Preface

The **Encyclopedia of Proof Systems** aims at providing a reliable, technically accurate, historically informative, concise and convenient central repository of proof systems for various logics. The goal is to facilitate the exchange of information among logicians with mathematical, computational or philosophical backgrounds; in order to foster and accelerate the development of new proof systems and automated deduction tools that rely on them.

Preparatory work for the creation of the Encyclopedia, such as the implementation of the LaTeX template and the setup of the Github repository, started in October 2014, triggered by the call for workshop proposals for the 25th Conference on Automated Deduction (CADE). Christoph Benzmüller, CADE's conference chair, and Jasmin Blanchette, CADE's workshop co-chair, encouraged me to submit a workshop proposal and supported my alternative idea to organize instead a special poster session based on encyclopedia entries. I am thankful for their encouragement and support.

In December 2014, Björn Lellmann, Giselle Reis and Martin Riener kindly accepted my request to beta-test the template and the instructions I had created. They submitted the first few example entries to the encyclopedia and provided valuable feedback, for which I am grateful. Their comments were essential for improving the templates and instructions before the public announcement of the encyclopedia.

Discussions with Lev Beklemishev, Björn Lellmann, Roman Kuznets, Sergei Soloviev and Anna Zamansky brainstormed many ideas for improving the organization and structure of the encyclopedia. Many of these ideas still need to be fully implemented.

May 2015 *Bruno Woltzenlogel Paleo*

Contents

Part I
Introductions

ToDo:

We plan to add a few short introductory chapters addressing various aspects that are orthogonal to several entries, such as:

- basic technical notions for each type of proof system (e.g. tableaux, natural deduction systems, sequent calculi, resolution calculi ...),
- logical languages
- logics (classical, intuitionistic, modal, substructural, linear, paraconsistent, ...)
- application domains

The goals of these introductory chapters will be to:

- provide a global technical and historical view of the entries,
- reduce repetition of basic notions in the entries,
- increase the understandability of the entries,
- make the encyclopedia more self-contained.

The exact structure and content of these chapters will be decided later, after the collection of sufficiently many entries.

For now, entries are sorted in chronological order only, and various indexes are provided in the backmatter. If the need arises, we might consider grouping entries according to various criteria.

Part II
Proof Systems

Intuitionistic Natural Deduction NJ (1935)

$$\frac{\mathfrak{A} \quad \mathfrak{B}}{\mathfrak{A}\&\mathfrak{B}}\ UE \qquad \frac{\mathfrak{A}\&\mathfrak{B}}{\mathfrak{A}}\ UB \qquad \frac{\mathfrak{A}\&\mathfrak{B}}{\mathfrak{B}}\ UB$$

$$\frac{\mathfrak{A}}{\mathfrak{A}\lor\mathfrak{B}}\ OE \qquad \frac{\mathfrak{B}}{\mathfrak{A}\lor\mathfrak{B}}\ OE \qquad \frac{\mathfrak{A}\lor\mathfrak{B} \quad \begin{array}{c}[\mathfrak{A}]\\ \vdots\\ \mathfrak{C}\end{array} \quad \begin{array}{c}[\mathfrak{B}]\\ \vdots\\ \mathfrak{C}\end{array}}{\mathfrak{C}}\ OB$$

$$\frac{\mathfrak{F}\mathfrak{a}}{\forall\mathfrak{x}\mathfrak{F}\mathfrak{x}}\ AE \qquad \frac{\forall\mathfrak{x}\mathfrak{F}\mathfrak{x}}{\mathfrak{F}\mathfrak{a}}\ AB \qquad \frac{\mathfrak{F}\mathfrak{a}}{\exists\mathfrak{x}\mathfrak{F}\mathfrak{x}}\ EE \qquad \frac{\exists\mathfrak{x}\mathfrak{F}\mathfrak{x} \quad \begin{array}{c}[\mathfrak{F}\mathfrak{a}]\\ \vdots\\ \mathfrak{C}\end{array}}{\mathfrak{C}}\ EB$$

$$\frac{\begin{array}{c}[A]\\ \vdots\\ B\end{array}}{\mathfrak{A}\supset\mathfrak{B}}\ FE \qquad \frac{\mathfrak{A} \quad \mathfrak{A}\supset\mathfrak{B}}{\mathfrak{B}}\ FB \qquad \frac{\begin{array}{c}[\mathfrak{A}]\\ \vdots\\ \curlywedge\end{array}}{\neg\mathfrak{A}}\ NE \qquad \frac{\mathfrak{A} \quad \neg\mathfrak{A}}{\curlywedge}\ NB \qquad \frac{\curlywedge}{\mathfrak{D}}$$

The eigenvariable $\mathfrak{a}$ of an AE must not occur in the formula designated in the schema by $\forall\mathfrak{x}\mathfrak{F}\mathfrak{x}$; nor in any assumption formula upon which that formula depends. The eigenvariable $\mathfrak{a}$ of an EB must not occur in the formula designated in the schema by $\exists\mathfrak{x}\mathfrak{F}\mathfrak{x}$; nor in any assumption formula upon which that formula depends, with the exception of the assumption formulae designated by $\mathfrak{F}\mathring{\mathfrak{a}}$.

Clarifications: The names of the rules are those originally given by Gentzen [1]: U = und (and), O = oder (or), A = all, E = es-gibt (exists), F = folgt (follows), N = nicht (not), E = Einführung (introduction), B = Beseitigung (elimination).

History: The main novelty introduced by Gentzen in this proof system is its *assumption* handling mechanism, which allows formal proofs to reflect more naturally the logical reasoning involved in mathematical proofs.

Remarks: In [1], completeness of **NJ** is proven by showing how to translate proofs in the Hilbert-style calculus **LHJ** to **NJ**-proofs, and soundness is proven by showing how to translate **NJ**-proofs to **LJ**-proofs {3}.

[1] Gerhard Gentzen. "Untersuchungen über das logische Schließen I". In: *Mathematische Zeitschrift* 39.1 (Dec. 1935), pp. 176–210.

Entry 1 by: Bruno Woltzenlogel Paleo

Classical Sequent Calculus LK (1935)

$$\frac{}{A \vdash A} \qquad \frac{\Gamma \vdash \Lambda, A \quad A, \Delta \vdash \Theta}{\Gamma, \Delta \vdash \Lambda, \Theta}\ cut$$

$$\frac{\Gamma \vdash \Theta}{A, \Gamma \vdash \Theta}\ w_l \qquad \frac{\Gamma \vdash \Theta}{\Gamma \vdash \Theta, A}\ w_r$$

$$\frac{\Gamma, B, A, \Delta \vdash \Theta}{\Gamma, A, B, \Delta \vdash \Theta}\ e_l \quad \frac{A, A, \Gamma \vdash \Theta}{A, \Gamma \vdash \Theta}\ c_l \qquad \frac{\Gamma \vdash \Theta, B, A, \Delta}{\Gamma \vdash \Theta, A, B, \Delta}\ e_r \quad \frac{\Gamma \vdash \Theta, A, A}{\Gamma \vdash \Theta, A}\ c_r$$

$$\frac{\Gamma \vdash \Theta, A}{\neg A, \Gamma \vdash \Theta}\ \neg_l \qquad \frac{A, \Gamma \vdash \Theta}{\Gamma \vdash \Theta, \neg A}\ \neg_r$$

$$\frac{A_i, \Gamma \vdash \Theta}{A_1 \wedge A_2, \Gamma \vdash \Theta}\ \wedge_l \qquad \frac{\Gamma \vdash \Theta, A \quad \Gamma \vdash \Theta, B}{\Gamma \vdash \Theta, A \wedge B}\ \wedge_r$$

$$\frac{A, \Gamma \vdash \Theta \quad B, \Gamma \vdash \Theta}{A \vee B, \Gamma \vdash \Theta}\ \vee_l \qquad \frac{\Gamma \vdash \Theta, A_i}{\Gamma \vdash \Theta, A_1 \vee A_2}\ \vee_r$$

$$\frac{\Gamma \vdash \Lambda, A \quad B, \Delta \vdash \Theta}{A \rightarrow B, \Gamma, \Delta \vdash \Lambda, \Theta}\ \rightarrow_l \qquad \frac{A, \Gamma \vdash \Theta, B}{\Gamma \vdash \Theta, A \rightarrow B}\ \rightarrow_r$$

$$\frac{A[\alpha], \Gamma \vdash \Theta}{\exists x.A[x], \Gamma \vdash \Theta}\ \exists_l \quad \frac{A[t], \Gamma \vdash \Theta}{\forall x.A[x], \Gamma \vdash \Theta}\ \forall_l \quad \frac{\Gamma \vdash \Theta, A[\alpha]}{\Gamma \vdash \Theta, \forall x.A[x]}\ \forall_r \quad \frac{\Gamma \vdash \Theta, A[t]}{\Gamma \vdash \Theta, \exists x.A[x]}\ \exists_r$$

The eigenvariable α should not occur in Γ, Θ or $A[x]$.
The term t should not contain variables bound in $A[t]$.

History: This is a modern presentation of Gentzen's original **LK** calculus[1], using modern notations and rule names.

Remarks: **LK** is complete relative to **NK** (i.e. **NJ** {1} with the axiom of excluded middle) and sound relative to a Hilbert-style calculus **LHK** [2]. Cut is eliminable (*Hauptsatz* [1]), and hence classical predicate logic is consistent. Any *prenex* cut-free proof may be further transformed into a shape with only propositional inferences above and only quantifier and structural inferences below a *midsequent* [2].

[1] Gerhard Gentzen. "Untersuchungen über das logische Schließen I". In: *Mathematische Zeitschrift* 39.1 (Dec. 1935), pp. 176–210.

[2] Gerhard Gentzen. "Untersuchungen über das logische Schließen II". In: *Mathematische Zeitschrift* 39.1 (Dec. 1935), pp. 405–431.

Entry 2 by: Martin Riener

Intuitionistic Sequent Calculus LJ (1935)

$$\frac{}{A \vdash A} \qquad \frac{\Gamma \vdash A \quad A, \Delta \vdash \Theta}{\Gamma, \Delta \vdash \Theta}\ cut$$

$$\frac{\Gamma \vdash \Theta}{A, \Gamma \vdash \Theta}\ w_l \qquad \frac{\Gamma \vdash}{\Gamma \vdash A}\ w_r$$

$$\frac{\Gamma, B, A, \Delta \vdash \Theta}{\Gamma, A, B, \Delta \vdash \Theta}\ e_l \qquad \frac{A, A, \Gamma \vdash \Theta}{A, \Gamma \vdash \Theta}\ c_l$$

$$\frac{\Gamma \vdash A}{\neg A, \Gamma \vdash}\ \neg_l \qquad \frac{A, \Gamma \vdash}{\Gamma \vdash \neg A}\ \neg_r$$

$$\frac{A_i, \Gamma \vdash \Theta}{A_1 \wedge A_2, \Gamma \vdash \Theta}\ \wedge_l \qquad \frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\ \wedge_r$$

$$\frac{A, \Gamma \vdash \Theta \quad B, \Gamma \vdash \Theta}{A \vee B, \Gamma \vdash \Theta}\ \vee_l \qquad \frac{\Gamma \vdash A_i}{\Gamma \vdash A_1 \vee A_2}\ \vee_r$$

$$\frac{\Gamma \vdash A \quad B, \Delta \vdash \Theta}{A \to B, \Gamma, \Delta \vdash \Theta}\ \to_l \qquad \frac{A, \Gamma \vdash B}{\Gamma \vdash A \to B}\ \to_r$$

$$\frac{A[\alpha], \Gamma \vdash \Theta}{\exists x.A[x], \Gamma \vdash \Theta}\ \exists_l \qquad \frac{\Gamma \vdash A[t]}{\Gamma \vdash \exists x.A[x]}\ \exists_r$$

$$\frac{A[t], \Gamma \vdash \Theta}{\forall x.A[x], \Gamma \vdash \Theta}\ \forall_l \qquad \frac{\Gamma \vdash A[\alpha]}{\Gamma \vdash \forall x.A[x]}\ \forall_r$$

The eigenvariable α should not occur in Γ, Θ or $A[x]$.
The term t should not contain variables bound in $A[t]$.

Clarifications: LJ and **LK** {2} have exactly the same rules, but in **LJ** the succedent of every sequent may have at most one formula. This restriction is equivalent to forbidding the axiom of excluded middle in natural deduction.

Remarks: The cut rule is eliminable (*Hauptsatz* [1]), and hence intuitionistic predicate logic is consistent and its propositional fragment is decidable [2]. **LJ** is complete relative to **NJ** {1} and sound relative to the Hilbert-style calculus **LHJ** [2].

[1] Gerhard Gentzen. “Untersuchungen über das logische Schließen I”. In: *Mathematische Zeitschrift* 39.1 (Dec. 1935), pp. 176–210.

[2] Gerhard Gentzen. “Untersuchungen über das logische Schließen II”. In: *Mathematische Zeitschrift* 39.1 (Dec. 1935), pp. 405–431.

