
An Extended Framework for Specifying and Reasoning about Proof Systems

Vivek Nigam, *Computer Science Department, Universidade Federal da Paraíba, Brazil. E-mail: vivek.nigam@gmail.com*

Elaine Pimentel, *Departamento de Matemática, Universidade Federal do Rio Grande do Norte, Brazil. Departamento de Matemática, Universidade Federal de Minas Gerais, Brazil. E-mail: elaine.pimentel@gmail.com*

Giselle Reis, *Institute of Computer Languages, TUWien, Austria. E-mail: giselle@logic.at*

Abstract

It has been shown that linear logic can be successfully used as a framework for both specifying proof systems for a number of logics, as well as proving fundamental properties about the specified systems. This paper shows how to extend the framework with subexponentials in order to declaratively encode a wider range of proof systems, including a number of non-trivial proof systems such as multi-conclusion intuitionistic logic, classical modal logic S4, intuitionistic Lax logic, and Negri's labelled proof systems for different modal logics. Moreover, we propose methods for checking whether an encoded proof system has important properties, such as if it admits cut-elimination, the completeness of atomic identity rules, and the invertibility of its inference rules. Finally, we present a tool implementing some of these specification/verification methods.

1 Introduction

Designing suitable proof systems for specific applications has become one of the main tasks of many applied logicians working in computer science. Proof theory has been applied in different fields including programming languages, knowledge representation, automated reasoning, access control, among many others. Such proof systems should also have *good properties*, e.g. the admissibility of the cut-rule (which leads to other important properties such as the sub-formula property and the consistency of the system) as well as the completeness of atomic identity rules and the invertibility of inference rules. It is therefore of interest to develop techniques and *automated* tools that can help logicians (and possibly non-logicians) in specifying and reasoning about proof systems.

In recent years, a series of papers [19, 18, 25, 31] have shown that linear logic [13] can be used as a framework for *specifying* and *reasoning* about proof systems. In particular, [31, 25] showed how to specify not only sequent calculus systems, but also natural deduction systems for different logics, such as minimal, intuitionistic and classical logics. Moreover, [19, 18] showed how to check whether an encoded proof system enjoys important properties by simply analyzing its linear logic specification. This existing material provides sufficient conditions for guaranteeing cut-elimination of the specified systems.

In our previous work [26], we proposed using linear logic with *subexponentials* as a framework for specifying proof systems. The motivation for this step comes from the fact that,

2 An Extended Framework for Specifying and Reasoning about Proof Systems

since exponentials in linear logic are not canonical [23, 7], one can construct linear logic proof systems containing as many subexponentials as one needs. This feature made it possible to declaratively encode a wide range of proof systems, such as a multi-conclusion proof system for intuitionistic logic [16]. In addition, since the proposed encoding is natural and direct, we were able to use the rich linear logic meta-level theory in order to reason about the specified systems in an elegant and simple way.

The contribution of this paper is three-fold. First, in Section 4, we demonstrate how to declaratively specify proof systems with more involved structural and logical inferences rules using linear logic theories with subexponentials. We encode proof systems that have structural restrictions that are much more interesting and challenging than those of the systems specified in [26]. Besides the multi-conclusion system for intuitionistic logic specified in our previous work, we specify proof systems for intuitionistic lax logic [10], focused intuitionistic logic LJQ^* , classical modal logic S4 as well as Negri's labelled proof systems for different modal logics. These examples provide evidence that linear logic with subexponentials can be successfully used as a framework for a number of proof systems including systems for modal logics.

Our second contribution, in Section 5, follows and enhances the ideas presented in [19]. We provide sufficient conditions for guaranteeing three properties of systems specified using subexponentials: (1) the admissibility of the cut-rule; (2) the completeness of the system when using only atomic instances of the initial rule; and (3) the invertibility of each inference rule. The main difference from what is presented here and the work developed in [19] is the establishment of some criteria for *permutation of rules*. Such analysis is needed for checking whether cuts can be transformed into *principal cuts*. Since our framework enables the encoding of much more complicated proof systems, the behavioral analysis is more involved and it leads to more general conditions when compared to [19].

Finally, we have implemented a tool, described in Section 6, that accepts a linear logic with subexponentials specification and automatically checks whether principals cuts can be reduced to atomic cuts and whether initial rules can be atomic only. Our tool is able to show that most of the systems mentioned¹ above satisfy these conditions. Furthermore it also can check cases for when the cut-rule can be permuted over an introduction rule and when an introduction rule can permute over another introduction rule. Such analysis can greatly help to discover corner cases for when the reduction of a proof with cuts into a proof with principal cuts only is not immediate.

This paper is structured as follows. Section 2 introduces the proof system for linear logic with subexponentials, called *SELLF*, which is the basis of the proposed logical framework. In Section 3, we describe how to encode a proof system in our framework. Section 4 describes the encoding of a number of proof systems, namely, the proof system *G1m* for minimal logic [33], the multi-conclusion proof system for intuitionistic logic *mLJ* [16], the focused proof system *LJQ** for intuitionistic logic [8], a proof system for the classical modal logic S4, a proof system for intuitionistic lax logic [10], and the labelled proof system G3K for modal logics [21]. Section 5 introduces the conditions for verifying whether an encoded proof system satisfies the properties mentioned before, which can be checked using our tool described in Section 6. Finally, in Sections 7 and 8, we end by discussing related and future work.

This is an improved and expanded version of the workshop paper [26].

¹The exception is LJQ^* , whose cut-elimination argument is quite involved.

2 Linear Logic with Subexponentials

Although we assume that the reader is familiar with linear logic, we review some of its basic proof theory. *Literals* are either atomic formulas (A) or their negations (A^\perp). The connectives \otimes and \wp and their units 1 and \perp are *multiplicative*; the connectives \oplus and $\&$ and their units 0 and \top are *additive*; \forall and \exists are (first-order) quantifiers; and $!$ and $?$ are the exponentials (called bang and question mark, respectively). We shall assume that all formulas are in *negation normal form*, meaning that all negations have atomic scope.

