Atomic Cut Elimination for Classical Logic

Technische Universität Dresden – Technical Report WV-2002-11

Kai Brünnler
kai.bruennler@inf.tu-dresden.de

Technische Universität Dresden, Fakultät Informatik, D - 01062 Dresden, Germany

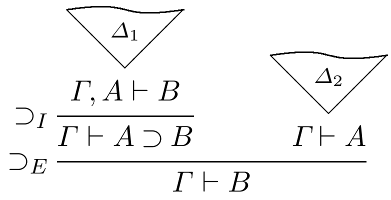
Abstract. System SKS is a set of rules for classical propositional logic presented in the calculus of structures. Like sequent systems and unlike natural deduction systems, it has an explicit cut rule, which is admissible. In contrast to sequent systems, the cut rule can easily be restricted to atoms. This allows for a very simple cut elimination procedure based on plugging in parts of a proof, like normalisation in natural deduction and unlike cut elimination in the sequent calculus. It should thus be a good common starting point for investigations into both proof search as computation and proof normalisation as computation.

1 Introduction

The two well-known connections between proof theory and language design, *proof search as computation* and *proof normalisation as computation*, have mainly used different proof-theoretic formalisms. While designers of functional programming languages prefer natural deduction, because of the close correspondence between proof normalisation and reduction in related term calculi [3, 7], designers of logic programming languages prefer the sequent calculus [6], because infinite choice and much of the unwanted non-determinism is limited to the cut rule, which can be eliminated.

System SKS [1] is a set of inference rules for classical propositional logic presented in a new formalism, the *calculus of structures* [4, 5]. This system admits the good properties usually found in sequent systems: in particular, all rules that induce infinite choice in proof search are admissible. Thus, in principle, it is as suitable for proof search as systems in the sequent calculus. In this paper I will present a cut elimination procedure for SKS that is very similar to normalisation in natural deduction. It thus allows us to develop, at least for the case of classical logic, both the proof search and the proof normalisation paradigm of computation in the same formalism and starting from the same system of rules.

Normalisation in natural deduction and cut elimination in the sequent calculus, widely perceived as ‘morally the same’, differ quite a bit, technically. Compared to cut elimination, normalisation is simpler, involving neither permutation of a multicut rule, nor induction on the cut-rank. The equivalent of a cut in natural deduction, for example,



,

is eliminated as follows: first, assumption A and all its copies are removed from Δ_1. Second, the derivation Δ_2, with the context strengthened accordingly, is plugged into all the leaves of Δ_1 where assumption A was used.

This method relies on the fact that no rule inside Δ_1 can change the premise A, which is why it does not work for the sequent calculus. However, given a cut with an *atomic* cut formula a inside a sequent calculus proof, we can trace the occurrence of a and its copies produced by contraction, identify all the leaves where they are used in identity axioms, and plug in subproofs in very much the same way as in natural deduction. The problem for the sequent calculus is that cuts are not atomic, in general.

The calculus of structures, which generalises the one-sided sequent calculus, was conceived to express a logical system with a self-dual non-commutative connective resembling sequential composition in process algebras [4, 5, 2]. It has also led to inference systems for existing logics, like classical and linear logic, with interesting new properties [1, 8, 9].

Derivations in the calculus of structures enjoy a top-down symmetry that is not available in the sequent calculus: they are chains of one-premise inference rules. ‘Meta-level conjunction’ (the branching of the proof tree) and ‘object-level conjunction’ (the connective in a formula) are identified. The two notions of formulae and sequent are also identified, they merge into the notion of *structure*, which is a formula subject to equivalences that are usually imposed on sequents. This simplification allows to observe the exact duality between the cut rule and the identity axiom [4]:

$$\text{identity}\;\frac{S\{\mathit{true}\}}{S\{R \vee \bar{R}\}} \qquad\qquad \text{cut}\;\frac{S\{R \wedge \bar{R}\}}{S\{\mathit{false}\}}$$

The identity rule is read bottom-up as: if inside a structure there occurs a disjunction of a structure R and its negation, then it can be replaced by the constant *true*. The notion of duality between cut and identity is precisely the *contrapositive* one.