Entry 3 by: Giselle Reis

Multi-Conclusion Sequent Calculus LJ′ (1954)

$$\frac{}{A \vdash A} \qquad \frac{\Gamma \vdash \Theta, A \quad A, \Delta \vdash \Lambda}{\Gamma, \Delta \vdash \Theta, \Lambda}\ cut$$

$$\frac{A_i, \Gamma \vdash \Theta}{A_1 \wedge A_2, \Gamma \vdash \Theta}\ \wedge_l \qquad \frac{\Gamma \vdash \Theta, A \quad \Gamma \vdash \Theta, B}{\Gamma \vdash \Theta, A \wedge B}\ \wedge_r$$

$$\frac{A, \Gamma \vdash \Theta \quad B, \Gamma \vdash \Theta}{A \vee B, \Gamma \vdash \Theta}\ \vee_l \qquad \frac{\Gamma \vdash \Theta, A_i}{\Gamma \vdash \Theta, A_1 \vee A_2}\ \vee_r$$

$$\frac{\Gamma \vdash \Theta, A \quad B, \Delta \vdash \Lambda}{A \to B, \Gamma, \Delta \vdash \Theta, \Lambda}\ \to_l \qquad \frac{A, \Gamma \vdash B}{\Gamma \vdash A \to B}\ \to_r$$

$$\frac{A\alpha, \Gamma \vdash \Theta}{\exists x.Ax, \Gamma \vdash \Theta}\ \exists_l \quad \frac{\Gamma \vdash \Theta, At}{\Gamma \vdash \Theta, \exists x.Ax}\ \exists_r \quad \frac{At, \Gamma \vdash \Theta}{\forall x.Ax, \Gamma \vdash \Theta}\ \forall_l \quad \frac{\Gamma \vdash A\alpha}{\Gamma \vdash \forall x.Ax}\ \forall_r$$

$$\frac{\Gamma \vdash \Theta, A}{\neg A, \Gamma \vdash \Theta}\ \neg_l \quad \frac{A, \Gamma \vdash}{\Gamma \vdash \neg A}\ \neg_r \quad \frac{\Gamma, B, A, \Delta \vdash \Theta}{\Gamma, A, B, \Delta \vdash \Theta}\ e_l \quad \frac{\Gamma\Delta \vdash \Theta, B, A, \Lambda}{\Gamma \vdash \Theta, A, B, \Lambda}\ e_r$$

$$\frac{A, A, \Gamma \vdash \Theta}{A, \Gamma \vdash \Theta}\ c_l \quad \frac{\Gamma \vdash \Theta, A, A}{\Gamma \vdash \Theta, A}\ c_r \quad \frac{\Gamma \vdash \Theta}{A, \Gamma \vdash \Theta}\ w_l \quad \frac{\Gamma \vdash}{\Gamma \vdash A}\ w_r$$

The eigenvariable α should not occur in Γ, Θ or $A[x]$.
The term t should not contain variables bound in $A[t]$.

Clarifications: While **LJ** {3} is defined by restricting **LK** {2} to single conclusion, in **LJ′** only the rules $\neg_r$, $\to_r$ and $\forall_r$ have this restriction.

History: **LJ′** was proposed in [6] and used to prove the completeness of **LJ** {3} in [4]. It also appears in [3] (as GHPC) and [5] (as L').

Remarks: **LJ′** is equivalent to **LJ**, and this is established by translating sequents of the form $\Gamma \vdash A_1, ..., A_n$ into sequents of the form $\Gamma \vdash A_1 \vee ... \vee A_n$. Cut can be eliminated by using a combination of the rewriting rules for cut-elimination in **LJ** and **LK** and permutation of inferences, as shown by Schellinx [2] and Reis [1].

[1] Giselle Reis. "Cut-elimination by resolution in intuitionistic logic". PhD thesis. Vienna University of Technology, July 2014.

[2] Harold Schellinx. "Some Syntactical Observations on Linear Logic". In: *J. Log. Comput.* 1.4 (1991), pp. 537–559.

[3] A. G. Dragalin. *Mathematical Intuitionism: Introduction to Proof Theory*. American Mathematical Society, 1988.

[4] Gaisi Takeuti. *Proof Theory*. 2nd Edition. North Holland, 1987.

[5] Michael Dummett. *Elements of Intuitionism*. Oxford: Clarendon Press, 1977.

[6] Shôji Maehara. "Eine Darstellung der intuitionistischen Logik in der klassischen". In: *Nagoya Math. J.* 7 (1954), pp. 45–64.

Entry 4 by: Giselle Reis and Valeria de Paiva

Second Order λ-Calculus (System F) (1971)

$$\frac{(x:T)\in E}{\Gamma;E\vdash x:T}\ \text{assumption}$$

$$\frac{\Gamma;E,(x:T)\vdash e:S}{\Gamma;E\vdash\lambda(x:T.e):(T\to S)}\to I \qquad \frac{\Gamma;E\vdash f:(T\to S)\quad \Gamma;E\vdash e:T}{\Gamma;E\vdash(fe):\tau}\to E$$

$$\frac{\Gamma X;E\vdash e:T}{\Gamma;E\vdash(\Lambda X:Tp.e):(\forall X:Tp.T)}\ \forall I^{*} \qquad \frac{\Gamma;E\vdash f:(\forall X:Tp.T)\quad \Gamma\vdash S:Tp}{\Gamma;E\vdash fS:[S/X]T}\ \forall E$$

* X must be not free in the type of any free term variable in E.

Clarifications: The presentation from [3] with minor corrections is used. Below $X,Y,Z...$ are type-variables and $x,y,...$ term variables.

Type expressions: $T := X|(T\to S)|(\forall X:Tp.T)$.

Term expressions: $e := x|(ee)|(eT)|(\lambda x:T.e)|(\Lambda X:Tp.e)$.

$\forall,\Lambda$ and λ are variable binders. All expressions are considered up to renaming of bound variables (α-conversion). An unbound variable is free. $FV(R)$ is the set of free variables for any (type or term) expression; $[e/x]$, $[S/X]$ mean capture-avoiding substitution in term- and type-expressions respectively (defined by induction). A context is a finite set Γ of type variables; ΓX stands for $\Gamma\cup X$. A type T is legal in Γ iff $FV(T)\subseteq FV(\Gamma)$. A type assignment in Γ is a finite list $E=(x_1:T_1),...,(x_n:T_n)$ where any T_i is legal in Γ. The typing relation $\Gamma;E\vdash e:T$, where E is a type assignmentlegal in Γ, e is a term expression and T is a type expression, is defined by the rules above.

The *conversion relation* between well-typed terms is very important. It is defined by the following axioms: (β) $(\lambda x:T.f)e=[e/x]f$; (β_2) $(\Lambda X:Tp.e)S=[S/X]e$; (η) $\lambda x:T.(ex)=e$ if $x\notin FV(e)$; (η_2) $\Lambda X:Tp.(eX)=e$ if $X\notin FV(e)$, and by usual rules that turn "=" into congruence. The system $\mathbf{F}_c$ is obtained if one more equality axiom is added: (**C**) $\quad eT=eT'$ for $\Gamma;E\vdash e:\forall X.S$ and $X\notin FV(S)$.

History: Introduced by Girard [8] and Reynolds [6]. Inspired works on higher order type systems. Included by Barendregt in his λ-cube [4]. Various extensions were considered, for example, $\mathbf{F}_c$ [2], **F** with subtyping [5, 1]. Important for functional programming languages.

Remarks: A strong normalization theorem for **F** was proved by Girard [7]. It implies a normalization theorem and consistency for second order arithmetic PA_2. For $\mathbf{F}_c$, a *genericity theorem* holds [2].

[1] G. Longo, K. Milsted, and S. Soloviev. "Coherence and Transitivity of Subtyping as Entailment". In: *Journal of Logic and Computation* 10 (2000).

Entry 5 by: Giuseppe Longo, Kathleen Milsted, Sergei Soloviev

[2] G. Longo, K. Milsted, and S. Soloviev. “The Genericity Theorem and the Notion of Parametricity in the Polymorphic λ-calculus”. In: *Theoretical Computer Science* 121 (1993).

[3] Andrea Asperti and Giuseppe Longo. *Categories, Types and Structures*. Cambridge, Mass., London, England: The MIT Press, 1991.

[4] H.P. Barendregt. “Introduction to generalized type systems”. In: *Journal of Functional Programming* 2 (1991).

[5] L. Cardelli, S. Martini, J.C. Mitchell, and A. Scedrov. “An Extension of System F with Subtyping”. In: *Lecture Notes in Computer Science* 526 (1991).

[6] J.C. Reynolds. “Towards a Theory of Type Structure”. In: *Lecture Notes in Computer Science* 19 (1974).

[7] J.-Y. Girard. “Interprétation fonctionelle et élimination des coupures de l’arithmétique d’ordre supérieur”. PhD thesis. Université Paris VII, 1972.

[8] J.-Y. Girard. “Une extension de l’interpretation fonctionelle de Gödel à l’analyse et son application à l’élimination des coupures dans et la thèorie des types’”. In: *Proc. 2nd Scandinavian Logic Symposium.* North-Holland (1971).

Expansion Proofs (1983)

Expansion trees, *eigenvariables*, and the function $\mathsf{Sh}(-)$ (read *shallow formula of*), that maps an expansion tree to a formula, are defined as follows:

1. If A is $\top$ (true), $\bot$ (false), or a literal, then A is an expansion tree with top node A, and $\mathsf{Sh}(A) = A$.
2. If E is an expansion tree with $\mathsf{Sh}(E) = [y/x]A$ and y is not an eigenvariable of any node in E, then $E' = \forall x.A +^y E$ is an expansion tree with top node $\forall x.A$ and $\mathsf{Sh}(E') = \forall x.A$. The variable y is called an *eigenvariable* of (the top node of) E'. The set of eigenvariables of all nodes in an expansion tree is called the *eigenvariables of* the tree.
3. If $\{t_1, \ldots, t_n\}$ (with $n \geq 0$) is a set of terms and $E_1, \ldots, E_n$ are expansion trees with pairwise disjoint eigenvariable sets and with $\mathsf{Sh}(E_i) = [t_i/x]A$ for $i \in \{1, \ldots, n\}$, then $E' = \exists x.A +^{t_1} E_1 \ldots +^{t_n} E_n$ is an expansion tree with top node $\exists x.A$ and $\mathsf{Sh}(E') = \exists x.A$. The terms $t_1, \ldots, t_n$ are known as the *expansion terms* of (the top node of) E'.
4. If E_1 and E_2 are expansion trees that share no eigenvariables and $\circ \in \{\wedge, \vee\}$, then $E_1 \circ E_2$ is an expansion tree with top node $\circ$ and $\mathsf{Sh}(E_1 \circ E_2) = \mathsf{Sh}(E_1) \circ \mathsf{Sh}(E_2)$.

In the expansion tree $\forall x.A +^x E$ (resp. in $\exists x.A +^{t_1} E_1 \ldots +^{t_n} E_n$), we say that x (resp. t_i) *labels* the top node of E (resp. E_i, for any $i \in \{1, \ldots, n\}$). A term t *dominates* a node in an expansion tree if it labels a parent node of that node in the tree.

For an expansion tree E, the quantifier-free formula $\mathsf{Dp}(E)$, called the *deep formula of E*, is defined as:

- $\mathsf{Dp}(E) = E$ if E is $\top$, $\bot$, or a literal;
- $\mathsf{Dp}(E_1 \circ E_2) = \mathsf{Dp}(E_1) \circ \mathsf{Dp}(E_2)$ for $\circ \in \{\wedge, \vee\}$;
- $\mathsf{Dp}(\forall x.A +^y E) = \mathsf{Dp}(E)$; and
- $\mathsf{Dp}(\exists x.A +^{t_1} E_1 \cdots +^{t_n} E_n) = \mathsf{Dp}(E_1) \vee \ldots \vee \mathsf{Dp}(E_n)$ if $n > 0$, and $\mathsf{Dp}(\exists x.A) = \bot$.

Let $\mathcal{E}$ be an expansion tree and let $<^0_{\mathcal{E}}$ be the binary relation on the occurrences of expansion terms in $\mathcal{E}$ defined by $t <^0_{\mathcal{E}} s$ if there is an x which is free in s and which is the eigenvariable of a node dominated by t. Then $<_{\mathcal{E}}$, the transitive closure of $<^0_{\mathcal{E}}$, is called the *dependency relation* of $\mathcal{E}$.

An expansion tree $\mathcal{E}$ is said to be an *expansion proof* if $<_{\mathcal{E}}$ is acyclic and $\mathsf{Dp}(\mathcal{E})$ is a tautology; in particular, $\mathcal{E}$ is an *expansion proof of* $\mathsf{Sh}(\mathcal{E})$.

Clarifications: The soundness and completeness theorem for expansion trees is the following. A formula B is a theorem of first-order logic if and only if there is an expansion proof Q such that $\mathsf{Sh}(Q) = B$.

History: Expansion trees and expansion proofs [3, 4] provide a simple generalization of both Herbrand's disjunctions and Gentzen's mid-sequent theorem to formulas that are not necessarily in prenex-normal form. These proof structures were

Entry 6 by: Dale Miller

originally defined for higher-order classical logic and used to provide a generalization of Herbrand's theorem for higher-order logic as well as a soundness proof for skolemization in the presence of higher-order quantification. Expansion trees are an early example of a matrix-based proof system that emphasizes parallelism within proof structures in a manner similar to that found in linear logic proof nets [2]. That parallelism is explicitly analyzed in [1] using a multi-focused version of LKF {16}.

---

[1] Kaustuv Chaudhuri, Stefan Hetzl, and Dale Miller. "A Multi-Focused Proof System Isomorphic to Expansion Proofs". In: *J. of Logic and Computation* (June 2014).

[2] Jean-Yves Girard. "Linear Logic". In: *Theoretical Computer Science* 50 (1987), pp. 1–102.

[3] Dale Miller. "A Compact Representation of Proofs". In: *Studia Logica* 46.4 (1987), pp. 347–370.

