



# Deep Proof Search in MELL

Ozan Kahramanođulları

University of Trento, Department of Mathematics  
The Microsoft Research - University of Trento  
Centre for Computational and Systems Biology

## Abstract

The deep inference presentation of multiplicative exponential linear logic (MELL) benefits from a rich combinatoric analysis with many more proofs in comparison to its sequent calculus presentation. In the deep inference setting, all the sequent calculus proofs are preserved. Moreover, many other proofs become available, and some of these proofs are much shorter. However, proof search in deep inference is subject to a greater nondeterminism, and this nondeterminism constitutes a bottleneck for applications. To this end, we address the problem of reducing nondeterminism in MELL by refining and extending our technique that has been previously applied to multiplicative linear logic and classical logic. We show that, besides the nondeterminism in commutative contexts, the nondeterminism in exponential contexts can be reduced in a proof theoretically clean manner. The method conserves the exponential speed-up in proof construction due to deep inference, exemplified by Statman tautologies. We validate the improvement in accessing the shorter proofs by experiments with our implementations.

## 1 Introduction

Deep inference [7] is now broadly established as a proof theoretical framework that sheds new light on formal properties of proofs. With deep inference, computational phenomena such as atomicity [2, 8] and locality [17, 3, 21] of information processing as well as sequentiality [7, 9] find natural interpretations within logic. Other explorations with deep inference bring new perspectives to theoretical concepts such as cut elimination [7, 8, 20] and normal forms of proofs [19, 20]. Some of these are provably impossible without deep inference [22]. However, despite such a theoretically rich landscape that should benefit applications, proof search with deep inference faces an obstacle, which can be considered a ramification, or the price, of the combinatoric wealth that gives rise to all these properties. Namely, proof search, which would constitute the backbone of many applications, is difficult in deep inference due to a much broader breadth of the search space. This is because the inference rules are applied not only at the top level connectives, but to connectives at any position inside the formulae.

Consider, for example, the multiplicative linear logic (MLL) formula  $[(a \wp \bar{a}) \otimes \bar{b}] \wp b$ . With a one sided sequent calculus, the immediate breadth of the search space has the left and

middle choices below, whereas deep inference adds the third rule instance on the right to these.

$$\text{s} \frac{([a \wp \bar{a}] \otimes [b \wp \bar{b}])}{[[a \wp \bar{a}] \otimes \bar{b}] \wp b} \quad \text{s} \frac{([a \wp \bar{a} \wp b] \otimes \bar{b})}{[[a \wp \bar{a}] \otimes \bar{b}] \wp b} \quad \text{ai}\downarrow \frac{[(1 \otimes \bar{b}) \wp b]}{[[a \wp \bar{a}] \otimes \bar{b}] \wp b}$$

Although this fortunate example benefits from the increased breadth of the search space, the situation easily becomes different with other formulae. For example, even after eliminating the nondeterminism due to equations for units [13], the simple formula  $[(\bar{a} \otimes \bar{b}) \wp a \wp b]$  has 6 immediate rule instances and only 2 result in proofs. With  $[(\bar{a} \otimes \bar{b} \otimes \bar{c}) \wp a \wp b \wp c]$ , we get 42 instances, only 6 deliver proofs, and 3 of these are redundant. More examples are provided in [11], and the reader can try others by using the implementations available at our web-site.

Given this high non-determinism, even with simpler formulae, the immediate question is “why would one consider deep inference for proof search?” An answer for this can be motivated by the fact that deep inference provides short proofs: sequent calculus proofs can be straight-forwardly mapped to deep inference proofs, providing polynomial simulation results [4, 6], whereas deep inference provides proofs that are impossible in the sequent calculus. For example, the inference step on the right above results in a shorter proof. In some cases, these deep inference proofs are exponentially shorter, as it was shown for Statman tautologies in propositional logic [4]. However, in proof search, even an improvement of few steps can determine if the search will be successful or not. This is because, the breadth of the search space propagates through the search tree, and this can cause a combinatoric explosion after only a few steps, thereby preventing a thorough exploration of the search space that would lead to a proof. In such a setting, before committing to a heuristic, two criteria determine the outcome of a proof search: (i) how many proof steps are required to find a proof, that is, the proof length; and (ii) how broad is the breadth of the search space. Thus, addressing these two criteria simultaneously in a satisfactory manner can expand the reach of proof search applications.

The excessive nondeterminism in deep inference systems is mainly due to the context management rules. In previous work [10, 11], we have shown that the nondeterminism due to the context management rules can be controlled to a great extent without sacrificing a clean proof theory. For this we have introduced a technique that is sensitive to interactions between complementary formulae and its correctness relies on the similarity between cut-elimination and completeness. We have used this technique on the switch rule (s), exemplified above, which is responsible for managing the interactions of disjunctions and conjunction. This was first proved [14, 15] on logic BV [7], that is, the logic that introduced deep inference, extending MLL with a self-dual non-commutative logical operator. The technique was later used as a proof theoretical tool to show the NP-completeness of BV [10]. In [11], we have refined this technique to reduce the nondeterminism in MLL and classical logic. This refinement, for example, reduces the number of rule instances in the formulae above from 6 to 2 and from 42 to 3, respectively.

We extend and refine our technique further to control the nondeterminism in multiplicative exponential linear logic (MELL) in deep inference [19]. In the following, besides the nondeterminism in the commutative contexts resulting from the interaction of conjunctions and disjunctions, we apply our technique to the exponentials of MELL. For this, we refine the promotion rule that is responsible for managing the interactions in the contexts that contain formulae with exponentials. This way, we introduce a control mechanism that is sensitive to complementarity of the subformulae, on which the promotion rule acts, similar to the mechanism that we use for the switch rule. The inference rules refined this way act only on pairs of subformulae that have complementary atoms. We show that introducing this mechanism simultaneously on the switch and promotion rules preserves completeness and access to short proofs, while providing a significant reduction in nondeterminism as it has been the case for MLL and classical logic [11].

We prove the correctness of the technique by using a proof theoretical method, that is, *splitting*, [7, 19, 9] originally introduced for proving cut elimination, which also serves as a certificate of formal hygiene. In this respect, besides the reduction in nondeterminism that should impact proof search applications, the systems we present here are of independent theoretical interest from the point of view of the proof theory of MELL; the decision problem for this logic is an open problem and similar methods were used to address NP-hardness of BV [10].

We conclude with a discussion on illustrative examples, and quantify the improvement in accessing the shorter proofs with the aid of experiments with our implementations on these examples. The implementations [16] of all these systems are available at our website.<sup>1</sup>

## 2 MELL with Deep Inference

In deep inference deductive systems, the laws such as associativity and commutativity are imposed on the logical expressions by means of an underlying equational system [7, 17, 19, 2, 21, 3, 9, 20]. This is done by working with congruence classes of formulae instead of formulae.

**Definition 1.** *There are countably many atoms, denoted by  $a, b, c, \dots$ . The formulae  $P, Q, R, \dots$  of multiplicative exponential linear logic are generated by*

$$R ::= a \mid 1 \mid \perp \mid [R \wp R] \mid (R \otimes R) \mid !R \mid ?R \mid \bar{R} \quad ,$$

where  $a$  stands for any atom.  $1$  and  $\perp$  are special atoms, called one and bottom. A formula  $[R \wp R]$  is a par formula,  $(R \otimes R)$  is a copar (times) formula,  $!R$  is a of-course formula,  $?R$  is a why-not formula, and  $\bar{R}$  is the negation of  $R$ . Different kinds of brackets for par and copar are used to enhance readability. Formulae are considered to be equivalent modulo the relation  $\approx$ , which is the smallest congruence relation induced by the equational system consisting of the equations for associativity and commutativity for par and copar, and the equations

$$\overline{[R \wp T]} \approx (\bar{R} \otimes \bar{T}), \quad \overline{(R \otimes T)} \approx [\bar{R} \wp \bar{T}], \quad \overline{?R} \approx !\bar{R}, \quad \overline{!R} \approx ?\bar{R}, \quad \overline{\bar{R}} \approx R, \quad \overline{1} \approx 1, \quad \overline{\perp} \approx \perp$$

for negation. We denote the formulae in the same equivalence class by picking a formula from the equivalence class. If there is no ambiguity, we drop the superfluous brackets in the formulae by resorting to the equations for associativity. We consider the formulae to be in negation normal form, which are obtained by applying the equations for negation above from left to right exhaustively. This allows us to disregard the equations for negation.

**Definition 2.** A formula context,  $S\{ \}$ , is a formula with a hole.  $R$  is a subformula of  $S\{R\}$  and  $S\{ \}$  is its context. A context is empty if it is  $\{ \}$ . Context braces are omitted if no ambiguity is possible. For example, if  $S\{ \} = \{ \} \wp b$ , and  $R = (\bar{b} \otimes \perp)$ ,  $S\{R\} = [(\bar{b} \otimes \perp) \wp b]$ .

**Definition 3.** A deep inference rule is scheme of the kind  $\rho \frac{S\{T\}}{S\{R\}}$ , where  $\rho$  is the name of the rule,  $S\{T\}$  is its premise, and  $S\{R\}$  is its conclusion. Such a rule determines the linear implication  $T \multimap R$ , or equivalently  $[\bar{T} \wp R]$ , inside a generic context  $S\{ \}$ . In an instance of  $\rho$ , we say that  $R$  is the redex and  $T$  is the contractum. A system  $\mathcal{S}$  is a set of inference rules.

**Definition 4.** The rules of the deep inference multiplicative Exponential Linear logic System with unit rules or ELSu are depicted in Figure 1. The rules are called atomic interaction ( $\text{ai}\downarrow$ ), switch ( $\text{s}$ ), promotion ( $\text{p}\downarrow$ ), absorption ( $\text{b}\downarrow$ ), digging ( $\text{g}\downarrow$ ), weakening ( $\text{w}\downarrow$ ), empty ( $\text{e}\downarrow$ ), unit-one ( $\text{u}_1\downarrow$ ), and unit-two ( $\text{u}_2\downarrow$ ).