As a consequence of this duality, R can be restricted to atomic formulas not only in the identity axiom, as in the sequent calculus, but with the same ease also in the cut. For classical logic, the calculus of structures therefore admits a very simple cut elimination procedure similar to normalisation in natural deduction.

After introducing basic notions of the calculus of structures, I show system SKS with atomic contraction, weakening, identity and, most significantly, atomic cut. Then, after establishing some lemmas, I present the cut elimination procedure.

2 Structures and Derivations

Definition 1. *Atoms* are denoted by a, b, ... The *structures* of the language KS are generated by

$$S ::= \mathsf{t} \mid \mathsf{f} \mid a \mid [\,\underbrace{S,\dots,S}_{>0}\,] \mid (\,\underbrace{S,\dots,S}_{>0}\,) \mid \bar{S} \quad ,$$

where t and f are the constants *true* and *false*, $[S_1,\dots,S_h]$ is a *disjunction* and $(S_1,\dots,S_h)$ is a *conjunction.* $\bar{S}$ is the *negation* of the structure S. The negation of an atom is again an atom. Structures are denoted by S, R, T, U and V. *Structure contexts*, denoted by $S\{\ \}$, are structures with one occurrence of $\{\ \}$, the *empty context* or *hole*, that does not appear in the scope of a negation. $S\{R\}$ denotes the structure obtained by filling the hole in $S\{\ \}$ with R. We drop the curly braces when they are redundant: for example, $S[R,T]$ stands for $S\{[R,T]\}$. Structures are equivalent modulo the smallest congruence relation induced by the equations shown in Fig. 1. In the following, we do not distinguish between a congruence class and one of its representatives: both of them are structures.

Associativity

$$[\boldsymbol{R},[\boldsymbol{T}]] = [\boldsymbol{R},\boldsymbol{T}]$$
$$(\boldsymbol{R},(\boldsymbol{T})) = (\boldsymbol{R},\boldsymbol{T})$$

Commutativity

$$[\boldsymbol{R},\boldsymbol{T}] = [\boldsymbol{T},\boldsymbol{R}]$$
$$(\boldsymbol{R},\boldsymbol{T}) = (\boldsymbol{T},\boldsymbol{R})$$

Singleton

$$[R] = R = (R)$$

Constants

$$[\mathsf{f},\boldsymbol{R}] = [\boldsymbol{R}]$$
$$(\mathsf{t},\boldsymbol{R}) = (\boldsymbol{R})$$
$$[\mathsf{t},\mathsf{t}] = \mathsf{t}$$
$$(\mathsf{f},\mathsf{f}) = \mathsf{f}$$

Negation

$$\bar{\mathsf{t}} = \mathsf{f}$$
$$\bar{\mathsf{f}} = \mathsf{t}$$
$$\overline{[R_1,\dots,R_h]} = (\bar{R}_1,\dots,\bar{R}_h)$$
$$\overline{(R_1,\dots,R_h)} = [\bar{R}_1,\dots,\bar{R}_h]$$
$$\bar{\bar{R}} = R$$

$\boldsymbol{R}$ and $\boldsymbol{T}$ are finite, non-empty sequences of structures.

Fig. 1. Equations on structures

Definition 2. An *inference rule* is a scheme of the kind

$$\rho \frac{S\{T\}}{S\{R\}} \quad ,$$

where ρ is the *name* of the rule, $S\{T\}$ is its *premise* and $S\{R\}$ is its *conclusion*. In an instance of ρ, the structure taking the place of R is called *redex* and the structure taking the place of T is called *contractum*. A (*formal*) *system* $\mathscr{S}$ is a set of inference rules. To clarify the use of the equational theory where it is not obvious, there is a rule $= \frac{T}{R}$ where R and T are different representatives of the same structure.

