:

$$\dfrac{\dfrac{\mathsf{n}\!\downarrow\dfrac{Pa}{\exists x Px}}{\dfrac{\exists x\left[\dfrac{Px}{Px \wedge Px}\right]}{\exists x Px \wedge \exists x Px} \wedge \dfrac{\forall y\left(\dfrac{\mathsf{t}}{\bar{P}y \vee \dfrac{Py}{\exists z Pz}}\right)}{\forall y \bar{P}y \vee \exists z Pz}}}{\dfrac{\exists x\left[\dfrac{Px}{\mathsf{t}}\right]}{\mathsf{t}} \wedge \mathsf{s}\,\dfrac{\exists x Px \wedge [\forall y \bar{P}y \vee \exists z Pz]}{\mathsf{gr}\!\uparrow\dfrac{\exists x Px \wedge \forall y \bar{P}y}{\exists x\left[\dfrac{Px \wedge \dfrac{\forall y \bar{P}y}{\bar{P}x}}{\mathsf{f}}\right]} \vee \exists z Pz}}$$
$$\exists z Pz$$



One simple observation to make is that the quantifiers create boxes in the derivation, a feature that has already been compared to nested sequents proof systems. As discussed above, these pose an immediate problem to achieving local inference rules and normalisation procedures. In fact, the atomicity of the structural rules inherited from SKS makes for clumsiness rather than elegance in a predicate logic, with quantifiers having to be pushed out for the structural rules and in for the logical rules.

I believe that it is the boxed approach to quantification that impedes the desired sort of proof theory, and that it is necessary to look beyond the quantifier for solutions. Thus, taking inspiration from a wide range of sources—Hilbert's ϵ-calculus [1], Herbrand's Theorem [22], Miller's expansion trees [28], Hintikka's hyperclassical (or independence-friendly) logic [23] to name a few—I hope to develop a syntax for first-order logic without the standard quantifiers that satisfies the requirements and specifications described above.

At this juncture, I have conducted preliminary investigations into an approach I call the *æ-calculus* (read "alpha-epsilon calculus").[3] It is a simplified version of the ϵ-calculus that replaces quantification with term-level quantifier dependency information:

$$\forall x \forall y \forall z (Pxy \wedge Qxz \wedge Ryz) \quad \longrightarrow \quad P\alpha_x\alpha_y \wedge Q\alpha_x\alpha_z \wedge R\alpha_y\alpha_z$$
$$\forall x \exists y [Pxy \vee Qy] \wedge \exists z Rz \quad \longrightarrow \quad [P\alpha_x\epsilon_y(x) \vee Q\epsilon_y(x)] \wedge R\epsilon_z$$
$$\forall x \exists y [Px \vee Qy] \wedge \exists z Rz \quad \longrightarrow \quad [P\alpha_x\epsilon_y \vee Q\epsilon_y] \wedge R\epsilon_z$$

[3] Some of these ideas were presented to the research group here in Bath. The slides for the presentation can be found online at http://people.bath.ac.uk/bdr25/files/expres.pdf

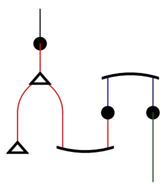
$$\forall x \exists y \forall z \exists w[(Pw \wedge Qwz) \vee (Rzy \wedge Ryx)]$$
$$\downarrow$$
$$[(P\epsilon_w(z,x) \wedge Q\epsilon_w(z,x)\alpha_z(y)) \vee (R\alpha_z(y)\epsilon_y(x) \wedge R\epsilon_y(x)\alpha_x)]$$

Moving quantification information from the propositional level to the term level allows much more freedom for permutation of the propositional rules of deep inference and the possibility of an atomic equivalent to the $\mathsf{n}{\downarrow}$ rule, making it simpler to extend the local cut elimination techniques for propositional logic. The improvement can be seen clearly from the translation of the earlier proof into the new calculus, and motivates an obvious extension to the atomic flow:

$$\dfrac{\dfrac{\mathsf{n}{\downarrow}\dfrac{Pa}{\exists x Px}}{\dfrac{\exists x\left[\dfrac{Px}{Px \wedge Px}\right]}{\exists x Px \wedge \exists x Px} \wedge \dfrac{\forall y\left(\dfrac{\mathsf{t}}{\bar{P}y \vee \dfrac{Py}{\exists z Pz}}\right)}{\forall y \bar{P}y \vee \exists z Pz}}}{\dfrac{\exists x\left[\dfrac{Px}{\mathsf{t}}\right]}{\mathsf{t}} \wedge \mathsf{s}\dfrac{\exists x Px \wedge [\forall y \bar{P}y \vee \exists z Pz]}{\mathsf{gr}{\uparrow}\dfrac{\exists x Px \wedge \forall y \bar{P}y}{\exists x\left[\dfrac{Px \wedge \dfrac{\forall y \bar{P}y}{\bar{P}x}}{\mathsf{f}}\right]} \vee \exists z Pz}} \over \exists z Pz$$

$$\longrightarrow$$

$$\dfrac{\dfrac{\dfrac{\text{æ}{\downarrow}\dfrac{Pa}{P\epsilon_x}}{\mathsf{ac}{\uparrow}\dfrac{P\epsilon_x}{P\epsilon_x \wedge P\epsilon_x} \wedge \mathsf{ai}{\downarrow}\dfrac{\mathsf{t}}{\bar{P}\alpha_z \vee \text{æ}{\downarrow}\dfrac{P\alpha_z}{P\epsilon_y}}}}{P\epsilon_x \wedge [\bar{P}\alpha_z \vee P\epsilon_y]}}{\dfrac{P\epsilon_x}{\mathsf{t}} \wedge \dfrac{P\epsilon_x \wedge \dfrac{\bar{P}\alpha_z}{\bar{P}\epsilon_x}}{\mathsf{f}} \vee P\epsilon_y} \over P\epsilon_y$$



Hopefully, this inquiry will lead to a logical syntax that can express first-order logic (and perhaps more) equipped with a sound and complete proof system and a natural and confluent cut elimination procedure. But even partial results could be of interest: for example it should be very straightforward to construct an æ-style system for the (decidable) monadic fragment of first-order predicate logic. Results that are likely to be interesting and useful include analogues to the first and second ϵ-theorems [1], and an appropriate formulation of Herbrand's Theorem—probably a similar version to that described above.