[4] Dale Miller. "Proofs in Higher-order Logic". PhD thesis. Carnegie-Mellon University, Aug. 1983.

Bledsoe's Natural Deduction - Prover (1973-78)

SPLIT: basic rules of Natural Deduction(see {1}), for example
To prove $A \wedge B$, prove A and prove B
To prove $p \to A \wedge B$, prove $(p \to A) \wedge (p \to B)$
To prove $p \vee q \to A$, prove $(p \to A) \wedge (q \to A)$
To prove $\exists x P(x) \to D$, prove $P(y) \to D$, where y is a new variable
REDUCE: conversion rules, for example
To prove $x \in A \cap B$, prove $x \in A \wedge x \in B$
To prove $S \in \mathcal{P}(A)$, prove $S \subset A \wedge S \in \mathcal{U}$
To prove $x \in \sigma F$, prove $\exists y(y \in F \wedge x \in y)$
DEFINITIONS, example
$A \subset B$ is defined by $\forall x(x \in A \to x \in B)$ and is replaced by $x \in A \to x \in B$ or by $x_o \in A \to x_o \in B$, depending on the position of the formula in the theorem.
IMPLY: in addition to SPLIT and REDUCE rules,
- search for substitutions which unify some hypotheses and a conclusion and compose them until obtaining the empty substitution (theorem proved) or failing
- forward chaining : if P and P' are unified by θ ($P\theta{=}P'\theta$), then a hypothesis $P' \wedge (P \to Q)$ is converted into $P' \wedge (P \to Q) \wedge Q\theta$
- PEEK forward chaining : if $P\theta{=}P'\theta$ and A has the definition $(P \to Q)$, then a hypothesis $P' \wedge A$ is converted into $P' \wedge A \wedge Q\theta$
- backward chaining : if $A \to D$ and $D\theta = C\theta$, replace the conclusion C by $A\theta$

Clarifications: Bledsoe's natural deduction may be seen as both an extension and a restriction of formal natural deduction {1}. In SPLIT and REDUCE, there is reduction but not expansion. Some subroutines convert expressions into forms convenient for applying the rules. The notions of hypothesis and conclusion are privileged.

History: After having applied the rules of IMPLY and REDUCE, the first version of **Prover** [3] called a resolution program if necessary. Then, in [2], these calls to resolution are completely replaced by IMPLY. **Prover** has been working in set theory, limit theorems, topology and program verification.

Remarks: The system is sound but not complete. Bledsoe emphasizes the fact that, with these methods, provers may succeed because they proceed in a natural human-like way [1].

---

[1] W. W. Bledsoe. "Non-resolution theorem proving". In: *Artificial Intelligence* 9 (1977), pp. 1–35.

[2] R. S. Boyer W. W. Bledsoe and W. H. Henneman. "Computer proofs of Limit theorems". In: 3 (1972).

[3] W. W. Bledsoe. "Splitting and reduction heuristics in automatic theorem proving". In: *Artificial Intelligence* 2 (1971), pp. 55–77.

Entry 7 by: Dominique Pastre

Natural Knowledge Bases - Muscadet (1984)

Some of the rules given to the system :
Basic rules of Natural Deduction (similar to Bledsoe's SPLIT rules {7}).
Flatten : Replace $P(f(x))$ by $\exists y(y : f(x) \wedge P(y))$ or by $\forall y(y : f(x) \Rightarrow P(y))$ depending on the position (positive or negative) of the formula in the theorem to be proved and in the definitions and lemmas.
Rules automatically built by metarules from definitions :
If $A \subset B$ and $x \in A$ then $x \in B$ If $x \in \sigma E$, then $\exists y(y \in E \wedge x \in y)$
If $C : A \cap B$ and $x \in C$, then $x \in A$ If $C : A \cap B$, $x \in A$ and $x \in B$, then $x \in C$
in place of (and more general than) given REDUCE conversion rules of {7}.
and from universal hypotheses :
Universal hypotheses are removed and replaced by local rules (for a sub-theorem). This replaces and generalizes PEEK forward-chaining of {7}.

Clarifications: "If $C : A \cap B$" expresses that C is $A \cap B$ which has already been introduced. Flattening is used to recursively create and name objects such as $f(x)$, and in a certain manner to "eliminate" functional symbols since the expression $y : f(x)$ will be handled as if it was a predicate expression $F(x)$.
Rules are conditional actions. Actions may be defined by packs of rules. Metarules build rules from definitions, lemmas and universal hypotheses.

History: Muscadet [3, 2] is a knowledge-based system. Facts are hypotheses and the conclusion of a theorem or a sub-theorem to be proved, and all sorts of facts which give relevant information during the proof search process. Universal hypotheses are handled as local definitions (no skolemization). **Muscadet** worked in set theory, mappings and relations, topology and topological linear spaces, elementary geometry, discrete geometry, cellular automata, and TPTP problems. It attended CASC competitions. It is open software, freely available.
Muscadet is efficient for everyday mathematical problems which are expressed in a natural manner, and problems which involve many axioms, definitions or lemmas, but not for problems with only one large conjecture and few definitions.

Remarks: The system is sound but not complete (because of the use of many selective rules and heuristics). It displays proofs easily readable by a human reader.

[1] D. Pastre. *Muscadet version 4.1 : user's manual*. 2011, pp. 1–22. URL: http://www.normalesup.org/~pastre/muscadet/manual-en.pdf.

[2] D. Pastre. "Automated Theorem Proving in Mathematics". In: *Annals on Artificial Intelligence and Mathematics* 8.3-4 (1993), pp. 425–447.

[3] D. Pastre. "MUSCADET : An Automatic Theorem Proving System using Knowledge and Metaknowledge in Mathematics". In: *Artificial Intelligence* 38.3 (1989), pp. 257–318.

Entry 8 by: Dominique Pastre

Full Intuitionistic Linear Logic (FILL) (1990)

$$\frac{}{x : A \vdash x : A}\ Ax \qquad \frac{\Gamma \vdash t : A \mid \Delta \quad \Gamma', y : A \vdash \Delta'}{\Gamma, \Gamma' \vdash \Delta \mid [t/y]\Delta'}\ Cut$$

$$\frac{\Gamma \vdash \Delta}{\Gamma, x : \top \vdash \mathsf{let}\, x\, \mathsf{be} * \mathsf{in}\, \Delta}\ \top_L \qquad \frac{}{\cdot \vdash * : \top}\ \top_R$$

$$\frac{}{x : \bot \vdash \cdot}\ \bot_L \qquad \frac{\Gamma \vdash \Delta}{\Gamma \vdash \circ : \bot \mid \Delta}\ \bot_R$$

$$\frac{\Gamma, x : A, y : B \vdash \Delta}{\Gamma, z : A \otimes B \vdash \mathsf{let}\, z\, \mathsf{be}\, x \otimes y\, \mathsf{in}\, \Delta}\ \otimes_L \qquad \frac{\Gamma \vdash t_1 : A \mid \Delta \quad \Gamma' \vdash t_2 : B \mid \Delta'}{\Gamma, \Gamma' \vdash t_1 \otimes t_2 : A \otimes B \mid \Delta \mid \Delta'}\ \otimes_R$$

$$\frac{\Gamma \vdash t : A \mid \Delta \quad \Gamma', x : B \vdash t_i : C_i}{\Gamma, y : A \multimap B, \Gamma' \vdash [y\,t/x]t_i : C_i \mid \Delta}\ \multimap_L \qquad \frac{\Gamma, x : A \vdash t : B \quad x \notin \mathsf{FV}(\Delta)}{\Gamma \vdash \lambda x.t : A \multimap B \mid \Delta}\ \multimap_R$$

$$\frac{\Gamma, x : A \vdash t_i : C_i \quad \Gamma', y : B \vdash t_j : D_j}{\Gamma, \Gamma', z : A \parr B \vdash \mathsf{let\text{-}pat}\, z\, (x \parr -)\, t_i : C_i \mid \mathsf{let\text{-}pat}\, z\, (- \parr y)\, t_j : D_j}\ \parr_L$$

$$\frac{\Gamma \vdash \Delta \mid t_1 : A \mid t_2 : B \mid \Delta'}{\Gamma \vdash \Delta \mid t_1 \parr t_2 : A \parr B \mid \Delta'}\ \parr_R$$

Clarifications: The left-hand side and right-hand side of sequents are multisets of formulas denoted Γ and Δ. The terms annotating formulas are standard terms used in the simply typed λ-calculus. Capture avoiding substitution is denoted by $[t/x]t'$, and uniformly replaces every occurrence of x in t' with t. The definition of the let-pattern function used in the rule $\parr_L$ is defined as follows:

$$\begin{array}{lll} \mathsf{let\text{-}pat}\, z\, (x \parr -)\, t = t & \mathsf{let\text{-}pat}\, z\, (- \parr y)\, t = t & \mathsf{let\text{-}pat}\, z\, p\, t = \mathsf{let}\, z\, \mathsf{be}\, p\, \mathsf{in}\, t \\ \quad \text{where } x \notin \mathsf{FV}(t) & \quad \text{where } y \notin \mathsf{FV}(t) & \end{array}$$

We denote vectors of terms (resp. types) by t_i (resp. A_j). The function $\mathsf{FV}(\Delta)$ constructs the set of all free variables in each term found in Δ.

History: The formulation of **FILL** given here was defined by Martin Hyland and Valeria de Paiva [6]. This version was an improvement over an early version first defined by Valeria de Paiva in her thesis [7]. The improvement was the addition of the term assignment which was necessary to gain cut elimination. There was still a mistake in [6], which was corrected independently, with different proof methods, by Bierman [5], Bellin [4], Brauner/dePaiva [3], dePaiva/Ritter [2]. The version here is the minimal modification suggested by Bellin, who used proofnets in [4], here using a term assignment, described in Eades/dePaiva [1].

Entry 9 by: Harley Eades III, Valeria de Paiva

[1] Harley Eades III and Valeria de Paiva. "Multiple Conclusion Intuitionistic Linear Logic and Cut Elimination". `http://metatheorem.org/papers/FILL-cut-report.pdf`.

[2] Valeria de Paiva and Eike Ritter. "A Parigot-style Linear Lambda-calculus for Full Intuitionistic Linear Logic". In: *Theory and Applications of Categories* 17.3 (2006), pp. 127–152.

[3] Torben Braüner and Valeria de Paiva. "A formulation of linear logic based on dependency-relations". In: *Computer Science Logic*. Ed. by Mogens Nielsen and Wolfgang Thomas. Vol. 1414. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 1998, pp. 129–148.

[4] Gianluigi Bellin. "Subnets of proof-nets in multiplicative linear logic with MIX". In: *Mathematical Structures in Computer Science* 7 (Dec. 1997), pp. 663–669.

[5] Gavin Bierman. "A note on full intuitionistic linear logic". In: *Annals of Pure and Applied Logic* 79.3 (1996), pp. 281–287.

[6] Martin Hyland and Valeria de Paiva. "Full intuitionistic linear logic (extended abstract)". In: *Annals of Pure and Applied Logic* 64.3 (1993), pp. 273–291.

[7] Valeria de Paiva. "The Dialectica Categories". PhD thesis. University of Cambridge, 1990.

Z. Luo’s LF (1994)

$$\frac{}{<>\vdash \mathbf{valid}} \quad \frac{\Gamma \vdash K\mathbf{kind} \quad x \notin FV(\Gamma)}{\Gamma, x : K \vdash \mathbf{valid}} \quad \frac{\Gamma, x : K, \Gamma' \vdash \mathbf{valid}}{\Gamma, x : K, \Gamma' \vdash x : K} \quad (1)$$

$$\frac{\Gamma \vdash k : K \quad \Gamma \vdash K = K'}{\Gamma \vdash k : K'} \quad \frac{\Gamma \vdash k = k' : K \quad \Gamma \vdash K = K'}{\Gamma \vdash k = k' : K'} \quad (2)^{*}$$

$$\frac{\Gamma, x : K, \Gamma' \vdash J \quad \Gamma \vdash k : K}{\Gamma, [k/x]\Gamma' \vdash [k/x]J} \quad (3)^{**}$$

$$\frac{\Gamma \vdash K\mathbf{kind} \quad \Gamma, x : K \vdash K'\mathbf{kind}}{\Gamma \vdash (x : K)K'\mathbf{kind}} \quad \frac{\Gamma \vdash K_1 = K_2 \quad \Gamma, x : K_1 \vdash K_1' = K_2'}{\Gamma \vdash (x : K_1)K_1' = (x : K_2)K_2'}$$

$$\frac{\Gamma \vdash K\mathbf{kind} \quad \Gamma, x : K \vdash K'\mathbf{kind}}{\Gamma \vdash (x : K)K'\mathbf{kind}} \quad \frac{\Gamma \vdash K_1 = K_2 \quad \Gamma, x : K_1 \vdash K_1' = K_2'}{\Gamma \vdash (x : K_1)K_1' = (x : K_2)K_2'}$$

$$\frac{\Gamma \vdash K\mathbf{kind} \quad \Gamma, x : K \vdash K'\mathbf{kind}}{\Gamma \vdash (x : K)K'\mathbf{kind}} \quad \frac{\Gamma \vdash K_1 = K_2 \quad \Gamma, x : K_1 \vdash K_1' = K_2'}{\Gamma \vdash (x : K_1)K_1' = (x : K_2)K_2'}$$

$$\frac{\Gamma \vdash K\mathbf{kind} \quad \Gamma, x : K \vdash K'\mathbf{kind}}{\Gamma \vdash (x : K)K'\mathbf{kind}} \quad \frac{\Gamma \vdash K_1 = K_2 \quad \Gamma, x : K_1 \vdash K_1' = K_2'}{\Gamma \vdash (x : K_1)K_1' = (x : K_2)K_2'} \quad (4)$$

$$\frac{\Gamma \vdash \mathbf{valid}}{\Gamma \vdash \mathbf{Type}\,\mathbf{kind}} \quad \frac{\Gamma \vdash A : \mathbf{Type}}{\Gamma \vdash El(A)\mathbf{kind}} \quad (5)$$

Clarifications: We follow [3]. Terms of **LF** are of the forms **Type**, $El(A)$, $(x : K)K'$ (dependent product), $[x : K]K'$ (abstraction), $f(k)$, and judgements of the forms $\Gamma \vdash$ **valid** (validity of context), $\Gamma \vdash K\mathbf{kind}$, $\Gamma \vdash k : K$, $\Gamma \vdash k = k' : K$, $\Gamma \vdash K = K'$. Rule groups: (1) rules for contexts and assumptions; (2)* equality rules (reflexivity, symmetry and transitivity rules are ommitted); (3)** substitution rules (J denotes the right side of any of the five forms of judgement); (4) rules for dependent product kinds; (5) and the kind **Type**.