Definition 3. A *derivation* Δ in a certain formal system is a finite chain of instances of inference rules in the system:

$$\begin{array}{c} \pi' \dfrac{T}{V} \\ \pi \dfrac{}{\vdots} \\ \rho' \dfrac{}{U} \\ \rho \dfrac{}{R} \end{array} \quad .$$

$$\mathsf{ai}{\downarrow}\,\frac{S\{\mathsf{t}\}}{S[a,\bar{a}]} \qquad \mathsf{ai}{\uparrow}\,\frac{S(a,\bar{a})}{S\{\mathsf{f}\}}$$

$$\mathsf{s}\,\frac{S([R,U],T)}{S[(R,T),U]}$$

$$\mathsf{m}\,\frac{S[(R,U),(T,V)]}{S([R,T],[U,V])}$$

$$\mathsf{aw}{\downarrow}\,\frac{S\{\mathsf{f}\}}{S\{a\}} \qquad \mathsf{aw}{\uparrow}\,\frac{S\{a\}}{S\{\mathsf{t}\}}$$

$$\mathsf{ac}{\downarrow}\,\frac{S[a,a]}{S\{a\}} \qquad \mathsf{ac}{\uparrow}\,\frac{S\{a\}}{S(a,a)}$$

Fig. 2. System SKS

A derivation can consist of just one structure. The topmost structure in a derivation is called the *premise* of the derivation, and the structure at the bottom is called its *conclusion*. A derivation Δ whose premise is T, whose conclusion is R, and whose inference rules are in $\mathscr{S}$ will be indicated with $\begin{array}{c}T\\ \Delta\Vert\mathscr{S}\\ R\end{array}$. A *proof* Π in the calculus of structures is a derivation whose premise is the constant t. It will be denoted by $\begin{array}{c}\Pi\overline{\Vert}\mathscr{S}\\ R\end{array}$. A rule ρ is *derivable* in a system $\mathscr{S}$ if for every instance of $\rho\,\frac{T}{R}$ there is a derivation $\begin{array}{c}T\\ \Delta\Vert\mathscr{S}\\ R\end{array}$. A rule ρ is *admissible* for a system $\mathscr{S}$ if for every proof $\begin{array}{c}\Pi\overline{\Vert}\mathscr{S}\cup\{\rho\}\\ S\end{array}$ there is a proof $\begin{array}{c}\Pi'\overline{\Vert}\mathscr{S}\\ S\end{array}$.

3 System SKS

System SKS is shown in Fig. 2. The first S stands for "symmetric" or "self-dual", meaning that for each rule, its dual (or contrapositive) is also in the system. The K stands for "klassisch" as in Gentzen's LK and the last S says that it is a system on structures.

The rules $\mathsf{ai}\!\downarrow, \mathsf{s}, \mathsf{m}, \mathsf{aw}\!\downarrow, \mathsf{ac}\!\downarrow$ are called respectively *atomic identity*, *switch*, *medial*, *atomic weakening* and *atomic contraction*. Their dual rules carry the same name prefixed with a "co-", so e.g. $\mathsf{aw}\!\uparrow$ is called *atomic co-weakening*. The rules s and m are their own duals. The rule $\mathsf{ai}\!\uparrow$ is special, it is called *atomic cut*. Rules $\mathsf{ai}\!\downarrow, \mathsf{aw}\!\downarrow, \mathsf{ac}\!\downarrow$ are called *down-rules* and their duals are called *up-rules*. In [1], by a semantic argument, all up-rules were shown to be admissible. By removing them we obtain system KS, shown in Fig. 3, which is sufficient for proof search.

$$\mathsf{ai}\!\downarrow \frac{S\{\mathsf{t}\}}{S[a,\bar{a}]} \qquad \mathsf{aw}\!\downarrow \frac{S\{\mathsf{f}\}}{S\{a\}} \qquad \mathsf{ac}\!\downarrow \frac{S[a,a]}{S\{a\}}$$

$$\mathsf{s}\,\frac{S([R,T],U)}{S[(R,U),T]} \qquad \mathsf{m}\,\frac{S[(R,T),(U,V)]}{S([R,U],[T,V])}$$

Fig. 3. System KS

Identity, cut, weakening and contraction are atomic in system SKS, they do not *have to* be applied to arbitrarily large formulas. But they *can*, by the following theorem.

