A Deep Inference System for Differential Linear Logic

Matteo Acclavio
Univeristy of Luxembourg, Belval, Luxembourg

Giulio Guerrieri
University of Bath, Bath, United Kingdom

Differential linear logic (DiLL) provides a fine analysis of resource consumption in cut-elimination. We investigate the subsystem of DiLL without promotion in a deep inference formalism, where cuts are at an atomic level. In our system every provable formula admits a derivation in normal form, via a normalization procedure that commutes with the translation from sequent calculus to deep inference.

1   Introduction

Girard [13] introduced linear logic (LL) as a refinement of intuitionistic and classical logics, built around cut-elimination. In LL, a pair of dual modalities (the *exponentials* ! and ?) give a logical status to the operations of erasing and copying (sub-)proofs in the cut-elimination procedure. The idea is that *linear* proofs (*i.e.* proofs without exponentials) use their hypotheses exactly once, whilst *exponential* proofs may use their hypotheses at will. In particular, the *promotion rule* makes a (sub-)proof available to be erased or copied an unbounded number of times, provided that its hypotheses are as well (it is a contextual rule). Via Curry–Howard correspondence between programs and proofs, LL gives a logical status to the operations of erasing and copying data in the evaluation process. Linear proofs correspond to programs which call their arguments exactly once, exponential proofs to programs which call their arguments at will. The study of LL contributed to unveil the logical nature of resource consumption.

The importance of being differential.   A further tool for the analysis of resource consumption in cut-elimination came from Ehrhard and Regnier's work on differential λ-calculus [7] and *differential linear logic* (DiLL, [9, 28]). Despite the fact that DiLL is inconsistent (every sequent $\vdash \Gamma$ can be proved), it has a cut-elimination theorem [28, 12] and internalizes notions from denotational models of LL into the syntax. In particular, $DiLL_0$ (the *promotion-free* fragment of DiLL, [9]) is a logic corresponding to the semantic constructions defined by Ehrhard's finiteness spaces [4]. Finiteness spaces interpret linear proofs as linear functions on certain topological vector spaces, on which one can define an operation of derivative. Exponential proofs are interpreted as analytic functions, in the sense that they can be arbitrarily approximated by the semantic equivalent of a Taylor expansion [4, 5], which becomes available thanks to the presence of a derivative operator. In syntactic terms, these constructions take an interesting form: they correspond to "symmetrizing" the exponential modalities, *i.e.* in $DiLL_0$ the rules handling the dual exponential modalities ! and ? are perfectly symmetrical, although the logic is not self-dual. Indeed, in LL, only the promotion rule introduces the ! modality, creating inputs that can be called an unbounded number of times. In $DiLL_0$ the rules handling the ! modality (!-dereliction !d, !-contraction !c, !-weakening !w) are the duals of the usual rules dealing with the ? modality (?-dereliction ?d, ?-contraction ?c, ?-weakening ?w). In particular, !-*dereliction* expresses in the syntax the semantic derivative: it releases inputs of type !A that must be called exactly *once*, so that executing a program f on a "!-derelicted" input x (*i.e.* performing cut-elimination on a proof f cut with a "!-derelicted" sub-proof x) amounts to compute the best linear approximation of f on x. This imposes non-deterministic choices: if in an evaluation the program f needs several copies of the input x (*i.e.* if the proof f uses several times the hypothesis !A),

then there are different executions of f on x, depending on which sub-routine (*i.e.* hypothesis) of f is fed with the unique available copy of x. Thus we get a formal *sum*, where each term represents a possibility. The rules !-contraction and !-weakening put together a finite (possibly 0) number of copies of an input, so that it can be called a *bounded* number of times during execution.

What is also interesting is that LL promotion rule can be encoded in DiLL_0 through the notion of syntactic *Taylor expansion* [8, 10, 26, 29, 15, 3, 16, 17]: a proof in LL can be decomposed into a possibly infinite set of (promotion-free) proofs in DiLL_0. Given a proof in LL with exactly one promotion rule !p, the idea is to replace !p (which makes the resource π available at will) with an infinite set of "differential" proofs in DiLL_0, each of them taking $n \in \mathbb{N}$ copies of π so as to make the resource π available exactly n times. The potential infinity of the promotion rule becomes an actual infinite via the Taylor expansion.

Nets vs. sequents. The system DiLL_0 is usually presented in two formalisms: sequent calculus and Lafont's interaction nets [24] (a graphical representation of proofs similar to LL proof-nets). The *symmetry* of the rules handling the dual exponentials ! and ? in DiLL_0 is evident in interaction nets, but not at all in the sequent calculus. In interaction nets for DiLL_0, the rules for ? and ! have the *same geometry*:

?- and !-weakening   ?- and !-dereliction   ?- and !-contraction
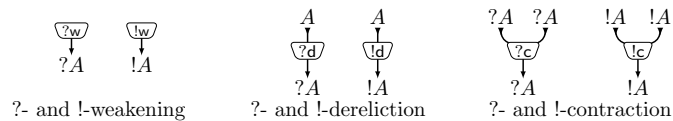
So, the distinction between ! and ? is given only by their different behaviors in correctness graphs (a geometrical characterization of the interaction nets corresponding to proofs in the DiLL_0 sequent calculus). But meaningful operations in DiLL_0 such as cut-elimination can be defined directly on interaction nets, regardless of being correct or not. The benefit is that DiLL_0 cut-elimination steps defined on interaction nets are perfectly symmetric: for instance, the step for a cut ?c/!d is exactly the *dual* of the step for a cut !c/?d, and similarly for the other steps (see [28, Fig. 4]).

This elegant symmetry in the presentation of cut-elimination steps is lost in DiLL_0 sequent calculus, see our Figure 3. Moreover, cut-elimination in DiLL_0 sequent calculus has to deal with (many) uninteresting commutative steps, while interaction nets get rid of them. Thus, interaction nets allow one to express DiLL_0 cut-elimination with a sharper account than in sequent calculus. Not by chance, all papers dealing with DiLL_0 cut-elimination use only interaction nets [9, 28, 34, 12, 35, 30].

However, the interaction net presentation of DiLL_0 has some flaws that do not affect the sequent calculus: interaction nets do not have an *inductive tree-like structure* and so it is not easy to handle them. Moreover, not all interaction nets correspond to a derivation in DiLL_0 sequent calculus, a *global* geometrical correctness criterion is required to identify them.

Our contribution. We define a proof system for DiLL_0 in the formalism of *open deduction* [20] following the principles of *deep inference* [21, 2, 19, 31, 38]. Such a formalism, which allows rules to be applied deep in a context, provides a more flexible composition of derivations and makes explicit the behavior of the cut-elimination process in DiLL_0 in a more fine-grained way, since it pushes cut-elimination at an *atomic* level. Besides, our deep inference system for DiLL_0 gathers good qualities of both sequent calculus and interaction nets formalisms: it restores the interaction net *symmetries* lost in the sequent calculus and its derivations keep a handy *inductive* tree-like (or better, sequence-like) structure as in the sequent calculus, without the need for a global correctness criterion like in interaction nets.

A first attempt in the direction of a deep inference system for DiLL_0 is in [11] where, however, the sum-rule is absent and, as a consequence, it is not suitable to represent the dynamic behavior of DiLL_0.

$$\cfrac{}{\vdash A,\overline{A}}\,\text{ax} \quad \cfrac{\vdash \Gamma,A,B}{\vdash \Gamma,A\,\invamp\,B}\,\invamp \quad \cfrac{\vdash \Gamma,A \quad \vdash B,\Delta}{\vdash \Gamma,A\otimes B,\Delta}\,\otimes \quad \cfrac{}{\vdash 1}\,1 \quad \cfrac{\vdash \Gamma}{\vdash \Gamma,\bot}\,\bot \quad \cfrac{\vdash \Gamma,A \quad \vdash \overline{A},\Delta}{\vdash \Gamma,\Delta}\,\text{cut}$$

$$\cfrac{\vdash \Gamma,A}{\vdash \Gamma,!A}\,!\text{d} \quad \cfrac{\vdash \Gamma,A}{\vdash \Gamma,?A}\,?\text{d} \quad \cfrac{\vdash \Gamma,!A \quad \vdash !A,\Delta}{\vdash \Gamma,!A,\Delta}\,!\text{c} \quad \cfrac{\vdash \Gamma,?A,?A}{\vdash \Gamma,?A}\,?\text{c} \quad \cfrac{}{\vdash !A}\,!\text{w} \quad \cfrac{\vdash \Gamma}{\vdash \Gamma,?A}\,?\text{w} \quad \cfrac{\vdash \Gamma \quad \vdash \Gamma}{\vdash \Gamma}\,\text{sum} \quad \cfrac{}{\vdash \Gamma}\,\text{zero} \quad \Big| \quad \cfrac{\vdash ?\Gamma,A}{\vdash ?\Gamma,!A}\,!\text{p}$$

Figure 1: Sequent calculus rules for DiLL_0 (on the left) and the promotion rule (on the right) [28].

To fully recover the expressiveness of this logic, we design our system to include a binary connective $+$ which represents the sum operation. The rules for $+$ (and for its unit 0) prevent the use of Guglielmi and Tubella's general result [36, 37] to show cut-elimination. However, we are able to define a normalization procedure by rule permutations which fully captures the dynamics of DiLL_0 cut-elimination, in a way similar to the one in [32, 33]. Our system is sound and complete with respect to DiLL_0 sequent calculus, through a translation that commutes with cut-elimination/normalization.

In the normalization procedure, we can classify our rule permutations depending on their behavior: some rule permutations correspond to multiplicative cut-elimination steps, other permutations correspond to "resource management" cut-elimination steps (involving the ? and ! rules), other permutations correspond to "slice management" operations (involving the propagation of $+$ and 0).

2   Differential linear logic

We present here the classical, propositional, one-sided sequent calculus for *differential linear logic* without promotion (DiLL_0). The formulas of DiLL_0 are exactly the same as in the multiplicative exponential fragment of linear logic (MELL). MELL *formulas* are defined by the grammar below, where $a,b,c,\dots$ range over a countably infinite set of propositional variables:

$$A,B ::= a \mid \overline{a} \mid 1 \mid \bot \mid A\otimes B \mid A\,\invamp\,B \mid !A \mid ?A$$

Linear negation $\overline{(\cdot)}$ is defined through De Morgan laws so as to be involutive ($\overline{\overline{A}} = A$ for any A):

$$\overline{(a)} = \overline{a} \quad \overline{(\overline{a})} = a \quad \overline{A\otimes B} = \overline{A}\,\invamp\,\overline{B} \quad \overline{A\,\invamp\,B} = \overline{A}\otimes\overline{B} \quad \overline{1} = \bot \quad \overline{\bot} = 1 \quad \overline{!A} = ?\overline{A} \quad \overline{?A} = !\overline{A}$$

Variables and their negations are *atomic*; $\otimes,\invamp$ are *multiplicative connectives* and $1,\bot$ are their respective *units*; $!,?$ are *exponential modalities*. A MELL *sequent* is a (finite) multiset of MELL formulas $A_1,\dots,A_n$ (for any $n \geq 0$), and it is ranged over by Γ,Δ,Σ.

Figure 1 gives the sequent calculus rules[1] for *differential linear logic* DiLL_0 (without *promotion* $!\text{p}$); the rules on the first line correspond to the *multiplicative linear logic* fragment MLL. We set:

$$\text{MELL} = \text{MLL}\cup\{?\text{w},?\text{d},?\text{c},!\text{p}\} \qquad \text{DiLL}_0^- = \text{DiLL}_0 \setminus \{\text{zero},\text{sum}\}$$

We define $\equiv$ as the least congruence on DiLL_0 derivations generated by the relations in Figure 2. Roughly, the rule zero plays the role of annihilating element with respect to all the other rules but sum, for which it is a neutral element; whilst the rule sum commutes with any rule below it. Clearly, $\equiv$ preserves conclusions and can be oriented so as to define a terminating rewriting relation that pushes down the rules zero and sum in a derivation. As a consequence, every derivation in DiLL_0 can be rewritten in a *canonical form* (with the same conclusion).

Definition 2.1 (Canonical form, slice). Let π be a derivation in DiLL_0:

---

[1] Usually, in the literature on LL and DiLL, the rules ?w, ?d, ?c, !w, !d, !c are called respectively weakening, dereliction, contraction, co-weakening, co-dereliction and co-contraction. To avoid clashes with the usual terminology in deep inference (see Footnote 4), we call them ?-weakening, ?-dereliction, ?-contraction, !-weakening, !-dereliction and !-contraction, respectively.

$$\dfrac{\dfrac{}{\vdash\Gamma}\text{zero}}{\vdash\Delta}r_1 \equiv \dfrac{}{\vdash\Delta}\text{zero} \qquad \dfrac{\dfrac{}{\vdash\Gamma}\text{zero}\quad \vdash\Delta}{\vdash\Sigma}r_2 \equiv \dfrac{}{\vdash\Sigma}\text{zero} \equiv \dfrac{\vdash\Gamma\quad\dfrac{}{\vdash\Delta}\text{zero}}{\vdash\Sigma}r_2 \qquad \dfrac{\dfrac{}{\vdash\Gamma}\text{zero}\quad\vdash\Gamma}{\vdash\Gamma}\text{sum}\equiv\vdash\Gamma\equiv\dfrac{\vdash\Gamma\quad\dfrac{}{\vdash\Gamma}\text{zero}}{\vdash\Gamma}\text{sum} \qquad \dfrac{\dfrac{\vdash\Gamma\quad\vdash\Gamma}{\vdash\Gamma}\text{sum}}{\vdash\Delta}r_1\equiv\dfrac{\dfrac{\vdash\Gamma}{\vdash\Delta}r_1\quad\dfrac{\vdash\Gamma}{\vdash\Delta}r_1}{\vdash\Delta}\text{sum}$$

$$\dfrac{\dfrac{\vdash\Gamma\quad\vdash\Gamma}{\vdash\Gamma}\text{sum}\quad\vdash\Delta}{\vdash\Sigma}r_2\equiv\dfrac{\dfrac{\vdash\Gamma\quad\vdash\Delta}{\vdash\Sigma}r_2\quad\dfrac{\vdash\Gamma\quad\vdash\Delta}{\vdash\Sigma}r_2}{\vdash\Sigma}\text{sum}\equiv\dfrac{\vdash\Gamma\quad\dfrac{\vdash\Delta\quad\vdash\Delta}{\vdash\Delta}\text{sum}}{\vdash\Sigma}r_2\qquad \dfrac{\vdash\Gamma\quad\dfrac{\vdash\Gamma\quad\vdash\Gamma}{\vdash\Gamma}\text{sum}}{\vdash\Gamma}\text{sum}\equiv\dfrac{\dfrac{\vdash\Gamma\quad\vdash\Gamma}{\vdash\Gamma}\text{sum}\quad\vdash\Gamma}{\vdash\Gamma}\text{sum}$$

Figure 2: The congruence $\equiv$ on derivations generated by the rules zero and sum in DiLL_0, where r_1 is any unary rule in DiLL_0, and r_2 is any binary rule in DiLL_0 but sum.