<sup>1</sup>[http://sites.google.com/site/ozankahramanogullari/software/di\\_in\\_maude](http://sites.google.com/site/ozankahramanogullari/software/di_in_maude)

|   |  |   |  |  |
|---|--|---|--|--|
| $\text{ai}\downarrow \frac{S\{1\}}{S[a \wp \bar{a}]}$ | $\text{s} \frac{S([U \wp R] \otimes T)}{S[U \wp (R \otimes T)]}$ | $\text{p}\downarrow \frac{S\{![R \wp T]\}}{S\{![R \wp ?T]\}}$ | $\text{b}\downarrow \frac{S\{?R \wp R\}}{S\{?R\}}$   |  |
| $\text{g}\downarrow \frac{S\{??R\}}{S\{?R\}}$         | $\text{w}\downarrow \frac{S\{\perp\}}{S\{?R\}}$                  | $\text{e}\downarrow \frac{S\{1\}}{S\{!1\}}$                   | $\text{u}_1\downarrow \frac{S\{R\}}{S[R \wp \perp]}$ | $\text{u}_2\downarrow \frac{S\{R\}}{S(R \otimes 1)}$ |

Figure 1: Deep Inference System ELSu

An inference rule of the form in Definition 3 specifies a step of rewriting. In this paper, these rewritings are those that rewrite the conclusion to the premise. This is because we consider the inference rules for proof-search, thus we consider their bottom-up applications, which result in proofs that grow from the conclusion to the premise.

**Definition 5.** A derivation  $\Delta$  is a finite chain of instances of inference rules. A derivation can consist of just one formula. The top-most formula in a derivation is called the premise, and the bottom-most formula is called its conclusion. A derivation  $\Delta$  whose premise is  $T$ , conclusion is  $R$ , and inference rules are in  $\mathcal{S}$  will be written as  $\mathcal{S} \parallel \frac{T}{\Delta}$ . A proof  $\Pi$  is either the unit 1 or a finite derivation whose premise is the unit 1.

**Definition 6.** Two systems  $\mathcal{S}$  and  $\mathcal{S}'$  are equivalent if they prove the same formulae. A rule  $\rho$  is admissible for a system  $\mathcal{S}$  if for every proof in  $\mathcal{S} \cup \{\rho\}$  there is a proof in  $\mathcal{S}$  with the same conclusion. A rule  $\rho$  permutes over a rule  $\beta$  if for every derivation  $\frac{\beta \frac{Q}{U}}{\rho \frac{Q}{P}}$  there exists  $\frac{\rho \frac{Q}{V}}{\beta \frac{Q}{P}}$ .

The deep inference presentation of multiplicative exponential linear logic that we use here differs in two ways from system ELS introduced in [19]. The first difference is that we employ an explicit treatment of the units with the inference rules *unit-one* ( $\text{u}_1\downarrow$ ), and *unit-two* ( $\text{u}_2\downarrow$ ), instead of equations for unit as in system ELS. The second is in the treatment of exponentials, which are again done by the inference rules *digging* ( $\text{g}\downarrow$ ) and *empty* ( $\text{e}\downarrow$ ) as in [14, 9], instead of equations, whereas the presentation in [19] employs equations instead of these rules.

**Theorem 7.** A formula  $R$  has a proof in ELSu if and only if it has a proof in MELL.

*Proof.* The soundness of ELSu rules is proven as in [19] by showing that each rule of ELSu, as a linear implication, is a theorem in MELL. The completeness of ELSu for multiplicative exponential linear logic is shown by inductively translating MELL proofs in the sequent calculus into ELSu proofs, similar to the proof in [19].  $\square$

**Proposition 8.** For any formula  $R$  and context  $S$ ,  $S\{R\}$  has a proof in ELSu if and only if (i)  $S[R \wp \perp]$  has a proof; and (ii)  $S(R \otimes 1)$  has a proof.

*Proof.* If  $R$  has a proof, we prove the statements (i) and (ii) with instances of the rules  $\text{u}_1\downarrow$  and  $\text{u}_2\downarrow$ . To show that these rules are invertible, in ELSu we prove  $\{[R \wp \perp] \multimap R\} \otimes \{R \multimap [R \wp \perp]\}$  and  $\{(R \otimes 1) \multimap R\} \otimes \{R \multimap (R \otimes 1)\}$ .  $\square$

### 3 Refining Context Management

The deep inference presentation of MELL in [19] employs equations for the treatment of units. Although this can bring an ease in the proof theoretic analysis, it has implications that result in an increase in non-determinism in proof search. We have elaborated on this with examples in [11] within the setting of MLL, and thus replaced these equations with inference rules. We have above analogously introduced a controlled treatment of units via inference rules  $u_1\downarrow$  and  $u_2\downarrow$  in system ELSu. However, an important part of the non-determinism in proof search with deep inference systems is due to the context management rules, which are the rules  $s$  and  $p\downarrow$  in ELSu. Below, we incrementally reduce the nondeterminism introduced by these rules by exploiting the interaction pattern among logical expressions, presented in [7] and used in [10, 11].

#### 3.1 Non-determinism and the Switch Rule

As the first step, we re-design the switch rule in ELSu to control the non-determinism in proof search by extending and refining the procedure for multiplicative linear logic in [10, 11].

**Definition 9.** *Given a formula  $R$ ,  $\text{at } R$  is the set of the atoms in  $R$ .*

**Example 10.** *For  $R = [a \wp \bar{a} \wp b \wp (\bar{a} \otimes \perp) \wp (a \otimes \bar{b})]$ , we have  $\text{at } R = \{a, \bar{a}, b, \bar{b}, \perp\}$ .*

**Definition 11.** *For any two formulae  $R$  and  $U$ , we say that  $R$  and  $U$  can interact if  $\text{at } \bar{R} \cap \text{at } U \neq \emptyset$ , that is,  $R$  and  $U$  contain complementary atoms.*

**Definition 12.** [11] *An instance of the switch rule as in Figure 1*

1. *is an instance of interaction switch (is) if  $R$  and  $U$  can interact,*
2. *it is lazy switch (ls) if  $U$  is not a par,*
3. *it is deep switch (ds) if  $R$  is not a copar.*

*By combining these restrictions, we thus obtain 7 new inference rules. For example, by imposing all these restrictions, we obtain the rule deep lazy interaction switch (dlis). The systems obtained by replacing in ELSu the switch rule with one of these 7 rules are denoted by replacing the suffix  $u$  of ELSu with the prefix of the switch rule. For example, the system ELSu with deep lazy interaction switch, denoted with ELSdli, is  $\{a_i\downarrow, \text{dlis}, p\downarrow, b\downarrow, g\downarrow, w\downarrow, e\downarrow, u_1\downarrow, u_2\downarrow\}$ .*

Let us now discuss the contribution of the rules  $ls$  and  $ds$  in reducing the nondeterminism. Consider the rule  $s$ , where  $U$  is a par formula with  $n$  disjuncts. In an instance of  $s$ , because every combination of  $n$  disjuncts can match  $U$ , this results in  $2^n$  different choices for the redex, that is, the size of the powerset of a set with  $n$  elements. The rule  $ls$  reduces this to  $n$  different choices. The same explanation applies to the rule  $ds$  for the part of the redex that matches  $R$ . If  $R$  is a copar with  $m$  conjuncts, the rule  $ds$  reduces the number of choices from  $2^m$  to  $m$ . We refer to [11] for examples on this. We now show that replacing the rule  $s$  with the 7 alternatives result in equivalent systems. We start with introducing notions from [19, 20] that we use below.

**Definition 13.** *We define system cELSd, that is, the core system ELSd, to be the system obtained by removing the rules  $\{g\downarrow, b\downarrow, w\downarrow\}$  from system ELSd. We define core systems for all the other systems similarly. For example, cELSdli denotes the core system obtained by removing the rules  $\{g\downarrow, b\downarrow, w\downarrow\}$  from cELSdli.*

The following theorem is a refined version of a theorem in [18] and another similar result in [20]. Here, we additionally emphasize the role played by the units and exponentials in terms of explicit inference rules instead of equations as it is the case in [18, 19, 20].

**Theorem 14.** *Every proof  $\Pi$  in ELSu can be rewritten as shown below*

$$\begin{array}{c}
 \frac{1}{\text{ELSu} \parallel \Pi} \\
 R
 \end{array}
 \xrightarrow{i.}
 \begin{array}{c}
 \frac{1}{\text{cELSu} \parallel} \\
 R_3 \\
 \{w\downarrow\} \parallel \\
 R_2 \\
 \{b\downarrow\} \parallel \\
 R_1 \\
 \{g\downarrow\} \parallel \\
 R
 \end{array}
 \xrightarrow{ii.}
 \begin{array}{c}
 \frac{1}{\text{cELSD} \parallel} \\
 R_3 \\
 \{w\downarrow\} \parallel \\
 R_2 \\
 \{b\downarrow\} \parallel \\
 R_1 \\
 \{g\downarrow\} \parallel \\
 R
 \end{array}
 \xrightarrow{iii.}
 \begin{array}{c}
 \frac{1}{\{e\downarrow, u_1\downarrow, u_2\downarrow\} \parallel} \\
 R_5 \\
 \{a\downarrow, u_1\downarrow, u_2\downarrow\} \parallel \\
 R_4 \\
 \{ds, p\downarrow, u_1\downarrow, u_2\downarrow\} \parallel \\
 R_3 \\
 \{w\downarrow, b\downarrow, g\downarrow\} \parallel \\
 R
 \end{array}$$

*Proof.* As in [18, 19, 20], (i.) permute the instances of the inference rules  $g\downarrow$ ,  $b\downarrow$ ,  $w\downarrow$  down in the proof; (ii.) permute all the instances of  $s$  that are not instances of the rule  $ds$  up in the proof until they become  $ds$ , (iii.) permute all the instances of  $a\downarrow$ , and  $e\downarrow$  up in the proof. The only cases that are not covered in [18, 19, 20] are the permutation of the rules  $u_1\downarrow$  and  $u_2\downarrow$ , which trivially permute up in the proof above  $g\downarrow$ ,  $b\downarrow$ , and  $w\downarrow$ , since they do not modify contexts.  $\square$

**Corollary 15.** [19, 18] *Systems ELSu and ELSd are equivalent.*

**Proposition 16.** (i) *Systems cELSu, cELSD, cELSI, and cELSdl are equivalent.* (ii) *Systems ELSu, ELSd, ELSI, and ELSdl are equivalent.*

*Proof.* We prove (i), since (ii) follows from (i): every proof in cELSdl is a proof in cELSD and cELSI, and every proof in cELSD or cELSI is a proof in cELSu. For the proof of the other direction, we transform proofs in cELSD into proofs in cELSdl, which by Corollary 15 proves the statement. For this, it suffices to show for every instance of the rule  $ds$ , where  $U$  consists of  $n \geq 1$  disjuncts, we can replace it with  $n$  instances of  $dls$  as follows.

$$ds \frac{([R \wp U_1 \wp U_2 \wp \dots \wp U_n] \otimes T)}{[(R \otimes T) \wp U_1 \wp U_2 \wp \dots \wp U_n]} \sim \frac{dls \frac{([R \wp U_1 \wp \dots \wp U_n] \otimes T)}{\vdots}}{dls \frac{([(R \wp U_1] \otimes T) \wp U_2 \wp \dots \wp U_n]}{[(R \otimes T) \wp U_1 \wp U_2 \wp \dots \wp U_n]}}$$

$\square$

We postpone the discussion on the completeness of ELSdli; below we prove a more powerful result, which includes this statement.