Theorem 4. General identity, weakening, contraction and their duals, i.e. the rules shown in Fig. 4, are derivable in system SKS. In particular, the rules $\mathsf{i}\!\downarrow$, $\mathsf{w}\!\downarrow$ and $\mathsf{c}\!\downarrow$ are derivable in $\{\mathsf{ai}\!\downarrow, \mathsf{s}\}$, $\{\mathsf{aw}\!\downarrow\}$ and $\{\mathsf{ac}\!\downarrow, \mathsf{m}\}$, respectively. Dually, the rules $\mathsf{i}\!\uparrow$, $\mathsf{w}\!\uparrow$ $\mathsf{c}\!\uparrow$ are derivable in $\{\mathsf{ai}\!\uparrow, \mathsf{s}\}$, $\{\mathsf{aw}\!\uparrow\}$ and $\{\mathsf{ac}\!\uparrow, \mathsf{m}\}$, respectively.

Proof. By an easy structural induction on the structure that is cut, weakened or contracted. Details are in [1]. The case for the cut is shown here. A cut introducing the structure (R,T) together with its dual structure $[\bar{R},\bar{T}]$ is replaced by two cuts on smaller structures:

$$
\mathsf{i}\!\downarrow \frac{S\{\mathsf{t}\}}{S[R,\bar{R}]} \qquad \mathsf{i}\!\uparrow \frac{S(R,\bar{R})}{S\{\mathsf{f}\}}
$$

$$
\mathsf{w}\!\downarrow \frac{S\{\mathsf{f}\}}{S\{R\}} \qquad \mathsf{w}\!\uparrow \frac{S\{R\}}{S\{\mathsf{t}\}}
$$

$$
\mathsf{c}\!\downarrow \frac{S[R,R]}{S\{R\}} \qquad \mathsf{c}\!\uparrow \frac{S\{R\}}{S(R,R)}
$$

Fig. 4. General identity, weakening, contraction and their duals

$$
\mathsf{i}\!\uparrow \frac{S(R,T,[\bar{R},\bar{T}])}{S\{\mathsf{f}\}} \quad \leadsto \quad \mathsf{i}\!\uparrow \frac{\mathsf{i}\!\uparrow \frac{\mathsf{s} \frac{\mathsf{s} \frac{S(R,T,[\bar{R},\bar{T}])}{S(R,[\bar{R},(T,\bar{T})])}}{S[(R,\bar{R}),(T,\bar{T})]}}{S(R,\bar{R})}}{S\{\mathsf{f}\}} \quad .
$$

□

So, while general identity, weakening, contraction and their duals do not belong to SKS, they will be freely used in derivations in SKS to denote multiple instances of the corresponding rules in SKS according to Theorem 4. Because the decomposition of down-rules does not introduce up-rules, by the same argument, general identity, weakening and contraction (but not their duals) will be used in derivations in KS.

Remark 5. Sequent calculus derivations easily correspond to derivations in system SKS. For instance, the cut of sequent systems in Gentzen-Schütte form [10]:

$$
\mathsf{Cut} \frac{\vdash \Phi, A \quad \vdash \Psi, \bar{A}}{\vdash \Phi, \Psi} \qquad \text{corresponds to} \qquad \mathsf{i}\!\uparrow \frac{\mathsf{s} \frac{\mathsf{s} \frac{([\Phi,A],[\Psi,\bar{A}])}{[\Phi,(A,[\Psi,\bar{A}])]}}{[\Phi,\Psi,(A,\bar{A})]}}{[\Phi,\Psi]} \quad .
$$

4 Cut Elimination

In contrast to the sequent calculus, the cut is not the only problematic rule in system SKS. The rule $\mathsf{aw}\uparrow$ also induces infinite choice in proof-search. Fortunately, we can not only eliminate the cut rule, but also the other up-rules. Each up-rule individually can be shown to be admissible for system KS. However, since we are going to eliminate the cut anyway, to eliminate rules $\mathsf{aw}\uparrow$ and $\mathsf{ac}\uparrow$ the following lemma is sufficient.