1. π is a *slice* if it is in DiLL_0^- (*i.e.* the rules zero and sum do not occur in π);

2. π is *canonical* or in *canonical form* if either it consists of a zero rule, or it is a slice, or if its last rule is sum with a canonical form as left premise and a slice as right premise.

   A *canonical form* of π is any canonical derivation π' in DiLL_0 such that $\pi \equiv \pi'$.

Fact 2.2 (Canonicity). *Any derivation in DiLL_0 has a canonical form (with the same conclusion).*

Intuitively, considering only canonical derivations, slices—*i.e.* derivations in the subsystem DiLL_0^-—are the "real and meaningful" proofs in DiLL_0 (corresponding to simple nets in [9, 26, 28, 34]), while the rules sum and zero are needed to define cut-elimination in DiLL_0, in particular they ensure that the conclusion of a derivation is preserved after a cut-elimination step (the subsystem DiLL_0^- is not closed under cut-elimination, see below). The rule sum puts together slices with the same conclusions $\vdash\Gamma$, similarly to multiset union: it expresses the possibility of several "real proofs" of $\vdash\Gamma$. The rule zero then corresponds to the empty multiset of "real proofs" of $\vdash\Gamma$: it claims $\vdash\Gamma$ without a proof (it is reminiscent of daimon in ludics [14]). Because of it, any MELL sequent (also the empty one) is provable in DiLL_0.

Let π be a derivation in DiLL_0. We say that π is *with atomic axioms* (or η-*expanded*) if every instance of the rule ax introduces a MELL sequent of the form $\vdash a, \bar{a}$, where a is a propositional variable. We say that π is *cut-free* is it does not contain any instance of the rule cut, *i.e.* π is a derivation in $\mathsf{DiLL}_0 \setminus \{\mathsf{cut}\}$.

Proposition 2.3 (Atomic axioms). *For every derivation π in DiLL_0 with conclusion $\vdash\Gamma$, there exists a η-expanded derivation π' in DiLL_0 with conclusion $\vdash\Gamma$. If, moreover, π is canonical (resp. a slice) then π' is canonical (resp. a slice).*

Proof. Rewrite any non-atomic instance of the rule ax according to the η-*expansion* relation $\leadsto_\eta$ below:

$$\dfrac{}{\vdash A\otimes B, \overline{A}\,\mathbin{\rotatebox[origin=c]{180}{\&}}\,\overline{B}}\text{ax} \leadsto_\eta \dfrac{\dfrac{\dfrac{}{\vdash A, \overline{A}}\text{ax}\quad \dfrac{}{\vdash B, \overline{B}}\text{ax}}{\vdash A\otimes B, \overline{A},\overline{B}}\otimes}{\vdash A\otimes B, \overline{A}\,\mathbin{\rotatebox[origin=c]{180}{\&}}\,\overline{B}}\mathbin{\rotatebox[origin=c]{180}{\&}} \qquad \dfrac{}{\vdash 1, \bot}\text{ax}\leadsto_\eta \dfrac{\dfrac{}{\vdash 1}1}{\vdash 1, \bot}\bot \qquad \dfrac{}{\vdash\,!A, ?\overline{A}}\text{ax}\leadsto_\eta\dfrac{\dfrac{\dfrac{}{\vdash A, \overline{A}}\text{ax}}{\vdash A, ?\overline{A}}?\text{d}}{\vdash\,!A, ?\overline{A}}!\text{d} \tag{1}$$

It is immediate to prove that the relation $\leadsto_\eta$ on the derivations of DiLL_0 is terminating. $\qquad\square$

Cut-elimination. Despite its incoherence, DiLL_0 provides a fine analysis of resource consumption in cut-elimination. Rewriting rules $\leadsto_{\text{cut}}$ for cut-elimination in DiLL_0 sequent calculus are defined in Figure 3. They are just the formulation in the sequent calculus formalism of the cut-elimination steps defined and studied in [9, 34, 12] and [28, Fig. 4] within the interaction nets formalism. We represent in Figure 3 only the *key cases*, where the principal connectives in the cut formulas are dual (the pairs of dual connectives are $\otimes/\mathbin{\rotatebox[origin=c]{180}{\&}}$, $1/\bot$, $!/?$). The way DiLL_0 deals with the *commutative cases* is omitted since is analogous to usual sequent calculi. With these cut-elimination steps it has been proved in [9, 28, 12] that the rule cut is admissible in DiLL_0 (and even in DiLL, *i.e.*, the system DiLL_0 plus MELL promotion $!\mathsf{p}$).

Figure 3: Key cases of cut-elimination rewriting rules for DiLL_0 sequent calculus (colors highlight cut-relations between formula occurrences).

Theorem 2.4 (Cut-elimination, [9, 28, 12]). *For every derivation π in DiLL_0 with conclusion $\vdash \Gamma$, there exists a cut-free derivation π' in DiLL_0 with conclusion $\vdash \Gamma$ such that $\pi \leadsto^*_{\mathsf{cut}} \pi'$.*

Cut-elimination preserves atomic axioms: if $\pi \leadsto_{\mathsf{cut}} \pi'$ and π is η-expanded, then π' is η-expanded. Note that if $\pi \leadsto_{\mathsf{cut}} \pi'$ with π canonical then π' is not necessarily canonical (*e.g.* if in π a cut ?c/!d or ?d/!w is above another rule), but π' can be rewritten in a canonical form (see Fact 2.2 above). Indeed, DiLL_0^- is not closed under cut-elimination: steps ?c/!d or ?d/!w create instances of the rule sum or zero.

To explain the importance of the rules sum and zero as *resource management*, we give an informal

account of the cut-elimination steps in Figure 3 for the key cases involving !/?. Roughly, they follow the "law of supply and demand" so as to be *resource-sensitive*: in each slice no duplication or erasure is allowed. The rules for ? (?w, ?d, ?c) *ask for* a number of resources of type $!A$ (0, 1, and the sum of the numbers asked by its premises, respectively), while the rules for ! (!w, !d, !c) *supply* a number of resources of type $!A$ (0, 1, and the sum of the numbers supplied by its premises, respectively). Cases:

1. If the numbers of demanded and supplied resources match, the cut-elimination proceeds normally (see the steps ?d/!d and ?w/!w).

2. The step ?c/!c is slightly more complex: intuitively, it connects the dual premises of a ?-contraction and of a !-contraction in all possible ways.

3. The step ?c/!w duplicates the rule !w, spreading the information that there are no available resources to the premises of ?c.

4. The step ?d/!w represents a *mismatch* in supply and demand: ?-dereliction asks for a resource but !-weakening says that it is not available; the rule zero in the resulting derivation keeps track of this mismatch, as a sort of error in computation, and ensures that the conclusion is preserved.

5. In the step ?c/!d, ?-contraction represents two possible demands for a resource, but according to !-dereliction only one resource is available, so there is a *non-deterministic choice* on which request will be fed, the other one will receive a !-weakening; the rule sum has to be intended as a way to keep track of all possible choices, *not* as a way to *duplicate resources*; said differently, in the step ?c/!d a derivation reduces to a pair of derivations (of slices, if we consider their canonical forms).

By duality, the discussion above about resource management is similar for the steps ?w/!c, ?w/!d and ?d/!c, respectively. Figure 4 provides an example of the cut-elimination procedure in DiLL_0.

It is worth comparing cut-elimination steps as defined for DiLL_0 sequent calculus (Figure 3) and for DiLL_0 interaction nets ([9, Sect. 2], [28, Fig. 4]): symmetry and duality in the latter are lost in the former.

As there is no promotion rule !p, in DiLL_0 transforming a derivation in DiLL_0 into one with atomic axioms does not commute with cut-elimination. For instance, derivation π below reduces to π' via cut-elimination; but derivation π_η with atomic axioms, obtained from π through η-expansion (the procedure described in the proof of Proposition 2.3) reduces to $\pi'_\eta \neq \pi'$ via cut-elimination.

$$\pi = \cfrac{\cfrac{}{\vdash !\overline{a}, ?a}\, {}^{ax} \quad \cfrac{}{\vdash ?a}\, {}^{?w}}{\vdash ?a}\, {}_{cut} \quad \rightsquigarrow_{cut} \quad \cfrac{}{\vdash ?a}\, {}^{?w} = \pi' \qquad\qquad \pi_\eta = \cfrac{\cfrac{\cfrac{\cfrac{}{\vdash \overline{a}, a}\, {}^{ax}}{\vdash \overline{a}, ?a}\, {}^{?d}}{\vdash !\overline{a}, ?a}\, {}^{!d} \quad \cfrac{}{\vdash ?a}\, {}^{?w}}{\vdash ?a}\, {}_{cut} \quad \rightsquigarrow_{cut} \quad \cfrac{}{\vdash ?a}\, {}^{zero} = \pi'_\eta$$

3   A calculus of structures for DiLL_0

In this section, we introduce a deep inference system [21, 2, 19, 38] suitable for DiLL_0, using the *open deduction* formalism [20]. As a first novelty, we internalize the rules zero and sum of DiLL_0 sequent calculus at the level of formulas. In fact, derivations in deep inference systems have a sequence structure instead of the more general tree-like structure of sequent calculus: every rule in deep inference has exactly one premise, consisting of one formula. This because the meta-connectives for sequent composition (the comma) and sequent juxtaposition (derivation branching) are internalized by $\parr$ and $\otimes$, respectively. To internalize the DiLL_0 meta-connective for sum, together with its unit, we introduce the (commutative and associative) binary connective $+$ and its unit 0.[2] In this way, the rule sum branches the derivation tree

---

[2] Here, the new connective $+$ has nothing to do with the additive disjunction $\oplus$ in LL; and the unit 0 for $+$ must not be confused with the additive unit 0 for $\oplus$ in LL.
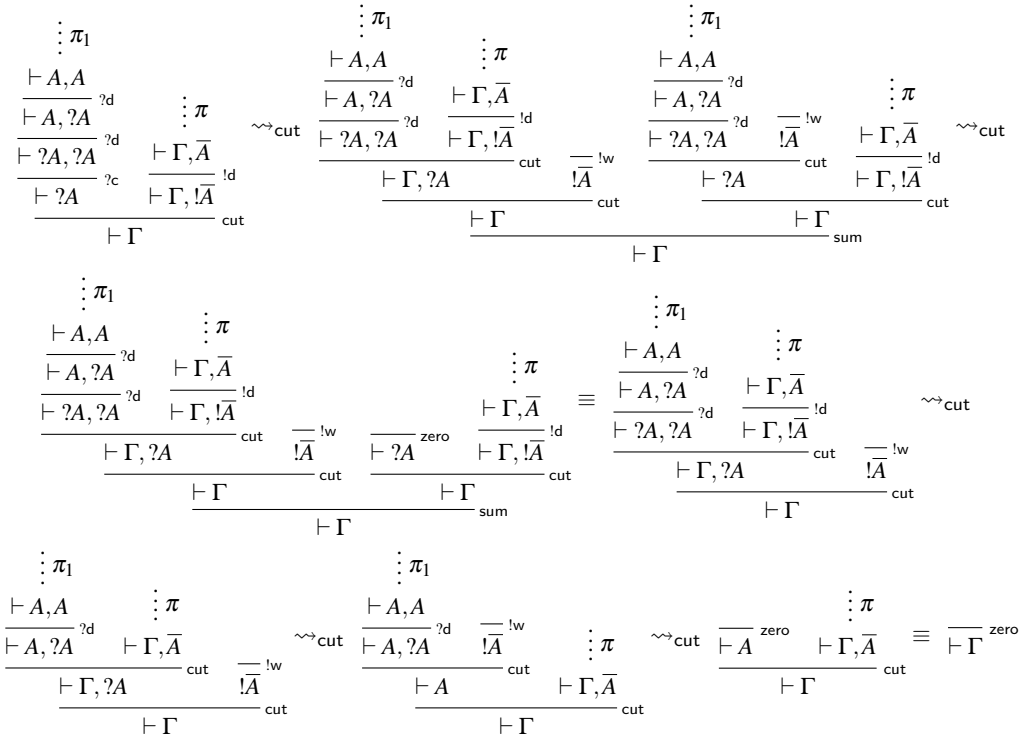
Figure 4: An example of the cut-elimination procedure in DiLL_0 sequent calculus.

with a connective, $+$; and similarly, the rule zero has is own premise, 0. Thus, *formulas* are defined by:

$$A, B ::= a \mid \bar{a} \mid A \otimes B \mid A \,\mathord{⅋}\, B \mid 1 \mid \bot \mid !A \mid ?B \mid 0 \mid A + B$$

where $a, b, c, \ldots$ range over the usual countably infinite set of propositional variables (so, a MELL formula as defined on p. 3 is a formula with no occurrences of $+$ and 0). Formulas are identified up to the *equivalence* $\simeq$ defined as the least congruence on formulas generated by the relations in (2).

$$
\begin{array}{lll}
A \,\mathord{⅋}\, B \simeq B \,\mathord{⅋}\, A & A \otimes B \simeq B \otimes A & A + B \simeq B + A \\
A \,\mathord{⅋}\, (B \,\mathord{⅋}\, C) \simeq (A \,\mathord{⅋}\, B) \,\mathord{⅋}\, C & A \otimes (B \otimes C) \simeq (A \otimes B) \otimes C & A + (B + C) \simeq (A + B) + C \\
A \,\mathord{⅋}\, \bot \simeq A & A \otimes 1 \simeq A & A + 0 \simeq A \\
A \,\mathord{⅋}\, (B + C) \simeq (A \,\mathord{⅋}\, B) + (A \,\mathord{⅋}\, C) & A \otimes (B + C) \simeq (A \otimes B) + (A \otimes C) & \\
!(A + B) \simeq {!A} + {!B} & ?(A + B) \simeq {?A} + {?B} & \\
0 \otimes A \simeq 0 \qquad 0 \,\mathord{⅋}\, A \simeq 0 & ?0 \simeq 0 \qquad !0 \simeq 0 &
\end{array}
\tag{2}
$$

Some equivalences in (2) correspond to well-known isomorphisms in MLL. With respect to $\simeq$, the formula 0 is an annihilating element for all other connectives but $+$, for which it is a neutral element; every connective other than $+$ distributes over $+$.