Due to the exponentials, one can distinguish in linear logic two kinds of formulas: the linear ones whose main connective is not a $?$ and the unbounded ones whose main connective is a $?$. The linear formulas can be seen as resources that can only be used once, while the unbounded formulas represent unlimited resources that can be used as many times as necessary. This distinction is usually reflected in syntax by using two different contexts in linear logic sequents ($\vdash \Theta : \Gamma$), one (Θ) containing only unbounded formulas and another (Γ) with only linear formulas [1]. Such distinction allows to incorporate structural rules, *i.e.*, weakening and contraction, into the introduction rules of connectives, as done in similar presentations for classical logic, *e.g.*, the $G3c$ system in [33]. In such presentation, the context (Θ) containing unbounded formulas is treated as a set of formulas, while the other context (Γ) containing only linear formulas is treated as a multiset of formulas.

It turns out that the exponentials are not canonical [7] with respect to the logical equivalence relation. In fact, if, for any reason, we decide to define a blue and red conjunctions (\wedge^b and \wedge^r respectively) with the standard classical rules:

$$\frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge^b B \vdash \Delta} [\wedge^b L] \qquad \frac{\Gamma \vdash \Delta, A \quad \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \wedge^b B} [\wedge^b R]$$

$$\frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge^r B \vdash \Delta} [\wedge^r L] \qquad \frac{\Gamma \vdash \Delta, A \quad \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \wedge^r B} [\wedge^r R]$$

then it is easy to show that, for any formulas A and B , $A \wedge^b B \equiv A \wedge^r B$. This means that all the symbols for classical conjunction belong to the same equivalence class. Hence, we can choose to use as the conjunction's *canonical* form any particular color, and provability is not affected by this choice. However, the same behavior does not hold with the linear logic exponentials. In fact, suppose we have red $!^r, ?^r$ and blue $!^b, ?^b$ sets of exponentials with the standard linear logic rules:

$$\frac{\vdash ?^r \Gamma, F}{\vdash ?^r \Gamma, !^r F} [!^r] \qquad \frac{\vdash \Gamma, F}{\vdash \Gamma, ?^r F} [D?^r] \qquad \frac{\vdash ?^b \Gamma, F}{\vdash ?^b \Gamma, !^b F} [!^b] \qquad \frac{\vdash \Gamma, F}{\vdash \Gamma, ?^b F} [D?^b]$$

We cannot show that $!^r F \equiv !^b F$ nor $?^r F \equiv ?^b F$. This opens the possibility of defining classes of exponentials, called *subexponentials* [24]. In this way, it is possible to build proof systems containing as many exponential-like operators, ($!^l, ?^l$) as one needs: they may or may not allow contraction and weakening, and are organized in a pre-order (\leq) specifying the entailment relation between these operators.

Formally, a proof system for linear logic with subexponentials, called $SELL_\Sigma$, is specified by a subexponential signature.

DEFINITION 2.1

A subexponential signature is a tuple $\Sigma = \langle I, \leq, \mathcal{U} \rangle$, where I is the set of labels for subexponentials, $\mathcal{U} \subseteq I$, specifying which subexponentials allow for weakening and contraction and

4 An Extended Framework for Specifying and Reasoning about Proof Systems

\leq is a preorder relation² among the elements of I upwardly closed with respect to the set \mathcal{U} , that is, if $x < y$ and $x \in \mathcal{U}$, then $y \in \mathcal{U}$.³

For a given a subexponential signature $\Sigma = \langle I, \leq, \mathcal{U} \rangle$, the proof system $SELL_\Sigma$ contains the same introduction rules as in linear logic for all connectives, except the exponentials. These are specified by the subexponential signature, Σ , as follows:⁴

$$\frac{\vdash C, \Delta}{\vdash ?^x C, \Delta} [D, \text{if } x \in I] \quad \frac{\vdash ?^y C, ?^y C, \Delta}{\vdash ?^y C, \Delta} [C, \text{if } y \in \mathcal{U}] \quad \frac{\vdash \Delta}{\vdash ?^z C, \Delta} [W, \text{if } z \in \mathcal{U}]$$

The first rule, called dereliction, can be applied to any subexponential, while contraction and weakening only to subexponentials that appear in the set \mathcal{U} . The promotion rule is given by the following inference rule:

$$\frac{\vdash ?^{x_1} C_1, \dots, ?^{x_n} C_n, C}{\vdash ?^{x_1} C_1, \dots, ?^{x_n} C_n, !^a C} [!^a]$$

where $a \leq x_i$ for all $i = 1, \dots, n$. The promotion rule will play an important role here, namely, to specify the structural restrictions of encoded proof systems. In particular, one can use a subexponential bang, $!^c$, to check whether there are only some type of formulas in the context, namely, those that are marked with subexponentials, $?^x$, such that $c \leq x$. If there is any formula $?^y F$ in the context such that $c \not\leq y$, then $!^c$ cannot be introduced.

We classify all the subexponential indices belonging to \mathcal{U} as *unrestricted* or *unbounded*, and the remaining indices as *restricted* or *bounded*.

Danos *et al.* showed that $SELL$ admits cut-elimination [7].

THEOREM 2.2

For any signature Σ , the cut-rule is admissible in $SELL_\Sigma$.

2.1 Focusing

First proposed by Andreoli [1] for linear logic, focused proof systems provide the normal form proofs for cut-free proofs. In this section, we review the focused proof system for $SELL$, called $SELLF$, proposed in [24].