Lemma 6. Each rule in SKS is derivable using identity, cut, switch and its dual rule.

Proof. An instance of $\rho\uparrow \dfrac{S\{T\}}{S\{R\}}$ can be replaced by

$$\mathsf{i}\uparrow \dfrac{\rho\downarrow \dfrac{\mathsf{s} \dfrac{\mathsf{i}\downarrow \dfrac{S\{T\}}{S(T,[R,\bar{R}])}}{S[R,(T,\bar{R})]}}{S[R,(T,\bar{T})]}}{S\{R\}} \quad .$$

The same holds for down-rules. □

When plugging in a derivation in natural deduction, its context has to be strengthened, to fit into the leaf into which it is plugged. Adding to a context in natural deduction is easy, since it is a flat object, a set or a multiset. In the calculus of structures, contexts are more general, nested objects. The following definition and lemma are used to strengthen contexts.

Definition 7. Given a derivation Δ, the derivation $S\{\Delta\}$ is obtained as follows:

$$\Delta = \rho \dfrac{\rho' \dfrac{\vdots}{\pi \dfrac{\pi' \dfrac{T}{V}}{}}}{U}\Big/ R \qquad S\{\Delta\} = \rho \dfrac{\rho' \dfrac{\vdots}{\pi \dfrac{\pi' \dfrac{S\{T\}}{S\{V\}}}{}}}{S\{U\}}\Big/ S\{R\} \quad .$$

Lemma 8. The following *super switch* rules are derivable using the switch:

$$\mathsf{ss}\downarrow \dfrac{S\{T\{R\}\}}{S[R,T\{\mathsf{f}\}]} \qquad \text{and} \qquad \mathsf{ss}\uparrow \dfrac{S(R,T\{\mathsf{t}\})}{S\{T\{R\}\}} \quad .$$

Proof. An easy structural induction on $T\{\ \}$ replaces either rule by a derivation consisting of switches. Details are in [1].

□

In the sequent calculus as well as in sequent-style natural deduction, a derivation is a tree and cuts (seen bottom-up) split the tree into two branches. A cut instance decides how to split the context among the two branches (or just copies the context, in the case of natural deduction or additive cut). In the calculus of structures, a cut rule does not split the proof. But we can do that during cut elimination, duplicating the proof above a cut and using the following lemma to remove atoms from the proofs.

Lemma 9. Each proof $\begin{array}{c}\Vert \mathsf{KS} \\ T\{a\}\end{array}$ can be transformed into a proof $\begin{array}{c}\Vert \mathsf{KS} \\ T\{\mathsf{t}\}\end{array}$.

Proof. Starting with the conclusion, going up in the proof, in each structure we replace the occurrence of a and its copies, that are produced by contractions, by the constant t.

Replacements inside the context of any rule instance do not affect the validity of this rule instance. Instances of the rules m and s remain valid, also in the case that atom occurrences are replaced inside redex and contractum. Instances of the other rules are replaced by the following derivations:

$$\mathsf{ac}\!\downarrow \frac{S[a,a]}{S\{a\}} \quad \rightsquigarrow \quad = \frac{S[\mathsf{t},\mathsf{t}]}{S\{\mathsf{t}\}}$$

$$\mathsf{aw}\!\downarrow \frac{S\{\mathsf{f}\}}{S\{a\}} \quad \rightsquigarrow \quad = \frac{= \frac{\mathsf{s}\frac{S\{\mathsf{f}\}}{S([\mathsf{t},\mathsf{t}],\mathsf{f})}}{S[\mathsf{t},(\mathsf{t},\mathsf{f})]}}{S\{\mathsf{t}\}}$$

$$\mathsf{ai}\!\downarrow \frac{S\{\mathsf{t}\}}{S[a,\bar{a}]} \quad \rightsquigarrow \quad \mathsf{aw}\!\downarrow \frac{= \frac{S\{\mathsf{t}\}}{S[\mathsf{t},\mathsf{f}]}}{S[\mathsf{t},\bar{a}]} \quad .$$

□

Properly equipped, we now turn to cut elimination.