An *additive normal* formula A is a sum of MELL formulas, *i.e.* $A = A_1 + \cdots + A_n$ ($n \in \mathbb{N}$) where all A_i's are MELL formulas ($A = 0$ for $n = 0$). For any $n \in \mathbb{N}$, we set $\mathsf{n} = \underbrace{1 + \cdots + 1}_{n \text{ times}}$. Note that, by the equivalences in (2), $\mathsf{n} \otimes \mathsf{m} = \mathsf{k}$ where $k = n \times m$.

A *context* (resp. MELL *context*) $\mathsf{C}\{\}$ is a formula (resp. MELL formula) with exactly one occurrence of the hole $\{\}$ (which can be thought of as a special propositional variable). We write $\mathsf{C}\{A\}$ for the formula obtained from the context $\mathsf{C}\{\}$ by replacing its hole with the formula A.

Remark 3.1 (Additive normal form)**.** By definition of $\simeq$, if $\mathsf{C}\{\}$ is a context, $\mathsf{C}\{A+0\} \simeq \mathsf{C}\{A\}$ and $\mathsf{C}\{A+B\} \simeq \mathsf{C}\{A\}+\mathsf{C}\{B\}$. If $\mathsf{C}\{\}$ is a MELL context, $\mathsf{C}\{0\} \simeq 0$. In general, any formula A has an additive normal formula A' such that $A' \simeq A$. Indeed, equivalences in (2) but the ones on the first line can be oriented to define a terminating rewriting relation whose normal forms are additive normal formulas.

Derivations. A *deep inference system* $\mathcal{S}$ is a set of unary inference rules. A *derivation* $\mathcal{D}$ from a *premise* B to a *conclusion* A in a deep inference system $\mathcal{S}$, noted $\mathcal{D}\|\mathcal{S}$ with $\genfrac{}{}{0pt}{}{B}{A}$ or $\mathcal{D} \triangleright B \overset{\mathcal{S}}{\vdash\!\!-} A$, is defined as follows:

- (*assumption*) a formula A is a derivation (denoted by A) with premise and conclusion A;
- (*horizontal composition*) if for all $i \in \{1,2\}$ $\mathcal{D}_i$ is a derivation from B_i to A_i, then for any $\bullet \in \{\mathfrak{V}, \otimes, +\}$, $\mathcal{D}_1 \bullet \mathcal{D}_2$ is a derivation from $B_1 \bullet B_2$ to $A_1 \bullet A_2$ (see (3) below on the left);
- (*vertical composition*) if $\rho\, \dfrac{A_1}{B_2} \in \mathcal{S}$ and, for all $i \in \{1,2\}$, $\mathcal{D}_i$ is a derivation from B_i to A_i, then $\mathcal{D}_1 \circ_\rho \mathcal{D}_2$ is a derivation from B_1 to A_2 (see (3) below on the right).[3]

$$
\begin{array}{c}
B_1 \bullet B_2 \\
\mathcal{D}_1 \bullet \mathcal{D}_2 \| \mathcal{S} \\
A_1 \bullet A_2
\end{array}
=
\boxed{\begin{array}{c} B_1 \\ \mathcal{D}_1 \| \mathcal{S} \\ A_1 \end{array}} \bullet \boxed{\begin{array}{c} B_2 \\ \mathcal{D}_2 \| \mathcal{S} \\ A_2 \end{array}}
\quad \text{for } \bullet \in \{\mathfrak{V}, \otimes, +\}
\qquad
\begin{array}{c}
B_1 \\
\mathcal{D}_1 \circ_\rho \mathcal{D}_2 \| \mathcal{S} \\
A_2
\end{array}
= \rho \,
\begin{array}{c}
B_1 \\
\mathcal{D}_1 \| \mathcal{S} \\
A_1 \\
\hline
B_2 \\
\mathcal{D}_2 \| \mathcal{S} \\
A_2
\end{array}
\quad \text{for } \rho \in \mathcal{S}
\qquad (3)
$$

We write $B \overset{\mathcal{S}}{\vdash\!\!-} A$ if there is a derivation $\mathcal{D} \triangleright B \overset{\mathcal{S}}{\vdash\!\!-} A$. A rule $\rho\, \dfrac{B}{A}$ is *derivable* in $\mathcal{S}$ if $B \overset{\mathcal{S}}{\vdash\!\!-} A$.

The system SDDI is defined by the rules in Figure 5. All rules in SDDI have exactly one premise, as usual in deep inference. The *down-fragment* and *up-fragment*[4] of SDDI are the following sets of rules:

$$\mathsf{DDI}^\downarrow = \{\mathsf{ai}^\downarrow, !\mathsf{d}^\downarrow, ?\mathsf{d}^\downarrow, !\mathsf{w}^\downarrow, ?\mathsf{w}^\downarrow, !\mathsf{c}^\downarrow, ?\mathsf{c}^\downarrow, +^\downarrow, 0^\downarrow, \mathsf{s}\} \qquad \mathsf{DDI}^\uparrow = \{\mathsf{ai}^\uparrow, !\mathsf{d}^\uparrow, ?\mathsf{d}^\uparrow, !\mathsf{w}^\uparrow, ?\mathsf{w}^\uparrow, !\mathsf{c}^\uparrow, ?\mathsf{c}^\uparrow, +^\uparrow, 0^\uparrow, \mathsf{s}\}$$

Note the up/down symmetry between $\mathsf{DDI}^\downarrow$ and $\mathsf{DDI}^\uparrow$, and that $\mathsf{SDDI} = \mathsf{DDI}^\downarrow \cup \mathsf{DDI}^\uparrow$ with $\mathsf{DDI}^\downarrow \cap \mathsf{DDI}^\uparrow = \{\mathsf{s}\}$. We set $\mathsf{DDI}^\downarrow_- = \mathsf{DDI}^\downarrow \setminus \{+^\downarrow, 0^\downarrow\}$. Note that in a $\mathsf{DDI}^\downarrow_-$ derivation only MELL formulas occur.

Roughly, rules in $\mathsf{DDI}^\downarrow$ somehow mimic the ones in $\mathsf{DiLL}_0 \setminus \{\mathsf{cut}\}$. Rules in $\mathsf{DDI}^\uparrow$ are their duals, turning them upside down. Derivations in $\mathsf{DDI}^\downarrow_-$ correspond to cut-free slices in DiLL_0^- (see Theorem 4.3).

Remark 3.2 (Deep)**.** The idea of deep inference is that inference rules can be applied "deep" in any context: in a deep inference system $\mathcal{S}$, if $\rho\, \dfrac{B}{A} \in \mathcal{S}$ then, for any context $\mathsf{C}\{\}$, $\rho\, \dfrac{\mathsf{C}\{B\}}{\mathsf{C}\{A\}}$ is derivable in $\mathcal{S}$. Therefore, a derivation in $\mathcal{S}$ can be seen as a finite *sequence* of "deep" rules: for instance, the derivation $\boxed{?\mathsf{d}^\downarrow\, \dfrac{a}{?a}} \otimes \boxed{!\mathsf{d}^\downarrow\, \dfrac{b}{!b}}$ in $\mathsf{DDI}^\downarrow$ (with parallel $?\mathsf{d}^\downarrow$ and $!\mathsf{d}^\downarrow$) can be "sequenced" as both $?\mathsf{d}^\downarrow\, \dfrac{!\mathsf{d}^\downarrow\, \dfrac{a \otimes b}{a \otimes !b}}{?a \otimes !b}$ and $!\mathsf{d}^\downarrow\, \dfrac{?\mathsf{d}^\downarrow\, \dfrac{a \otimes b}{?a \otimes b}}{?a \otimes !b}$. Often we implicitly identify a derivation in a deep inference system $\mathcal{S}$ with its sequenced presentations.

---

[3] We can write $\simeq \dfrac{A}{B}$ as a rule in a derivation if $A \simeq B$, although formally its use is implicit as formulas are identified up to $\simeq$.

[4] Usually in the literature on deep inference, the dual rule $\mathsf{r}^\uparrow$ of a rule $\mathsf{r}^\downarrow$ is called "co-r". We avoid these names because they clash with the usual terminology in the literature on DiLL_0, see Footnote 1.

$$\text{ai}^\downarrow\frac{1}{a\,\mathbin{\rotatebox[origin=c]{180}{\&}}\,\bar a}\qquad \text{!d}^\downarrow\frac{A}{!A}\qquad \text{?d}^\downarrow\frac{A}{?A}\qquad \text{!w}^\downarrow\frac{1}{!A}\qquad \text{?w}^\downarrow\frac{\bot}{?A}\qquad \text{!c}^\downarrow\frac{!A\otimes !A}{!A}\qquad \text{?c}^\downarrow\frac{?A\,\mathbin{\rotatebox[origin=c]{180}{\&}}\,?A}{?A}\qquad +^\downarrow\frac{A+A}{A}\qquad 0^\downarrow\frac{0}{A}\qquad \text{s}\,\frac{A\otimes(B\,\mathbin{\rotatebox[origin=c]{180}{\&}}\,C)}{(A\otimes B)\,\mathbin{\rotatebox[origin=c]{180}{\&}}\,C}$$

$$\text{ai}^\uparrow\frac{a\otimes\bar a}{\bot}\qquad \text{!d}^\uparrow\frac{?A}{A}\qquad \text{?d}^\uparrow\frac{!A}{A}\qquad \text{!w}^\uparrow\frac{?A}{\bot}\qquad \text{?w}^\uparrow\frac{!A}{1}\qquad \text{!c}^\uparrow\frac{?A}{?A\,\mathbin{\rotatebox[origin=c]{180}{\&}}\,?A}\qquad \text{?c}^\uparrow\frac{!A}{!A\otimes !A}\qquad +^\uparrow\frac{A}{A+A}\qquad 0^\uparrow\frac{A}{0}$$

Figure 5: The rules of the deep inference system SDDI (A,B,C are MELL formulas).

Remark 3.3 (Big one). For any formula A and $n\in\mathbb{N}$, if $\mathcal{D}\triangleright n\overset{\text{SDDI}}{\vdash}A$, then there is a derivation $\mathcal{D}'\triangleright 1\overset{\text{SDDI}}{\vdash}A$. Indeed, $\mathcal{D}'$ is built from $\mathcal{D}$ by adding one rule $0^\uparrow$ if $n=0$, or $n-1$ rules $+^\uparrow$ if $n>1$, on top of $\mathcal{D}$.

System SDDI has only the *atomic* introduction rules $\text{ai}^\downarrow$ and $\text{ai}^\uparrow$ (indeed a is a propositional variable in Figure 5): they can be seen as the atomic version of ax- and cut-rules of sequent calculus, respectively. The *non-atomic* versions of the rules $\text{ai}^\downarrow$ and $\text{ai}^\uparrow$ are respectively:

$$\text{i}^\downarrow\frac{1}{A\,\mathbin{\rotatebox[origin=c]{180}{\&}}\,\bar A}\qquad\qquad \text{i}^\uparrow\frac{A\otimes\bar A}{\bot}\qquad\qquad\text{(where A is a MELL formula)}$$

However, the rules $\text{i}^\downarrow$ and $\text{i}^\uparrow$ are derivable in SDDI (Lemma 3.4). Derivability of $\text{i}^\downarrow$ is analogous to the fact that a derivation can be transformed to one with atomic axioms in DiLL_0 sequent calculus (Proposition 2.3), but derivability of $\text{i}^\uparrow$ is a typical result in deep inference systems that does not have a corresponding result in the sequent calculus: it says that restricting cuts to an *atomic* level is not limiting.

Lemma 3.4 (Atomic axioms and atomic cuts). *The rule $\text{i}^\downarrow$ is derivable in $\{\text{ai}^\downarrow,\text{s},\text{?d}^\downarrow,\text{!d}^\downarrow\}$; and the rule $\text{i}^\uparrow$ is derivable in $\{\text{ai}^\uparrow,\text{s},\text{?d}^\uparrow,\text{!d}^\uparrow\}$.*

Proof. Concerning $\text{i}^\downarrow$, the proof is by induction on the MELL formula A in $\text{i}^\downarrow\dfrac{1}{A\,\mathbin{\rotatebox[origin=c]{180}{\&}}\,\bar A}$:

- if $A=a$ is a propositional variable (and similarly if $A=\bar a$), then $\text{ai}^\downarrow\dfrac{1}{a\,\mathbin{\rotatebox[origin=c]{180}{\&}}\,\bar a}$;

- if $A=1$ (and similarly if $A=\bot$), then $\simeq\dfrac{1}{1\,\mathbin{\rotatebox[origin=c]{180}{\&}}\,\bot}$;

- if $A=B\otimes C$ (and similarly for $A=B\,\mathbin{\rotatebox[origin=c]{180}{\&}}\,C$), then

$$\simeq\cfrac{1}{\;\cfrac{\left.\begin{array}{c}1\\[2pt]\text{IH}\|\\[2pt]B\,\mathbin{\rotatebox[origin=c]{180}{\&}}\,\bar B\end{array}\right.\;\otimes\;\left.\begin{array}{c}1\\[2pt]\text{IH}\|\\[2pt]C\,\mathbin{\rotatebox[origin=c]{180}{\&}}\,\bar C\end{array}\right.}{2\times\text{s}\;\;(B\otimes C)\,\mathbin{\rotatebox[origin=c]{180}{\&}}\,(\bar B\,\mathbin{\rotatebox[origin=c]{180}{\&}}\,\bar C)}}$$

- if $A=!B$ (and similarly if $A=?B$), then

$$\cfrac{1}{\left.\begin{array}{c}\text{IH}\|\\[2pt]\text{!d}^\downarrow\dfrac{B}{!B}\;\mathbin{\rotatebox[origin=c]{180}{\&}}\;\text{?d}^\downarrow\dfrac{\bar B}{?\bar B}\end{array}\right.}$$

The proof for $\text{i}^\uparrow$ is dual, using $\text{ai}^\uparrow$, $\text{!d}^\uparrow$ and $\text{?d}^\uparrow$ instead of $\text{ai}^\downarrow$, $\text{!d}^\downarrow$ and $\text{?d}^\downarrow$, respectively. $\square$

The rule $\text{i}^\uparrow$ plays a special role in deep inference systems, as the cut does in sequent calculi. Thanks to $\simeq$ and $0^\downarrow$, it makes superfluous all the rules in $\text{DDI}^\uparrow$ (second line in Figure 5) but $0^\uparrow$ and s. Note that $\text{ai}^\uparrow$ is not enough for that, because $\text{i}^\uparrow$ needs $\text{!d}^\uparrow$ and $\text{?d}^\uparrow$ to be simulated by $\text{ai}^\uparrow$, as seen in Lemma 3.4.