### 3.2 Non-determinism and the Promotion Rule

**Definition 17.** *We define the rules interaction promotion ( $ip\downarrow$ ) and encourage ( $ir\downarrow$ ),*

$$ip\downarrow \frac{S\{![R \wp T]\}}{S\{!R \wp ?T\}} \quad \text{and} \quad ir\downarrow \frac{S\{?[R \wp T]\}}{S\{?R \wp ?T\}},$$

which require that  $R$  and  $T$  can interact. Any system obtained by replacing the rule  $\mathfrak{p}\downarrow$  with the rules  $\text{ip}\downarrow$  and  $\text{ir}\downarrow$  is denoted by adding the suffix  $\mathfrak{p}$  to the end of the name of the system. For example,  $\text{ELSdlip}$  is  $\{\text{ai}\downarrow, \text{dlis}, \text{ip}\downarrow, \text{ir}\downarrow, \text{b}\downarrow, \text{g}\downarrow, \text{w}\downarrow, \text{e}\downarrow, \text{u}_1\downarrow, \text{u}_2\downarrow\}$ .

**Proposition 18.** *The rule  $\text{ir}\downarrow$  is sound for MELL.*

*Proof.* Follows from the fact that the rule  $\text{r}\downarrow$ , that is, the rule  $\text{ir}\downarrow$  without the interaction condition, is derivable for switch, interaction, promotion and cut rules as shown in [18].  $\square$

The interaction requirements imposed by the context management rules in Definition 12 and Definition 17 conserve the locality of the inference rules. This is because the computational cost of checking the inference steps with these conditions is linear in the size of the formula as it is the case without them. Moreover, these conditions bring about a drastic reduction in non-determinism, while conserving the shorter proofs. Our main result, given with the following theorem, states that the introduction of the conditions does not result in a loss of completeness. In what follows, we provide a constructive proof of this theorem that is closely related with a cut elimination technique that is commonly used for deep inference systems [7, 19, 18, 9, 10, 11]. In particular, we adopt the case analysis on exponentials in [9], whereby the interaction condition requires a more refined treatment, and cut elimination becomes a corollary.

**Theorem 19.** *Systems  $\text{ELSdlip}$  and  $\text{ELSu}$  are equivalent, and thus all the 14 intermediate systems defined in Definition 12 and Definition 17 are equivalent.*

Let us now proceed with the lemmata that we use in the proof of Theorem 19.

**Proposition 20.** *In  $\text{ELSdlip}$ , a formula  $(R \otimes T)$  has a proof if and only if  $R$  and  $T$  have proofs.*

*Proof.* If direction being trivial, for the only if direction construct the proofs of  $R$  and  $T$  by induction on the length of the proof of  $(R \otimes T)$ .  $\square$

**Lemma 21** (Independence for Modalities). *For any formulae  $R, U, P_1, \dots, P_h$ , there exists a derivation of the form*

$$\frac{!U}{\text{cELSdlip} \parallel} \frac{[!R \wp ?P_1 \wp \dots \wp ?P_h]}{\text{if and only if there exists a derivation}} \frac{U}{\text{cELSdlip} \parallel \Delta} \frac{[R \wp P_1 \wp \dots \wp P_h]}{.}$$

*Proof.* For the if part, we inductively remove the instances of the rules  $\text{ip}\downarrow, \text{ir}\downarrow$  that have the formulae  $R, T_1, \dots, T_h$  at their redex, and this way obtain the derivation  $\Delta$ . We prove the only if part by structural induction on the lexicographic ordering of the number of atoms in the conclusion of  $\Delta$  and the length of  $\Delta$ . The base case is given with either  $P_1 = \dots = P_h = \perp$  and  $R = 1$  or, without loss of generality,  $P_h = 1$  and  $R = P_1 = \dots = P_{h-1} = \perp$ . We single out the bottom most rule instance  $\rho$  in  $\Delta$  and do case analysis on the position of the redex of  $\rho$ .

1. If  $\rho$  is  $\text{u}_2\downarrow$  or  $\text{e}\downarrow$  or  $\text{ir}\downarrow$  or  $\text{ip}\downarrow$ , the only possibility for the redex is that it can be inside one of  $R, P_1, \dots, P_h$ , which result in similar cases. We prove for the case, where  $\rho$  is inside  $R$ :

$$\text{we have } \frac{U}{\parallel} \frac{[R' \wp P_1 \wp \dots \wp P_h]}{\rho \frac{[R \wp P_1 \wp \dots \wp P_h]}} \text{ . Then we prove with } \frac{!U}{\parallel \Delta'} \frac{[!R' \wp ?P_1 \wp \dots \wp ?P_h]}{\rho \frac{[!R \wp ?P_1 \wp \dots \wp ?P_h]}} ,$$

where  $\Delta'$  is delivered by the induction hypothesis.



2. If  $\rho$  is  $u_1\downarrow$ , we can apply the argument in case (1). The only case, which does not follow this pattern, occurs as follows: for any  $Q, T \in \{R, P_1, \dots, P_h\}$  and  $T'$  such that  $T = [T' \wp \perp]$  and the redex of  $u_1\downarrow$  is  $[Q \wp \perp]$ . We replace this with an instance of  $u_1\downarrow$  that takes  $T$  as its redex and  $T'$  as the contractum.
3. If  $\rho$  is  $ai\downarrow$  or  $dliis$ , we have similar cases. We prove for  $dliis$ , which is the more involved case. If the redex of  $dliis$  is inside any of  $R, P_1, \dots, P_h$ , we have a situation similar to case 1. Otherwise, the redex of  $dliis$  must consist of the subformulae of two formulae  $Q, T \in \{R, P_1, \dots, P_h\}$  such that  $Q = [Q_1 \wp Q_2]$  and  $T = [(T_1 \otimes T_2) \wp T_3]$  with

$$dliis \frac{[Q_2 \wp ((Q_1 \wp T_1) \otimes T_2) \wp T_3 \wp \dots]}{[Q_1 \wp Q_2 \wp (T_1 \otimes T_2) \wp T_3 \wp \dots]},$$

where  $T_1$  and  $Q_1$  can interact. For the case, where  $Q = R$  and  $T \in \{P_1, \dots, P_h\}$ , we build

$$dliis \frac{\begin{array}{c} !U \\ \parallel_{\Delta'} \\ [! [Q_2 \wp ((Q_1 \wp T_1) \otimes T_2) \wp T_3] \wp ?P_1 \wp \dots \wp ?P_{h-1}] \end{array}}{ip\downarrow \frac{[! [Q_1 \wp Q_2] \wp ?[(T_1 \otimes T_2) \wp T_3] \wp ?P_1 \wp \dots \wp ?P_{h-1}]}{[! [Q_1 \wp Q_2] \wp ?[(T_1 \otimes T_2) \wp T_3] \wp ?P_1 \wp \dots \wp ?P_{h-1}]}}},$$

where  $Q_1$  and  $T_1$  can interact and  $\Delta'$  is delivered by the induction hypothesis. For the case, where  $Q, T \in \{P_1, \dots, P_h\}$ , we replace the instance of  $ip\downarrow$  with an instance of  $ir\downarrow$ .  $\square$

Following two lemmata are extensions of similar results in [11] for multiplicative linear logic to the setting that includes the rules  $ip\downarrow$ ,  $ir\downarrow$  and  $e\downarrow$ , which result in the core system  $cELSDlip$ . Because these results concern the formulae that do not involve exponentials, and the rules  $ip\downarrow$ ,  $ir\downarrow$  and  $e\downarrow$  take formulae with exponentials in their redex, the proof arguments in [11] easily extend to the setting of  $cELSDlip$ .