History: The calculus was defined in [3], ch. 9. LF is a typed version of Martin-Löf’s logical framework [4]. Type theories specified in **LF** were used as basis of proof-assistants Lego and Plastic. Later the system was extended to include coercive subtyping [**LuoSolXue:14**, 1].

Remarks: The proof-theoretical analysis of LF above was used in meta-theoretical studies of larger theories defined on its basis, *e.g.*, UTT (Universal Type Theory) that includes inductive schemata, second order logic SOL with impredicative type *Prop* and a hierarchy of predicative universes [3]. H. Goguen defined a typed operational semantics for UTT and proved strong normalization theorem [2]. For **LF** with coercive subtyping conservativity results were obtained [**LuoSolXue:14**, 1].

Entry 10 by: Zhaohui Luo and Sergei Soloviev

[1] S. Soloviev and Z. Luo. “Coercion Completion and Conservativity in Coercive Subtyping”. In: *Annals of Pure and Applied Logic* 113.1-3 (2002), pp. 297–322.

[2] H. Goguen. “A Typed Operational Semantics for Type Theory”. PhD thesis. University of Edinburgh, 1994.

[3] Zhaohui Luo. *Computation and Reasoning. A Type Theory for Computer Science*. Oxford, UK: Clarendon Press, 1994.

[4] B. Nordström, G. Petersson, and J. Smith. *Programming in Martin-Löf's Type Theory: An Introduction*. Oxford, UK: Oxford University Press, 1990.

Sequent Calculus G3c (1996)

$$\frac{}{P,\varGamma \vdash \varDelta, P}\ \mathrm{Ax} \qquad \frac{}{\bot,\varGamma \vdash \varDelta}\ \mathrm{L}\bot$$

$$\frac{A,B,\varGamma \vdash \varDelta}{A \wedge B,\varGamma \vdash \varDelta}\ \mathrm{L}\wedge \qquad \frac{\varGamma \vdash \varDelta, A \quad \varGamma \vdash \varDelta, B}{\varGamma \vdash \varDelta, A \wedge B}\ \mathrm{R}\wedge$$

$$\frac{A,\varGamma \vdash \varDelta \quad B,\varGamma \vdash \varDelta}{A \vee B,\varGamma \vdash \varDelta}\ \mathrm{L}\vee \qquad \frac{\varGamma \vdash \varDelta, A, B}{\varGamma \vdash \varDelta, A \vee B}\ \mathrm{R}\vee$$

$$\frac{\varGamma \vdash \varDelta, A \quad B,\varGamma \vdash \varDelta}{A \to B,\varGamma \vdash \varDelta}\ \mathrm{L}\to \qquad \frac{A,\varGamma \vdash \varDelta, B}{\varGamma \vdash \varDelta, A \to B}\ \mathrm{R}\to$$

$$\frac{\forall xA, A[x/t],\varGamma \vdash \varDelta}{\forall xA,\varGamma \vdash \varDelta}\ \mathrm{L}\forall \qquad \frac{\varGamma \vdash \varDelta, A[x/y]}{\varGamma \vdash \varDelta, \forall xA}\ \mathrm{R}\forall$$

$$\frac{A[x/y],\varGamma \vdash \varDelta}{\exists xA,\varGamma \vdash \varDelta}\ \mathrm{L}\exists \qquad \frac{\varGamma \vdash \varDelta, A[x/t], \exists xA}{\varGamma \vdash \varDelta, \exists xA}\ \mathrm{R}\exists$$

P should be atomic in Ax and *y* should not be free in the conclusion of R∀ and L∃

Clarifications: Sequents are based on multisets. A formula $A[x/t]$ is the result of uniformly substituting the term t for the variable x in A, renaming bound variables to prevent clashes with the variables in t.

Remarks: **G3c** is sound and complete w.r.t. classical first-order logic. Weakening and contraction are depth-preserving admissible and all rules are depth-preserving invertible.

[1] Anne Sjerp Troelstra and Helmut Schwichtenberg. *Basic Proof Theory*. 2nd ed. Vol. 43. Cambridge Tracts In Theoretical Computer Science. Cambridge University Press, 2000.

Entry 11 by: Björn Lellmann

$\mathbf{LK}_{\mu\tilde{\mu}}$ (2000)

STRUCTURAL SUBSYSTEM

$$\frac{(a:A)\in\Gamma}{\Gamma\vdash a:A\mid\Delta}\;Ax_R \qquad \frac{\Gamma\vdash v:A\mid\Delta \quad \Gamma\mid e:A\vdash\Delta}{\langle v|e\rangle:(\Gamma\vdash\Delta)}\;Cut \qquad \frac{(\alpha:A)\in\Delta}{\Gamma\mid\alpha:A\vdash\Delta}\;Ax_L$$

$$\frac{c:(\Gamma,a:A\vdash\Delta)}{\Gamma\mid\tilde{\mu}a.c:A\vdash\Delta}\;Focus_L \qquad \frac{c:(\Gamma\vdash\alpha:A,\Delta)}{\Gamma\vdash\mu\alpha.c:A\mid\Delta}\;Focus_R$$

INTRODUCTION RULES

$$\frac{\Gamma\mid e:A_i\vdash\Delta}{\Gamma\mid\pi_i\cdot e:A_1\wedge A_2\vdash\Delta}\;\wedge^i_L \qquad \frac{\Gamma\vdash v_1:A_1\mid\Delta \quad \Gamma\vdash v_2:A_2\mid\Delta}{\Gamma\vdash\ \mid(v_1,v_2):A_1\wedge A_2\mid\Delta}\;\wedge_R$$

$$\frac{\Gamma\mid e_1:A_1\vdash\Delta \quad \Gamma\mid e_2:A_2\vdash\Delta}{\Gamma\mid[e_1,e_2]:A_1\vee A_2\vdash\Delta}\;\vee_L \qquad \frac{\Gamma\vdash v:A_i\mid\Delta}{\Gamma\vdash\iota_i(v):A_1\vee A_2\mid\Delta}\;\vee^i_R$$

$$\frac{\Gamma\vdash v:A\mid\Delta \quad \Gamma\mid e:B\vdash\Delta}{\Gamma\mid v\cdot e:A\to B\vdash\Delta}\;\to_L \qquad \frac{\Gamma,a:A\vdash v:B\mid\Delta}{\Gamma\vdash\lambda a.v:A\to B\mid\Delta}\;\to_R$$

$$\frac{\Gamma\mid e:A[y]\vdash\Delta}{\Gamma\mid\tilde{\lambda}x.e:\exists xA[x]\vdash\Delta}\;\exists_L \qquad \frac{\Gamma\vdash v:A[t]\mid\Delta}{\Gamma\vdash t\cdot v:\exists xA[x]\mid\Delta}\;\exists_R$$

$$\frac{\Gamma\vdash e:A[t]\mid\Delta}{\Gamma\mid t\cdot e:\forall xA[x]\vdash\Delta}\;\forall_L \qquad \frac{\Gamma\vdash v:A[y]\mid\Delta}{\Gamma\vdash\lambda x.v:\forall xA[x]\mid\Delta}\;\forall_R$$

$$\frac{}{\Gamma\mid[\,]:\bot\vdash\Delta}\;\bot_L \qquad \frac{}{\Gamma\vdash():\top\mid\Delta}\;\top_R$$

Clarifications: There are three kinds of sequents: first $\Gamma\vdash v:A\mid\Delta$ with a distinguished formula on the right for typing the program v, second $\Gamma\mid e:A\vdash\Delta$ with a distinguished formula on the left for typing the evaluation context e, and finally $c:(\Gamma\vdash\Delta)$ with no distinguished formula for typing command c, i.e. the interaction of a program within an evaluation context. The typing contexts Γ and Δ are lists of named formulas so that a non-ambiguous correspondence with λ-calculus is possible (if it were sets or multisets, there were e.g. no way to distinguish the two distinct proofs of $x:A,x:A\vdash x:A\mid$). Weakening rules are implemented implicitly at the level of axioms. Contraction rules are derived, using a cut against an axiom. No exchange rule is needed. Not all cuts are eliminable: only those not involving an axiom rule are. Negation $\neg A$ can be defined as $A\to\bot$. In the rules $\exists_E$ and $\forall_R$, y is assumed fresh in Γ, Δ and $A[x]$. The syntax of the underlying λ-calculus is:

$$\begin{aligned}
c &::= \langle v|e\rangle\\
e &::= \alpha\mid\tilde{\mu}a.c\mid\pi_i\cdot e\mid[e,e]\mid v\cdot e\mid(t,e)\mid\tilde{\lambda}x.e\mid[\,]\\
v &::= a\mid\mu\alpha.c\mid(v,v)\mid\iota_i(v)\mid\lambda a.v\mid\lambda x.v\mid(t,v)\mid()
\end{aligned}$$

Entry 12 by: Hugo Herbelin

History: The purpose of this system is to provide with a λ-calculus-style computational meaning to Gentzen's LK {2} and to highlight how the symmetries of sequent calculus show computationally. Seeing the rules as typing rules, the left/right symmetry is a symmetry between programs and their evaluation contexts. At the level of cut elimination, giving priority to the left-hand side relates to call-by-name evaluation while giving priority to the right-hand side relates to call-by-value evaluation [8]. Thanks to the presence of two dual axiom rules and implicit contraction rules, the system supports a tree-like sequent-free presentation like originally presented by Gentzen for natural deduction [5] (see {13}).

The structural subsystem can be adapted to various sequent calculi, such as $\mathbf{LK}_{pol}$ [3], and in particular **LKT** (emphasizing call-by-name) and **LKQ** (emphasizing call-by-value) [11, 10]. Restriction to intuitionistic logic can be obtained by demanding that the right-hand side has exactly one formula.

The presentation of this calculus with conjunctive and disjunctive additive connectives has been studied in [7, 6, 5]. A variant with only commands, called $\mathcal{X}$, has been studied in [4], based on previous work in [9]. Various extensions of the system emphasizing different symmetries can be found in [1] (general connectives), recursion/corecursion [2], ... An asymmetric variant of **LKT** with sequents of the form $\Gamma \mid A \vdash B \mid \Delta$, $\Gamma \vdash A \mid \Delta$ and $\Gamma \vdash \Delta$ can be found in [13], with the intuitionistic restriction **LJT** studied in [12].

Remarks: The system is obviously logically equivalent to Gentzen's **LK** when equipped with the corresponding connectives and observed through the sequents of the form $\Gamma \vdash \Delta$.

---

[1] Paul Downen and Zena M. Ariola. "The Duality of Construction". In: *Programming Languages and Systems - 23rd European Symposium on Programming, ESOP 2014, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2014, Grenoble, France, April 5-13, 2014, Proceedings*. Ed. by Zhong Shao. Vol. 8410. Lecture Notes in Computer Science. Springer, 2014, pp. 249–269.

[2] Daisuke Kimura and Makoto Tatsuta. "Call-by-Value and Call-by-Name Dual Calculi with Inductive and Coinductive Types". In: *Logical Methods in Computer Science* 9.1 (2013). DOI: `10.2168/LMCS-9(1:14)2013`. URL: `http://dx.doi.org/10.2168/LMCS-9(1:14)2013`.

[3] Guillaume Munch-Maccagnoni. "Focalisation and Classical Realisability". In: *Computer Science Logic, 23rd international Workshop, CSL 2009, 18th Annual Conference of the EACSL, Coimbra, Portugal, September 7-11, 2009. Proceedings*. Ed. by Erich Grädel and Reinhard Kahle. Vol. 5771. Lecture Notes in Computer Science. Springer, 2009, pp. 409–423. DOI: `10.1007/978-3-642-04027-6_30`. URL: `http://dx.doi.org/10.1007/978-3-642-04027-6_30`.

[4] Steffen van Bakel, Stephane Lengrand, and Pierre Lescanne. "The Language X: Circuits, Computations and Classical Logic". In: *Theoretical Computer Science, 9th Italian Conference, ICTCS 2005, Siena, Italy, October 12-14,*

2005, Proceedings. Ed. by Mario Coppo, Elena Lodi, and G. Michele Pinna. Vol. 3701. Springer, 2005, pp. 81–96. ISBN: 3-540-29106-7.

[5] Hugo Herbelin. "C'est maintenant qu'on calcule: au cœur de la dualité". Habilitation thesis. University Paris 11, Dec. 2005.