In order to explain $SELLF$, we first recall some more terminology. We classify as *positive* the formulas whose main connective is either \otimes, \oplus, \exists , the subexponential bang, the unit 1 and negated atoms (A^\perp)⁵. All other formulas are classified as *negative*. Figure 1 contains the focused proof system $SELLF$ that is a rather straightforward generalization of Andreoli's original system. There are two kinds of arrows in this proof system. Sequents with the \Downarrow belong to the *positive* phase and introduce the logical connective of the "focused" formula (the one to the right of the arrow): building proofs of such sequents may require non-invertible proof steps to be taken. Sequents with the \Uparrow belong to the *negative* phase and decompose the formulas on their right in such a way that only invertible inference rules are applied. The

²A preorder relation is a binary relation that is reflexive and transitive.

³This last condition on the pre-order is necessary to prove that $SELL_\Sigma$ admits cut-elimination see [7]. Moreover, it is possible to separate the set of labels into two sub-sets denoting the subexponential labels that allow contraction and weakening respectively. We opted to use a single set \mathcal{U} , as here we only use subexponentials that have none of these structural properties or both. See [23] for more details.

⁴Whenever it is clear from the context, we will elide the subexponential signature Σ .

⁵In the original presentation of $SELLF$ [24], one is allowed to assign the polarity of literals. Here for simplicity, we assume a fixed polarity, where negated atoms have always positive polarity.

structural rules $D_1, D_l, R \uparrow$, and $R \downarrow$ and the promotion rule make the transition between a negative and a positive phase.

Similar to the usual presentation of linear logic, there is a *pair* of contexts to the left of \uparrow and \downarrow of sequents, written here as $\mathcal{K} : \Gamma$. The second context, Γ , collects the formulas whose main connective is not a question-mark, behaving as the bounded context in linear logic. But differently from linear logic, where the first context is a set of formulas whose main connective is a question-mark, we generalize \mathcal{K} to be an *indexed context*, which is a mapping from each index in the set I (for some given and fixed subexponential signature) to a finite multiset of formulas, in order to accommodate for more than one subexponential in *SELLF*. In Andreoli's focused system for linear logic, the index set contains a single subexponential, ∞ , and $\mathcal{K}[\infty]$ contains the set of unbounded formulas. Figure 2 contains different operations used in such indexed contexts. For example, the operation $(\mathcal{K}_1 \otimes \mathcal{K}_2)$, used in the tensor rule, specifies the resulting indexed context obtained by merging two contexts \mathcal{K}_1 and \mathcal{K}_2 .

One of the main features of focusing is to allow the composition of a collection of inference rules of the same polarity into a “macro-rule.” Consider, for example, the formula $N_1 \oplus N_2 \oplus N_3$, where all N_1, N_2 , and N_3 are negative formulas. Once focused on, the only way to introduce such a formula is by using a “macro-rule” of the form:

$$\frac{\vdash \mathcal{K} : \Gamma \uparrow N_i}{\vdash \mathcal{K} : \Gamma \downarrow N_1 \oplus N_2 \oplus N_3}$$

where $i \in \{1, 2, 3\}$. In this paper, we will encode proof systems in *SELLF* in such a way that the “macro-rules” available using our specifications match exactly the inference rules of the encoded system.

This paper will make great use of the promotion rule, $!^l$, in order to specify the structural restrictions of a proof system. In particular, this rule determines two different operations when seen from the conclusion to premise. The first one arises by its side condition: a bang can be introduced only if the linear contexts that are not greater to l are all empty. This operation is similar to the promotion rule in plain linear logic: a bang can be introduced only if the linear context is empty. The second operation is specified by using the operation $\mathcal{K} \leq_l$: in the premise of the promotion rule all unbounded contexts that are not greater than l are erased. Notice that such operation is not available in plain linear logic.

Nigam in [23] proved that *SELLF* is sound and complete with respect to *SELL*.

THEOREM 2.3

For any subexponential signature Σ , *SELLF* $_{\Sigma}$ is sound and complete with respect to *SELL* $_{\Sigma}$.

Finally, to improve readability, we will often show explicitly the formulas appearing in the image of the indexed context, \mathcal{K} , of a sequent. For example, if the set of subexponential indices is $\{x_1, \dots, x_n\}$, then the following negative sequent

$$\vdash \Theta_1 \dot{x}_1 \Theta_2 \dot{x}_2 \cdots \Theta_n \dot{x}_n \Gamma \uparrow L$$

denotes the *SELLF* sequent $\vdash \mathcal{K} : \Gamma \uparrow L$, such that $\mathcal{K}[x_i] = \Theta_i$ for all $1 \leq i \leq n$. Note that each separator “:” is now annotated with a subexponential index. We will also assume the existence of a maximal unbounded subexponential called ∞ , which is greater than all other subexponentials. This subexponential is used to mark the linear logic specification of proof systems explained in the next section.