**Lemma 22** (Independence for Copar). *For any formulae  $P, U$  and  $R$ , if  $[P \wp U]$  has a proof*

$$in \ cELSDlip, \ then \ there \ is \ a \ derivation \ \frac{R}{cELSDlip \parallel} \frac{R}{[(R \otimes P) \wp U]}.$$

*Proof.* Let  $\Pi$  be the proof of the formula  $[P \wp U]$ . We prove by structural induction on the lexicographic ordering of the number of atoms in the conclusion of  $\Pi$  and the length of  $\Pi$ . The base case is given with (i)  $P = \perp$  and  $U = 1$  or (ii)  $P = 1$  and  $U = \perp$  or (iii)  $P = a$  and  $U = \bar{a}$  or vice versa. For the inductive cases, we single out the bottom most rule instance  $\rho$  in the proof and do case analysis on the position of the redex of  $\rho$ . If  $\rho$  is  $e\downarrow$  or  $ir\downarrow$  or  $ip\downarrow$ , the only possibility for the redex is that it can be inside  $P$  or  $U$ , which result in similar cases. We prove for  $P$ . If the redex of  $\rho$  is inside  $P$ ,

$$we \ have \ \frac{1}{\rho} \frac{[P' \wp U]}{[P \wp U]} \ . \quad \text{Then we prove with} \quad \frac{R}{\rho} \frac{\begin{array}{c} R \\ \parallel_{\Delta} \\ [(R \otimes P') \wp U] \end{array}}{[(R \otimes P) \wp U]},$$

where  $\Delta$  is delivered by the induction hypothesis. The other cases are as in the proof of a similar result for multiplicative linear logic (MLSdli) in [11], where  $MLSdli = \{ai\downarrow, dliis, u_1\downarrow, u_2\downarrow\}$ .  $\square$



**Lemma 23** (Splitting for Copar). *For any formulae  $R, T, P$ , if  $[(R \otimes T) \wp P]$  is provable in  $\text{cELSdli}$ , then (i) either  $[R \wp P]$  and  $T$  are provable in  $\text{cELSdli}$ , (ii) or  $[T \wp P]$  and  $R$  are provable in  $\text{cELSdli}$ , (iii) or there are formulae  $P_R$  and  $P_T$ , such that*

$$\begin{array}{c} [P_R \wp P_T] \\ \text{cELSdli} \parallel \\ P \end{array}, \quad \begin{array}{c} 1 \\ \text{cELSdli} \parallel \\ [R \wp P_R] \end{array} \quad \text{and} \quad \begin{array}{c} 1 \\ \text{cELSdli} \parallel \\ [T \wp P_T] \end{array}.$$

*Proof.* We prove by structural induction on the lexicographic ordering of the number of atoms in the conclusion of the proof and the length of the proof. The base case is trivially covered by (i) or (ii). For the inductive cases, we single out the bottom most rule instance  $\rho$  in the proof and do case analysis on the position of the redex of  $\rho$ . If  $\rho$  is  $\text{e}\downarrow$  or  $\text{ir}\downarrow$  or  $\text{ip}\downarrow$ , the only possibility for the redex is that it can be inside  $R$  or  $T$  or  $P$ . If the redex of  $\rho$  is inside  $R$  we have the following case, that is, if

$$\rho \frac{\begin{array}{c} 1 \\ \parallel \\ [(R' \otimes T) \wp P] \end{array}}{[(R \otimes T) \wp P]} \quad \text{then} \quad \rho \frac{\begin{array}{c} 1 \\ \parallel_{\Pi_1} \\ [R' \wp P] \end{array}}{[R \wp P]}, \quad \rho \frac{\begin{array}{c} 1 \\ \parallel_{\Pi_2} \\ R' \end{array}}{R} \quad \text{and} \quad \rho \frac{\begin{array}{c} 1 \\ \parallel_{\Pi_3} \\ [R' \wp P_1] \end{array}}{[R \wp P_1]},$$

where  $\Pi_1, \Pi_2$  and  $\Pi_3$  are delivered by the induction hypothesis for (i), (ii) and (iii). If the redex of  $\rho$  is inside  $T$  or  $P$ , we have a similar situation. The other cases, where  $\rho \in \text{MLSdli} = \{\text{ai}\downarrow, \text{dlis}, \text{u}_1\downarrow, \text{u}_2\downarrow\}$ , rely on Lemma 22, and are as in the proof of a similar result for multiplicative linear logic ( $\text{MLSdli}$ ) in [11].  $\square$

**Lemma 24** (Splitting for Modality !). *For any formulae  $R$  and  $P$ , if  $[!R \wp P]$  is provable in  $\text{cELSdli}$ , then (i) either there are formulae  $P_1, \dots, P_h$  for some  $h \geq 1$ , such that*

$$\begin{array}{c} [?P_1 \wp \dots \wp ?P_h] \\ \text{cELSdli} \parallel \\ P \end{array} \quad \text{and} \quad \begin{array}{c} 1 \\ \text{cELSdli} \parallel \\ [R \wp P_1 \wp \dots \wp P_h] \end{array} \quad \text{(i) or} \quad \begin{array}{c} \perp \\ \text{cELSdli} \parallel \\ P \end{array} \quad \text{and} \quad \begin{array}{c} 1 \\ \text{cELSdli} \parallel \\ R \end{array}.$$

*Proof.* Let  $\Pi$  be the proof of the formula  $[!R \wp P]$ . We prove by structural induction on the lexicographic ordering of the number of atoms in the conclusion of  $\Pi$  and the length of  $\Pi$ . The base case is given with  $P = \perp$ . For the inductive cases, we single out the bottom most rule instance  $\rho$  in  $\Pi$  and do case analysis on the position of the redex of  $\rho$ .

1. If  $\rho$  is  $\text{ai}\downarrow$  or  $\text{ir}\downarrow$  or  $\text{u}_1\downarrow$  or  $\text{u}_2\downarrow$  (assuming  $P \neq \perp$ ), the only possibility for the redex is that it can be inside  $R$  or  $P$ , which result in similar cases. We prove for  $R$ . If the redex of  $\rho$  is inside  $R$ , that is, if

$$\rho \frac{\begin{array}{c} 1 \\ \parallel \\ [!R' \wp P] \end{array}}{[!R \wp P]} \quad \text{then} \quad \rho \frac{\begin{array}{c} 1 \\ \parallel_{\Pi_1} \\ R' \end{array}}{R} \quad \text{and} \quad \rho \frac{\begin{array}{c} 1 \\ \parallel_{\Pi_2} \\ [R' \wp P_1 \wp \dots \wp P_h] \end{array}}{[R \wp P_1 \wp \dots \wp P_h]},$$

where  $\Pi_1$  and  $\Pi_2$  are delivered by the induction hypothesis for (i) and (ii).

2. If  $\rho$  is *dis* there are two possibilities: if the redex is inside  $R$  or  $P$ , we have a situation similar to the one in case (1). Otherwise, the only possibility is that  $!R$  is inside the redex of *dis*. In this case, for some  $P_1, P_2, P_3$ , either  $P = [(P_1 \otimes P_2) \wp P_3]$  or  $P = (P_1 \otimes P_2)$ . We prove the former, which is more general. For proving the latter, it suffices to apply Proposition 20 to the premise of *dis*. We thus have

$$\text{dis} \frac{[[(!R \wp P_1) \otimes P_2) \wp P_3]}{[!R \wp (P_1 \otimes P_2) \wp P_3]} .$$

We apply Lemma 23 to the premise and obtain three cases: either (i)  $P_2$  and  $[!R \wp P_1 \wp P_3]$  have proofs, or (ii)  $[!R \wp P_1]$  and  $[P_2 \wp P_3]$  have proofs, or

$$(iii) \quad \begin{array}{c} [Q_1 \wp Q_2] \\ \parallel_{\Delta_1} \\ P_3 \end{array} , \quad \begin{array}{c} 1 \\ \parallel_{\Pi_1} \\ [!R \wp P_1 \wp Q_1] \end{array} \quad \text{and} \quad \begin{array}{c} 1 \\ \parallel_{\Pi_2} \\ [P_2 \wp Q_2] \end{array} .$$

We prove for (iii) as this is the more involved case, and other cases are similar. We apply the induction hypothesis to the proof  $\Pi_1$ , as the conclusion has less number of atoms, and obtain one of the two cases of the lemma. For (i), we obtain

$$\begin{array}{c} [?P'_1 \wp \dots \wp ?P'_h] \\ \parallel_{\Delta_2} \\ [P_1 \wp Q_1] \end{array} \quad \text{and} \quad \begin{array}{c} 1 \\ \parallel_{\Pi_3} \\ [R \wp P'_1 \wp \dots \wp P'_h] \end{array} \quad \text{and build} \quad \begin{array}{c} [?P'_1 \wp \dots \wp ?P'_h] \\ \parallel_{\Delta_2} \\ [P_1 \wp Q_1] \\ \parallel_{\Delta'_2} \\ [(P_1 \otimes P_2) \wp Q_1 \wp Q_2] \\ \parallel_{\Delta_1} \\ [(P_1 \otimes P_2) \wp P_3] \end{array}$$

where  $\Delta'_2$  is delivered by Lemma 22 with proof  $\Pi_2$ . The proof of (ii) is obtained by replacing  $[?P'_1 \wp \dots \wp ?P'_h]$  with  $\perp$  above, since  $R$  has a proof.

3. If  $\rho$  is *ip* $\downarrow$  there are two possibilities: if the redex is inside  $R$  or  $P$ , we have a situation similar to the one in case (1). Otherwise, the only possibilities are (i)  $P = ?P_1$ , where there is nothing to prove, and (ii)  $P = [?P_1 \wp Q_2]$ , thus

$$\text{ip}\downarrow \frac{[![R \wp P_1] \wp Q_2]}{[!R \wp ?P_1 \wp Q_2]} .$$

We apply the induction hypothesis to the proof of the premise and obtain, as the conclusion has a shorter proof, one of the two cases in the statement of the lemma. For (i), we obtain

$$\begin{array}{c} [?P_2 \wp \dots \wp ?P_h] \\ \parallel_{\Delta} \\ Q_2 \end{array} \quad \text{and} \quad \begin{array}{c} 1 \\ \parallel \\ [R \wp P_1 \wp P_2 \wp \dots \wp P_h] \end{array} \quad \text{and build} \quad \begin{array}{c} [?P_1 \wp \dots \wp ?P_h] \\ \parallel_{\Delta} \\ [?P_1 \wp Q_2] \end{array} .$$

For (ii), it suffices to use the proof of  $[R \wp P_1]$  and replace  $[?P_2 \wp \dots \wp ?P_h]$  in  $\Delta$  with  $\perp$  to build the desired derivation with an instance of *u* $\downarrow$  as the top-most rule instance.

4. If  $\rho$  is  $e\downarrow$  there are two possibilities: if the redex is inside  $R$  or  $P$ , we have a situation similar to the one in case (1). Otherwise, the only possibility is the one where  $R = 1$ .