Lemma 3.5 (Getting rid of up-rules via $\text{i}^\uparrow$ and $0^\downarrow$). *Any rule $\rho^\uparrow\in\{\text{!d}^\uparrow,\text{?d}^\uparrow,\text{!c}^\uparrow,\text{?c}^\uparrow,\text{!w}^\uparrow,\text{?w}^\uparrow\}$ is derivable in $\{\rho^\downarrow,\text{i}^\uparrow,\text{i}^\downarrow,\text{s}\}$; the rule $+^\uparrow$ is derivable in $\{0^\downarrow\}$.*

Figure 6: Translation of η-expanded DiLL_0 sequent calculus derivations into $\mathsf{DDI}^{\downarrow} \cup \{\mathsf{i}^{\uparrow}\}$ derivations $(m = n_1 \times n_2)$.

Proof. For a rule $\rho^{\uparrow} \dfrac{\bar{B}}{\bar{A}}$ with $\rho^{\uparrow} \in \{!\mathsf{d}^{\uparrow}, ?\mathsf{d}^{\uparrow}, !\mathsf{c}^{\uparrow}, ?\mathsf{c}^{\uparrow}, !\mathsf{w}^{\uparrow}, ?\mathsf{w}^{\uparrow}\}$, see (4). For the rule $+^{\uparrow} \dfrac{A}{A+A}$, see (5).

$$(4)$$

$$(5)$$

4   Correspondence between DiLL_0 and SDDI

In this section we prove that SDDI is a *sound* and *complete* proof system for DiLL_0 sequent calculus. At first sight, this result is obvious because the rules zero in DiLL_0 and $0^{\downarrow}$ in SDDI make everything provable. But the interest is to show that the fragments without zero and $0^{\downarrow}$ correspond to each other.

If $\Gamma = A_1, \dots, A_n$ (with $n \in \mathbb{N}$) is a MELL sequent, we set $[\![\Gamma]\!] = A_1 \,\mathfrak{P} \cdots \mathfrak{P}\, A_n$ (so, $[\![\Gamma]\!] = \perp$ for $n = 0$).

Theorem 4.1 (Completeness). *Let Γ be a MELL sequent. If $\overset{\mathsf{DiLL}_0}{\vdash} \Gamma$ (resp. $\overset{\mathsf{DiLL}_0 \backslash \{\mathsf{cut}\}}{\vdash} \Gamma$), then* $\mathsf{n} \overset{\mathsf{DDI}^{\downarrow} \cup \{\mathsf{i}^{\uparrow}\}}{\vdash} [\![\Gamma]\!]$

$$\mathsf{ai}^{\downarrow}\frac{1}{a\,\mathscr{B}\,\bar{a}} \to \frac{\dfrac{}{\vdash a,\bar{a}}\,\mathsf{ax}}{\dfrac{\vdash a\,\mathscr{B}\,\bar{a}}{\vdash a\,\mathscr{B}\,\bar{a},\perp}\,\perp}\,\mathscr{B} \qquad \mathsf{i}^{\uparrow}\frac{A\otimes\bar{A}}{\perp} \to \frac{\dfrac{}{\vdash \bar{A},A}\,\mathsf{ax}}{\dfrac{\vdash \bar{A}\,\mathscr{B}\,A}{\vdash \bar{A}\,\mathscr{B}\,A,\perp}\,\perp}\,\mathscr{B} \qquad \mathsf{!w}^{\downarrow}\frac{1}{!A} \to \frac{\dfrac{}{\vdash !A}\,\mathsf{!w}}{\vdash !A,\perp}\,\perp \qquad \mathsf{?w}^{\downarrow}\frac{\perp}{?A} \to \frac{\dfrac{}{\vdash 1}\,1}{\vdash 1,?A}\,\mathsf{?w}$$

$$\mathsf{?d}^{\downarrow}\frac{A}{?A} \to \frac{\dfrac{}{\vdash \bar{A},A}\,\mathsf{ax}}{\vdash \bar{A},?A}\,\mathsf{?d} \qquad \mathsf{!d}^{\downarrow}\frac{A}{!A} \to \frac{\dfrac{}{\vdash \bar{A},A}\,\mathsf{ax}}{\vdash \bar{A},!A}\,\mathsf{!d} \qquad \mathsf{s}\frac{A\otimes(B\,\mathscr{B}\,C)}{(A\otimes B)\,\mathscr{B}\,C} \to \frac{\dfrac{}{\vdash \bar{A},A}\,\mathsf{ax}\quad \dfrac{\dfrac{}{\vdash \bar{B},B}\,\mathsf{ax}\quad \dfrac{}{\vdash \bar{C},C}\,\mathsf{ax}}{\vdash B,C,\bar{B}\otimes\bar{C}}\,\otimes}{\dfrac{\vdash A\otimes B,C,\bar{A},\bar{B}\otimes\bar{C}}{\vdash (A\otimes B)\,\mathscr{B}\,C,\bar{A}\,\mathscr{B}\,(\bar{B}\otimes\bar{C})}\,2\times\mathscr{B}}\,\otimes$$

$$\mathsf{?c}^{\downarrow}\frac{?A\,\mathscr{B}\,?A}{?A} \to \frac{\dfrac{\dfrac{}{\vdash !\bar{A},?A}\,\mathsf{ax}\quad \dfrac{}{\vdash !\bar{A},?A}\,\mathsf{ax}}{\vdash !\bar{A}\otimes!\bar{A},?A,?A}\,\otimes}{\vdash !\bar{A}\otimes!\bar{A},?A}\,\mathsf{?c} \qquad \mathsf{!c}^{\downarrow}\frac{!A\otimes!A}{!A} \to \frac{\dfrac{\dfrac{}{\vdash ?\bar{A},!A}\,\mathsf{ax}\quad \dfrac{}{\vdash ?\bar{A},!A}\,\mathsf{ax}}{\vdash ?\bar{A},?\bar{A},!A}\,\mathsf{!c}}{\vdash !A,?\bar{A}\,\mathscr{B}\,?\bar{A}}\,\mathscr{B}$$

$$\simeq\frac{A\,\mathscr{B}\,(B\,\mathscr{B}\,C)}{(A\,\mathscr{B}\,B)\,\mathscr{B}\,C} \to \frac{\dfrac{}{\vdash \bar{A},A}\,\mathsf{ax}\quad \dfrac{\dfrac{}{\vdash \bar{B},B}\,\mathsf{ax}\quad \dfrac{}{\vdash \bar{C},C}\,\mathsf{ax}}{\vdash B,C,\bar{B}\otimes\bar{C}}\,\otimes}{\dfrac{\vdash \bar{A}\otimes(\bar{B}\otimes\bar{C}),A,B,C}{\vdash \bar{A}\otimes(\bar{B}\otimes\bar{C}),(A\,\mathscr{B}\,B)\,\mathscr{B}\,C}\,2\times\mathscr{B}}\,\otimes \qquad \simeq\frac{(A\,\mathscr{B}\,B)\,\mathscr{B}\,C}{A\,\mathscr{B}\,(B\,\mathscr{B}\,C)} \to \frac{\dfrac{\dfrac{}{\vdash \bar{A},A}\,\mathsf{ax}\quad \dfrac{}{\vdash \bar{B},B}\,\mathsf{ax}}{\vdash B,A,\bar{A}\otimes\bar{B}}\,\otimes\quad \dfrac{}{\vdash \bar{C},C}\,\mathsf{ax}}{\dfrac{\vdash (\bar{A}\otimes\bar{B})\otimes\bar{C},A,B,C}{\vdash (\bar{A}\otimes\bar{B})\otimes\bar{C},A\,\mathscr{B}\,(B\,\mathscr{B}\,C)}\,2\times\mathscr{B}}\,\otimes$$

$$\simeq\frac{A\,\mathscr{B}\,B}{B\,\mathscr{B}\,A} \to \frac{\dfrac{\dfrac{}{\vdash \bar{A},A}\,\mathsf{ax}\quad \dfrac{}{\vdash \bar{B},B}\,\mathsf{ax}}{\vdash \bar{A}\otimes\bar{B},B,A}\,\otimes}{\vdash \bar{A}\otimes\bar{B},B\,\mathscr{B}\,A}\,\mathscr{B} \qquad \simeq\frac{A}{A\,\mathscr{B}\,\perp} \to \frac{\dfrac{\dfrac{}{\vdash \bar{A},A}\,\mathsf{ax}}{\vdash \bar{A},A,\perp}\,\perp}{\vdash \bar{A},A\,\mathscr{B}\,\perp}\,\mathscr{B} \qquad \simeq\frac{A\,\mathscr{B}\,\perp}{A} \to \frac{\dfrac{}{\vdash \bar{A},A}\,\mathsf{ax}\quad \dfrac{}{\vdash 1}\,1}{\vdash \bar{A}\otimes 1,A}\,\mathscr{B}$$

Figure 7: Interpretation of the rules in $\mathsf{DDI}^{\downarrow}_{-}\cup\{\mathsf{i}^{\uparrow}\}$ and of $\simeq$ as derivations in DiLL_0 sequent calculus.

(resp. $\mathsf{n}\overset{\mathsf{DDI}^{\downarrow}}{\vdash}[\![\Gamma]\!]$) for some $n\in\mathbb{N}$, and $1\overset{\mathsf{SDDI}}{\vdash}[\![\Gamma]\!]$. Moreover, if $\overset{\mathsf{DiLL}^{-}_0}{\vdash}\Gamma$ then $1\overset{\mathsf{DDI}^{\downarrow}_{-}\cup\{\mathsf{i}^{\uparrow}\}}{\vdash}[\![\Gamma]\!]$; if $\overset{\mathsf{DiLL}^{-}_0\setminus\{\mathsf{cut}\}}{\vdash}\Gamma$ then $1\overset{\mathsf{DDI}^{\downarrow}_{-}}{\vdash}[\![\Gamma]\!]$; if $\overset{\{\mathsf{zero}\}}{\vdash}\Gamma$ then $0\overset{\{0^{\downarrow}\}}{\vdash}[\![\Gamma]\!]$.

Proof. If we show that $\overset{\mathsf{DiLL}_0}{\vdash}\Gamma$ implies $\mathsf{n}\overset{\mathsf{DDI}^{\downarrow}\cup\{\mathsf{i}^{\uparrow}\}}{\vdash}[\![\Gamma]\!]$, then $\mathsf{n}\overset{\mathsf{SDDI}}{\vdash}[\![\Gamma]\!]$ by Lemma 3.4, and thus $1\overset{\mathsf{SDDI}}{\vdash}[\![\Gamma]\!]$ by Remark 3.3. So, let π be a derivation of $\vdash\Gamma$ in DiLL_0. By Proposition 2.3 we can assume that π is η-expanded. By induction on π, we define a derivation $[\![\pi]\!]\triangleright \mathsf{n}\overset{\mathsf{DDI}^{\downarrow}\cup\{\mathsf{i}^{\uparrow}\}}{\vdash}[\![\Gamma]\!]$ (for some $n\in\mathbb{N}$) as shown in Figure 6. According to this translation, if π is in $\mathsf{DiLL}_0\setminus\{\mathsf{cut}\}$ (resp. in $\mathsf{DiLL}^{-}_0\setminus\{\mathsf{cut}\}$; in DiLL^{-}_0; in $\{\mathsf{zero}\}$) then $[\![\pi]\!]$ is in $\mathsf{DDI}^{\downarrow}$ (resp. in $\mathsf{DDI}^{\downarrow}_{-}$ and $n=1$; in $\mathsf{DDI}^{\downarrow}_{-}\cup\{\mathsf{i}^{\uparrow}\}$ and $n=1$; in $\{0^{\downarrow}\}$ and $n=0$). $\square$

Completeness (Theorem 4.1) says that slices of a derivation in DiLL_0 (*i.e.* derivations in DiLL^{-}_0) with atomic axioms correspond to derivations in $\mathsf{DDI}^{\downarrow}_{-}\cup\{\mathsf{i}^{\uparrow}\}$ (and so in $\mathsf{SDDI}\setminus\{+^{\uparrow},0^{\uparrow},+^{\downarrow},0^{\downarrow}\}$, by rewriting $\mathsf{i}^{\uparrow}$ according to Lemma 3.4) with only MELL formulas, via the translation $[\![\cdot]\!]$ defined in Figure 6. Soundness (Theorem 4.2) says somehow that the converse holds too.

Theorem 4.2 (Soundness). *For any* MELL *sequent* Γ *and any* $n\in\mathbb{N}$, *if* $\mathsf{n}\overset{\mathsf{SDDI}}{\vdash}[\![\Gamma]\!]$ *then* $\overset{\mathsf{DiLL}_0}{\vdash}\Gamma$; *and more precisely, if* $1\overset{\mathsf{DDI}^{\downarrow}_{-}\cup\{\mathsf{i}^{\uparrow}\}}{\vdash}[\![\Gamma]\!]$, *then* $\overset{\mathsf{DiLL}^{-}_0}{\vdash}\Gamma$.

Proof. Clearly, for any MELL sequent Γ, there is a derivation in DiLL_0 sequent calculus: $\dfrac{}{\vdash\Gamma}\,\mathsf{zero}$.

Let us assume that we have a derivation $\mathcal{D}$ in $\mathsf{DDI}^{\downarrow}_{-}\cup\{\mathsf{i}^{\uparrow}\}$ from 1 to $[\![\Gamma]\!]$. To define the derivation of $\vdash\Gamma$ in DiLL^{-}_0, we consider the formulas occurring in $\mathcal{D}$ (which actually are MELL formulas) not up to

$\simeq$, so when $\simeq$ is required, its use is made explicit as if it were an inference rule (see also Footnote 3). For any $\rho \in \mathsf{DDI}^{\downarrow}_{\pm} \cup \{\mathsf{i}^{\uparrow}, \simeq\}$, if $\rho \dfrac{B}{A}$ then $\overset{\mathsf{DiLL}^{-}_0}{\vdash} \overline{B}, A$ as shown in Figure 7. Since A and B are MELL formulas, many equivalences in (2) cannot occur in $\mathcal{D}$. The cases for $\rho = \simeq$ corresponding to $A \otimes B \simeq B \otimes A$ and $A \otimes (B \otimes C) \simeq (A \otimes B) \otimes C$ and $A \otimes 1 \simeq A$ are omitted in Figure 7 as they are analogous to the ones for $\mathbin{⅋}$.