$$\begin{array}{c}
\text{NEGATIVE PHASE} \\
\frac{\frac{\vdash \mathcal{K} : \Gamma \uparrow L, A \quad \vdash \mathcal{K} : \Gamma \uparrow L, B}{\vdash \mathcal{K} : \Gamma \uparrow L, A \& B} [\&] \quad \frac{\vdash \mathcal{K} : \Gamma \uparrow L, A, B}{\vdash \mathcal{K} : \Gamma \uparrow L, A \wp B} [\wp] \quad \frac{}{\vdash \mathcal{K} : \Gamma \uparrow L, \top} [\top]}{\vdash \mathcal{K} : \Gamma \uparrow L} \\
\frac{\vdash \mathcal{K} : \Gamma \uparrow L}{\vdash \mathcal{K} : \Gamma \uparrow L, \perp} [\perp] \quad \frac{\vdash \mathcal{K} : \Gamma \uparrow L, A\{c/x\}}{\vdash \mathcal{K} : \Gamma \uparrow L, \forall x.A} [\forall] \quad \frac{\vdash \mathcal{K} +_l A : \Gamma \uparrow L}{\vdash \mathcal{K} : \Gamma \uparrow L, ?^l A} [?^l] \\
\text{POSITIVE PHASE} \\
\frac{\vdash \mathcal{K} : \Gamma \Downarrow A_i}{\vdash \mathcal{K} : \Gamma \Downarrow A_1 \oplus A_2} [\oplus_i] \quad \frac{\vdash \mathcal{K}_1 : \Gamma \Downarrow A \quad \vdash \mathcal{K}_2 : \Delta \Downarrow B}{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 : \Gamma, \Delta \Downarrow A \otimes B} [\otimes, \text{given } (\mathcal{K}_1 = \mathcal{K}_2)|_u] \\
\frac{}{\vdash \mathcal{K} : \cdot \Downarrow 1} [1, \text{given } \mathcal{K}[I \setminus \mathcal{U}] = \emptyset] \quad \frac{\vdash \mathcal{K} : \Gamma \Downarrow A\{t/x\}}{\vdash \mathcal{K} : \Gamma \Downarrow \exists x.A} [\exists] \\
\frac{\vdash \mathcal{K} \leq_l : \cdot \uparrow A}{\vdash \mathcal{K} : \cdot \Downarrow !^l A} [!^l, \text{given } \mathcal{K}[\{x \mid l \not\leq x \wedge x \notin \mathcal{U}\}] = \emptyset] \\
\text{INITIAL, REACTION AND DECISION RULES} \\
\frac{}{\vdash \mathcal{K} : \Gamma \Downarrow A_i^\perp} [I, \text{given } A_i \in (\Gamma \cup \mathcal{K}[I]) \text{ and } (\Gamma \cup \mathcal{K}[I \setminus \mathcal{U}]) \subseteq \{A_i\}] \\
\frac{\vdash \mathcal{K} +_l P : \Gamma \Downarrow P}{\vdash \mathcal{K} +_l P : \Gamma \uparrow \cdot} [D_l, \text{given } l \in \mathcal{U}] \quad \frac{\vdash \mathcal{K} : \Gamma \Downarrow P}{\vdash \mathcal{K} +_l P : \Gamma \uparrow \cdot} [D_l, \text{given } l \notin \mathcal{U}] \\
\frac{\vdash \mathcal{K} : \Gamma \Downarrow P}{\vdash \mathcal{K} : \Gamma, P \uparrow \cdot} [D_1] \quad \frac{\vdash \mathcal{K} : \Gamma \uparrow N}{\vdash \mathcal{K} : \Gamma \Downarrow N} [R \Downarrow] \quad \frac{\vdash \mathcal{K} : \Gamma, S \uparrow L}{\vdash \mathcal{K} : \Gamma \uparrow L, S} [R \uparrow]
\end{array}$$

FIG. 1: Focused linear logic system with subexponentials. We assume that all atoms are classified as negative polarity formulas and their negations as positive polarity formulas. Here, L is a list of formulas, Γ is a multi-set of positive formulas and literals, A_i is an atomic formula, P is not an atomic formula, S is a literal or a positive formula and N is a negative formula.

$$\begin{array}{l}
\bullet (\mathcal{K}_1 \otimes \mathcal{K}_2)[i] = \begin{cases} \mathcal{K}_1[i] \cup \mathcal{K}_2[i] & \text{if } i \notin \mathcal{U} \\ \mathcal{K}_1[i] & \text{if } i \in \mathcal{U} \end{cases} \quad \bullet \mathcal{K}[S] = \bigcup \{\mathcal{K}[i] \mid i \in S\} \\
\bullet (\mathcal{K} +_l A)[i] = \begin{cases} \mathcal{K}[i] \cup \{A\} & \text{if } i = l \\ \mathcal{K}[i] & \text{otherwise} \end{cases} \quad \bullet \mathcal{K} \leq_l [l] = \begin{cases} \mathcal{K}[l] & \text{if } i \leq l \\ \emptyset & \text{if } i \not\leq l \end{cases} \\
\bullet (\mathcal{K}_1 \star \mathcal{K}_2) \upharpoonright_S \text{ is true if and only if } (\mathcal{K}_1[j] \star \mathcal{K}_2[j]) \text{ for all } j \in S
\end{array}$$

FIG. 2: Specification of operations on contexts. Here, $i \in I$, $j \in S$, $S \subseteq I$, and the binary connective $\star \in \{=, \subset, \subseteq\}$.

3 Encoding Proof Systems in *SELLF*

3.1 Encoding Sequents

Similar as in Church's simple type theory [5], we assume that linear logic propositions have type o and that the object-logic quantifiers have type $(term \rightarrow form) \rightarrow form$, where $term$ and $form$ are respectively the types for an object-logic term and for object-logic formulas. Moreover, following [30, 31, 25], we encode a sequent in *SELLF* by using two meta-level atoms $[\cdot]$ and $\lceil \cdot \rceil$ of type $form \rightarrow o$. These meta-level atoms are used to mark, respectively, formulas appearing on the left and on the right of sequents. For example, the formulas appearing in the sequent $B_1, \dots, B_n \vdash C_1, \dots, C_m$ are specified by the meta-level atoms: $[B_1], \dots, [B_n], \lceil C_1 \rceil, \dots, \lceil C_m \rceil$. Both atoms are naturally generalized to sets and multisets of formulas, i.e. $\lfloor \Gamma \rfloor = \{\lfloor F \rfloor \mid F \in \Gamma\}$ and $\lceil \Delta \rceil = \{\lceil F \rceil \mid F \in \Delta\}$

Given such a collection of meta-level atoms, it remains to decide where exactly these atoms are going to appear in the meta-level sequents. When using linear logic without subexponentials, the number of possibilities is quite limited. As the sequents of linear logic without subexponentials ($\vdash \Theta : \Gamma$) have only two contexts, namely an unbounded context (Θ) (which is treated as a set of formulas) and a bounded context (Γ) (which is treated as a multiset of formulas), there are only two options: a meta-level formula either belongs to one context or to the other.