The value in such a calculus, if successful, would be to further explore the proof theory of classical logic; in particular, the calculus should help to distinguish proof theoretical properties central to the logic from those that are artefacts of any particular syntax. Unlike in Gentzen's time, the fear of inconsistency is no longer the driving force behind proof theory. Instead, we can turn our efforts towards issues of identity and complexity—a proof theory that might already have been developed had Hilbert's discarded twenty-fourth problem replaced his second [31].

References

[1] P. Bernays and D. Hilbert. *Grundlagen der mathematik*. Springer, 1939.

[2] P. Bruscoli and A. Guglielmi. On the proof complexity of deep inference. *ACM Transactions on Computational Logic (TOCL)*, 10(2):14, 2009.

[3] P. Bruscoli, A. Guglielmi, T. Gundersen, and M. Parigot. A quasipolynomial cut-elimination procedure in deep inference via atomic flows and threshold formulae. In *Logic for Programming, Artificial Intelligence, and Reasoning*, pages 136–153. Springer, 2010.

[4] K. Brünnler. *Deep inference and symmetry in classical proofs*. Logos Verlag, 2003.

[5] K. Brünnler. Two restrictions on contraction. *Logic Journal of IGPL*, 11(5):525–529, 2003.

[6] K. Brünnler. Cut elimination inside a deep inference system for classical predicate logic. *Studia Logica*, 82(1):51–71, 2006.

[7] K. Brünnler. *Deep inference and its normal form of derivations*, pages 65–74. Springer, 2006.

[8] K. Brünnler. Locality for classical logic. *Notre Dame Journal of Formal Logic*, 47(4):557–580, 2006.

[9] K. Brünnler. Deep sequent systems for modal logic. *Archive for Mathematical Logic*, 48(6):551–577, 2009.

[10] K. Brünnler. How to universally close the existential rule. In *Logic for Programming, Artificial Intelligence, and Reasoning*, pages 172–186. Springer, 2010.

[11] K. Brünnler and L. Straßburger. *Modular sequent systems for modal logic*, pages 152–166. Springer, 2009.

[12] K. Brünnler and A. F. Tiu. A local system for classical logic. In *Logic for Programming, Artificial Intelligence, and Reasoning*, pages 347–361. Springer, 2001.

[13] S. R. Buss. On herbrand's theorem. In *Logic and Computational Complexity*, pages 195–209. Springer, 1995.

[14] T. Coquand. A semantics of evidence for classical arithmetic. *The Journal of Symbolic Logic*, 60(01):325–337, 1995.

[15] A. Das. *Complexity of deep inference via atomic flows*, pages 139–150. Springer, 2012.

[16] G. Gentzen. Investigations into logical deduction. *American philosophical quarterly*, 1(4):288–306, 1964.

[17] J.-Y. Girard. A new constructive logic: Classical logic. *Mathematical Structures in Computer Science*, 1(03):255–296, 1991.

[18] J.-Y. Girard, P. Taylor, and Y. Lafont. *Proofs and types*, volume 7. Cambridge University Press Cambridge, 1989.

[19] A. Guglielmi. A system of interaction and structure. *Acm Transactions on Computational Logic*, 8(1), 2007. 138zq Times Cited:31 Cited References Count:36.

[20] A. Guglielmi, T. Gundersen, and M. Parigot. A proof calculus which reduces syntactic bureaucracy. *Leibniz International Proceedings in Informatics (LIPIcs)*, 6:16, 2010.

[21] T. E. Gundersen. *A general view of normalisation through atomic flows*. Thesis, Thesis (Ph.D.) - University of Bath, 2009., 2009. Thesis (Ph.D.) - University of Bath, 2009.

[22] J. Herbrand and W. D. Goldfarb. *Logical writings*. Springer, 1971.

[23] J. Hintikka. Hyperclassical logic (aka if logic) and its implications for logical theory. *Bulletin of Symbolic Logic*, 8(03):404–423, 2002.

[24] F. Lamarche and L. Straßburger. *Naming proofs in classical propositional logic*, pages 246–261. Springer, 2005.

[25] S. Marin and L. Straßburger. Label-free modular systems for classical and intuitionistic modal logics. In *Advances in Modal Logic 10*, 2014.

[26] R. McKinley. *Categorical models of first-order classical proofs*. Thesis, University of Bath, 2006.

[27] R. McKinley. A sequent calculus demonstration of herbrand's theorem. *arXiv preprint arXiv:1007.3414*, 2010.

[28] D. A. Miller. A compact representation of proofs. *Studia Logica*, 46(4):347–370, 1987.

[29] M. Parigot. $\lambda\mu$-calculus: An algorithmic interpretation of classical natural deduction. In *Logic programming and automated reasoning*, pages 190–201. Springer, 1992.

[30] D. Prawitz. Natural deduction: A proof-theoretical study. *Almqvist Wiksell, Stockholm*, 1965.

[31] R. Thiele. Hilbert's twenty-fourth problem. *American Mathematical Monthly*, pages 1–24, 2003.