Theorem 10. Each proof $\begin{array}{c}\| \mathsf{SKS} \\ T\end{array}$ can be transformed into a proof $\begin{array}{c}\| \mathsf{KS} \\ T\end{array}$.

Proof. By Lemma 6, the only rule left to eliminate is the cut. The topmost instance of cut, together with the proof above it, is singled out:

$$\begin{array}{c}\| \mathsf{KS} \cup \{\mathsf{ai}\!\uparrow\} \\ T\end{array} \quad = \quad \begin{array}{c} \Pi \| \mathsf{KS} \\ \mathsf{ai}\!\uparrow \dfrac{R(a,\bar{a})}{R\{\mathsf{f}\}} \\ \Delta \| \mathsf{KS} \cup \{\mathsf{ai}\!\uparrow\} \\ T \end{array} \quad .$$

Lemma 9 is applied twice on Π to obtain

$$\begin{array}{c} \Pi_1 \| \mathsf{KS} \\ R\{a\} \end{array} \quad \text{and} \quad \begin{array}{c} \Pi_2 \| \mathsf{KS} \\ R\{\bar{a}\} \end{array} \quad .$$

Starting with the conclusion, going up in proof Π_1, in each structure we replace the occurrence of a and its copies, that are produced by contractions, by the structure $R\{\mathsf{f}\}$.

Replacements inside the context of any rule instance do not affect the validity of this rule instance. Instances of the rules m and s remain valid, also in the case that atom occurrences are replaced inside redex and contractum. Instances of $\mathsf{ac}\!\downarrow$ and $\mathsf{aw}\!\downarrow$ are replaced by their general versions:

$$\mathsf{ac}\!\downarrow \frac{S[a,a]}{S\{a\}} \quad \leadsto \quad \mathsf{c}\!\downarrow \frac{S[R\{\mathsf{f}\},R\{\mathsf{f}\}]}{S\{R\{\mathsf{f}\}\}}$$

$$\mathsf{aw}\!\downarrow \frac{S\{\mathsf{f}\}}{S\{a\}} \quad \leadsto \quad \mathsf{w}\!\downarrow \frac{S\{\mathsf{f}\}}{S\{R\{\mathsf{f}\}\}} \quad .$$

Instances of $\mathsf{ai}\!\downarrow$ are replaced by $S\{\Pi_2\}$:

$$\mathsf{ai}\!\downarrow \frac{S\{\mathsf{t}\}}{S[a,\bar a]} \quad \leadsto \quad \mathsf{ss}\!\downarrow \frac{\begin{array}{c} S\{\mathsf{t}\} \\ S\{\Pi_2\} \Big\| \mathsf{KS} \\ S\{R\{\bar a\}\} \end{array}}{S[R\{\mathsf{f}\},\bar a]} \quad .$$

The result of this process of substituting Π_2 into Π_1 is a proof Π_3, from which we build

$$\begin{array}{c} \Pi_3 \Big\| \mathsf{KS} \\ \mathsf{ss}\!\downarrow \dfrac{R\{R\{\mathsf{f}\}\}}{[R\{\mathsf{f}\},R\{\mathsf{f}\}]} \\ \mathsf{c}\!\downarrow \dfrac{}{R\{\mathsf{f}\}} \\ \Delta \Big\| \mathsf{KS}\cup\{\mathsf{ai}\!\downarrow\} \\ T \end{array} \quad .$$

Proceed inductively downward with the remaining instances of cut. □

The proof is inspired by [4], where in a system without contraction the context $R\{\ \}$ is split into two disjoint parts: one, that comes together with a and one that comes together with $\bar a$. The proof of the context splitting lemma is hard. In system KS, to split the proof above a cut, no such context splitting is necessary: we can simply duplicate the proof and use contraction. In the sequent calculus, where the technique of permuting up a cut is used, contraction is an obstacle to cut elimination. Curiously, in the calculus of structures, when the technique of plugging subproofs is used, contraction *simplifies* cut elimination.