The use of subexponentials opens, on the other hand, a wider range of possibilities, as there is one context for *each* subexponential index. For instance, we can encode the object-level sequent above by using two subexponentials: l whose context stores $[\cdot]$ formulas and r whose context stores $\lceil \cdot \rceil$ formulas. The meta-level encoding of an object-level sequent would in this case have the following form $\vdash \mathcal{L} :_{\infty} [B_1], \dots, [B_n] :_l \lceil C_1 \rceil, \dots, \lceil C_m \rceil :_r \cdot \uparrow \cdot$, where \mathcal{L} is a theory specifying the proof system's introduction rules, which will be explained later. Moreover, if needed, one could further refine such specification and partition meta-level atoms in more contexts by using more subexponentials. For instance, the focused sequent of focused proof systems, such as *LJQ**, has an extra context, called *stoup*, where the focused formula is. To specify such a sequent, we use an additional subexponential index f , whose context contains the focused formula. As we show in the next subsection, when we describe how inference rules are specified, this refinement of linear logic sequents enables the specification of a number of structural properties of proof systems in an elegant fashion.

Furthermore, in *SELLF*, subexponential contexts can be configured so as to behave as sets or multisets. For instance, if we use the subexponentials signature $\langle \{l, r, \infty\}, \leq, \{l, \infty\} \rangle$, with some preorder \leq , the contexts for l and ∞ are treated as sets, while the context for r is treated as a multiset. Such situation would be useful for any proof system where the right-hand-side of its sequent behaves as a *multiset* of formulas and the left-hand-side behaves as a *set* of formulas, *e.g.*, the system *LJ* for intuitionistic logic. We could, alternatively, specify the contexts for both l and r to behave as multisets. In this case, l and r are bounded subexponentials. Such a specification is used when both sides of the object-level sequent behave as multisets, such as for the system *G1m* [33] for minimal logic, which has explicit weakening and contraction rules.

3.2 Encoding Inference Rules

The inference rules of a system are specified using *monopoles* and *bipoles* [19]. These concepts are generalized next.

8 An Extended Framework for Specifying and Reasoning about Proof Systems

DEFINITION 3.1

A *monopole* formula is a *SELLF* formula that is built up from atoms and occurrences of the negative connectives, with the restriction that, for any label t , $?^t$ has atomic scope and that all atomic formulas, A , are necessarily under the scope of a subexponential question-mark, $?^t A$. A *bipole* is a formula built from monopoles and negated atoms using only positive connectives, with the additional restriction that $!^s$, $s \in I$, can only be applied to a monopole. We shall also insist that a bipole is either a negated atom or has a top-level positive connective.⁶

The last restriction on bipoles forces them to be different from monopoles: bipoles are always positive formulas. Using the linear logic distributive properties, monopoles are equivalent to formulas of the form

$$\forall x_1 \dots \forall x_p [\&_{i=1, \dots, n} \wp_{j=1, \dots, m_i} ?^{t_{i,j}} A_{i,j}],$$

where $A_{i,j}$ is an atomic formula and $t_{i,j} \in I$. Similarly, bipoles can be rewritten as formulas of the form

$$\exists x_1 \dots \exists x_p [\oplus_{i=1, \dots, n} \otimes_{j=1, \dots, m_i} C_{i,j}],$$

where $C_{i,j}$ are either negated atoms, monopole formulas, or the result of applying $!^s$ to a monopole formula to some $s \in I$.

Throughout this paper, the following invariant holds: the linear context to the left of the \uparrow and \Downarrow on *SELLF* sequents is empty⁷. This invariant derives from the focusing discipline and from the definition of bipoles above, namely, from the fact that all atomic formulas are under the scope of a $?^t$. This is illustrated by the derivation below. In particular, according to the focusing discipline, a bipole is necessarily introduced by such a derivation containing a single alternation of phases. We call these derivations *bipole-derivations*.

$$\frac{\frac{\frac{\frac{\vdash \mathcal{K}'_i : \cdot \uparrow \cdot}{\vdash \mathcal{K}'_i <_s : \cdot \uparrow \wp_{j=1, \dots, m_i} ?^{t_{i,j}} A_{i,j}} [m_i \times (\wp, ?^t)] \dots}{\vdash \mathcal{K}'_i <_s : \cdot \uparrow \forall x_1 \dots \forall x_p [\&_{i=1, \dots, n} \wp_{j=1, \dots, m_i} ?^{t_{i,j}} A_{i,j}]} [p \times \forall, n \times \&]}{\vdash \mathcal{K}'_i : \cdot \Downarrow !^s \forall x_1 \dots \forall x_p [\&_{i=1, \dots, n} \wp_{j=1, \dots, m_i} ?^{t_{i,j}} A_{i,j}]} [!^s]} \dots}{\vdash \mathcal{K}' : \cdot \Downarrow \exists x_1 \dots \exists x_p [\oplus_{i=1, \dots, k} \otimes_{j=1, \dots, q_i} C_{i,j}]} [t \times \exists, k \times \oplus, q_i \times \otimes]} \dots \frac{\vdash \mathcal{K}' : \cdot \Downarrow \exists x_1 \dots \exists x_p [\oplus_{i=1, \dots, k} \otimes_{j=1, \dots, q_i} C_{i,j}]}{\vdash \mathcal{K} : \cdot \uparrow \cdot} [D]$$

Notice that the derivation above contains a single positive and a single negative trunk. Moreover, if the connective $!^s$ is not present, then the rule $!^s$ is replaced by the rule $R \Downarrow$.