□

**Lemma 25** (Splitting for Modality  $?$ ). *For any formulae  $R$  and  $P$ , if  $[?R \wp P]$  is provable in  $\text{cELSdlip}$ , then there are formulae  $T, Q_1, \dots, Q_h$ , such that*

$$\frac{\frac{1}{\text{cELSdlip}} \left\| \begin{array}{c} [?R \wp !T \wp ?Q_1 \wp \dots \wp ?Q_h] \end{array} \right.}{\frac{1}{\text{cELSdlip}} \left\| \begin{array}{c} [?R \wp P] \end{array} \right.} .$$

*Proof.* We prove by structural induction as in Lemma 24. We single out the bottom most rule instance  $\rho$  in the proof and do case analysis on the position of the redex of  $\rho$ .

1. If  $\rho$  is  $ai\downarrow$  or  $u_1\downarrow$  or  $u_2\downarrow$ , the situation is similar to the case (1) of the proof of Lemma 24.
2. If  $\rho$  is  $d\text{lis}$ , and the redex is inside  $R$  or  $P$ , we have a situation similar to the one in case (1). Otherwise, the only possibility is that  $?R$  is inside the redex of  $d\text{lis}$  as in the case (2) of the proof of Lemma 24. Again, we consider the most general case, where  $P = [(P_1 \otimes P_2) \wp P_3]$ , and apply Lemma 23 to the premise of  $d\text{lis}$ . We then obtain three cases: either (i)  $P_2$  and  $[?R \wp P_1 \wp P_3]$  are provable, or (ii)  $[?R \wp P_1]$  and  $[P_2 \wp P_3]$  are provable, or

$$(iii) \quad \frac{[K_1 \wp K_2]}{\left\| \begin{array}{c} \Delta_1 \\ P_3 \end{array} \right.}, \quad \frac{1}{\left\| \begin{array}{c} \Pi_1 \\ [?R \wp P_1 \wp K_1] \end{array} \right.} \quad \text{and} \quad \frac{1}{\left\| \begin{array}{c} \Pi_2 \\ [P_2 \wp K_2] \end{array} \right.} .$$

We prove for (iii). Other cases being similar, this is the more involved case. The conclusion of  $\Pi_1$  has less number of atoms, so we apply the induction hypothesis to  $\Pi_1$ , and obtain

$$\frac{\frac{1}{\left\| \begin{array}{c} \Pi_3 \\ [?R \wp !T \wp ?Q_1 \wp \dots \wp ?Q_h] \end{array} \right.}}{\left\| \begin{array}{c} \Delta_3 \\ [?R \wp P_1 \wp K_1] \end{array} \right.} \quad \text{and build} \quad \frac{\frac{1}{\left\| \begin{array}{c} \Pi_3 \\ [?R \wp !T \wp ?Q_1 \wp \dots \wp ?Q_h] \end{array} \right.}}{\left\| \begin{array}{c} \Delta_3 \\ [?R \wp P_1 \wp K_1] \end{array} \right.}, \quad \frac{\frac{1}{\left\| \begin{array}{c} \Pi_2 \\ [P_2 \wp K_2] \end{array} \right.}}{\left\| \begin{array}{c} \Delta_2 \\ [?R \wp (P_1 \otimes P_2) \wp K_1 \wp K_2] \end{array} \right.}, \quad \frac{\frac{1}{\left\| \begin{array}{c} \Pi_1 \\ [?R \wp P_1 \wp K_1] \end{array} \right.}}{\left\| \begin{array}{c} \Delta_1 \\ [?R \wp (P_1 \otimes P_2) \wp P_3] \end{array} \right.},$$

where  $\Delta_2$  is delivered by Lemma 22 with proof  $\Pi_2$ .

3. If  $\rho$  is  $ip\downarrow$  there are two possibilities: if the redex is inside  $R$  or  $P$ , we have a situation similar to the one in case (1). Otherwise, the only possibilities are (i)  $P = !P_1$ , where there is nothing to prove, and (ii)  $P = [!P_1 \wp P_2]$ , thus

$$ip\downarrow \frac{[![R \wp P_1] \wp P_2]}{[?R \wp !P_1 \wp P_2]} .$$

We apply Lemma 24 to the premise and obtain two cases of the lemma. For (i), we have

$$\begin{array}{c} [?Q_1 \wp \dots \wp ?Q_h] \\ \parallel_{\Delta} \\ P_2 \end{array}, \quad \begin{array}{c} 1 \\ \parallel_{\Pi} \\ [R \wp P_1 \wp Q_1 \wp \dots \wp Q_h] \end{array} \quad \text{and build} \quad \begin{array}{c} 1 \\ \parallel_{\Pi} \\ [?R \wp !P_1 \wp ?Q_1 \wp \dots \wp ?Q_h] \\ \parallel_{\Delta} \\ [?R \wp !P_1 \wp P_2] \end{array}$$

where  $\Pi'$  is delivered by Lemma 21 with proof  $\Pi$ . For (ii), since we have a proof of  $[R \wp P_1]$ , to prove the result, we replace  $[?Q_1 \wp \dots \wp ?Q_h]$  in the premise of  $\Delta$  with  $\perp$ , and with this we build the desired derivation as above with a top-most instance of  $\mathbf{u}_1\downarrow$ .

4. If  $\rho$  is  $\text{ir}\downarrow$ , the only possibility is  $P = [?P_1 \wp P_2]$  and the bottom-most instance is

$$\text{ir}\downarrow \frac{[?[R \wp P_1] \wp P_2]}{[?R \wp ?P_1 \wp P_2]} .$$

We are immediately done, since the induction hypothesis delivers the derivation

$$\begin{array}{c} 1 \\ \parallel_{\Pi} \\ [?[R \wp P_1] \wp !T \wp ?Q_1 \wp \dots \wp ?Q_h] \\ \parallel_{\Delta} \\ [?[R \wp P_1] \wp P_2] \end{array} .$$

5. If  $\rho$  is  $\mathbf{e}\downarrow$ , the only possibility is that the redex is inside  $R$  or  $P$ . We prove as in case (1).  $\square$

The following lemma provides access to rule instances in arbitrary contexts. This result is similar to those in [7, 19, 18, 9, 10, 11], however the interaction requirement makes the independence argument of Lemmata 21 and 22 necessary.

**Lemma 26** (Context reduction). *For any formula  $R$  and context  $S\{ \}$ , if  $S\{R\}$  is provable in  $\text{cELSdlip}$ , then either there is a formula  $P_R$  such that*

$$(i) \quad \begin{array}{c} !\dots![R \wp P_R] \\ \text{cELSdlip} \parallel \\ S\{R\} \end{array} \quad \text{and} \quad \begin{array}{c} 1 \\ \text{cELSdlip} \parallel \\ [R \wp P_R] \end{array} \quad \text{or} \quad (ii) \quad \begin{array}{c} !\dots!R \\ \text{cELSdlip} \parallel \\ S\{R\} \end{array} \quad \text{and} \quad \begin{array}{c} 1 \\ \text{cELSdlip} \parallel \\ R \end{array} .$$

*Proof.* We prove by structural induction on the context  $S\{ \}$ . The base case with the empty context is trivial, so we proceed with the inductive cases.

1. If  $S\{ \} = [!S'\{ \} \wp P]$ , we apply Lemma 24 to the proof of  $S\{R\}$ . The statement (ii) of Lemma 24 is possible only if  $R = 1$ , which is the easier case to prove. We proceed with the more involved case (i), as the same argument applies to (ii), and obtain the derivation

$$\begin{array}{c} [?P_1 \wp \dots \wp ?P_h] \\ \parallel_{\Delta_P} \\ P \end{array} \quad \text{and the proof} \quad \begin{array}{c} 1 \\ \parallel_{\Pi_S} \\ [S'\{R\} \wp P_1 \wp \dots \wp P_h] \end{array} .$$

We apply the induction hypothesis to  $\Pi_S$ . It delivers two cases. With the first, we obtain

$$\begin{array}{c} !\dots![R \wp P_R] \\ \parallel_{\Delta'} \\ [S'\{R\} \wp P_1 \wp \dots \wp P_h] \end{array} \quad \text{and} \quad \begin{array}{c} 1 \\ \parallel_{\Pi} \\ [R \wp P_R] \end{array} \quad \text{and build} \quad \begin{array}{c} ![R \wp P_R] \\ \parallel_{\Delta''} \\ [!S'\{R\} \wp ?P_1 \wp \dots \wp ?P_h] \\ \parallel_{\Delta_P} \\ [!S'\{R\} \wp P] \end{array} ,$$

where  $\Delta''$  is delivered by Lemma 21 with derivation  $\Delta'$ . In the second,  $P_R$  disappears and we replace the premise of the left and right derivations with  $!\dots!R$  instead of  $!\dots![R \wp P_R]$ .

2. If  $S\{\} = [?S'\{\} \wp P]$ , we first apply Lemma 25 to the proof of  $S\{R\}$ , which delivers

$$\begin{array}{c} 1 \\ \parallel_{\Pi_P} \\ [?S'\{R\} \wp !T \wp ?Q_1 \wp \dots \wp ?Q_h] \\ \parallel_{\Delta_P} \\ [?S'\{R\} \wp P] \end{array} \quad \text{and obtain} \quad \begin{array}{c} 1 \\ \parallel_{\Pi'_P} \\ [S'\{R\} \wp T \wp Q_1 \wp \dots \wp Q_h] \end{array}$$

by Lemma 21 with  $\Pi_P$ . We can apply the induction hypothesis to  $\Pi_S$ , which delivers two cases. With the first, we obtain

$$\begin{array}{c} !\dots![R \wp P_R] \\ \parallel_{\Delta'} \\ [S'\{R\} \wp T \wp Q_1 \wp \dots \wp Q_h] \end{array} \quad \text{and} \quad \begin{array}{c} 1 \\ \parallel_{\Pi} \\ [R \wp P_R] \end{array} \quad \text{and build} \quad \begin{array}{c} ![R \wp P_R] \\ \parallel_{\Delta''} \\ [?S'\{R\} \wp !T \wp ?Q_1 \wp \dots \wp ?Q_h] \\ \parallel_{\Delta_P} \\ [?S'\{R\} \wp P] \end{array}$$

where  $\Delta''$  is delivered by Lemma 21 with derivation  $\Delta'$ . In the second case,  $P_R$  disappears and we replace the premise of the left and right derivations with  $!\dots!R$  as before.