The cut elimination procedure has not been extended to the predicative first-order case yet, however, we do know that in this case the cut is admissible, either by a semantic argument or by translation to the sequent calculus, cf. [1]. So, system SKS should be relevant for proof search also in the first-order predicative case, because for proof search it is just important *that* the cut is admissible, not *how* this admissibility is shown.

Because of its simplicity, the cut elimination procedure presented here seems a good starting point for the endeavour outlined in the introduction. In the proof search as computation realm, given the admissibility of cut, a suitable notion of *uniform proof* as in [6] should be obtainable. For proof normalisation as computation, natural questions to be considered are strong normalisation and confluence of the cut elimination procedure when imposing as little strategy as possible. Similarly to [7], a term calculus should be developed and its computational meaning be made precise. The possibility of treating intuitionistic logic should be explored.

Acknowledgements
This work has been supported by the DFG Graduiertenkolleg 334 and the IQN 'Rational Mobile Agents and Systems of Agents'. I would like to thank Alessio Guglielmi, for inspiration, Alwen Fernanto Tiu, for simplifying an earlier version of the cut elimination procedure, and Michel Parigot, for helpful discussions. I am grateful to Alessio Guglielmi and Charles Stewart who helped me to improve previous drafts of this work.

References

1. Kai Brünnler and Alwen Fernanto Tiu. A local system for classical logic. In R. Nieuwenhuis and A. Voronkov, editors, *LPAR 2001*, volume 2250 of *Lecture Notes in Artificial Intelligence*, pages 347–361. Springer-Verlag, 2001. Available as technical report at http://www.wv.inf.tu-dresden.de/~kai/LocalClassicalLogic-tr.pdf.
2. Paola Bruscoli. A purely logical account of sequentiality in proof search. Technical Report WV-02-06, Dresden University of Technology, 2002. Accepted at ICLP 2002.
3. Jean Gallier. Constructive logics. Part I: A tutorial on proof systems and typed λ-calculi. *Theoretical Computer Science*, 110:249–339, 1993.
4. Alessio Guglielmi. A calculus of order and interaction. Technical Report WV-99-04, Dresden University of Technology, 1999. Revised version available at http://www.wv.inf.tu-dresden.de/~guglielm/Research/.
5. Alessio Guglielmi and Lutz Straßburger. Non-commutativity and MELL in the calculus of structures. In L. Fribourg, editor, *CSL 2001*, volume 2142 of *Lecture Notes in Computer Science*, pages 54–68. Springer-Verlag, 2001.
6. Dale Miller, Gopalan Nadathur, Frank Pfenning, and Andre Scedrov. Uniform proofs as a foundation for logic programming. *Annals of Pure and Applied Logic*, 51:125–157, 1991.
7. M. Parigot. $\lambda\mu$-calculus: an algorithmic interpretation of classical natural deduction. In *LPAR 1992*, volume 624 of *Lecture Notes in Computer Science*, pages 190–201. Springer-Verlag, 1992.
8. Lutz Straßburger. MELL in the calculus of structures. Technical Report WV-2001-03, Dresden University of Technology, 2001. On the web at: http://www.ki.inf.tu-dresden.de/~lutz/els.pdf.
9. Lutz Straßburger. A local system for linear logic. Technical Report WV-2002-01, Dresden University of Technology, 2002.
10. Anne Sjerp Troelstra and Helmut Schwichtenberg. *Basic Proof Theory*. Cambridge University Press, 1996.