[6] Philip Wadler. "Call-by-Value Is Dual to Call-by-Name - Reloaded". In: *Term Rewriting and Applications, 16th International Conference, RTA 2005, Nara, Japan, April 19-21, 2005, Proceedings*. Ed. by Jürgen Giesl. Vol. 3467. Lecture Notes in Computer Science. Springer, 2005, pp. 185–203. ISBN: 3-540-25596-6.

[7] Philip Wadler. "Call-by-value is dual to call-by-name". In: *Proceedings of the Eighth ACM SIGPLAN International Conference on Functional Programming, ICFP 2003, Uppsala, Sweden, August 25-29, 2003*. Ed. by Colin Runciman and Olin Shivers. Vol. 38(9). SIGPLAN Notices. ACM, 2003, pp. 189–201. ISBN: 1-58113-756-7.

[8] Pierre-Louis Curien and Hugo Herbelin. "The duality of computation". In: *Proceedings of the Fifth ACM SIGPLAN International Conference on Functional Programming, ICFP 2000, Montreal, Canada, September 18-21, 2000*. SIGPLAN Notices 35(9). ACM, 2000, pp. 233–243. ISBN: 1-58113-202-6.

[9] Christian Urban. "Classical Logic and Computation". Ph.D. Thesis. University of Cambridge, Oct. 2000.

[10] Vincent Danos, Jean-Baptiste Joinet, and Harold Schellinx. "A new deconstructive logic: Linear Logic". In: 62.3 (1997), pp. 755–807.

[11] Vincent Danos, Jean-Baptiste Joinet, and Harold Schellinx. "LKQ and LKT: sequent calculi for second order logic based upon dual linear decompositions of the classical implication". In: *Advances in Linear Logic*. Vol. 222. Cambridge University Press, 1995, pp. 211–224.

[12] Hugo Herbelin. "A Lambda-Calculus Structure Isomorphic to Gentzen-Style Sequent Calculus Structure". In: *Computer Science Logic, 8th International Workshop, CSL '94, Kazimierz, Poland, September 25-30, 1994, Selected Papers*. Ed. by Leszek Pacholski and Jerzy Tiuryn. Vol. 933. Springer, 1995, pp. 61–75. ISBN: 3-540-60017-5.

[13] Hugo Herbelin. "Séquents qu'on calcule: de l'interprétation du calcul des séquents comme calcul de λ-termes et comme calcul de stratégies gagnantes". Ph.D. thesis. University Paris 7, Jan. 1995.

LK$\mu\tilde{\mu}$ in sequent-free tree form (2005)

STRUCTURAL SUBSYSTEM

$$\frac{\vdash A \quad A \vdash}{\vdash}\ Cut$$

$$\frac{\begin{array}{c}[\vdash A]\\ \vdots \\ \vdash\end{array}}{A \vdash}\ Focus_L \qquad \frac{\begin{array}{c}[A \vdash]\\ \vdots \\ \vdash\end{array}}{\vdash A}\ Focus_R$$

INTRODUCTION RULES

$$\frac{A_i \vdash}{A_1 \wedge A_2 \vdash}\ \wedge_L^i \qquad \frac{\vdash A_1 \quad \vdash A_2}{\vdash A_1 \wedge A_2}\ \wedge_R$$

$$\frac{A_1 \vdash \quad A_2 \vdash}{A_1 \vee A_2 \vdash}\ \vee_L \qquad \frac{\vdash A_i}{\vdash A_1 \vee A_2}\ \vee_R^i$$

$$\frac{\vdash A \quad B \vdash}{A \to B \vdash}\ \to_L \qquad \frac{\begin{array}{c}[\vdash A]\\ \vdots \\ \vdash B\end{array}}{\vdash A \to B}\ \to_R$$

$$\frac{A[y] \vdash}{\exists x A[x] \vdash}\ \exists_L \qquad \frac{\vdash A[t]}{\vdash \exists x A[x]}\ \exists_R$$

$$\frac{A[t] \vdash}{\forall x A[x] \vdash}\ \forall_L \qquad \frac{\vdash A[y]}{\vdash \forall x A[x]}\ \forall_R$$

$$\frac{}{\bot \vdash}\ \bot_L \qquad \frac{}{\vdash \top}\ \top_R$$

Clarifications: There are three kinds of nodes, $\vdash A$ for asserting formulas, $A \vdash$ for refuting formulas, and $\vdash$ for expressing a contradiction. Annotation by proof-terms can optionally be added as in {12}. Negation $\neg A$ can be defined as $A \to \bot$.

History: The purpose of this system is to show that the original distinction in Gentzen [6] between natural deduction presented as a tree of formulas and sequent calculus presented as a tree of sequents is no longer relevant. It is known from at least Howard [3] that natural deduction can be presented with sequents. The above formulation shows that systems based on left and right introductions ("sequent-calculus style") can be presented as a sequent-free tree of formulas [2].

The terminology "sequent calculus" seems to have become popular from [5] followed then e.g. by [4] who were associating the term "sequents" to Gentzen's LJ and LK systems. The terminology having lost the connection to its etymology, this motivated some authors to use alternative terminologies such as "L" systems [1].

Entry 13 by: Hugo Herbelin

[1] Guillaume Munch-Maccagnoni. “Focalisation and Classical Realisability”. In: *Computer Science Logic, 23rd international Workshop, CSL 2009, 18th Annual Conference of the EACSL, Coimbra, Portugal, September 7-11, 2009. Proceedings*. Ed. by Erich Grädel and Reinhard Kahle. Vol. 5771. Lecture Notes in Computer Science. Springer, 2009, pp. 409–423. DOI: 10.1007/978-3-642-04027-6_30. URL: http://dx.doi.org/10.1007/978-3-642-04027-6_30.

[2] Hugo Herbelin. “C’est maintenant qu’on calcule: au cœur de la dualité”. Habilitation thesis. University Paris 11, Dec. 2005.

[3] William A. Howard. “The formulae-as-types notion of constructions”. In: *to H.B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*. Unpublished manuscript of 1969. Academic Press, 1980.

[4] Anne S. Troelstra. *Metamathematical Investigation of Intuitionistic Arithmetic and Analysis*. Vol. 344. Lecture Notes in Mathematics. Berlin: Springer-Verlag, 1973.

[5] Dag Prawitz. *Natural Deduction, a Proof-Theoretical Study*. Almqvist and Wiksell, Stockholm, 1965.

[6] Gerhard Gentzen. “Untersuchungen über das logische Schließen”. In: *Mathematische Zeitschrift* 39 (1935). English Translation in [**Szabo69**], “Investigations into logical deduction”, pages 68-131, pp. 176–210, 405–431.

HO Sequent Calculi $\mathcal{G}_\beta$ and $\mathcal{G}_{\beta\mathfrak{fb}}$ (2003-2009)

Basic Rules

$$\frac{\Delta, s}{\Delta, \neg\neg s}\,\mathcal{G}(\neg) \qquad \frac{\Delta, \neg s \quad \Delta, \neg t}{\Delta, \neg(s \vee t)}\,\mathcal{G}(\vee_-) \qquad \frac{\Delta, s, t}{\Delta, (s \vee t)}\,\mathcal{G}(\vee_+)$$

$$\frac{\Delta, \neg(sl){\downarrow_\beta} \quad l_\alpha \text{ closed term}}{\Delta, \neg\Pi^\alpha s}\,\mathcal{G}(\Pi^l_-) \qquad \frac{\Delta, (sc){\downarrow_\beta} \quad c_\delta \text{ new symbol}}{\Delta, \Pi^\alpha s}\,\mathcal{G}(\Pi^c_+)$$

Initialization

$$\frac{s \text{ atomic (and } \beta\text{-normal)}}{\Delta, s, \neg s}\,\mathcal{G}(init) \qquad \frac{\Delta, (s \doteq^o t) \quad s,t \text{ atomic}}{\Delta, \neg s, t}\,\mathcal{G}(Init^{\doteq})$$

Extensionality

$$\frac{\Delta, (\forall X_\alpha sX \doteq^\beta tX){\downarrow_\beta}}{\Delta, (s \doteq^{\alpha\to\beta} t)}\,\mathcal{G}(\mathfrak{f}) \qquad \frac{\Delta, \neg s, t \quad \Delta, \neg t, s}{\Delta, (s \doteq^o t)}\,\mathcal{G}(\mathfrak{b})$$

Decomposition

$$\frac{\Delta, (s^1 \doteq^{\alpha_1} t^1) \ \cdots \ \Delta, (s^n \doteq^{\alpha_n} t^n) \quad n \geq 1, \beta \in \{o, \iota\}, h_{\overline{\alpha^n}\to\beta} \in \Sigma}{\Delta, (h\overline{s^n} \doteq^\beta h\overline{t^n})}\,\mathcal{G}(d)$$

One-sided sequent calculus $\mathcal{G}_\beta$ is defined by the rules $\mathcal{G}(init)$, $\mathcal{G}(\neg)$, $\mathcal{G}(\vee_-)$, $\mathcal{G}(\vee_+)$, $\mathcal{G}(\Pi^l_-)$ and $\mathcal{G}(\Pi^c_+)$.
Calculus $\mathcal{G}_{\beta\mathfrak{fb}}$ extends $\mathcal{G}_\beta$ by the additional rules $\mathcal{G}(\mathfrak{b})$, $\mathcal{G}(\mathfrak{f})$, $\mathcal{G}(d)$, and $\mathcal{G}(Init^{\doteq})$.

Clarifications: Δ and Δ' are finite sets of β-normal closed formulas of classical higher-order logic (HOL; Church's Type Theory) [1]. Δ, s denotes the set $\Delta \cup \{s\}$. Let $\alpha, \beta, o \in T$. HOL *terms* are defined by the grammar (c_α denotes typed constants and X_α typed variables distinct from c_α): $s,t ::= c_\alpha \mid X_\alpha \mid (\lambda X_\alpha s_\beta)_{\alpha\to\beta} \mid (s_{\alpha\to\beta}\, t_\alpha)_\beta \mid (\neg_{o\to o}\; s_o)_o \mid (s_o \vee_{o\to o\to o} t_o)_o \mid (\Pi_{(\alpha\to o)\to o}\; s_{\alpha\to o})_o$. *Leibniz equality* $\doteq^\alpha$ at type α is defined as $s_\alpha \doteq^\alpha t_\alpha := \forall P_{\alpha\to o}(\neg Ps \vee Pt)$. For each simply typed λ-term s there is a unique *β-normal form* (denoted $s{\downarrow_\beta}$). HOL formulas are defined as terms of type o. A *non-atomic formula* is any formula whose β-normal form is of the form $[c\overline{\mathbf{A}^n}]$ where c is a logical constant. An *atomic formula* is any other formula.

Theorem proving in these calculi works as follows: In order to prove that a (closed) conjecture formula c logically follows from a (possibly empty) set of (closed) axioms $\{a^1, \ldots, a^n\}$, we start from the initial sequent $\Delta := \{c, \neg a^1, \ldots, \neg a^n\}$ and reason backwards by applying the respective calculus rules.

History: The calculi have been presented in [2]. Earlier (two-sided) versions and further related sequent calculi for HOL have been presented in [4] and [3].

Remarks: $\mathcal{G}_\beta$ is sound and complete for elementary type theory ($\mathcal{G}_\beta$ is thus also sound for HOL). $\mathcal{G}_{\beta\mathfrak{fb}}$ is sound and complete for HOL. Moreover, both calculi are cut-free and they do not admit cut-simulation [2].

Entry 14 by: Christoph Benzmüller

[1] Peter Andrews. “Church’s Type Theory”. In: *The Stanford Encyclopedia of Philosophy*. Ed. by Edward N. Zalta. Spring 2014. 2014.

[2] Christoph Benzmüller, Chad Brown, and Michael Kohlhase. “Cut-Simulation and Impredicativity”. In: *Logical Methods in Computer Science* 5.1:6 (2009), pp. 1–21.

[3] Chad E. Brown. “Set Comprehension in Church’s Type Theory”. See also Chad E. Brown, *Automated Reasoning in Higher-Order Logic*, College Publications, 2007. PhD thesis. Department of Mathematical Sciences, Carnegie Mellon University, 2004.

[4] Christoph Benzmüller, Chad Brown, and Michael Kohlhase. *Semantic Techniques for Cut-Elimination in Higher Order Logic.* Tech. rep. Saarland University, Saarbrücken, Germany and Carnegie Mellon University, Pittsburgh, USA, 2003.