It turns out that one can match exactly the shape of a bipole-derivation with the shape of the inference rule the bipole encodes. Consider, for example, the following bipole $F = \exists A \exists B. [A \supset B]^\perp \otimes (!^l ?^r [A] \otimes ?^l [B])$ encoding the \supset left-introduction rule for intuitionistic logic, assuming the signature $\langle \{l, r, \infty\}, \{l < \infty, r < \infty\}, \{l, \infty\} \rangle$. The only way to introduce F in *SELLF* is by using a bipole-derivation of the following form, where $F \in \mathcal{L}$:

$$\frac{\frac{\frac{\vdash \mathcal{L} \overset{\circ}{\circ} [\Gamma], [A \supset B] \dot{\dot{}} [A] \dot{\dot{}} \uparrow \cdot \quad \vdash \mathcal{L} \overset{\circ}{\circ} [\Gamma], [A \supset B], [B] \dot{\dot{}} [G] \dot{\dot{}} \uparrow \cdot}{\vdash \mathcal{L} \overset{\circ}{\circ} [\Gamma], [A \supset B] \dot{\dot{}} [G] \dot{\dot{}} \Downarrow F}}{\vdash \mathcal{L} \overset{\circ}{\circ} [\Gamma], [A \supset B] \dot{\dot{}} [G] \dot{\dot{}} \uparrow \cdot}}$$

⁶Notice that our definition of monopole and bipole is more restrictive than the definition given by Andreoli [1].

⁷That is, the context Γ in $\vdash \mathcal{K} : \Gamma \uparrow \cdot$ and in $\vdash \mathcal{K} : \Gamma \Downarrow F$ is empty.

right-hand side, while the context for \diamond_l and \Box_r store formulas whose main connective is a diamond and box on the left and on the right-hand side, respectively. For instance, the sequent $\Box\Gamma, \Gamma', \diamond\Gamma'' \vdash \Box\Delta, \Delta', \diamond\Delta''$ is encoded as $\vdash \Theta \overset{\circ}{\circ} [\Box\Gamma] \overset{\circ}{\circ}_l [\Gamma', \diamond\Gamma''] \overset{\circ}{\circ}_r [\Box\Delta, \Delta'] \overset{\circ}{\circ}_r [\diamond\Delta''] \overset{\circ}{\circ}_r \cdot \uparrow \cdot$, where Θ is the theory specifying the inference rules of the system. The following clauses, classified as structural clauses (see Definition 3.2), specify the relation among object-logic formulas whose main connective is a \Box and a \diamond and the context of the indexes \Box_l and \diamond_r .

$$(\Box_S) \ [\Box A]^\perp \otimes ?^{\Box_l}[\Box A] \quad \text{and} \quad (\diamond_S) \ [\diamond A]^\perp \otimes ?^{\diamond_r}[\diamond A]$$

From these clauses we obtain the equivalences⁸ $\forall A. [\Box A] \equiv ?^{\Box_l}[\Box A]$ and $\forall A. [\diamond A] \equiv ?^{\diamond_r}[\diamond A]$. That is, any formula of the form $[\Box A]$ can be placed in the context of \Box_l and any formula of the form $[\diamond A]$ in the context of \diamond_r . Furthermore, we specify e as follows: $e < \Box_l$, $e < \diamond_r$, and $e < \infty$ and e is unrelated to the remaining subexponentials. Hence, the connective $!^e$ can play a similar role for the specification of the rule \Box_R as the $!^l$ in the specification of the \supset_R rule above. In particular, to introduce $!^e$, all contexts but \Box_l , \diamond_r and ∞ have to be erased. It is easy to check that this operation is exactly the one needed for specifying the modal logic rule above. In Section 4, we show this specification in detail.

In combination to the use of bounded subexponentials, whose contexts behave as multisets, subexponentials can also be used to check whether a formula is present in the sequent. These type of requirement also often appears in inference rules, such as the one below for intuitionistic lax logic [10]:

$$\frac{F, \Gamma \longrightarrow \circ G}{\circ F, \Gamma \longrightarrow \circ G} [\circ_L]$$

The connective \circ on the left can be introduced only if the main connective of the formula on the right is also a \circ . To specify this rule, we use the following subexponentials indexes: $\{l, r, \circ_r, \infty\}$, where l and ∞ are unbounded, while r and \circ_r are bounded. Moreover, $r < \circ_r$, $\circ_r < l$, and $\circ_r, l < \infty$. Similarly as in the modal logic example above, a formula $[H]$ is stored in the context of the subexponential \circ_r only if H 's main connective is \circ , *i.e.*, $H = \circ H'$ for some H' . This is also accomplished by using an analogous logical equivalence, namely, $\forall A. [\circ A] \equiv ?^{\circ_r}[\circ A]$, which is obtained by using the clause (\circ_S) in Figure 13. It is then easy to check that the formula $\exists F. [\circ F]^\perp \otimes !^{\circ_r}[F]$ specifies the rule above. In particular, the $!^{\circ_r}$ is used to check whether the formula on the right has \circ as main connective: if this is the case, then some formula of the form $[\circ G]$ will be in the context \circ_r , while the context for r will be empty. Notice, that this specification does not mention any side-formulas of the sequent, not even the formula appearing on the right-hand-side of the sequent. As we argue later, the use of such declarative specifications will help us reason about proof systems.

3.3 Adding Labels and Relations

Labelled proof systems have been proposed in [21] in order to syntactically capture the Kripke semantics of modal logics and to obtain contraction and cut-free sequent calculus for some of these logics. But it has also been used in hybrid and temporal logics. As the name suggests, formulas F in these proof systems are annotated with labels l , written $l : F$. Moreover, labelled proof systems may also specify a relation R among these labels, which can be used in inference rules. The following is an example of a rule in a labelled proof

⁸ $F \equiv G$ denotes the formula $(F \wp G^\perp) \otimes (F^\perp \wp G)$.

$$\begin{array}{cc}
 \frac{xRx, \Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta} \text{ [Ref]} & \frac{xRz, xRy, yRz, \Gamma \Rightarrow \Delta}{xRy, yRz, \Gamma \Rightarrow \Delta} \text{ [Trans]} \\
 \frac{yRx, xRy, \Gamma \Rightarrow \Delta}{xRy, \Gamma \Rightarrow \Delta} \text{ [Sym]} & \frac{yRz, xRy, xRz, \Gamma \Rightarrow \Delta}{xRy, xRz, \Gamma \Rightarrow \Delta} \text{ [Eucl]}
 \end{array}$$

 FIG. 3. Rules for the relation R

system:

$$\frac{xRy, \Gamma \Rightarrow \Delta, y : A}{\Gamma \Rightarrow \Delta, x : \Box A} \text{ } [\Box_r]$$

where the label y does not appear in Γ nor in Δ . The semantic of \Box is the following: if $\Box F$ is true in a world x , then F is true in every world y that is accessible from x . If we interpret the labels as worlds and R as the accessibility relation, it is easy to see that the rule above specifies exactly the intended semantic.