3. If  $S\{\} = (S'\{\} \otimes T)$ , or  $S\{\} = [(S'\{\} \otimes T) \wp P]$ , the proof procedure is as for the one for multiplicative linear logic presented in [11]. For the former, applying Proposition 20 and the induction hypothesis delivers the desired proof. For the latter, we apply Lemma 23 and the induction hypothesis.

□

**Theorem 27.** *Systems cELSdlip and cELSdl are equivalent.*

*Proof.* Observe that every proof in cELSdlip is also a proof in cELSdl. For the other direction, we inductively remove each instance of the rule in the cELSdl proof that is not an instance of a cELSdlip. Before going into the case analysis, let us illustrate the method for this. Let  $\Pi$  be a proof of a formula  $R$  in cELSdlip. We single out the topmost rule instance  $\rho$  in  $\Pi$  such that either (i)  $\rho$  is an instance of  $p\downarrow$ , but it is not an instance of  $ip\downarrow$  or (ii)  $\rho$  is an instance of  $dls$ , but it is not an instance of  $dlis$ . We then map the proof of the conclusion of  $\rho$  to a proof in

cELSdlip as illustrated below for  $\Pi'$  with the conclusion  $S\{T\}$  to  $\Pi''$  with the conclusion  $S\{T\}$ .

$$\Pi = \left\{ \begin{array}{ccc} \frac{1}{\text{cELSdlip} \parallel \Pi'} & & \frac{1}{\text{cELSdlip} \parallel \Pi''} \\ \rho \frac{S\{Q\}}{S\{T\}} & \rightsquigarrow & S\{T\} \\ \text{cELSdl} \parallel & & \text{cELSdl} \parallel \\ R & & R \end{array} \right.$$

We map proof  $\Pi'$  to proof  $\Pi''$  by using Lemmata 21, 22, 23, 24, 25 and 26 as described below. We repeat this procedure inductively until all of the proof is transformed into a cELSdlip proof.

1.  $\rho$  is an instance of  $\text{p}\downarrow$  but not an instance of  $\text{ip}\downarrow$  such that

$$\text{p}\downarrow \frac{S\{[R \wp T]\}}{S\{[!R \wp ?T]\}} .$$

From Lemma 26, we have that (i) either  $[R \wp T]$  has a proof  $\Pi_1$  in cELSdlip and

$$\frac{! \dots ! [R \wp T]}{\text{cELSdlip} \parallel \Delta_1} \quad \text{or} \quad (ii) \quad \frac{! \dots ! [! [R \wp T] \wp V]}{\text{cELSdlip} \parallel \Delta_2} \quad \text{and} \quad \frac{1}{\text{cELSdlip} \parallel \Pi_2} \frac{! [R \wp T] \wp V}{S\{[!R \wp T]\}} .$$

For the case (i), we apply Lemma 21 to the proof  $\Pi_1$  and obtain a proof of  $[!R \wp ?T]$  in cELSdlip, which we use together with  $\Delta_1$  to build the desired proof. For the more involved case (ii), we apply Lemma 24 to  $\Pi_2$ , and obtain either (ii.i) a proof  $\Pi_3$  of  $[R \wp T]$  and

$$\frac{\perp}{\text{cELSdlip} \parallel \Delta_3} \frac{V}{V} \quad \text{or} \quad (ii.ii) \quad \frac{[?P_1 \wp \dots \wp ?P_h]}{\text{cELSdlip} \parallel \Delta_4} \frac{V}{V} \quad \text{and} \quad \frac{1}{\text{cELSdlip} \parallel \Pi_4} \frac{[R \wp T \wp P_1 \wp \dots \wp P_h]}{S\{[!R \wp ?T]\}} .$$

We construct the proofs below for the cases (ii.i) and (ii.ii), where  $\Delta_5$  is delivered by Lemma 22 with the proof  $\Pi_3$ , and  $\Delta_6$  is delivered by Lemma 22 with the proof  $\Pi_4$ .

$$\frac{\frac{\frac{\frac{1}{\{e\downarrow\}} \parallel \dots \parallel 1}{! \dots ! [!1 \wp \perp]}{\text{cELSdlip} \parallel \Delta_5}}{! \dots ! [!R \wp ?T \wp \perp]}{\text{cELSdlip} \parallel \Delta_3}}{! \dots ! [!R \wp ?T \wp V]}{\text{cELSdlip} \parallel \Delta_2} \frac{S\{[!R \wp ?T]\}}{S\{[!R \wp ?T]\}} \quad \frac{\frac{\frac{\frac{1}{\{e\downarrow\}} \parallel \dots \parallel 1}{! \dots ! 1}}{\text{cELSdlip} \parallel \Delta_6}}{! \dots ! [!R \wp ?T \wp ?P_1 \wp \dots \wp ?P_h]}{\text{cELSdlip} \parallel \Delta_4}}{! \dots ! [!R \wp ?T \wp V]}{\text{cELSdlip} \parallel \Delta_2} \frac{S\{[!R \wp ?T]\}}{S\{[!R \wp ?T]\}} .$$

2.  $\rho$  is an instance of  $\text{dls}$  but not an instance of  $\text{dlis}$  such that

$$\text{dls} \frac{S\{([R \wp U] \otimes T)\}}{S\{([R \otimes T] \wp U)\}} .$$

From Lemma 26, we have that (i) either  $([R \wp U] \otimes T)$  has a proof in cELSdlip and

$$\begin{array}{c} ! \dots ! ([R \wp U] \otimes T) \\ \text{cELSdlip} \parallel_{\Delta} \end{array} \quad \text{or} \quad (ii) \quad \begin{array}{c} ! \dots ! ([R \wp U] \otimes T) \wp V \\ \text{cELSdlip} \parallel_{\Delta} \end{array} \quad \text{and} \quad \begin{array}{c} 1 \\ \text{cELSdlip} \parallel_{\Delta} \end{array} \cdot \\ S([R \wp U] \otimes T) \quad S([R \wp U] \otimes T) \quad [(R \wp U] \otimes T) \wp V$$

We prove the more involved case (ii); case (i) is proved similarly. We apply Lemma 23, and obtain either (ii.i) a proof  $\Pi_1$  of  $[R \wp U \wp V]$  and a proof  $\Pi_2$  of  $T$  in cELSdlip, or (ii.ii) a proof  $\Pi_3$  of  $[R \wp U]$  together with a proof  $\Pi_4$  of  $[T \wp V]$  in cELSdlip, or

$$(ii.iii) \quad \begin{array}{c} [K_1 \wp K_2] \\ \text{cELSdlip} \parallel_{\Delta_1} \\ V \end{array}, \quad \begin{array}{c} 1 \\ \text{cELSdlip} \parallel_{\Pi_5} \\ [R \wp U \wp K_1] \end{array} \quad \text{and} \quad \begin{array}{c} 1 \\ \text{cELSdlip} \parallel_{\Pi_6} \\ [K_2 \wp T] \end{array}.$$

We construct the proofs below for the cases (ii.i), (ii.ii) and (ii.iii), where  $\Delta_2$  is delivered by Lemma 22 with the proof  $\Pi_4$ , and  $\Delta_3$  is delivered by Lemma 22 with the proof  $\Pi_6$ .

$$\begin{array}{c} 1 \\ \text{cELSdlip} \parallel_{\Pi_1} \\ ! \dots ! [R \wp U \wp V] \\ \text{u}_2 \downarrow \frac{\quad}{! \dots ! [(R \otimes 1) \wp U \wp V]} \\ \text{cELSdlip} \parallel_{\Pi_2} \\ ! \dots ! [(R \otimes T) \wp U \wp V] \\ \text{cELSdlip} \parallel_{\Delta} \\ S[(R \otimes T) \wp U] \end{array} \quad \begin{array}{c} 1 \\ \text{cELSdlip} \parallel_{\Pi_3} \\ ! \dots ! [R \wp U] \\ \text{u}_2 \downarrow \frac{\quad}{! \dots ! [(R \otimes 1) \wp U]} \\ \text{cELSdlip} \parallel_{\Delta_2} \\ ! \dots ! [(R \otimes T) \wp U \wp V] \\ \text{cELSdlip} \parallel_{\Delta} \\ S[(R \otimes T) \wp U] \end{array} \quad \begin{array}{c} 1 \\ \text{cELSdlip} \parallel_{\Pi_5} \\ ! \dots ! [R \wp U \wp K_1] \\ \text{u}_2 \downarrow \frac{\quad}{! \dots ! [(R \otimes 1) \wp U \wp K_1]} \\ \text{cELSdlip} \parallel_{\Delta_3} \\ ! \dots ! [(R \otimes T) \wp U \wp K_1 \wp K_2] \\ \text{cELSdlip} \parallel_{\Delta_1} \\ ! \dots ! [(R \otimes T) \wp U \wp V] \\ \text{cELSdlip} \parallel_{\Delta} \\ S[(R \otimes T) \wp U] \end{array}$$

□

**Theorem 28** (Cut elimination). *The cut rule is admissible for system ELSdlip.*

*Proof.* The result is proved with a procedure that is similar to those in [7, 19, 18, 9, 10, 11], by applying the splitting and context reduction lemmata above as in [9] and the independence lemmata as in [11]. A weaker result can be obtained by first applying the cut elimination procedure in [19], and then applying Theorem 19. □

## 4 Discussion

The interaction condition that we use in the design of the context management rules  $\text{dli}\downarrow$ ,  $\text{ir}\downarrow$  and  $\text{ip}\downarrow$  requires the complementarity of the pairs of subformulae that these rules act on. This simple condition provides a significant reduction in non-determinism in proof search as proofs are constructed by annihilating dual atoms in the instances of the atomic interaction rule ( $\text{ai}\downarrow$ ). Because the context management rules perform the selection of the subformulae that will maintain their interaction, the conditions carry over the interaction of dual atoms eventually to the instances of the rule  $\text{ai}\downarrow$ . Moreover, these conditions provide a more immediate access to shorter proofs when they are exploited in deeper contexts. This way, by constructing the



proofs ‘from the inside out’, subformulae that would otherwise be carried along the proof, and sometimes copied, can be annihilated in advance. Such deeper instances also provide the additional benefit of reducing the size of the subformulae that are processed by the conditions. In this respect, an open problem related to reducing nondeterminism in deep inference is related to completeness of the systems, where the rule instances are applied only at the “deepest” contexts. In such a setting, the depth and laziness requirements of the rule  $\text{dlis}$  would provide an additional advantage in the computational accounting of proof search as they further reduce the size of the processed subformulae.