Consider $\mathcal{D}$ as sequenced (Remark 3.2). By induction on the MELL context $\mathsf{C}\{\}$, we prove that if $\rho \dfrac{\mathsf{C}\{B\}}{\mathsf{C}\{A\}}$ occurs in $\mathcal{D}$, then $\overset{\mathsf{DiLL}^{-}_0}{\vdash} \overline{\mathsf{C}\{B\}}, \mathsf{C}\{A\}$. We have just shown the case $\mathsf{C}\{\} = \{\}$. Other cases:

$\mathsf{C}\{A\} = D \mathbin{⅋} \mathsf{C}'\{A\}$ (or similarly $\mathsf{C}\{A\} = D \otimes \mathsf{C}'\{A\}$)

$$\cfrac{\cfrac{}{\vdash D, \bar{D}}\; ^{\mathsf{ax}} \qquad \cfrac{^{\mathsf{IH}\,\|\!|}}{\vdash \overline{\mathsf{C}'\{B\}}, \mathsf{C}'\{A\}}}{\cfrac{\vdash \bar{D} \otimes \overline{\mathsf{C}'\{B\}}, D, \mathsf{C}'\{A\}}{\vdash \bar{D} \otimes \overline{\mathsf{C}'\{B\}}, D \mathbin{⅋} \mathsf{C}'\{A\}}\; ^{\mathbin{⅋}}}\; ^{\otimes}$$

$\mathsf{C}\{A\} = {!}\,\mathsf{C}'\{A\}$ (or similarly $\mathsf{C}\{A\} = {?}\,\mathsf{C}'\{A\}$)

$$\cfrac{\cfrac{\cfrac{^{\mathsf{IH}\,\|\!|}}{\vdash \overline{\mathsf{C}'\{B\}}, \mathsf{C}'\{A\}}}{\vdash {?}\,\overline{\mathsf{C}'\{B\}}, \mathsf{C}'\{A\}}\; ^{?\mathsf{d}}}{\vdash {?}\,\overline{\mathsf{C}'\{B\}}, {!}\,\mathsf{C}'\{A\}}\; ^{!\mathsf{d}}$$

We define a derivation of $\vdash [\![\Gamma]\!]$ in DiLL^{-}_0 by induction on the number of rules in $\mathcal{D}$ as follows:

$$1 \to \cfrac{}{\vdash 1}\, 1 \qquad\qquad \rho \cfrac{\underset{\|\, \mathsf{DDI}^{\downarrow}_{\pm} \cup \{\mathsf{i}^{\uparrow}\}}{\overset{1}{\underset{[\![\Delta]\!]}{}}}}{[\![\Gamma]\!]} \;\to\; \cfrac{\overset{\mathsf{IH}\,\|\!|\,\mathsf{DiLL}^{-}_0}{\vdash [\![\Delta]\!]} \qquad \overset{\|\!|\,\mathsf{DiLL}^{-}_0}{\vdash \overline{[\![\Delta]\!]}, [\![\Gamma]\!]}}{\vdash [\![\Gamma]\!]}\; ^{\mathsf{cut}}$$

By reversibility of $\mathbin{⅋}$ (if $\overset{\mathsf{DiLL}^{-}_0}{\vdash} A \mathbin{⅋} B$ then $\overset{\mathsf{DiLL}^{-}_0}{\vdash} A, B$), we have $\overset{\mathsf{DiLL}^{-}_0}{\vdash} \Gamma$. $\square$

Let us sum up the correspondence between DiLL_0 sequent calculus and SDDI deep inference system.

Theorem 4.3 (Sequent calculus vs. deep inference). *Let Γ be a* MELL *sequent.*

1. DiLL_0 *vs.* SDDI: $\overset{\mathsf{DiLL}_0}{\vdash} \Gamma$ *if and only if* $1 \overset{\mathsf{SDDI}}{\vdash} [\![\Gamma]\!]$.

2. DiLL_0 *cut-free vs.* $\mathsf{DDI}^{\downarrow}$: $\overset{\mathsf{DiLL}_0 \setminus \{\mathsf{cut}\}}{\vdash} \Gamma$ *if and only if* $\mathsf{n} \overset{\mathsf{DDI}^{\downarrow}}{\vdash} [\![\Gamma]\!]$ *for some* $n \in \mathbb{N}$.

3. DiLL^{-}_0 *cut-free vs.* $\mathsf{DDI}^{\downarrow}_{\pm}$: *if* $\overset{\mathsf{DiLL}^{-}_0 \setminus \{\mathsf{cut}\}}{\vdash} \Gamma$ *then* $1 \overset{\mathsf{DDI}^{\downarrow}_{\pm}}{\vdash} [\![\Gamma]\!]$.

Proof.   1. For $\Rightarrow$, by completeness (Theorem 4.1); for $\Leftarrow$, by soundness (Theorem 4.2).

2.–3. For $\Rightarrow$, see Theorem 4.1. For $\Leftarrow$ (only for Item 2), by Theorems 2.4 and 4.3.1. $\square$

5   Normalization in SDDI

In this section we define a *standard* form for derivations in SDDI and a *normalization* procedure to obtain a "cut-free" standard derivation in $\mathsf{DDI}^{\downarrow}$ for any formula A provable in SDDI. The usual approach to prove normalization in deep inference system relies on the *splitting technique* [19, 23, 1, 22, 32]. However, the presence in our syntax of the connective $+$ and its unit 0 prevents us to use Guglielmi and Tubella's normalization result [36, 37], which covers and generalizes the splitting proofs. This is mainly due to the fact that 0 is an absorbing element for $\otimes$, $\mathbin{⅋}$, $?$ and $!$, together with the distributivity over $+$.

For this reason, following [32, 33], we define the normalization process in terms of *rule permutations*, which play the same role as cut-elimination steps in DiLL_0. In some cases, their definition relies on the rules for the connective $+$ and its unit 0. This behavior is coherent with the dynamics of cut-elimination in DiLL_0 [9, 28, 12], where the rules sum and zero step in to deal with non-deterministic choices or mismatches between "supply and demand" (see Section 2). Interestingly, these permutations in SDDI

mimic the elegant symmetries of cut-elimination steps as defined for interaction nets [28, Fig. 4], instead of the awkward rewrite rules defined for the sequent calculus (Figure 3).

The fact that the syntax for SDDI is more flexible and symmetric than the sequent calculus, and internalizes the connective $+$ and its unit 0, allows for a more fine-grained analysis of the normalization process than in DiLL_0. In particular, we can distinguish three kinds of rule permutations corresponding to three distinct phases in normalizing: MLL *cut-elimination steps* (involving $\mathsf{ai}^\uparrow$, $\mathsf{ai}^\downarrow$ and s only), *resource management steps* (involving the ?- and the !-rules only) and *slice operations* (the process of duplicating or removing a slice, which is less evident in the DiLL_0 sequent calculus and DiLL_0 interaction nets).

Definition 5.1 (Permutation)**.** In SDDI, a rule ρ *permutes over* a rule σ (or σ *permutes under* ρ) if, for

any derivation $\sigma \dfrac{C}{B}$ $\rho \dfrac{}{A}$, one of the following holds:

$$A = C\,; \qquad \sigma\dfrac{C}{A}\,; \qquad \rho\dfrac{C}{A}\,; \qquad \rho\dfrac{C}{B'}\ \text{for some formula } B'\,; \qquad 2\times\rho\dfrac{C}{B'}\ \sigma\dfrac{}{A}\ \text{for some formula } B'.$$

A rule ρ *permutes over* a rule σ (or σ *permutes under* ρ) *by* a set of rules $\mathcal{S}$ if ρ permutes over σ, or

for any derivation $\sigma \dfrac{C}{B}$ $\rho \dfrac{}{A}$, one of the following holds: $\begin{array}{c}C\\\|s\\A\end{array}$, or $\begin{array}{c}\rho\dfrac{C}{B''}\\\|s\\\sigma\dfrac{B'}{A}\end{array}$ for some formulas B',B'' .

Roughly, permuting σ under ρ means that σ can be pushed below ρ in a derivation with same premise and conclusion. In this operation, ρ or σ might disappear or other rules might appear in between. The definition of rule permutation is asymmetric: two ρ's can be above one σ, but not two σ's below one ρ.

We call *trivial* the rule permutations identified by the open deduction syntax, such as the one below.

$$\rho_1\dfrac{B_1\otimes B_2}{\rho_2\dfrac{A_1\otimes B_2}{A_1\otimes A_2}} = \boxed{\rho_1\dfrac{B_1}{A_1}}\otimes\boxed{\rho_2\dfrac{B_2}{A_2}} = \rho_2\dfrac{B_1\otimes B_2}{\rho_1\dfrac{B_1\otimes A_2}{A_1\otimes A_2}}$$

The following lemma is analogous to canonicity (Fact 2.2) for DiLL_0 sequent calculus. It means that, in SDDI, rules $0^\downarrow$ and $+^\downarrow$ can be pushed down in a derivation, and rules $0^\uparrow$ and $+^\uparrow$ can be pushed up.

Lemma 5.2 (Permuting 0 and +)**.** *Any rule in* SDDI *permutes over* $0^\downarrow$ *and* $+^\downarrow$, *and under* $+^\uparrow$ *and* $0^\uparrow$.

Proof. We define the rule permutations below, for $\rho, \tau, \sigma \in$ SDDI with $\rho \neq +^\uparrow$, $\tau \neq 0^\uparrow$ and $\sigma \notin \{+^\downarrow, 0^\downarrow\}$.

$$+^\downarrow\dfrac{B+B}{\boxed{\rho\dfrac{B}{A}}} \rightsquigarrow_{\mathsf{norm}} +^\downarrow\dfrac{\boxed{\rho\dfrac{B}{A}}+\boxed{\rho\dfrac{B}{A}}}{A} \qquad +^\uparrow\dfrac{+^\downarrow\dfrac{B+B}{B}}{B+B} \rightsquigarrow_{\mathsf{norm}} B+B \qquad \tau\dfrac{0^\downarrow\dfrac{0}{B}}{A} \rightsquigarrow_{\mathsf{norm}} 0^\downarrow\dfrac{0}{A}$$

$$0^\uparrow\dfrac{0^\downarrow\dfrac{0}{B}}{0} \rightsquigarrow_{\mathsf{norm}} 0 \qquad +^\uparrow\dfrac{\boxed{\sigma\dfrac{B}{A}}}{A+A} \rightsquigarrow_{\mathsf{norm}} +^\uparrow\dfrac{B}{\boxed{\sigma\dfrac{B}{A}}+\boxed{\sigma\dfrac{B}{A}}} \qquad 0^\uparrow\dfrac{\sigma\dfrac{B}{A}}{0} \rightsquigarrow_{\mathsf{norm}} 0^\uparrow\dfrac{B}{0}$$

(6)

Note that some of the rule permutations in (6) may implicitly use formula equivalence $\simeq$ in order to be applied, for example in the following rule permutations concerning $\mathsf{ai}^\downarrow$ and $+^\uparrow$, or $\mathsf{ai}^\downarrow$ and $0^\uparrow$:

$$
\mathsf{ai}^\downarrow \frac{1}{\boxed{+^\uparrow \dfrac{a}{a+a}}\,\mathfrak{V}\,\bar a}
=
\mathsf{ai}^\downarrow \frac{1}{\boxed{+^\uparrow \dfrac{a\,\mathfrak{V}\,\bar a}{(a+a)\,\mathfrak{V}\,\bar a}}}
\simeq
\mathsf{ai}^\downarrow \frac{1}{\boxed{+^\uparrow \dfrac{a\,\mathfrak{V}\,\bar a}{(a\,\mathfrak{V}\,\bar a)+(a\,\mathfrak{V}\,\bar a)}}}
\rightsquigarrow_{\mathsf{norm}}
+^\uparrow \frac{1}{\boxed{\mathsf{ai}^\downarrow \dfrac{1}{a\,\mathfrak{V}\,\bar a}}+\boxed{\mathsf{ai}^\downarrow \dfrac{1}{a\,\mathfrak{V}\,\bar a}}}
\tag{7}
$$

$$
\mathsf{ai}^\downarrow \frac{1}{\boxed{0^\uparrow \dfrac{a}{0}}\,\mathfrak{V}\,\bar a}
=
\mathsf{ai}^\downarrow \frac{1}{\boxed{0^\uparrow \dfrac{a\,\mathfrak{V}\,\bar a}{0\,\mathfrak{V}\,\bar a}}}
\simeq
\mathsf{ai}^\downarrow \frac{1}{\boxed{0^\uparrow \dfrac{a\,\mathfrak{V}\,\bar a}{0}}}
\rightsquigarrow_{\mathsf{norm}}
0^\uparrow \frac{1}{0}
$$

Similar permutations take place when $\mathsf{ai}^\downarrow$ is replaced with $!\mathsf{c}^\uparrow$ or $?\mathsf{c}^\uparrow$, and in their dual configurations. $\square$

As a consequence of Lemma 5.2, pushing up the rules $0^\uparrow$, $+^\uparrow$ and down $0^\downarrow$, $+^\downarrow$ can be interpreted as *slice management* operations: it duplicates and discards the "slices" (the subderivations without the rule $0^\uparrow$, $+^\uparrow$, $0^\downarrow$, $+^\downarrow$) and extends them as much as possible, propagating the non-deterministic choice $+$ and the resource mismatch 0 all along the derivation. It generalizes Remark 3.1 (for formulas) to derivations.

Corollary 5.3 (Slice management). *Let A, A', B, B' be MELL formulas, and $C\{\,\}$ be a MELL context. Let $\rightsquigarrow_{\mathsf{norm}}$ be one of the steps defined in (6), and $\rightsquigarrow_{\mathsf{norm}}^n$ be a sequence of $n \in \mathbb{N}$ of such steps. Then,*

where $m = |\mathcal{D}_1| + |\mathcal{D}_2|$ and $k = |\mathcal{D}_A| + |\mathcal{D}_B|$ ($|\mathcal{D}|$ is the number of inference rules in the derivation $\mathcal{D}$).

Actually, we can further structure SDDI derivations so as to separate an initial up-segment and a final down-segment (Theorem 5.7). To prove this, we use the two following lemmas.