It is also possible to specify the properties of the relation R . For instance, the following rule specifies that R is symmetric.

$$\frac{yRx, xRy, \Gamma \Rightarrow \Delta}{xRy, \Gamma \Rightarrow \Delta} \text{ [Sym]}$$

The rules in Figure 3 specify other properties, such as transitivity, reflexivity and euclidean-ness⁹ of R .

As investigated by Negri in [21], a basic labelled proof system plus a subset of the rules in Figure 3 result in proof systems for different modal logics, such as T, 4, B, S4, TB and S5. This illustrates the power of labelled proof systems.

While labelled formulas can be encoded as shown previously, using the $[\cdot]$ and $[\cdot]$ meta-level atoms (e.g., $[l : F]$ and $[l : \neg F]$), the encoding of relations require some extra machinery. In particular, relations are not encoded as object-logic formulas with the $[\cdot]$ and $[\cdot]$, as they are not really formulas. Relations are specified by logical theories in the meta-logic. For the example above, the relation R is specified by the logical formulas:

$$\begin{array}{ll}
 \text{(Ref)} & \exists x.[R(x, x)^\perp] \\
 \text{(Sym)} & \exists x \exists y.[R(x, y)^\perp \otimes ?^R R(y, x)] \\
 \text{(Trans)} & \exists x \exists y \exists z.[R(x, y)^\perp \otimes R(y, z)^\perp \otimes ?^R R(x, z)] \\
 \text{(Eucl)} & \exists x \exists y \exists z.[R(x, y)^\perp \otimes R(x, z)^\perp \otimes ?^R R(y, z)]
 \end{array}$$

where $R(\cdot, \cdot)$ is a meta-level atomic formula. From a simple inspection, it is easy to see that each clause corresponds to one of the relations' properties shown in Figure 3.

Also, in order not to mix relations and object-logic formulas, we define an *unbounded* subexponential index R that is unrelated to all other indexes, which will store all $R(\cdot, \cdot)$ formulas as well as the theories specifying the properties of the relation. Whenever we want to check whether a formula $R(a, b)$ follows from the existing relations and its theory, \mathcal{R} , we use a $!^R$ as follows:

$$\frac{\vdash \cdot \overset{\cdot}{\circlearrowleft} \mathcal{R}, R(a_1, b_1), \dots, R(a_n, b_n) \overset{\cdot}{R} \cdot \overset{\cdot}{i} \cdots \overset{\cdot}{m-1} \cdot \overset{\cdot}{m} : \cdot \uparrow R(a, b)^\perp}{\vdash \mathcal{L} \overset{\cdot}{\circlearrowleft} \mathcal{R}, R(a_1, b_1), \dots, R(a_n, b_n) \overset{\cdot}{R} \mathcal{T}_1 \overset{\cdot}{i} \cdots \overset{\cdot}{m-1} \mathcal{T}_m \overset{\cdot}{m} : \cdot \downarrow !^R R(a, b)^\perp} \text{ } [!^R]$$

⁹Meaning that R is an Euclidian relation.

12 An Extended Framework for Specifying and Reasoning about Proof Systems

Notice that in the premise, the formula $R(a, b)^\perp$ should be provable using only the set of formulas $\mathcal{R}, R(a_1, b_1), \dots, R(a_n, b_n)$, that is, it should be provable only using the relations until then constructed and the theory \mathcal{R} specifying its properties.

Finally, as with the right introduction for \square shown above, new relations may be created. Whenever this is needed, the new relation predicate should be added to the context of the subexponential R by using a formula of the form $?^R(R(a, b))$.

3.4 Canonical Proof System Theories

The definition below classifies clauses into three different categories, namely the identity rules (Cut and Init rules), introduction rules, and structural rules, following usual terminology in proof theory literature [33].

DEFINITION 3.2

- i.* In its most general form, the clause specifying the *cut rule* has the form to the left, while the clause specifying the *initial rule* has the form to the right:

$$\text{Cut} = \exists A. !^a ?^b [A] \otimes !^c ?^d [A] \quad \text{and} \quad \text{Init} = \exists A. [A]^\perp \otimes [A]^\perp$$

where a, c are subexponentials that may or may not appear depending on the structural restrictions imposed by the proof system.

- ii.* The *structural rules* are specified by clauses of the form below, where $i, j \in I$:

$$\exists A. [A]^\perp \otimes \underbrace{(?^i [A] \wp \dots \wp ?^i [A])}_n \quad \text{or} \quad \exists A. [A]^\perp \otimes \underbrace{(?^j [A] \wp \dots \wp ?^j [A])}_n.$$

Moreover, we classify a structural rule as *contextual* if $n = 1$.

- iii.* Finally, an *introduction clause* is a closed bipole formula of the form

$$\exists x_1 \dots \exists x_n [(q(\diamond(x_1, \dots, x_n)))^\perp \otimes B \otimes R]$$

where \diamond is an object-level connective of arity n ($n \geq 0$) and $q \in \{[\cdot], [\cdot]\}$. Furthermore, B does not contain negated atoms and an atom occurring in B is either of the form $p(x_i)$ or $p(x_i(y))$ where $p \in \{[\cdot], [\cdot]\}$ and $1 \leq i \leq n$. In the first case, x_i has type *obj* while in the second case x_i has type $d \rightarrow \text{obj}$ and y is a variable (of type d) quantified (universally or existentially) in B (in particular, y is not in $\{x_1, \dots, x_n\}$). Finally, when encoding labelled proof systems, R is either the formula $!^R R(x, y)^\perp$ or $?^R R(x, y)$ or the formula 1 if no relation is mentioned.