In fact, from such a point of view, the main flaw of the sequent calculus in proof search can be seen as its complementary perspective in proof construction that is blind to the size of the subformulae, processed by the rule instances. The bottom up construction of the sequent calculus proofs implicitly relies on carrying interacting subformulae to the leaves of the proof tree. However, contrary to the situation delivered by deep rule instances, applying the inference rules only at the main connectives delays accessing the subformulae, and this often introduces an otherwise unnecessary cost in proof length. A simple illustration of this is provided by the example in the introduction. However, a more sophisticated characterization is given by the Statman tautologies: this class of formulae have polynomial size deep inference proofs in classical logic in contrast to their exponential size sequent calculus proofs [4].

Because the modality  $?$  permits contraction as in classical logic, Statman tautologies can be used as an example for a class of formulae that enjoys exponential speed up with deep inference systems for MELL that we have discussed above, including system  $\text{ELSdlip}$ . To see this, we consider the first two Statman tautologies; for the definition of this class of formulae and an in-depth discussion of the exponential speed-up provided by deep inference, we refer to [4].

The simplest multiplicative linear logic representation of the first Statman tautology is the formula  $[a \wp b \wp (\bar{a} \otimes \bar{b})]$ , which we have discussed in the introduction. Let us consider the formula  $[?a \wp ?b \wp ?c \wp (!\bar{a} \otimes !\bar{b} \otimes !\bar{c})]$ . This formula can be proved in system  $\text{ELSdlip}$  as below, where we adopt a syntactic sugar to denote multiple applications of the inference rules.

$$\begin{array}{c}
 \text{ai}\downarrow; \text{e}\downarrow \frac{1}{![c \wp \bar{c}]} \\
 \text{ip}\downarrow \frac{\phantom{1}}{[?c \wp !\bar{c}]} \\
 \text{ai}\downarrow; \text{e}\downarrow; \text{u}_2\downarrow \frac{\phantom{1}}{[?c \wp (![b \wp \bar{b}] \otimes !\bar{c})]} \\
 \text{dlis}; \text{ip}\downarrow \frac{\phantom{1}}{[?b \wp ?c \wp (!\bar{b} \otimes !\bar{c})]} \\
 \text{ai}\downarrow; \text{e}\downarrow; \text{u}_2\downarrow \frac{\phantom{1}}{[?b \wp ?c \wp (![a \wp \bar{a}] \otimes !\bar{b} \otimes !\bar{c})]} \\
 \text{dlis}; \text{ip}\downarrow \frac{\phantom{1}}{[?a \wp ?b \wp ?c \wp (!\bar{a} \otimes !\bar{b} \otimes !\bar{c})]}
 \end{array}$$

For a comparison of the breadth-size between the systems with and without the condition, we have extended the system  $\text{ELSu}$  with the rule  $r\downarrow$ ; we denote this system with  $\text{cELSu}^*$ . The table below lists the size of the immediate breadth of the search space at every step of the proof above for the systems  $\text{cELSu}$ ,  $\text{cELSu}^*$ , and  $\text{ELSdlip}$ . As this comparison demonstrates, system  $\text{ELSdlip}$  provides a greater reduction in the breadth of the search space when the non-determinism is greater. It is important to note here that all the nodes in the search spaces of system  $\text{ELSdlip}$  are provable, whereas the search space of  $\text{ELSu}$  has dead nodes.

| proof step       | 1. | 2. | 3. | 4. | 5. | 6. | 7. | 8. | 9. | 10. | 11. | 12. | 13. |
|------------------|----|----|----|----|----|----|----|----|----|-----|-----|-----|-----|
| $\text{cELSu}$   | 42 | 19 | 19 | 19 | 19 | 6  | 3  | 3  | 3  | 3   | 1   | 1   | 1   |
| $\text{cELSu}^*$ | 45 | 20 | 20 | 20 | 20 | 7  | 3  | 3  | 3  | 3   | 1   | 1   | 1   |
| $\text{ELSdlip}$ | 3  | 3  | 3  | 3  | 3  | 2  | 2  | 2  | 2  | 2   | 1   | 1   | 1   |

The formulae in the examples above consists of distinct pairs of atoms. Although the conditions of the rules  $\text{dlis}$ ,  $\text{ir}\downarrow$  and  $\text{ip}\downarrow$  are more sensitive to the non-determinism for the cases with distinct atoms, the formulae with identical atoms also benefit from a reduced nondeterminism, however to a lesser extent. In order to see this on a simple example, consider the formula  $[a \wp a \wp a \wp (\bar{a} \otimes \bar{a} \otimes \bar{a})]$ . In system  $\text{ELSu}$ , this formula has six immediate rule instances, however only two of these instances result in proofs, and one of these instances is redundant. On the other hand, system  $\text{ELSdlip}$  has only one instance, which is the only one amongst the six instances that results in a short proof. This is because, here, the laziness and depth requirements of the rule  $\text{dlis}$  rule filter out the non-determinism that is not due to complementary atoms.

Let us now consider the second Statman tautology, a simple MELL representation of which can be given as  $[?(a \otimes b) \wp ((\bar{a} \wp \bar{b}) \otimes c \otimes [\bar{a} \wp \bar{b}] \otimes d) \wp \bar{c} \wp \bar{d}]$ . To better illustrate the construction of the proofs ‘from the inside out’ in a way that contrasts with the sequent calculus which can only access the top-level connective, we introduce an additional depth to it.

$$[(a \otimes b) \wp (((\bar{a} \wp \bar{b}) \otimes c \otimes [\bar{a} \wp \bar{b}] \otimes d) \wp \bar{c} \wp \bar{d}) \otimes [e \wp \bar{e}]]$$

One of the shorter proofs of this formula that is impossible in the sequent calculus is as follows.

$$\begin{array}{c} \text{dlis} \frac{([\bar{a} \wp \bar{b} \wp (a \otimes b)] \otimes [\bar{a} \wp \bar{b} \wp (a \otimes b)])}{[(a \otimes b) \wp ([\bar{a} \wp \bar{b}] \otimes [\bar{a} \wp \bar{b} \wp (a \otimes b)])]} \\ \text{dlis} \frac{[(a \otimes b) \wp ([\bar{a} \wp \bar{b}] \otimes [\bar{a} \wp \bar{b} \wp (a \otimes b)])]}{[(a \otimes b) \wp (a \otimes b) \wp ([\bar{a} \wp \bar{b}] \otimes [\bar{a} \wp \bar{b}])]} \\ \text{ai}\downarrow; \text{u}_2\downarrow \frac{[(a \otimes b) \wp (a \otimes b) \wp ([\bar{a} \wp \bar{b}] \otimes [\bar{a} \wp \bar{b}])]}{[(a \otimes b) \wp (a \otimes b) \wp ([\bar{a} \wp \bar{b}] \otimes [\bar{a} \wp \bar{b}] \otimes [e \wp \bar{e}])]} \\ \text{ai}\downarrow; \text{u}_2\downarrow \frac{[(a \otimes b) \wp (a \otimes b) \wp ([\bar{a} \wp \bar{b}] \otimes [\bar{a} \wp \bar{b}] \otimes [e \wp \bar{e}])]}{[(a \otimes b) \wp (a \otimes b) \wp ([\bar{a} \wp \bar{b}] \otimes [\bar{a} \wp \bar{b}] \otimes [d \wp \bar{d}] \otimes [e \wp \bar{e}])]} \\ \text{ai}\downarrow; \text{u}_2\downarrow \frac{[(a \otimes b) \wp (a \otimes b) \wp ([\bar{a} \wp \bar{b}] \otimes [c \wp \bar{c}] \otimes [\bar{a} \wp \bar{b}] \otimes [d \wp \bar{d}] \otimes [e \wp \bar{e}])]}{[(a \otimes b) \wp (a \otimes b) \wp ([\bar{a} \wp \bar{b}] \otimes [c \wp \bar{c}] \otimes [\bar{a} \wp \bar{b}] \otimes d) \wp \bar{c} \wp \bar{d}] \otimes [e \wp \bar{e}]} \\ \text{dlis} \frac{[(a \otimes b) \wp (a \otimes b) \wp ([\bar{a} \wp \bar{b}] \otimes [c \wp \bar{c}] \otimes [\bar{a} \wp \bar{b}] \otimes d) \wp \bar{c} \wp \bar{d}] \otimes [e \wp \bar{e}]}{[(a \otimes b) \wp (a \otimes b) \wp ([\bar{a} \wp \bar{b}] \otimes [c \wp \bar{c}] \otimes [\bar{a} \wp \bar{b}] \otimes d) \wp \bar{c} \wp \bar{d}] \otimes [e \wp \bar{e}]} \\ \text{dlis} \frac{[(a \otimes b) \wp (a \otimes b) \wp ([\bar{a} \wp \bar{b}] \otimes [c \wp \bar{c}] \otimes [\bar{a} \wp \bar{b}] \otimes d) \wp \bar{c} \wp \bar{d}] \otimes [e \wp \bar{e}]}{[(a \otimes b) \wp (a \otimes b) \wp ([\bar{a} \wp \bar{b}] \otimes c \otimes [\bar{a} \wp \bar{b}] \otimes d) \wp \bar{c} \wp \bar{d}] \otimes [e \wp \bar{e}]} \\ \text{b}\downarrow; \text{b}\downarrow; \text{w}\downarrow; \text{u}_1\downarrow \frac{[(a \otimes b) \wp (a \otimes b) \wp ([\bar{a} \wp \bar{b}] \otimes c \otimes [\bar{a} \wp \bar{b}] \otimes d) \wp \bar{c} \wp \bar{d}] \otimes [e \wp \bar{e}]}{[(a \otimes b) \wp (([\bar{a} \wp \bar{b}] \otimes c \otimes [\bar{a} \wp \bar{b}] \otimes d) \wp \bar{c} \wp \bar{d}) \otimes [e \wp \bar{e}]]} \end{array}$$