Lemma 5.4 (Permutations of rules for $!$ and $?$). *In SDDI, the following rule permutations hold:*

1. Interaction-net permutations: *The rules in $\{!\mathsf{d}^\uparrow, ?\mathsf{d}^\uparrow, !\mathsf{w}^\uparrow, ?\mathsf{w}^\uparrow, !\mathsf{c}^\uparrow, ?\mathsf{c}^\uparrow\}$ permute over the rules in $\{!\mathsf{d}^\downarrow, ?\mathsf{d}^\downarrow, !\mathsf{w}^\downarrow, ?\mathsf{w}^\downarrow, !\mathsf{c}^\downarrow, ?\mathsf{c}^\downarrow\}$ by the rules in $\{+^\downarrow, +^\uparrow, 0^\downarrow, 0^\uparrow\}$;*

2. *The rules in $\{?\mathsf{d}^\downarrow, !\mathsf{d}^\downarrow, ?\mathsf{w}^\downarrow, !\mathsf{w}^\downarrow, ?\mathsf{c}^\downarrow, !\mathsf{c}^\downarrow\}$ permutes under any rule in $\{\mathsf{ai}^\uparrow, \mathsf{ai}^\downarrow, \mathsf{s}\}$;*

3. *The rules in $\{?\mathsf{d}^\uparrow, !\mathsf{d}^\uparrow, ?\mathsf{w}^\uparrow, !\mathsf{w}^\uparrow, ?\mathsf{c}^\uparrow, !\mathsf{c}^\uparrow\}$ permutes over any rule in $\{\mathsf{ai}^\uparrow, \mathsf{ai}^\downarrow, \mathsf{s}\}$.*

Proof.

1. By the (non-trivial) rule permutations in Figure 8 or by their duals obtained by up/down symmetry.

2. First, note that all rule permutations involving $\mathsf{ai}^\downarrow$ in $\{?\mathsf{d}^\downarrow, !\mathsf{d}^\downarrow, ?\mathsf{w}^\downarrow, !\mathsf{w}^\downarrow, ?\mathsf{c}^\downarrow, !\mathsf{c}^\downarrow\}$ are trivial. Moreover, any $\rho \in \{?\mathsf{d}^\downarrow, !\mathsf{d}^\downarrow, ?\mathsf{w}^\downarrow, !\mathsf{w}^\downarrow, ?\mathsf{c}^\downarrow, !\mathsf{c}^\downarrow\}$ permutes under s as follows.

$$
\mathsf{s}\frac{\boxed{\rho\dfrac{A'}{A}}\otimes(B\,\mathfrak{V}\,C)}{(A\otimes B)\,\mathfrak{V}\,C}
\rightsquigarrow_{\mathsf{norm}}
\mathsf{s}\frac{A'\otimes(B\,\mathfrak{V}\,C)}{\left(\boxed{\rho\dfrac{A'}{A}}\otimes B\right)\mathfrak{V}\,C}
\tag{8}
$$

$$?d^\downarrow \frac{\dfrac{A}{?A}}{A} \rightsquigarrow_{norm} A \qquad !d^\downarrow \frac{\dfrac{A}{!A}}{A} \rightsquigarrow_{norm} A \qquad ?w^\downarrow \frac{\dfrac{\bot}{?A}}{\bot} \rightsquigarrow_{norm} \bot \qquad !w^\downarrow \frac{\dfrac{1}{!A}}{1} \rightsquigarrow_{norm} 1 \qquad ?w^\downarrow \frac{\dfrac{\bot}{?A}}{A} \rightsquigarrow_{norm} 0^\uparrow \frac{\dfrac{\bot}{0}}{A} \qquad !w^\downarrow \frac{\dfrac{1}{!A}}{A} \rightsquigarrow_{norm} 0^\uparrow \frac{\dfrac{1}{0}}{A}$$

(Figure 8 derivations — complex deep-inference diagrams)

Figure 8: Non-trivial rule permutations for ! and ? by $\{0^\uparrow, 0^\downarrow, +^\uparrow, +^\downarrow\}$ in SDDI.

We conclude by the following rule permutations for $?d^\downarrow$, $?c^\downarrow$ and $?w^\downarrow$ (permutations for $!d^\downarrow$, $!c^\downarrow$ and $!w^\downarrow$ are defined similarly).

(permutation diagrams for $?d^\downarrow$, $?c^\downarrow$, $?w^\downarrow$)

3. It can be obtained dually from Item 2, by the up/down symmetry of rules. $\qquad\square$

Permutations in Figure 8 and their duals correspond to the cut-elimination steps for modalities ? and ! in the *interaction-nets* presentation of DiLL$_0$, see [28, Fig. 4] and [9, 34, 12]. They take place when a down-rule for ! meets an up-rule for ?, or vice-versa, and deal with the *resource management* (see Section 2). Akin to interaction-nets and unlike the sequent calculus, these permutations on SDDI are perfectly symmetric. Note the key role of the rules $+^\uparrow$, $+^\downarrow$, $0^\uparrow$, $0^\downarrow$ in some permutations. In particular,

- formulas 0 appear when there is a mismatch between "supply and demand" ($?w^\downarrow/!d^\uparrow$ and $!w^\downarrow/?d^\uparrow$),

- formulas with $+$ appear when there is a non-deterministic choice on which request will be fed ($?c^\downarrow/!d^\uparrow$ and $!c^\downarrow/?d^\uparrow$).

Lemma 5.5 (Linear permutations). *Let A and B be* MELL *formulas.*

1. *If* $B \overset{\{ai^\uparrow, ai^\downarrow, s\}}{\vdash\!\!-} A$ *then* $B \overset{\{ai^\downarrow\}}{\vdash\!\!-} B' \overset{\{s\}}{\vdash\!\!-} A' \overset{\{ai^\uparrow\}}{\vdash\!\!-} A$ *for some* MELL *formulas B' and A'.*

2. *If* $1 \overset{\{ai^\uparrow, ai^\downarrow, s\}}{\vdash\!\!-} A$ *then* $1 \overset{\{ai^\downarrow\}}{\vdash\!\!-} A' \overset{\{s\}}{\vdash\!\!-} A$ *for some* MELL *formula A'.*

Proof (sketch). This is a standard result in deep inference systems. Nowadays, it is usually proved via splitting [37, 36, 38], as it is a consequence of cut-elimination, but the hypotheses to apply the splitting technique do not hold in SDDI. In [18, 32, 33, 19], which were written before the splitting technique was found, Items 1 and 2 are proved using some sort of rule permutations. To prove Item 1 it is enough

to use the non-trivial rule permutations of $\mathsf{ai}^\downarrow$ over s shown in (9) below, and the dual rule permutations of $\mathsf{ai}^\uparrow$ under s obtained from (9) by up/down symmetry.

$$\mathsf{s}\frac{1\otimes(B\,\mathcal{B}\,C)}{\left(\boxed{\mathsf{ai}^\downarrow\dfrac{1}{a\,\mathcal{B}\,\bar a}}\otimes B\right)\mathcal{B}\,C}\ \rightsquigarrow_{\mathsf{norm}}\ \mathsf{s}\frac{\boxed{\mathsf{ai}^\downarrow\dfrac{1}{a\,\mathcal{B}\,\bar a}}\otimes(B\,\mathcal{B}\,C)}{((a\,\mathcal{B}\,\bar a)\otimes B)\,\mathcal{B}\,C}\qquad\mathsf{s}\frac{A\otimes(B\,\mathcal{B}\,1)}{(A\otimes B)\,\mathcal{B}\,\boxed{\mathsf{ai}^\downarrow\dfrac{1}{a\,\mathcal{B}\,\bar a}}}\ \rightsquigarrow_{\mathsf{norm}}\ \mathsf{s}\frac{A\otimes\left(B\,\mathcal{B}\,\boxed{\mathsf{ai}^\downarrow\dfrac{1}{a\,\mathcal{B}\,\bar a}}\right)}{(A\otimes B)\,\mathcal{B}\,1}\qquad(9)$$

To have an intuition for the proof of Item 2, it is enough to remark that if $1\overset{\{\mathsf{ai}^\downarrow\}}{\vdash\!\!-}A'$, then there is a derivation of A' with *shallow* $\mathsf{ai}^\downarrow$, that is, with $\mathsf{ai}^\downarrow$ applied only in $\otimes$-context as the one below on the left:

$$\boxed{\mathsf{ai}^\downarrow\dfrac{1}{a_1\,\mathcal{B}\,\bar a_1}}\otimes\cdots\otimes\boxed{\mathsf{ai}^\downarrow\dfrac{1}{a_n\,\mathcal{B}\,\bar a_n}}$$
$$\|\,\{\mathsf{s}\}$$
$$A'$$

$$\simeq\frac{1}{\boxed{\mathsf{ai}^\downarrow\dfrac{1}{a\,\mathcal{B}\,\bar a}}\otimes\boxed{\mathsf{ai}^\downarrow\dfrac{1}{a\,\mathcal{B}\,\bar a}}}$$
$$2\times\mathsf{s}\frac{}{a\,\mathcal{B}\,\boxed{\mathsf{ai}^\uparrow\dfrac{\bar a\,\mathcal{B}\,a}{\bot}}\,\mathcal{B}\,\bar a}\ \rightsquigarrow_{\mathsf{norm}}\ \mathsf{ai}^\downarrow\dfrac{1}{a\,\mathcal{B}\,\bar a}$$
$$\simeq\frac{}{a\,\mathcal{B}\,\bar a}$$

Hence by Item 1, if $1\overset{\{\mathsf{ai}^\uparrow,\mathsf{ai}^\downarrow,\mathsf{s}\}}{\vdash\!\!-}A$ then A is provable by starting from shallow $\mathsf{ai}^\downarrow$ rules; and if there is a rule $\mathsf{ai}^\uparrow$, then there is at least one $\mathsf{ai}^\uparrow$ that can be permuted up in the derivation, until we can obtain a configuration as the one above on the right, which can be replaced by a rule $\mathsf{ai}^\downarrow$.   □

Rule permutations involved in the proof of Lemma 5.5.2 essentially correspond to MLL *cut-elimination steps* in the DiLL_0 sequent calculus, indeed modalities ! and ? do not play any active role there.

Definition 5.6 (Normalization step)**.** Any rewrite relation on SDDI derivations that is a non-trivial rule permutation used in the proofs of Lemmas 5.2, 5.4 and 5.5 is a *normalization step* and denoted by $\rightsquigarrow_{\mathsf{norm}}$.

Normalization steps rearrange rules in a $\mathsf{DDI}^\downarrow$ or SDDI derivation in a fixed order, leaving unchanged its premise and conclusion. So, derivability in $\mathsf{DDI}^\downarrow$ and SDDI can be decomposed in several segments. More precisely, every derivation in SDDI can be rearranged in a symmetrical way so that:

1. on the top there is an *up-segment* where:
    (a) the first part consists of rules $0^\uparrow$ and $+^\uparrow$, which decompose the derivation in vertical slices;
    (b) the second part consists of up-rules for ! and ?, which deal with non-linear resources;

2. in the middle there is a *linear segment*, roughly corresponding to MLL and to linear resources;

3. on the bottom there is a *down-segment* where:
    (a) the first part consists of down-rules for ! and ?, which deal with non-linear resources;
    (b) the second part consists of rules $0^\downarrow$ and $+^\downarrow$, which merge the vertical slices of the derivation.

The decomposition in $\mathsf{DDI}^\downarrow$ follows the same pattern but takes only down-rules, so there is no up-segment.

Theorem 5.7 (Decomposition)**.** *Let A and B be formulas.*

1. $\mathsf{DDI}^\downarrow$*-decomposition: If* $\mathcal{D}\triangleright\mathsf{n}\overset{\mathsf{DDI}^\downarrow}{\vdash\!\!-}A$*, then (for some additive normal formulas A',A'',A''') there is a derivation $\mathcal{D}'$ in $\mathsf{DDI}^\downarrow$ such that $\mathcal{D}\rightsquigarrow^*_{\mathsf{norm}}\mathcal{D}'$ and*

$$\mathcal{D}'\triangleright\mathsf{n}\overset{\{\mathsf{ai}^\downarrow\}}{\vdash\!\!-}A'''\overset{\{\mathsf{s}\}}{\vdash\!\!-}A''\overset{\{?\mathsf{d}^\downarrow,!\mathsf{d}^\downarrow,?\mathsf{w}^\downarrow,!\mathsf{w}^\downarrow,?\mathsf{c}^\downarrow,!\mathsf{c}^\downarrow\}}{\vdash\!\!-}A'\overset{\{0^\downarrow,+^\downarrow\}}{\vdash\!\!-}A.$$

2. *SDDI-decomposition: If $\mathcal{D} \triangleright B \overset{\text{SDDI}}{\vdash\!\!\!-} A$, then there is a derivation $\mathcal{D}'$ (called* standard*) in* SDDI *from B to A such that $\mathcal{D} \rightsquigarrow^*_{\text{norm}} \mathcal{D}'$ and (for some additive normal formulas $B', B'', B''', A''', A'', A'$):*

$$\mathcal{D}' \triangleright B \overset{\{0^\uparrow,+^\uparrow\}}{\vdash\!\!\!-} B' \overset{\{?d^\uparrow,!d^\uparrow,!w^\uparrow,?w^\uparrow,?c^\uparrow,!c^\uparrow\}}{\vdash\!\!\!-} B'' \overset{\{ai^\downarrow\}}{\vdash\!\!\!-} B''' \overset{\{s\}}{\vdash\!\!\!-} A''' \overset{\{ai^\uparrow\}}{\vdash\!\!\!-} A'' \overset{\{?d^\downarrow,!d^\downarrow,?w^\downarrow,!w^\downarrow,?c^\downarrow,!c^\downarrow\}}{\vdash\!\!\!-} A' \overset{\{0^\downarrow,+^\downarrow\}}{\vdash\!\!\!-} A.$$

Proof. The decomposition of $DDI^\downarrow$ derivations follows from Lemma 5.2, Lemma 5.4.2 and Lemma 5.5.2.