Extensional HO RUE-Resolution (1999-2013)

Normalisation Rules

$$\frac{\mathbf{C}\vee[\mathbf{A}\vee\mathbf{B}]^{\mathtt{tt}}}{\mathbf{C}\vee[\mathbf{A}]^{\mathtt{tt}}\vee[\mathbf{B}]^{\mathtt{tt}}}\,\vee^{\mathtt{tt}}\qquad \frac{\mathbf{C}\vee[\mathbf{A}\vee\mathbf{B}]^{\mathtt{ff}}}{\begin{array}{c}\mathbf{C}\vee[\mathbf{A}]^{\mathtt{ff}}\\ \mathbf{C}\vee[\mathbf{B}]^{\mathtt{ff}}\end{array}}\,\vee^{\mathtt{ff}}\qquad \frac{\mathbf{C}\vee[\neg\mathbf{A}]^{\mathtt{tt}}}{\mathbf{C}\vee[\mathbf{A}]^{\mathtt{ff}}}\,\neg^{\mathtt{tt}}\qquad \frac{\mathbf{C}\vee[\neg\mathbf{A}]^{\mathtt{ff}}}{\mathbf{C}\vee[\mathbf{A}]^{\mathtt{tt}}}\,\neg^{\mathtt{ff}}$$

$$\frac{\mathbf{C}\vee[\Pi^{\tau}\mathbf{A}]^{\mathtt{tt}}\quad X^{\tau}\text{ fresh variable}}{\mathbf{C}\vee[\mathbf{A}\,X]^{\mathtt{tt}}}\,\Pi^{\mathtt{tt}}\qquad \frac{\mathbf{C}\vee[\Pi^{\tau}\mathbf{A}]^{\mathtt{ff}}\quad \mathtt{sk}^{\tau}\text{ Skolem term}}{\mathbf{C}\vee[\mathbf{A}\,\mathtt{sk}^{\tau}]^{\mathtt{ff}}}\,\Pi^{\mathtt{ff}}$$

Resolution, Factorisation and Primitive Substitution

$$\frac{[\mathbf{A}]^{p_1}\vee\mathbf{C}\quad [\mathbf{B}]^{p_2}\vee\mathbf{D}\quad p_1\neq p_2}{\mathbf{C}\vee\mathbf{D}\vee[\mathbf{A}=\mathbf{B}]^{\mathtt{ff}}}\,\mathsf{res}\qquad \frac{\mathbf{C}\vee[\mathbf{A}]^{p}\vee[\mathbf{B}]^{p}}{\mathbf{C}\vee[\mathbf{A}]^{p}\vee[\mathbf{A}=\mathbf{B}]^{\mathtt{ff}}}\,\mathsf{fac}$$

$$\frac{[Q_{\tau}\overline{\mathbf{A}}^{n}]^{p}\vee\mathbf{C}\quad \mathbf{P}\in\mathcal{AB}_{\tau}^{(k)}\text{ for logic connective }k}{([Q_{\tau}\overline{\mathbf{A}}^{n}]^{p}\vee\mathbf{C})[\mathbf{P}/Q]}\,\mathsf{prim\_subst}$$

Extensionality and Pre-unification

$$\frac{\mathbf{C}\vee[\mathbf{A}^{\sigma\tau}=\mathbf{B}^{\sigma\tau}]^{\mathtt{tt}}\quad X^{\tau}\text{ fresh variable}}{\mathbf{C}\vee[\mathbf{A}\,X=\mathbf{B}\,X]^{\mathtt{tt}}}\,\textsc{FuncPos}\qquad \frac{\mathbf{C}\vee[\mathbf{A}^{o}=\mathbf{B}^{o}]^{\mathtt{tt}}}{\mathbf{C}\vee[\mathbf{A}^{o}\longleftrightarrow\mathbf{B}^{o}]^{\mathtt{tt}}}\,\textsc{BoolPos}$$

$$\frac{\mathbf{C}\vee[\mathbf{A}^{\sigma\tau}=\mathbf{B}^{\sigma\tau}]^{\mathtt{ff}}\quad \mathtt{sk}^{\tau}\text{ Skol. term}}{\mathbf{C}\vee[\mathbf{A}\,\mathtt{sk}=\mathbf{B}\,\mathtt{sk}]^{\mathtt{ff}}}\,\textsc{FuncNeg}\qquad \frac{\mathbf{C}\vee[\mathbf{A}^{o}=\mathbf{B}^{o}]^{\mathtt{ff}}}{\mathbf{C}\vee[\mathbf{A}^{o}\longleftrightarrow\mathbf{B}^{o}]^{\mathtt{ff}}}\,\textsc{BoolNeg}$$

$$\frac{\mathbf{C}\vee[h^{\sigma\tau}\overline{\mathbf{A}^{\alpha}}^{k}\ =\ h^{\sigma\tau}\overline{\mathbf{B}^{\alpha}}^{k}]^{\mathtt{ff}}}{\mathbf{C}\vee\overline{[\mathbf{A}_i\ =\ \mathbf{B}_i]^{\mathtt{ff}}}^{i\leq k}}\,\textsc{Dec}\qquad \frac{\mathbf{C}\vee[X=\mathbf{A}]^{\mathtt{ff}}\quad X\notin\mathrm{FV}(\mathbf{A})}{\mathbf{C}[\mathbf{A}/X]}\,\textsc{Subst}$$

$$\frac{\mathbf{C}\vee[\mathbf{A}=\mathbf{A}]^{\mathtt{ff}}}{\mathbf{C}}\,\textsc{Triv}\qquad \frac{\mathbf{C}\vee[F^{\tau}\overline{\mathbf{A}}^{n}\ =\ h\overline{\mathbf{B}}^{m}]^{\mathtt{ff}}\quad \mathbf{G}\in\mathcal{AB}_{\tau}^{(h)}}{\mathbf{C}\vee[F=\mathbf{G}]^{\mathtt{ff}}\ \vee\ [F\overline{\mathbf{A}}^{n}=h\overline{\mathbf{B}}^{m}]^{\mathtt{ff}}}\,\textsc{FlexRigid}$$

Choice

$$\frac{C:=\mathbf{C}'\vee[\mathbf{A}[E_{(\alpha\to o)\to\alpha}\mathbf{B}]]^{p}\quad \begin{array}{l}\epsilon\in\mathsf{CFs},\ E=\epsilon\ \mathit{or}\ E\in\mathit{freeVars(C)},\\ \mathit{freeVars}(\mathbf{B})\subseteq\mathit{freeVars}(C),Y\mathit{fresh}\end{array}}{[\mathbf{B}\ Y]^{\mathtt{ff}}\vee[\mathbf{B}\ (\epsilon_{\alpha(\alpha o)}\mathbf{B})]^{\mathtt{tt}}}\,\mathsf{choice}$$

$$\frac{[PX]^{\mathtt{ff}}\vee[P(f_{(\alpha\to o)\to\alpha}P)]^{\mathtt{tt}}}{\mathsf{CFs}\longleftarrow\mathsf{CFs}\cup\{f_{(\alpha\to o)\to\alpha}\}}\,\mathsf{detectChoiceFn}$$

Optional additional rules include (a) exhaustive universal instantion rule for (selective) finite domains, (b) detection and removal of Leibniz equations and Andrews equations, and (c) splitting. Like detectChoiceFn these rules are admissible.

Clarifications: $\mathbf{A}$ and $\mathbf{B}$ are metavariables ranging over terms of HOL [1]; see also {14}). The logical connectives are $\neg$, $\vee$, Π^{τ} (universal quantification over variables of type τ), and $=^{\tau}$ (equality on terms of type τ). Types are shown only if unclear in context. For example, in rule choice the variable $E^{\alpha(\alpha o)}$ is of function type, also

Entry 15 by: Christoph Benzmüller

written as $(\alpha \to o) \to \alpha$. Variables like F are presented as upper case symbols and constant symbols like h are lower case. α equality and $\beta\eta$-normalisation are treated implicit, meaning that all clauses are implicitly normalised. $\mathbf{C}$ and $\mathbf{D}$ are metavariables ranging over clauses, which are disjunctions of literals. These disjunctions are implicitly assumed associative and commutative; the latter also applies to all equations. Literals are formulas shown in square brackets and labelled with a *polarity* (either $\mathtt{tt}$ or $\mathtt{ff}$), e.g. $[\neg X]^{\mathtt{ff}}$ denotes the negation of $\neg X$. $\mathrm{FV}(\mathbf{A})$ denotes the free variables of term $\mathbf{A}$. $\mathcal{AB}_{\tau}^{(h)}$ is the set of approximating bindings for head h and type τ. $\epsilon_{\alpha(\alpha o)}$ is a choice operator and CFs is a set of dynamically collected choice functions symbols; CFs is initialised with a single choice function.

History: The original calculus (without choice) has been presented in [5] and [4]. Recent modifications and extensions (e.g. choice) are discussed in [3] and [2]. The calculus is inspired by and extends Huet's constrained resolution [7, 8] and the extensional resolution calculus in [6].

Remarks: The calculus works for classical higher-order logic with Henkin semantics and choice. Soundness and completeness has been discussed in [5] and [4]. In the prover LEO-II, the factorisation rule is for performance reasons restricted to binary clauses and a (parametrisable) depth limit is employed for pre-unification. Such restrictions are a (deliberate) source for incompleteness.

---

[1] Peter Andrews. "Church's Type Theory". In: *The Stanford Encyclopedia of Philosophy*. Ed. by Edward N. Zalta. Spring 2014. 2014.

[2] Christoph Benzmüller and Nik Sultana. "LEO-II Version 1.5". In: *PxTP 2013*. Vol. 14. EPiC Series. EasyChair, 2013, pp. 2–10.

[3] Nik Sultana and Christoph Benzmüller. "Understanding LEO-II's Proofs". In: *IWIL 2012*. Vol. 22. EPiC Series. EasyChair, 2013, pp. 33–52.

[4] Christoph Benzmüller. "Comparing Approaches to Resolution based Higher-Order Theorem Proving". In: *Synthese* 133.1-2 (2002), pp. 203–235.

[5] Christoph Benzmüller. "Extensional Higher-Order Paramodulation and RUE-Resolution". In: *Automated Deduction - CADE-16*. LNCS 1632. Springer, 1999, pp. 399–413.

[6] Christoph Benzmüller and Michael Kohlhase. "Extensional Higher-Order Resolution". In: *Automated Deduction - CADE-15*. LNAI 1421. Springer, 1998, pp. 56–71.

[7] Gérard P. Huet. "A Mechanization of Type Theory". In: *Proceedings of the 3rd International Joint Conference on Artificial Intelligence*. 1973, pp. 139–146.

[8] Gérard P. Huet. "Constrained Resolution: A Complete Method for Higher Order Logic". PhD thesis. Case Western Reserve University, 1972.

Focused LK (2007)

ASYNCHRONOUS INTRODUCTION RULES

$$\frac{}{\vdash \Gamma \Uparrow t^-, \Theta} \qquad \frac{\vdash \Gamma \Uparrow B_1, \Theta \quad \vdash \Gamma \Uparrow B_2, \Theta}{\vdash \Gamma \Uparrow B_1 \wedge^- B_2, \Theta} \qquad \frac{\vdash \Gamma \Uparrow \Theta}{\vdash \Gamma \Uparrow f^-, \Theta} \qquad \frac{\vdash \Gamma \Uparrow B_1, B_2, \Theta}{\vdash \Gamma \Uparrow B_1 \vee^- B_1, \Theta}$$

$$\frac{\vdash \Gamma \Uparrow [y/x]B, \Theta}{\vdash \Gamma \Uparrow \forall x.B, \Theta}$$

SYNCHRONOUS INTRODUCTION RULES

$$\frac{}{\vdash \Gamma \Downarrow t^+} \qquad \frac{\vdash \Gamma \Downarrow B_1 \quad \vdash \Gamma \Downarrow B_2}{\vdash \Gamma \Downarrow B_1 \wedge^+ B_2} \qquad \frac{\vdash \Gamma \Downarrow B_i}{\vdash \Gamma \Downarrow B_1 \vee^+ B_2}\ i \in \{1,2\} \qquad \frac{\vdash \Gamma \Downarrow [t/x]B}{\vdash \Gamma \Downarrow \exists x.B}$$

IDENTITY RULES

$$\frac{P \text{ atomic}}{\vdash \neg P, \Gamma \Downarrow P}\ init \qquad \frac{\vdash \Gamma \Uparrow B \quad \vdash \Gamma \Uparrow \neg B}{\vdash \Gamma \Uparrow \cdot}\ cut$$

STRUCTURAL RULES

$$\frac{\vdash \Gamma, C \Uparrow \Theta}{\vdash \Gamma \Uparrow C, \Theta}\ store \qquad \frac{\vdash \Gamma \Uparrow N}{\vdash \Gamma \Downarrow N}\ release \qquad \frac{\vdash P, \Gamma \Downarrow P}{\vdash P, \Gamma \Uparrow \cdot}\ decide$$

Here, Γ ranges over multisets of polarized formulas; Θ ranges over lists of polarized formulas; P denotes a positive formula; N denotes a negative formula; C denotes either a negative formula or a positive atom; and B denotes an unrestricted polarized formula. The negation in $\neg B$ denotes the negation normal form of the de Morgan dual of B. The right introduction rule for $\forall$ has the the usual eigenvariable restriction that y is not free in any formula in the conclusion sequent.

Clarifications: This proof system involves *polarized* (negative normal) formulas of first-order classical logic: in order to polarize a formula B, one must assign the status of "positive" or "negative" bias to all atomic formulas and replace all occurrences of truth with either t^+ or t^- and replace all occurrences of conjunctions with either $\wedge^+$ or $\wedge^-$; similarly, all occurrences of false and disjunctions must be polarized into f^+, f^-, $\vee^+$, and $\vee^-$. If there are n occurrences of propositional connectives in B, there are 2^n ways to polarize B. The *positive connectives* are f^+, $\vee^+$, t^+, $\wedge^+$, and $\exists$ while the *negative connectives* are t^-, $\wedge^-$, f^-, $\vee^-$, and $\forall$. A formula is *positive* it is a positive atom or has a top-level positive connective; similarly a formula is *negative* if it is a negative atom or has a top-level negative connective.

There are two kinds of sequents in this proof system, namely, $\vdash \Gamma \Uparrow \Theta$ and $\vdash \Gamma \Downarrow B$, where Γ is a multiset of polarized formulas, B is a polarized formula, and Θ is a list of polarized formulas. The list structure of Θ can be replaced by a multiset.