The table below lists the size of the immediate breadth of the search space at every proof step for the systems  $\text{cELSu}$  and  $\text{cELSdlip}$  as this proof does not involve any instances of  $\text{r}\downarrow$ .

| proof step        | 2. | 3. | 4. | 5. | 6. | 7. | 8. | 9. | 10. | 11. |
|-------------------|----|----|----|----|----|----|----|----|-----|-----|
| $\text{cELSu}$    | 41 | 22 | 53 | 53 | 28 | 28 | 15 | 15 | 8   | 10  |
| $\text{cELSdlip}$ | 6  | 6  | 6  | 6  | 5  | 5  | 4  | 4  | 3   | 6   |

In our last example, we observe the effect of the interaction encourage rule ( $\text{ir}\downarrow$ ).

$$\begin{array}{c} \text{ai}\downarrow; \text{ai}\downarrow; \text{u}_2\downarrow; \text{e}\downarrow \frac{1}{!([\bar{b} \wp b] \otimes [c \wp \bar{c}])} \\ \text{dlis}; \text{dlis} \frac{!([\bar{b} \wp b] \otimes [c \wp \bar{c}])}{![\bar{b} \wp (b \otimes c) \wp \bar{c}]} \\ \text{ip}\downarrow; \text{ip}\downarrow \frac{![\bar{b} \wp (b \otimes c) \wp \bar{c}]}{[?\bar{b} \wp !(b \otimes c) \wp ?\bar{c}]} \\ \text{u}_2\downarrow; \text{u}_2\downarrow \frac{[?\bar{b} \wp !(b \otimes c) \wp ?\bar{c}]}{[?(1 \otimes \bar{b}) \wp !(b \otimes c) \wp ?(\bar{c} \otimes 1)]} \\ \text{ai}\downarrow; \text{ai}\downarrow \frac{[?(1 \otimes \bar{b}) \wp !(b \otimes c) \wp ?(\bar{c} \otimes 1)]}{[?([\bar{a} \wp a] \otimes \bar{b}) \wp !(b \otimes c) \wp ?(\bar{c} \otimes [d \wp \bar{d}])]} \\ \text{dlis}; \text{dlis} \frac{[?([\bar{a} \wp a] \otimes \bar{b}) \wp !(b \otimes c) \wp ?(\bar{c} \otimes [d \wp \bar{d}])]}{[?\bar{a} \wp (a \otimes \bar{b}) \wp !(b \otimes c) \wp ?(\bar{c} \otimes d) \wp \bar{d}]} \\ \text{ir}\downarrow; \text{ir}\downarrow \frac{[?\bar{a} \wp (a \otimes \bar{b}) \wp !(b \otimes c) \wp ?(\bar{c} \otimes d) \wp \bar{d}]}{[?\bar{a} \wp ?(a \otimes \bar{b}) \wp !(b \otimes c) \wp ?(\bar{c} \otimes d) \wp ?\bar{d}]} \end{array}$$

This proof is impossible in ELSu as this system does not include the rule  $\text{ir}\downarrow$  as well as its non interaction version  $\text{r}\downarrow$ . Thus, the table below lists the size of the immediate breadth of the search space at every step of the proof above only for the systems cELSu\* and ELSdlip. Again, all the nodes in the search spaces of system ELSdlip are provable, whereas the search space of ELSu has dead nodes.

| proof step | 1. | 2. | 3. | 4. | 5. | 6. | 7. | 8. | 9. | 10. | 11. | 12. | 13. | 14. |
|------------|----|----|----|----|----|----|----|----|----|-----|-----|-----|-----|-----|
| cELSu*     | 10 | 7  | 6  | 5  | 5  | 5  | 4  | 3  | 3  | 6   | 3   | 2   | 2   | 1   |
| cELSdlip   | 4  | 4  | 4  | 4  | 4  | 4  | 4  | 3  | 2  | 2   | 2   | 2   | 2   | 1   |

In conclusion, the method we have presented above provides a significant reduction in non-determinism, and in combination with other techniques it can pave the way for ambitious applications. In this respect, our long term goal is developing analytic deep inference theorem provers for different logics. Due to their similarity with MELL, various modal logics are immediate candidates for the technique that we have used here on the promotion rule as well as the switch rule. Moreover, the completeness argument that we employ is closely related with cut elimination. In that respect, our technique is pure from a proof theoretical point of view, and it is thus of independent theoretical interest; similar results were used to prove the NP-hardness of system BV [10], and the decision problem for MELL is an open problem.

For the development of applications, other proof theoretical techniques as well as heuristic methods can be combined to benefit from proof search in a way that is impossible for sequent calculus systems. Future work, along these lines, includes exploiting the concurrency in proofs [12] in proof search together with importing orthogonal techniques from focusing [1, 5]. The development of a technique for prioritizing the rule instances in proof search in a theoretically clean manner as we have described above should also provide another important milestone for reducing nondeterminism in deep inference systems.

## References

- [1] Jean-Marc Andreoli. Focussing and proof construction. *Ann. Pure Appl. Logic*, 107(1-3):131–163, 2001.
- [2] Kai Brünnler. Atomic cut elimination for classical logic. In M. Baaz and J. A. Makowsky, editors, *CSL 2003*, volume 2803 of *LNCS*, pages 86–97. Springer, 2003.
- [3] Kai Brünnler. Locality for classical logic. *Notre Dame Journal of Formal Logic*, 47(4):557–580, 2006.
- [4] Paola Bruscoli and Alessio Guglielmi. On the proof complexity of deep inference. *ACM Transactions on Computational Logic*, 2(14):1–34, 2009.
- [5] Kaustuv Chaudhuri, Nicolas Guenot, and Lutz Straßburger. The focused calculus of structures. In Marc Bezem, editor, *Computer Science Logic (CSL’11) - 25th International Workshop/20th Annual Conference of the EACSL*, volume 12, pages 159–173. LIPICS, 2011.
- [6] Anupam Das. On the relative proof complexity of deep inference via atomic flows. *Logical Methods in Computer Science*, 11(1:4):1–27, 2015.
- [7] Alessio Guglielmi. A system of interaction and structure. *ACM Transactions on Computational Logic*, 8(1):1–64, 2007.
- [8] Alessio Guglielmi and Tom Gundersen. Normalisation control in deep inference via atomic flows. *Logical Methods in Computer Science*, 4(1:9):1–36, 2008.
- [9] Alessio Guglielmi and Lutz Strassburger. A system of interaction and structure V: The exponentials and splitting. *Mathematical Structures in Computer Science*, 21(3):563–584, 2010.
- [10] Ozan Kahramanoğulları. System BV is NP-complete. *Annals of Pure and Applied Logic*, 152(1–3):107–121, 2008.

- [11] Ozan Kahramanoğulları. Interaction and depth against nondeterminism in proof search. *Logical Methods in Computer Science*, 10 (2:5):1–49, 2014.
- [12] Ozan Kahramanoğulları. True concurrency of deep inference proofs. In Jouko Vaananen, Asa Hirvonen, and Ruy de Queiroz, editors, *Logic, Language, Information, and Computation 23rd International Workshop, WoLLIC 2016, Puebla, Mexico, August 16-19th, 2016. Proceedings*, volume 9803 of *LNCS*. Springer, 2016.
- [13] Ozan Kahramanoğulları. System BV without the equalities for unit. In *Proceedings of the 19th International Symposium on Computer and Information Sciences, ISCIS'04*, volume 3280 of *LNCS*, pages 986–995, Antalya, Turkey, 2004. Springer.
- [14] Ozan Kahramanoğulları. *Nondeterminism and Language Design in Deep Inference*. PhD thesis, TU Dresden, 2006.
- [15] Ozan Kahramanoğulları. Reducing nondeterminism in the calculus of structures. In Miki Hermann and Andrei Voronkov, editors, *Logic for Programming, Artificial Intelligence, and Reasoning, Proceedings of the 13th International Conference, LPAR 2006, Phnom Penh, Cambodia*, volume 4246 of *LNCS*, pages 272–286. Springer, 2006.
- [16] Ozan Kahramanoğulları. Maude as a platform for designing and implementing deep inference systems. In *Proc. of the Eighth International Workshop on Rule-Based Programming, RULE'07*, volume 219 of *ENTCS*, pages 35–50. Elsevier, 2008.
- [17] Lutz Straßburger. A local system for linear logic. In M. Baaz and A. Voronkov, editors, *LPAR 2002*, volume 2514 of *LNAI*, pages 388–402. Springer, 2002.
- [18] Lutz Straßburger. *Linear Logic and Noncommutativity in the Calculus of Structures*. PhD thesis, TU Dresden, 2003.
- [19] Lutz Straßburger. MELL in the calculus of structures. *Theoretical Computer Science*, 309:213–285, 2003.
- [20] Lutz Strassburger and Alessio Guglielmi. A system of interaction and structure IV: The exponentials and decomposition. *ACM Trans. on Comp. Logic*, 12(4), 2011.
- [21] Alwen Fernanto Tiu. A local system for intuitionistic logic. In Miki Hermann and Andrei Voronkov, editors, *Logic for Programming, Artificial Intelligence, and Reasoning, Proceedings of the 13th International Conference, LPAR 2006, Phnom Penh, Cambodia*, volume 4246 of *LNCS*, pages 242–256. Springer, 2006.
- [22] Alwen Fernanto Tiu. A system of interaction and structure II: the need for deep inference. *Logical Methods in Computer Science*, 2 (2:4):1–24, April 2006.