To prove decomposition of SDDI derivations, we alternate applications of Lemmas 5.2 and 5.4 until we obtain a derivation of the shape below. Then we conclude by applying Lemma 5.5.1.

$$B \overset{\{0^\uparrow,+^\uparrow\}}{\vdash\!\!\!-} B' \overset{\{?d^\uparrow,!d^\uparrow,!w^\uparrow,?w^\uparrow,?c^\uparrow,!c^\uparrow\}}{\vdash\!\!\!-} B'' \overset{\{ai^\downarrow,s,ai^\uparrow\}}{\vdash\!\!\!-} A'' \overset{\{?d^\downarrow,!d^\downarrow,?w^\downarrow,!w^\downarrow,?c^\downarrow,!c^\downarrow\}}{\vdash\!\!\!-} A' \overset{\{0^\downarrow,+^\downarrow\}}{\vdash\!\!\!-} A. \qquad \square$$

As a consequence, the up-fragment $DDI^\uparrow$ of SDDI is *superfluous* (Corollary 5.8): all that can be proved in SDDI, is already derivable in the down-fragment $DDI^\downarrow$ of SDDI by a standard derivation. The existence of a standard derivation in $DDI^\downarrow$ is obvious because the rule $0^\downarrow$ makes every MELL formula derivable. The interesting part is that a standard derivation in $DDI^\downarrow$ can be reached via normalization steps, hence in a computational way that is *internal* to SDDI. Indeed, normalization of SDDI derivations follows from SDDI decomposition (Theorem 5.7.2), so it relies on the normalization steps defined on SDDI derivations (Definition 5.6). This normalization result is the deep inference version of *cut-elimination*, since in $DDI^\downarrow$ there is no analogue of the rule cut ($DDI^\downarrow$ is the "cut-free" fragment of SDDI).

Corollary 5.8 (Normalization). *Let A be a formula and $n \in \mathbb{N}$. If $\mathcal{D} \triangleright n \overset{\text{SDDI}}{\vdash\!\!\!-} A$ then, for some $n' \in \mathbb{N}$, there exists a standard $\mathcal{D}' \triangleright n \overset{\{0^\uparrow,+^\uparrow\}}{\vdash\!\!\!-} n' \overset{DDI^\downarrow}{\vdash\!\!\!-} A$ such that $\mathcal{D} \rightsquigarrow^*_{\text{norm}} \mathcal{D}'$. In particular, $n' \overset{DDI^\downarrow}{\vdash\!\!\!-} A$ for some $n' \in \mathbb{N}$.*

Proof. By Theorem 5.7.2, if $n \overset{\text{SDDI}}{\vdash\!\!\!-} A$ then there is a standard derivation

$$n \overset{\{0^\uparrow,+^\uparrow\}}{\vdash\!\!\!-} B \overset{\{?d^\uparrow,!d^\uparrow,!w^\uparrow,?w^\uparrow,?c^\uparrow,!c^\uparrow\}}{\vdash\!\!\!-} B' \overset{\{ai^\downarrow,s,ai^\uparrow\}}{\vdash\!\!\!-} A'' \overset{\{?d^\downarrow,!d^\downarrow,?w^\downarrow,!w^\downarrow,?c^\downarrow,!c^\downarrow\}}{\vdash\!\!\!-} A' \overset{\{0^\downarrow,+^\downarrow\}}{\vdash\!\!\!-} A.$$

Moreover, $n \overset{\{0^\uparrow,+^\uparrow\}}{\vdash\!\!\!-} B$ implies that $B = n'$ for some $n' \in \mathbb{N}$. As no rule in $\{?w^\uparrow,!w^\uparrow,?c^\uparrow,!c^\uparrow,?d^\uparrow,!d^\uparrow\}$ can be applied to a formula of the form n', we have $n' = B'$ and we conclude by Lemma 5.5.2. $\qquad \square$

6 Relation between cut-elimination in $DiLL_0$ and normalization in SDDI.

In this section we investigate the correspondence between the cut-elimination procedure in $DiLL_0$ sequent calculus (Section 2) and the normalization procedure in SDDI (Section 5).

We provided a translation $[\![\cdot]\!]$ of η-expanded $DiLL_0$ derivations to derivations in $DDI^\downarrow \cup \{i^\uparrow\}$ (Figure 6) and so in SDDI (via Lemma 3.4). The translation preserves "cut-freeness" (Theorem 4.1), and exhibits a one-to-one correspondence between weakening, contraction and dereliction rules of the two systems. However, the translation $[\![\cdot]\!]$ does not commute with cut-elimination/normalization: an η-expanded derivation π in $DiLL_0$ might reduce to a cut-free derivation $\widehat{\pi}$ via cut-elimination, but its translation $[\![\pi]\!]$ in SDDI (including the transformation of the rules $i^\uparrow$ into $ai^\uparrow$, as described in Lemma 3.4) normalizes to a $DDI^\downarrow$ derivation $\widehat{[\![\pi]\!]}$ other than $[\![\widehat{\pi}]\!]$. That is, diagram (10) does *not commute*:

$$
\begin{array}{ccc}
\pi & \xrightarrow{\;[\![\ ]\!]\;} & [\![\pi]\!] \\
{\scriptstyle\text{cut}}\downarrow{\scriptstyle *} & & \downarrow{\scriptstyle\text{norm}}{\scriptstyle *} \\
\widehat{\pi} & \xrightarrow[\;[\![\ ]\!]\;]{} & [\![\widehat{\pi}]\!] \neq \widehat{[\![\pi]\!]}
\end{array}
\qquad (10)
$$

Technically, the lack of commutation is because the rule cut is translated as an instance of $i^\uparrow$, which is not a rule of SDDI (it is not an atomic cut) and hence has to be rewritten according to Lemma 3.4. But this rewriting in SDDI might not match the resource distribution of the corresponding cut. Consider the derivation π below in DiLL_0 (with π' cut-free and η-expanded) and its translation $\mathcal{D}_\pi$ in SDDI:

According to cut-elimination for DiLL_0, $\pi \leadsto_{\mathsf{cut}} \pi'$ (one step). But $\mathcal{D}_\pi$ in SDDI normalizes as follows:

We observe that the transformation of the general $i^\uparrow$-rule into $\mathsf{ai}^\uparrow$ (Lemma 3.4) converts any potential interaction of weakening, contraction and dereliction up- and down-rules to an interaction of a (weakening, contraction or dereliction) down-rule with a dereliction up-rule: it arbitrarily chooses to asks for a resource, or to make it available, exactly once. In our example, the translation creates the "mismatches" $?\mathsf{w}^\downarrow/!\mathsf{d}^\uparrow$ and $!\mathsf{w}^\downarrow/?\mathsf{d}^\uparrow$ even if in the original DiLL_0 derivation we had a "matched" interaction of a $?\mathsf{w}$ with a $!\mathsf{w}$. Due to these mismatches, the normal form of the derivation $\mathcal{D}_\pi$ is $0^\downarrow \dfrac{0}{[\![\Gamma]\!]}$, which is not the translation of π' (the normal form of π with respect to cut-elimination in DiLL_0) if $\pi' \neq \overline{\vdash \Gamma}^{\,\mathsf{zero}}$. More generally, this problem is related to the fact that DiLL_0 misses the promotion rule $!\mathsf{p}$ (Figure 1), which would make a resource available at will [13] (see [28, 34, 12] for its cut-elimination in $\mathrm{DiLL} = \mathrm{DiLL}_0 \cup \{!\mathsf{p}\}$), while in the realm of DiLL_0 resources can be used only a finite number of times during cut-elimination.

In order to provide an *internal* solution (*i.e.*, in DiLL_0 without adding the rule $!\mathsf{p}$ to the system) to the commutation problem, we need to define a more sophisticated translation of cut-rules into SDDI.

A commutative translation.   We define a new translation $(\!|\cdot|\!)$ from DiLL_0 to SDDI so that diagram (10) commutes, when $[\![\cdot]\!]$ is replaced by $(\!|\cdot|\!)$. The idea is that the translation $(\!|\cdot|\!)$ "bends" a derivation π of

$\overset{\mathsf{DiLL_0}}{\vdash} \Gamma, A$ to a derivation $\overset{\overline{A}}{(\!|\pi|\!)} \| \mathsf{SDDI}$ so as to avoid using the rule $\mathsf{i}^{\uparrow}$. In this way, roughly, the translation $(\!|\cdot|\!)$
$\qquad\qquad\qquad\qquad [\![\Gamma]\!]$

converts the rule cut below (where π_1 and π_2 are η-expanded and cut-free $\mathsf{DiLL_0}$ derivations) as follows:

$$\cfrac{[\![\pi_1]\!] \quad [\![\pi_2]\!]}{\vdash \Gamma, A \quad \vdash \Delta, \overline{A}} \; \Longrightarrow \; \begin{array}{c} \mathsf{n_1} \\ [\![\pi_1]\!] \, \| \\ \boxed{\begin{array}{c} A \\ (\!|\pi_2|\!) \, \| \\ [\![\Delta]\!] \end{array}} \\ [\![\Gamma]\!] \, \mathfrak{N} \end{array} \qquad \text{where } [\![\pi_1]\!] \text{ is the translation of } \pi_1 \text{ defined in Figure 6.}$$

To define properly the translation $(\!|\pi|\!)$ of an η-expanded $\mathsf{DiLL_0}$ derivation, we first need to declare which formulas in π have to be "bent", selecting one of the two cut formulas for each cut in π. Formally, given an η-expanded $\mathsf{DiLL_0}$ derivation π, a *translation* $(\!|\pi|\!)$ is defined in two steps.

1. For each occurrence of the rule cut in π, we mark exactly one of its two cut formulas, say A, with $A^{\bullet}$. We propagate this mark bottom-up in π to the subformula occurrences of A in π: if the principal formula of a rule is marked, so are the active formulas in the premises (the other formulas in the sequent preserve their mark, if any). For instance, if the conclusion of a rule $\mathfrak{N}$ is $\vdash A^{\bullet}, (B \mathfrak{N} C)^{\bullet}, D$ with $B \mathfrak{N} C$ as principal formula, then its premise is $\vdash A^{\bullet}, B^{\bullet}, C^{\bullet}, D$.

2. We translate π decorated with marks $(\cdot)^{\bullet}$ into a derivation $(\!|\pi|\!)$ in SDDI according to the definition in Figure 9 (given by induction on π).

Note that an η-expanded derivation π in $\mathsf{DiLL_0}$ may have several translations $(\!|\pi|\!)$, depending on the initial selection of cut formulas in π to mark. We omit this dependency in the notation $(\!|\pi|\!)$. When we state a property of a translation $(\!|\pi|\!)$, we mean that it holds for *any* initial selection of cut formulas in π.

Lemma 6.1 (Target of the translation $(\!|\cdot|\!)$). *Let π be an η-expanded derivation in $\mathsf{DiLL_0}$. Then, $(\!|\pi|\!)$ is a derivation in SDDI. If, moreover, π is cut-free, then $(\!|\pi|\!) = \mathcal{D} \circ [\![\pi]\!]$ where $\mathcal{D} \triangleright 1 \overset{\{0^{\uparrow},+^{\uparrow}\}}{\vdash} \mathsf{n}$ for some $n \in \mathbb{N}$, and $[\![\pi]\!]$ (defined in Figure 6) is a derivation in $\mathsf{DDI}^{\downarrow}$.*

Proof. By induction on π. Each step in Figure 9 introduces only rules in SDDI, in particular no step introduces the rule $\mathsf{i}^{\uparrow}$. If π is cut-free, then no formula in π is marked; thus, each step in Figure 9 acts like $[\![\cdot]\!]$ defined in Figure 6 except possibly for adding some rules $+^{\uparrow}$ and $0^{\uparrow}$ on top, and it does not introduce any other rule in $\mathsf{DDI}^{\uparrow}$ except s (which is also in $\mathsf{DDI}^{\downarrow}$). $\qquad\square$

There is trivial way to "bend" a derivation π of $\overset{\mathsf{DiLL_0}}{\vdash} \Gamma, A$ to a derivation in SDDI: take $\overset{\overline{A}}{\underset{0^{\downarrow}}{0^{\uparrow}\cfrac{\;0\;}{[\![\Gamma]\!]}}}$. However,

such a translation does not make diagram (10) commute, because it does not keep track of resources. The translation $(\!|\pi|\!)$, instead, is *resource-sensitive*, thanks to a one-to-one correspondence between the occurrences of rules for weakening, dereliction, and contraction in π and in $(\!|\pi|\!)$ (proved by induction on π).

Lemma 6.2 (Resources). *For any η-expanded derivation π in $\mathsf{DiLL_0}$, there is a one-to-one correspondence between the rule occurrences of $\mathsf{!d}$ in π and the rule occurrences of $\{\mathsf{!d}^{\downarrow}, \mathsf{!d}^{\uparrow}\}$ in $(\!|\pi|\!)$, and similarly between $\mathsf{?d}$ and $\{\mathsf{?d}^{\downarrow}, \mathsf{?d}^{\uparrow}\}$, $\mathsf{!c}$ and $\{\mathsf{!c}^{\downarrow}, \mathsf{!c}^{\uparrow}\}$, $\mathsf{?c}$ and $\{\mathsf{?c}^{\downarrow}, \mathsf{?c}^{\uparrow}\}$, $\mathsf{!w}$ and $\{\mathsf{!w}^{\downarrow}, \mathsf{!w}^{\uparrow}\}$, $\mathsf{?w}$ and $\{\mathsf{?w}^{\downarrow}, \mathsf{?w}^{\uparrow}\}$.*

Let $\simeq_{\mathsf{norm}}$ be the reflexive, transitive and symmetric closure of $\rightsquigarrow_{\mathsf{norm}}$.

Theorem 6.3 (Simulation). *If π is an η-expanded $\mathsf{DiLL_0}$ derivation and $\pi \rightsquigarrow_{\mathsf{cut}} \pi'$, then $(\!|\pi|\!) \simeq_{\mathsf{norm}} (\!|\pi'|\!)$.*

Proof (sketch). By case analysis of cut-elimination steps for $\mathsf{DiLL_0}$, and the corresponding normalization steps in SDDI. By Lemma 6.2, cut-elimination steps for the rules in $\{\mathsf{!d}, \mathsf{?d}, \mathsf{!c}, \mathsf{?c}, \mathsf{!w}, \mathsf{?w}\}$ correspond to the (non-trivial) interaction-net rule permutations in Figure 8 (Lemma 5.4.1). A cut-elimination step $\otimes/\mathfrak{N}$ corresponds to linear permutations to prove Lemma 5.5. Commutative cut-elimination steps correspond to trivial rule permutations. All permutations are interleaved by steps (6) and (8), in both directions. $\quad\square$

Figure 9: Translation of η-expanded DiLL_0 sequent calculus derivations into $\mathsf{SDDI} \setminus \{\mathsf{ai}^\uparrow\}$ derivations (where if $\Gamma = A_1, \ldots, A_n$, then $\Gamma^\bullet = A_1^\bullet, \ldots, A_n^\bullet$).