Entry 16 by: Dale Miller

History: This focused proof system is a slight variation of the proof systems in [2, 3]. A multifocus variant of **LKF** has been described in [1]. The design of **LKF** borrows strongly by Andreoli's focused proof system for linear logic [5] and Girard's LC proof system [6]. The first-order versions of the LKT and LKQ proof systems of [4] can be seen subsystems of **LKF**.

---

[1] Kaustuv Chaudhuri, Stefan Hetzl, and Dale Miller. "A Multi-Focused Proof System Isomorphic to Expansion Proofs". In: *J. of Logic and Computation* (June 2014).

[2] Chuck Liang and Dale Miller. "Focusing and Polarization in Linear, Intuitionistic, and Classical Logics". In: *Theoretical Computer Science* 410.46 (2009), pp. 4747–4768.

[3] Chuck Liang and Dale Miller. "Focusing and Polarization in Intuitionistic Logic". In: *CSL 2007: Computer Science Logic*. Ed. by J. Duparc and T. A. Henzinger. Vol. 4646. LNCS. Springer, 2007, pp. 451–465.

[4] V. Danos, J.-B. Joinet, and H. Schellinx. "LKT and LKQ: sequent calculi for second order logic based upon dual linear decompositions of classical implication". In: *Advances in Linear Logic*. Ed. by J.-Y. Girard, Y. Lafont, and L. Regnier. London Mathematical Society Lecture Note Series 222. Cambridge University Press, 1995, pp. 211–224.

[5] Jean-Marc Andreoli. "Logic Programming with Focusing Proofs in Linear Logic". In: 2.3 (1992), pp. 297–347.

[6] Jean-Yves Girard. "A new constructive logic: classical logic". In: *Math. Structures in Comp. Science* 1 (1991), pp. 255–296.

Focused LJ (2007)

Asynchronous Introduction Rules

$$\frac{\Gamma \Uparrow B_1 \vdash B_2 \Uparrow}{\Gamma \Uparrow \cdot \vdash B_1 \supset B_2 \Uparrow} \qquad \frac{\Gamma \Uparrow \cdot \vdash B_1 \Uparrow \quad \Gamma \Uparrow \cdot \vdash B_2 \Uparrow}{\Gamma \Uparrow \cdot \vdash B_1 \wedge^- B_2 \Uparrow} \qquad \frac{}{\Gamma \Uparrow \cdot \vdash t^- \Uparrow}$$

$$\frac{\Gamma \Uparrow \cdot \vdash [y/x]B \Uparrow}{\Gamma \Uparrow \cdot \vdash \forall x.B \Uparrow} \qquad \frac{\Gamma \Uparrow [y/x]B, \Theta \vdash \mathcal{R}}{\Gamma \Uparrow \exists x.B, \Theta \vdash \mathcal{R}} \qquad \frac{}{\Gamma \Uparrow f^+, \Theta \vdash \mathcal{R}}$$

$$\frac{\Gamma \Uparrow B_1, B_2, \Theta \vdash \mathcal{R}}{\Gamma \Uparrow B_1 \wedge^+ B_2, \Theta \vdash \mathcal{R}} \qquad \frac{\Gamma \Uparrow \Theta \vdash \mathcal{R}}{\Gamma \Uparrow t^+, \Theta \vdash \mathcal{R}} \qquad \frac{\Gamma \Uparrow B_1, \Theta \vdash \mathcal{R} \quad \Gamma \Uparrow B_2, \Theta \vdash \mathcal{R}}{\Gamma \Uparrow B_1 \vee^+ B_2, \Theta \vdash \mathcal{R}}$$

Synchronous Introduction Rules

$$\frac{\Gamma \vdash B_1 \Downarrow \quad \Gamma \Downarrow B_2 \vdash E}{\Gamma \Downarrow B_1 \supset B_2 \vdash E} \qquad \frac{\Gamma \Downarrow [t/x]B \vdash E}{\Gamma \Downarrow \forall x.B \vdash E} \qquad \frac{\Gamma \Downarrow B_i \vdash E}{\Gamma \Downarrow B_1 \wedge^- B_2 \vdash E}\ i \in \{1,2\}$$

$$\frac{\Gamma \vdash B_i \Downarrow}{\Gamma \vdash B_1 \vee^+ B_2 \Downarrow} \qquad \frac{}{\Gamma \vdash t^+ \Downarrow} \qquad \frac{\Gamma \vdash B_1 \Downarrow \quad \Gamma \vdash B_2 \Downarrow}{\Gamma \vdash B_1 \wedge^+ B_2 \Downarrow} \qquad \frac{\Gamma \vdash [t/x]B \Downarrow}{\Gamma \vdash \exists x.B \Downarrow}$$

Identity rules

$$\frac{N \text{ atomic}}{\Gamma \Downarrow N \vdash N}\ I_l \qquad \frac{P \text{ atomic}}{\Gamma, P \vdash P \Downarrow}\ I_r \qquad \frac{\Gamma \Uparrow \cdot \vdash B \Uparrow \cdot \qquad \Gamma \Uparrow B \vdash \cdot \Uparrow E}{\Gamma \Uparrow \cdot \vdash \cdot \Uparrow E}\ Cut$$

Structural rules

$$\frac{\Gamma, N \Downarrow N \vdash E}{\Gamma, N \Uparrow \cdot \vdash \cdot \Uparrow E}\ D_l \qquad \frac{\Gamma \vdash P \Downarrow}{\Gamma \Uparrow \cdot \vdash \cdot \Uparrow P}\ D_r \qquad \frac{\Gamma \Uparrow P \vdash \cdot \Uparrow E}{\Gamma \Downarrow P \vdash E}\ R_l \qquad \frac{\Gamma \Uparrow \cdot \vdash N \Uparrow \cdot}{\Gamma \vdash N \Downarrow}\ R_r$$

$$\frac{C, \Gamma \Uparrow \Theta \vdash \mathcal{R}}{\Gamma \Uparrow C, \Theta \vdash \mathcal{R}}\ S_l \qquad \frac{\Gamma \Uparrow \cdot \vdash \cdot \Uparrow E}{\Gamma \Uparrow \cdot \vdash E \Uparrow \cdot}\ S_r$$

Here, Θ ranges over multisets of polarized formulas; Γ ranges over lists of polarized formulas; P denotes a positive formula; N denotes a negative formula; C denotes either a negative formula or a positive atom; and E denotes either a positive formula or a negative atom; and B denotes an unrestricted polarized formula. The introduction rule for $\forall$ has the usual eigenvariable restriction that y is not free in any formula in the conclusion sequent.

Clarifications: This proof system involves *polarized* formulas of first-order intuitionistic logic: in order to polarize a formula B, one must assign the status of "positive" or "negative" bias to all atomic formulas and replace all occurrences of truth with either t^+ or t^- and all occurrences of conjunction with either $\wedge^+$ or $\wedge^-$. If there are n occurrences of truth and conjunction in B, there are 2^n ways to do this replacement. Similarly, we replace the false and disjunction with f^+ and $\vee^+$ since only the

Entry 17 by: Dale Miller

positive polarization for these connectives are available in **LJF**. (Assigning polarization in classical logic is different: see the **LKF** proof system {16}.) The *positive connectives* are f^+, $\vee^+$, t^+, $\wedge^+$, and $\exists$ while the *negative connectives* are t^-, $\wedge^-$, $\supset$, and $\forall$. A formula is *positive* if it is a positive atom or has a top-level positive connective; similarly a formula is *negative* if it is a negative atom or has a top-level negative connective.

There are two kinds of sequents in this proof system. One kind contains a single $\Downarrow$ on either the right or the left of the turnstyle ($\vdash$) and are of the form $\Gamma \Downarrow B \vdash E$ or $\Gamma \vdash B \Downarrow$: in both of these cases, the formula B is the *focus* of the sequent. The other kind of sequent has an occurrence of $\Uparrow$ on each side of the turnstyle, eg., $\Gamma \Uparrow \Theta \vdash \Delta_1 \Uparrow \Delta_2$, and is such that the union of the two multisets Δ_1 and Δ_2 contains exactly one formula: that is, one of these multisets is empty and the other is a singleton. When writing asynchronous rules that introduce a connective on the left-hand side, we write $\mathcal{R}$ to denote $\Delta_1 \Downarrow \Delta_2$.

Note that in the asynchronous phase, a right introduction rule is applied only when the left asynchronous zone Γ is empty. Similarly, a left-introduction rule in the async phase introduces the connective at the top-level of the first formula in that context. The scheduling of introduction rules during this phase can be assigned arbitrarily and the zone Γ can be interpreted as a multiset instead of a list.

The choice of how to polarize an unpolarized formula does not affect provability in LJF but can make a big impact on the structure of LJF proofs that can be built.

History: This focused proof system is a slight variation of the proof system in [1, 2]. **LJF** can be seen as a generalization to the MJ sequent system of Howe [5]. Other focused proof systems, such as LJT [6], LJQ/LJQ' [3], and λRCC [4] can be directly emulated within **LJF** by making the appropriate choice of polarization.

---

[1] Chuck Liang and Dale Miller. "Focusing and Polarization in Linear, Intuitionistic, and Classical Logics". In: *Theoretical Computer Science* 410.46 (2009), pp. 4747–4768.

[2] Chuck Liang and Dale Miller. "Focusing and Polarization in Intuitionistic Logic". In: *CSL 2007: Computer Science Logic*. Ed. by J. Duparc and T. A. Henzinger. Vol. 4646. LNCS. Springer, 2007, pp. 451–465.

[3] R. Dyckhoff and S. Lengrand. "LJQ: a strongly focused calculus for intuitionistic logic". In: *Computability in Europe 2006*. Ed. by A. Beckmann and *et al.* Vol. 3988. Springer, 2006, pp. 173–185.

[4] Radha Jagadeesan, Gopalan Nadathur, and Vijay Saraswat. "Testing concurrent systems: An interpretation of intuitionistic logic". In: *FSTTCS05*. Vol. 3821. LNCS. Hyderabad, India: Springer, 2005, pp. 517–528.

[5] J. M. Howe. "Proof Search Issues in Some Non-Classical Logics". Available as University of St Andrews Research Report CS/99/1. PhD thesis. University of St Andrews, Dec. 1998.

[6] Hugo Herbelin. "Séquents qu'on calcule: de l'interprétation du calcul des séquents comme calcul de lambda-termes et comme calcul de stratégies gagnantes". PhD thesis. Université Paris 7, 1995.

Counterfactual Sequent Calculi I (1983,1992,2012,2013)

$$\frac{\begin{array}{c}\{\, B_k \vdash A_1,\dots,A_n,D_1,\dots,D_m \mid k \le n \,\}\\ \cup\, \{\, C_k \vdash A_1,\dots,A_n,D_1,\dots,D_{k-1} \mid k \le m \,\}\end{array}}{\Gamma,(C_1 \preccurlyeq D_1),\dots,(C_m \preccurlyeq D_m) \vdash \Delta,(A_1 \preccurlyeq B_1),\dots,(A_n \preccurlyeq B_n)}\ R_{n,m}$$

$$\frac{\{\, C_k \vdash D_1,\dots,D_{k-1} \mid k \le m \,\} \qquad \Gamma \vdash \Delta, D_1,\dots,D_m}{\Gamma,(C_1 \preccurlyeq D_1),\dots,(C_m \preccurlyeq D_m) \vdash \Delta}\ T_m$$

$$\frac{\{\, C_k \vdash A_1,\dots,A_n,D_1,\dots,D_{k-1} \mid k \le m \,\} \qquad \Gamma \vdash \Delta, A_1,\dots,A_n,D_1,\dots,D_m}{\Gamma,(C_1 \preccurlyeq D_1),\dots,(C_m \preccurlyeq D_m) \vdash \Delta,(A_1 \preccurlyeq B_1),\dots,(A_n \preccurlyeq B_n)}\ W_{n,m}$$

$$\frac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta,(A \preccurlyeq B)}\ R_{C1} \qquad \frac{\Gamma, A \vdash \Delta \qquad \Gamma \vdash \Delta, B}{\Gamma,(A \preccurlyeq B) \vdash \Delta}\ R_{C2}$$

$$\frac{\begin{array}{c}\{\, \Gamma^{\preccurlyeq}, B_k \vdash \Delta^{\preccurlyeq}, A_1,\dots,A_n,D_1,\dots,D_m \mid k \le n \,\}\\ \cup\, \{\, \Gamma^{\preccurlyeq}, C_k \vdash \Delta^{\preccurlyeq}, A_1,\dots,A_n,D_1,\dots,D_{k-1} \mid k \le m \,\}\end{array}}{\Gamma,(C_1 \preccurlyeq D_1),\dots,(C_m \preccurlyeq D_m) \vdash \Delta,(A_1, \preccurlyeq B_1),\dots,(A_n \preccurlyeq B_n)}\ A_{n,m}$$

$$\mathcal{R}_{\mathbb{V}_{\preccurlyeq}} = \{R_{n,m} \mid n \ge 1, m \ge 0\}$$

$$\begin{array}{llll}
\mathcal{R}_{\mathbb{VN}_{\preccurlyeq}} & = \{R_{n,m} \mid n+m \ge 1\} & \mathcal{R}_{\mathbb{VC}_{\preccurlyeq}} & = \mathcal{R}_{\mathbb{V}} \cup \{R_{C1}, R_{C2}\} \\
\mathcal{R}_{\mathbb{VT}_{\preccurlyeq}} & = \mathcal{R}_{\mathbb{V}_{\preccurlyeq}} \cup \{T_m \mid m \ge 1\} & \mathcal{R}_{\mathbb{VA}_{\preccurlyeq}} & = \{A_{n,m} \mid n \ge 1, m \ge 0\} \\
\mathcal{R}_{\mathbb{VW}_{\preccurlyeq}} & = \mathcal{R}_{\mathbb{V}_{\preccurlyeq}} \cup \{W_{n,m} \mid n+m \ge 1\} & \mathcal{R}_{\mathbb{VNA}_{\preccurlyeq}} & = \{A_{n,m} \mid n+m \ge 1\}
\end{array}$$

Clarifications: Sequents are based on multisets. The rules $\mathcal{R}_{\mathcal{L}_{\preccurlyeq}}$ form a calculus for a counterfactual logic $\mathcal{L}$ described in [6], where $\preccurlyeq$ is the *comparative plausibility* operator. Besides the rules shown above, these calculi also include the propositional rules of **G3c** {11} and contraction rules. The contexts $\Gamma^{\preccurlyeq}$ and $\Delta^{\preccurlyeq}$ contain all formulae of resp. Γ and Δ of the form $A \preccurlyeq B$.

History: The calculus for $\mathbb{VC}$ was introduced in the tableaux setting [5, 4]. The remaining calculi were introduced in [3, 2] and corrected in [1].

Remarks: Soundness and completeness are shown by proving equivalence to Hilbert-style calculi and (syntactical) cut elimination. These calculi yield PSPACE decision procedures (EXPTIME for $\mathbb{VA}_{\preccurlyeq}$ and $\mathbb{VNA}_{\preccurlyeq}$) and, in most cases, enjoy Craig Interpolation. Contraction can be made admissible.

Entry 18 by: Björn Lellmann

[1] Björn Lellmann. "Sequent Calculi with Context Restrictions and Applications to Conditional Logic". PhD thesis. Imperial College London, 2013. URL: http://hdl.handle.net/10044/1/18059.

[2] Björn Lellmann and Dirk Pattinson. "Constructing Cut Free Sequent Systems With Context Restrictions Based on Classical or Intuitionistic Logic". In: *ICLA 2013*. Ed. by Kamal Lodaya. Vol. 7750. LNAI. Springer-Verlag Berlin Heidelberg, 2013, pp. 148–160.

[3] Björn Lellmann and Dirk Pattinson. "Sequent Systems for Lewis' Conditional Logics". In: *JELIA 2012*. Ed. by Luis Fariñas del Cerro, Andreas Herzig, and Jerome Mengin. Vol. 7519. LNCS. Springer-Verlag Berlin Heidelberg, 2012, pp. 320–332.

[4] Ian P. Gent. "A Sequent- or Tableau-style System for Lewis's Counterfactual Logic VC". In: *Notre Dame J. Form. Log.* 33.3 (1992), pp. 369–382.

[5] Harrie C.M. de Swart. "A Gentzen- or Beth-Type System, a Practical Decision Procedure and a Constructive Completeness Proof for the Counterfactual Logics VC and VCS". In: *J. Symb. Log.* 48.1 (1983), pp. 1–20.

[6] David Lewis. *Counterfactuals*. Blackwell, 1973.

Counterfactual Sequent Calculi II (2012, 2013)

$$\frac{\begin{array}{c}\{\, C_k, \mathbf{B}^I \vdash \mathbf{A}^{[n]\smallsetminus I}, \mathbf{C}^J, \mathbf{D}^{[k-1]\smallsetminus J} \mid 1 \le k \le m, I \subseteq [n], J \subseteq [k-1] \,\} \\ \cup \{\, A_k, B_k, \mathbf{B}^I \vdash \mathbf{A}^{[n]\smallsetminus I}, \mathbf{C}^J, \mathbf{D}^{[m]\smallsetminus J} \mid k \le n, I \subseteq [n], J \subseteq [m] \,\}\end{array}}{\Gamma, (A_1 \boxRight B_1), \dots, (A_n \boxRight B_n) \vdash \Delta, (C_1 \boxRight D_1), \dots, (C_m \boxRight D_m)}\ R_{n,m}$$

$$\frac{\{\, \Gamma \vdash \Delta, \mathbf{C}^J, \mathbf{D}^{[m]\smallsetminus J} \mid J \subseteq [m] \,\} \cup \{\, C_k \vdash D_k, \mathbf{C}^J, \mathbf{D}^{[k-1]\smallsetminus J} \mid 1 \le k \le m, J \subseteq [k-1] \,\}}{\Gamma \vdash \Delta, (C_1 \boxRight D_1), \dots, (C_m \boxRight D_m)}\ T_m$$

$$\frac{\begin{array}{c}\{\, C_k, \mathbf{B}^I \vdash \mathbf{A}^{[n]\smallsetminus I}, \mathbf{C}^J, \mathbf{D}^{[k-1]\smallsetminus J} \mid 1 \le k \le m, I \subseteq [n], J \subseteq [k-1] \,\} \\ \cup \{\, \Gamma, \mathbf{B}^I \vdash \mathbf{A}^{[n]\smallsetminus I}, \mathbf{C}^J, \mathbf{D}^{[m]\smallsetminus J} \mid I \subseteq [n], J \subseteq [m] \,\}\end{array}}{\Gamma, (A_1 \boxRight B_1), \dots, (A_n \boxRight B_n) \vdash \Delta, (C_1 \boxRight D_1), \dots, (C_m \boxRight D_m)}\ W_{n,m}$$

$$\frac{\Gamma \vdash \Delta, A \qquad \Gamma, B \vdash \Delta}{\Gamma, (A \boxRight B) \vdash \Delta}\ R_{C1} \qquad \frac{\Gamma \vdash \Delta, A \qquad \Gamma, A \vdash \Delta, B}{\Gamma \vdash \Delta, (A \boxRight B)}\ R_{C2}$$

For $n > 0$ the set $[n]$ is $\{1, \dots, n\}$ and $[0]$ is $\emptyset$. For a set I of indices, $\mathbf{A}^I$ contains all A_i with $i \in I$.

$$\mathcal{R}_{\mathbb{V}_{\boxRight}} = \{R_{n,m} \mid n \ge 1, m \ge 0\}$$

$$\mathcal{R}_{\mathbb{VN}_{\boxRight}} = \{R_{n,m} \mid n + m \ge 1\} \qquad \mathcal{R}_{\mathbb{VW}_{\boxRight}} = \mathcal{R}_{\mathbb{VT}_{\boxRight}} \cup \{W_{n,m} \mid n + m \ge 1\}$$

$$\mathcal{R}_{\mathbb{VT}_{\boxRight}} = \mathcal{R}_{\mathbb{V}_{\boxRight}} \cup \{T_m \mid m \ge 1\} \qquad \mathcal{R}_{\mathbb{VC}_{\boxRight}} = \mathcal{R}_{\mathbb{V}_{\boxRight}} \cup \{R_{C1}, R_{C2}\}$$

Clarifications: Sequents are based on multisets. The rules $\mathcal{R}_{\mathcal{L}_{\boxRight}}$ form a calculus for a counterfactual logic $\mathcal{L}$ described in [3], where $\boxRight$ is the *strong counterfactual implication* operator. Besides the rules shown above, these calculi also include the propositional rules of **G3c** {11} and contraction rules.

History: These calculi were introduced in [2] and corrected in [1].

Remarks: The calculi are translations of the calculi in {18} to the language with $\boxRight$. They inherit cut elimination and yield PSPACE decision procedures. Contraction can be made admissible.

[1] Björn Lellmann. "Sequent Calculi with Context Restrictions and Applications to Conditional Logic". PhD thesis. Imperial College London, 2013. URL: http://hdl.handle.net/10044/1/18059.

[2] Björn Lellmann and Dirk Pattinson. "Sequent Systems for Lewis' Conditional Logics". In: *JELIA 2012*. Ed. by Luis Fariñas del Cerro, Andreas Herzig, and Jerome Mengin. Vol. 7519. LNCS. Springer-Verlag Berlin Heidelberg, 2012, pp. 320–332.

[3] David Lewis. *Counterfactuals*. Blackwell, 1973.

Entry 19 by: Björn Lellmann

Contextual Natural Deduction (2013)

$$\overline{\Gamma, a : A \vdash a : A}$$

$$\frac{\Gamma, a : A \vdash b : C_\pi[B]}{\Gamma \vdash \lambda_\pi a^A.b : C_\pi[A \to B]} \to_I (\pi)$$

$$\frac{\Gamma \vdash f : C^1_{\pi_1}[A \to B] \quad \Gamma \vdash x : C^2_{\pi_2}[A]}{\Gamma \vdash (f\ x)^{\rightharpoonup}_{(\pi_1;\pi_2)} : C^1_{\pi_1}[C^2_{\pi_2}[B]]} \to^{\rightharpoonup}_E (\pi_1;\pi_2)$$

$$\frac{\Gamma \vdash f : C^1_{\pi_1}[A \to B] \quad \Gamma \vdash x : C^2_{\pi_2}[A]}{\Gamma \vdash (f\ x)^{\leftharpoonup}_{(\pi_1;\pi_2)} : C^2_{\pi_1}[C^1_{\pi_2}[B]]} \to^{\leftharpoonup}_E (\pi_1;\pi_2)$$

π, π_1 and π_2 must be positive positions. a is allowed to occur in b only if π is strongly positive.

Clarifications: $C_\pi[F]$ denotes a formula with F occurring in the hole of a *context* $C_\pi[]$. π is the position of the hole. It is: *positive* iff it is in the left side of an even number of implications; *strongly positive* iff this number is zero.

History: Contextual Natural Deduction [1] combines the idea of deep inference with Gentzen's natural deduction {1}.

Remarks: Soundness and completeness w.r.t. minimal logic are proven [1] by providing translations between $\mathbf{ND^c}$ and the minimal fragment of **NJ** {1}. $\mathbf{ND^c}$ proofs can be quadratically shorter than proofs in the minimal fragment of **NJ**.

---

[1] Bruno Woltzenlogel Paleo. "Contextual Natural Deduction". In: *Logical Foundations of Computer Science, International Symposium, LFCS 2013, San Diego, CA, USA, January 6-8, 2013. Proceedings*. Ed. by Sergei N. Artёmov and Anil Nerode. Vol. 7734. Lecture Notes in Computer Science. Springer, 2013, pp. 372–386. ISBN: 978-3-642-35721-3. DOI: 10.1007/978-3-642-35722-0_27. URL: http://dx.doi.org/10.1007/978-3-642-35722-0_27.

Entry 20 by: Bruno Woltzenlogel Paleo

IR (2014)

C is a non-tautological clause from the matrix.
$\tau = \{0/u \mid u \text{ is universal in } C\}$, where the notation $0/u$ for literals u is shorthand for $0/y$ if $u = y$ and $1/y$ if $u = \neg y$. We define $\mathsf{restr}(\tau, x)$ as $\{c/u \mid c/u \in \tau, \mathsf{lv}(u) < \mathsf{lv}(x)\}$.
τ is a partial assignment to universal variables with $\mathsf{rng}(\tau) \subseteq \{0, 1\}$. $\xi = \sigma \cup \{c/u \mid c/u \in \mathsf{restr}(\tau, x), u \notin \mathsf{dom}(\sigma)\}$

The rules of IR [2]

Clarifications: The calculus aims to refute a quantified Boolean formula (QBF) of the form $Q_1 x_1 \ldots Q_n x_n . \varphi$ where $Q_i \in \{\forall, \exists\}$ and φ is a Boolean formula in conjunctive normal form (CNF). The formula φ is referred to as the *matrix*. We write $\mathsf{lv}(x)$ for the *quantification level* of x, i.e. $\mathsf{lv}(x_i) = i$. A variable x_i is *existential* (resp. *universal*) if $Q_i = \exists$ (resp. $Q_i = \forall$).

The calculus works by introducing clauses as *annotated clauses*, which are sets of annotated literals. Annotated literals consist of an existential literal and an annotation – a partial assignment to universal variables in $\{0, 1\}$. Two literals are identical if and only if both the existential literal and annotation are equal. The calculus enables deriving the empty clause if and only if the given formula is false.

Remarks: Soundness was shown by extracting valid Herbrand functions. Completeness is shown by p-simulation of another known QBF system Q-Resolution .

History: The name of the calculus comes from the two pivotal operations *instantiation* and *resolution*. The calculus naturally generalizes an older calculus $\forall$Exp+Res [1], which requires all clauses to be introduced into the proof by using a complete assignment.

[1] Mikoláš Janota and Joao Marques-Silva. “Expansion-based {QBF} solving versus Q-resolution”. In: *Theoretical Computer Science* 577 (2015), pp. 25–42. DOI: `http://dx.doi.org/10.1016/j.tcs.2015.01.048`.

[2] Olaf Beyersdorff, Leroy Chew, and Mikoláš Janota. “On Unification of QBF Resolution-Based Calculi”. In: *Mathematical Foundations of Computer Science (MFCS)*. 2014.

Entry 21 by: Leroy Chew, Mikoláš Janota

Part III
Indexes

List of Contributors

Appendix A
Contributors

ToDo: Use an index instead.

Appendix B
Authors

ToDo: Use an index instead.

Appendix C
Acronyms

Use the template *acronym.tex* together with the Springer document class SVMono (monograph-type books) or SVMult (edited books) to style your list(s) of abbreviations or symbols in the Springer layout.

Lists of abbreviations symbolsand the like are easily formatted with the help of the Springer-enhanced `description` environment.

ABC Spelled-out abbreviation and definition
BABI Spelled-out abbreviation and definition
CABR Spelled-out abbreviation and definition

Logics

Proof Systems Grouped by Type