Theorem 6.3 says that normalization steps in SDDI (Definition 5.6) mimic DiLL_0 cut-elimination via translation $(\!|\cdot|\!)$. As a consequence, cut-elimination/normalization commutes with translation $(\!|\cdot|\!)$.

Corollary 6.4 (Commutation). *If π is an η-expanded DiLL_0 derivation and $\pi \rightsquigarrow^*_{\mathsf{cut}} \hat{\pi}$ with $\hat{\pi}$ cut-free, then $(\!|\pi|\!) \rightsquigarrow^*_{\mathsf{norm}} (\!|\hat{\pi}|\!)$ and $(\!|\hat{\pi}|\!)$ is normal for $\rightsquigarrow_{\mathsf{norm}}$.*

Proof. By simulation (Theorem 6.3), from $\pi \rightsquigarrow^*_{\mathsf{cut}} \hat{\pi}$ it follows that $(\!|\pi|\!) \simeq_{\mathsf{norm}} (\!|\hat{\pi}|\!)$. As $\hat{\pi}$ is cut-free, $(\!|\hat{\pi}|\!)$ is normal for $\rightsquigarrow_{\mathsf{norm}}$ according to Lemma 6.1, hence $(\!|\pi|\!) \rightsquigarrow^*_{\mathsf{norm}} (\!|\hat{\pi}|\!)$. □

7 Conclusions and future work

In this paper we introduced the first sound and complete deep inference system, SDDI, for the promotion-free fragment of differential linear logic, DiLL_0 [9]. The deep inference syntax recovers the symmetry of this logic lacking in DiLL_0 sequent calculus—but which can be found in the interaction-net formalism for DiLL_0 [9]—and keeps the inductive and handy tree-like structure of sequent calculus derivations—missing in interaction nets. The deep inference formalism allows us to reduce cuts to atomic formulas, and provides a tool for a more fine-grained study of cut-elimination. Moreover, the syntax explicitly represents and internalizes the notion of slices of a derivation.

The inference rules of SDDI present an up/down symmetry and we proved that the up-fragment of SDDI is derivable from the down-fragment $\mathsf{DDI}^{\downarrow}$. To prove this result we provided a normalization procedure based on rule permutations. In fact, the presence of the connective $+$ and its unit 0 prevent the use of the general normalization result for splittable systems [37]. In our normalization procedure for SDDI, we are able to distinguish different kinds of rule permutations depending on their computational behavior: some rule permutations correspond to linear (in terms of resource) cut-elimination steps, some to resource management cut-elimination steps and some to slice management operations. Thanks to Corollary 5.3 we could implement a reduction strategy alternating slice management and proper cut-elimination steps inside each slice. The internal normalization procedure in SDDI to prove Corollary 5.8 provides "cut-free" derivations. And the translation $[\![\cdot]\!]$ defined in Figure 6 maps cut-free DiLL_0 derivations to $\mathsf{DDI}^{\downarrow}$, the "cut-free" fragment of SDDI (Theorem 4.3.2). We showed that cut-elimination/normalization does not commute with translation $[\![\cdot]\!]$, but it does with the translation $(\!|\cdot|\!)$ defined in Figure 9, a resource-sensitive refinement of $[\![\cdot]\!]$.

Translation of DiLL **proof-nets.**   Another ongoing work is to extend our deep inference system in order to represent the *full* differential linear logic $\mathsf{DiLL} = \mathsf{DiLL}_0 \cup \{!\mathsf{p}\}$ [28, 34, 30] (including the promotion rule), possibly with the rule mix which allows one to derive $A \,\invamp\, B$ from $A \otimes B$. The presence of promotion $!\mathsf{p}$ allows us to define a translation that commutes with cut-elimination for the reasons discussed above.

In this extended deep inference system, we can translate not only the DiLL sequent calculus but also DiLL proof-nets, via a direct embedding that does not pass through the sequent calculus. Indeed, the open deduction formalism [20] allows a direct encoding of proof-nets, plus a handy and inductive syntax.

Computational meaning and non-determinism.   In DiLL_0 interaction nets, when a cut-elimination step creates a new construct sum (expressing a non-deterministic choice) or zero (expressing mismatch on demanded and supplied resources), this construct is instantaneously propagated to all the interaction net where it is plugged in, without any computational step. It is like DiLL_0 interaction nets allow one to deal with canonical forms only, in the sense of Definition 2.1.

A feature of our deep inference formalism is that the constructs $+$ (non-determinism) and 0 (resource mismatch) are internalized in the syntax, and when they appear during the normalization process, they are propagated all along the derivation by means of normalization steps (slice management, Lemma 5.2 and Corollary 5.3). Is there a computational meaning in these kind of steps? Is it possible to interpret them in a model of computation which intrinsically represents non-determinism, parallelism and concurrency?

The π-calculus [27] (a model of concurrent computation) can be encoded in DiLL$_0$ [6], but Mazza [25] pointed out that the non-determinism expressed by DiLL$_0$ is too weak for true concurrency. Deep inference may shed new light on the quest for a convincing proof-theoretic counterpart of concurrency.

References

[1] Matteo Acclavio, Ross Horne & Lutz Straßburger (2020): *Logic beyond formulas: a proof system on graphs*. In: *35th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS'20*, ACM, pp. 38–52, doi:10.1145/3373718.3394763.

[2] Kai Brünnler & Alwen Fernanto Tiu (2001): *A Local System for Classical Logic*. In: *Logic for Programming, Artificial Intelligence, and Reasoning , 8th International Conference, LPAR 2001*, Lecture Notes in Computer Science 2250, Springer, pp. 347–361, doi:10.1007/3-540-45653-8_24.

[3] Daniel de Carvalho (2018): *Taylor expansion in linear logic is invertible*. Log. Methods Comput. Sci. 14(4), doi:10.23638/LMCS-14(4:21)2018.

[4] Thomas Ehrhard (2005): *Finiteness spaces*. Mathematical Structures in Computer Science 15(4), pp. 615–646, doi:10.1017/S0960129504004645.

[5] Thomas Ehrhard (2018): *An introduction to differential linear logic: proof-nets, models and antiderivatives*. Mathematical Structures in Computer Science 28(7), pp. 995–1060, doi:10.1017/S0960129516000372.

[6] Thomas Ehrhard & Olivier Laurent (2010): *Interpreting a finitary pi-calculus in differential interaction nets*. Inf. Comput. 208(6), pp. 606–633, doi:10.1016/j.ic.2009.06.005.

[7] Thomas Ehrhard & Laurent Regnier (2003): *The differential lambda-calculus*. Theor. Comput. Sci. 309(1-3), pp. 1–41, doi:10.1016/S0304-3975(03)00392-X.

[8] Thomas Ehrhard & Laurent Regnier (2006): *Böhm Trees, Krivine's Machine and the Taylor Expansion of Lambda-Terms*. In: *Second Conference on Computability in Europe, CiE 2006*, Lecture Notes in Computer Science 3988, Springer, pp. 186–197, doi:10.1007/11780342_20.

[9] Thomas Ehrhard & Laurent Regnier (2006): *Differential interaction nets*. Theor. Comput. Sci. 364(2), pp. 166–195, doi:10.1016/j.tcs.2006.08.003.

[10] Thomas Ehrhard & Laurent Regnier (2008): *Uniformity and the Taylor expansion of ordinary lambda-terms*. Theor. Comput. Sci. 403(2-3), pp. 347–372, doi:10.1016/j.tcs.2008.06.001.

[11] Stéphane Gimenez (2009): *Programming, Computation and their Analysis using Nets from Linear Logic*. Theses, Université Paris Diderot. Available at https://tel.archives-ouvertes.fr/tel-00629013.

[12] Stéphane Gimenez (2011): *Realizability Proof for Normalization of Full Differential Linear Logic*. In: *Typed Lambda Calculi and Applications - 10th International Conference, TLCA 2011*, Lecture Notes in Computer Science 6690, Springer, pp. 107–122, doi:10.1007/978-3-642-21691-6_11.

[13] Jean-Yves Girard (1987): *Linear Logic*. Theor. Comput. Sci. 50, pp. 1–102, doi:10.1016/0304-3975(87)90045-4.

[14] Jean-Yves Girard (2001): *Locus Solum: From the rules of logic to the logic of rules*. Mathematical Structures in Computer Science 11(3), pp. 301–506, doi:10.1017/S096012950100336X.

[15] Giulio Guerrieri, Luc Pellissier & Lorenzo Tortora de Falco (2016): *Computing Connected Proof(-Structure)s From Their Taylor Expansion*. In: *1st International Conference on Formal Structures for Computation and Deduction, FSCD 2016*, LIPIcs 52, Schloss Dagstuhl, pp. 20:1–20:18, doi:10.4230/LIPIcs.FSCD.2016.20.

[16] Giulio Guerrieri, Luc Pellissier & Lorenzo Tortora de Falco (2019): *Proof-Net as Graph, Taylor Expansion as Pullback*. In: *Logic, Language, Information, and Computation - 26th International Workshop, WoLLIC 2019*, Lecture Notes in Computer Science 11541, Springer, pp. 282–300, doi:10.1007/978-3-662-59533-6_18.

[17] Giulio Guerrieri, Luc Pellissier & Lorenzo Tortora de Falco (2020): *Glueability of Resource Proof-Structures: Inverting the Taylor Expansion*. In: *28th EACSL Annual Conference on Computer Science Logic, CSL 2020*, LIPIcs 152, Schloss Dagstuhl, pp. 24:1–24:18, doi:10.4230/LIPIcs.CSL.2020.24.

[18] Alessio Guglielmi (1999): *A calculus of order and interaction*. Technical Report WV-99-04, Dresden University of Technology.

[19] Alessio Guglielmi (2007): *A System of Interaction and Structure*. ACM Trans. Comput. Log. 8(1), pp. 1–64, doi:10.1145/1182613.1182614.

[20] Alessio Guglielmi, Tom Gundersen & Michel Parigot (2010): *A Proof Calculus Which Reduces Syntactic Bureaucracy*. In: *Proceedings of the 21st International Conference on Rewriting Techniques and Applications, RTA 2010*, LIPIcs 6, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, pp. 135–150, doi:10.4230/LIPIcs.RTA.2010.135.

[21] Alessio Guglielmi & Lutz Straßburger (2001): *Non-commutativity and MELL in the Calculus of Structures*. In Laurent Fribourg, editor: *Computer Science Logic, 15th International Workshop, CSL 2001*, Lecture Notes in Computer Science 2142, Springer, pp. 54–68, doi:10.1007/3-540-44802-0_5.

[22] Alessio Guglielmi & Lutz Straßburger (2002): *A Non-commutative Extension of MELL*. In: *Logic for Programming, Artificial Intelligence, and Reasoning, 9th International Conference, LPAR 2002*, Lecture Notes in Computer Science 2514, Springer, pp. 231–246, doi:10.1007/3-540-36078-6_16.

[23] Ross Horne, Alwen Tiu, Bogdan Aman & Gabriel Ciobanu (2019): *De Morgan Dual Nominal Quantifiers Modelling Private Names in Non-Commutative Logic*. ACM Transactions on Computational Logic (TOCL) 20(4), pp. 1–44.

[24] Yves Lafont (1990): *Interaction Nets*. In: *Seventeenth Annual ACM Symposium on Principles of Programming Languages, POPL 1990*, ACM Press, pp. 95–108, doi:10.1145/96709.96718.

[25] Damiano Mazza (2018): *The true concurrency of differential interaction nets*. Math. Struct. Comput. Sci. 28(7), pp. 1097–1125, doi:10.1017/S0960129516000402.

[26] Damiano Mazza & Michele Pagani (2007): *The Separation Theorem for Differential Interaction Nets*. In: *Logic for Programming, Artificial Intelligence, and Reasoning, 14th International Conference, LPAR 2007*, Lecture Notes in Computer Science 4790, Springer, pp. 393–407, doi:10.1007/978-3-540-75560-9_29.

[27] Robin Milner (1999): *Communicating and mobile systems: the pi calculus*. Cambridge university press.

[28] Michele Pagani (2009): *The Cut-Elimination Theorem for Differential Nets with Promotion*. In: *Typed Lambda Calculi and Applications, 9th International Conference, TLCA 2009*, Lecture Notes in Computer Science 5608, Springer, pp. 219–233, doi:10.1007/978-3-642-02273-9_17.

[29] Michele Pagani & Christine Tasson (2009): *The Inverse Taylor Expansion Problem in Linear Logic*. In: *Proceedings of the 24th Annual IEEE Symposium on Logic in Computer Science, LICS 2009, 11-14 August 2009, Los Angeles, CA, USA*, IEEE Computer Society, pp. 222–231, doi:10.1109/LICS.2009.35.

[30] Michele Pagani & Paolo Tranquilli (2017): *The conservation theorem for differential nets*. Mathematical Structures in Computer Science 27(6), pp. 939–992, doi:10.1017/S0960129515000456.

[31] Benjamin Ralph (2019): *Modular Normalisation of Classical Proofs*. Ph.D. thesis, University of Bath.

[32] Lutz Straßburger (2003): *Linear Logic and Noncommutativity in the Calculus of Structures*. Ph.D. thesis, Technische Universität Dresden.

[33] Lutz Straßburger (2003): *MELL in the calculus of structures*. Theoretical Computer Science 309(1), pp. 213 – 285, doi:https://doi.org/10.1016/S0304-3975(03)00240-8.

[34] Paolo Tranquilli (2009): *Confluence of Pure Differential Nets with Promotion*. In: *Computer Science Logic, 23rd international Workshop, CSL 2009, 18th Annual Conference of the EACSLS, Lecture Notes in Computer Science* 5771, Springer, pp. 500–514, doi:10.1007/978-3-642-04027-6_36.

[35] Paolo Tranquilli (2011): *Intuitionistic differential nets and lambda-calculus*. Theor. Comput. Sci. 412(20), pp. 1979–1997, doi:10.1016/j.tcs.2010.12.022.

[36] Andrea Aler Tubella (2017): *A study of normalisation through subatomic logic*. Ph.D. thesis, University of Bath.

[37] Andrea Aler Tubella & Alessio Guglielmi (2018): *Subatomic Proof Systems: Splittable Systems*. ACM Trans. Comput. Log. 19(1), pp. 5:1–5:33, doi:10.1145/3173544.

[38] Andrea Aler Tubella & Lutz Straßburger (2019): *Introduction to Deep Inference*. Available at `https://hal.inria.fr/hal-02390267`. Lecture.