In the remainder of this paper, we restrict our discussion to the so called *canonical systems* [3].

DEFINITION 3.3

A *canonical clause* is an introduction clause restricted so that, for every pair of atoms of the form $[T]$ and $[S]$ in a body, the head variable of T differs from the head variable of S . A *canonical proof system theory* is a set \mathcal{X} of formulas such that (i) the *Init* and *Cut* clauses are members of \mathcal{X} ; (ii) structural clauses may be members of \mathcal{X} ; and (iii) all other clauses in \mathcal{X} are canonical introduction clauses.

Remark: In the definition above of bipoles all atoms are under the scope of a question-mark $?^\perp$. However, as discussed in [30], another definition also works where bipoles do not contain

any question-mark, but the theory contains contextual structural rules which specify which context a formula belongs to. For instance, the contextual structural rule

$$\exists A.[A] \otimes ?^l[A]$$

specifies that all meta-level atoms of the form $[A]$ also belong to the context of the subexponential l . One advantage of the latter approach is that the bipole theories become more modular, as one just needs to rewrite the contextual structural rules, when one needs to change the specification. On the other hand, adding more clauses turns the adequacy theorems more complicated as one has to consider more cases. Since we want to stick to the adequacy on the level of derivations [25], we chose to include question-marks in the definition of bipoles.

4 Examples of Proof Systems encoded in *SELLF*

This section contains the specification of a number of proof systems that do not seem possible to be encoded in linear logic without the use of subexponentials or without mentioning side-formulas explicitly. In our specifications, we assume all free variables to be existentially quantified. Moreover, all the encodings below have the strongest level of adequacy, namely adequacy on the level of derivations [25].

Finally some of the systems, namely *mLJ* and *S4*, are the variants of the corresponding systems, where the structural rules are incorporated into the introduction rules and where the principal formula is always contracted in the premises.

4.1 *G1m*

The system *G1m* [33] (Figure 4) for minimal logic contains explicit rules for weakening and contraction of formulas appearing on the left-hand-side of sequents. The encoding of this system illustrates how to use subexponentials to specify proof systems whose sequents contain two or more linear contexts. Here, in particular, both the left and the right-hand-side of *G1m* sequents are treated as multisets of formulas.

We specify *G1m* by using the following subexponential signature: $\langle \{\infty, l, r\}, \{r < l < \infty\}, \{\infty\} \rangle$. The subexponentials l and r allow neither contraction nor weakening. Their contexts will store, respectively, object-logic formulas appearing on the left and on the right of the sequent. The theory \mathcal{L}_{G1m} , depicted in Figure 5, specifies *G1m*'s introduction rules in *SELLF*. This theory is stored in the context of ∞ . Thus, a *G1m* sequent of the form $\Gamma \vdash C$ is encoded as the *SELLF* sequent $\vdash \mathcal{L}_{G1m} \dot{\infty} [\Gamma] \dot{i} [C] \dot{i} \cdot \uparrow \cdot$.

Each clause in \mathcal{L}_{G1m} corresponds to one introduction rule of *G1m*. To obtain such strong correspondence, we need to capture precisely the structural restrictions in the system. In particular, the use of the $!$ in the clauses (\supset_L) , specifying the rule \supset_L , and (Cut), specifying the Cut rule, is necessary. It forces that the side-formula, C , appearing in the right-hand-side of their conclusion is moved to the correct premise. This is illustrated by the following derivation:

$$\frac{\frac{\frac{\vdash \mathcal{L}_{G1m} \dot{\infty} [\Gamma_1] \dot{i} [A] \dot{i} \cdot \uparrow}{\vdash \mathcal{L}_{G1m} \dot{\infty} [\Gamma_1] \dot{i} \cdot \dot{i} \cdot \downarrow \cdot !^r[A]} \quad [\uparrow, ?^r]}{\vdash \mathcal{L}_{G1m} \dot{\infty} [\Gamma_1, \Gamma_2] \dot{i} [C] \dot{i} \cdot \downarrow \cdot !^r[A] \otimes ?^l[A]} \quad [\uparrow, ?^l]}{\vdash \mathcal{L}_{G1m} \dot{\infty} [\Gamma_1, \Gamma_2] \dot{i} [C] \dot{i} \cdot \uparrow} \quad [D_{\infty}, \exists]$$

$$\begin{array}{c}
\frac{\Gamma_1 \longrightarrow A \quad \Gamma_2, B \longrightarrow C}{\Gamma_1, \Gamma_2, A \supset B \longrightarrow C} [\supset L] \quad \frac{\Gamma, A \longrightarrow B}{\Gamma \longrightarrow A \supset B} [\supset R] \quad \frac{\Gamma, A, B \longrightarrow C}{\Gamma, A \wedge B \longrightarrow C} [\wedge L] \\
\frac{\Gamma_1 \longrightarrow A \quad \Gamma_2 \longrightarrow B}{\Gamma_1, \Gamma_2 \longrightarrow A \wedge B} [\wedge R] \quad \frac{\Gamma, A\{t/x\} \longrightarrow C}{\Gamma, \forall x A \longrightarrow C} [\forall L] \quad \frac{\Gamma \longrightarrow A\{c/x\}}{\Gamma \longrightarrow \forall x A} [\forall R] \\
\frac{\Gamma, A\{c/x\} \longrightarrow C}{\Gamma, \exists x A \longrightarrow C} [\exists L] \quad \frac{\Gamma \longrightarrow A\{t/x\}}{\Gamma \longrightarrow \exists x A} [\exists R] \quad \frac{\Gamma, A \longrightarrow C \quad \Gamma, B \longrightarrow C}{\Gamma, A \vee B \longrightarrow C} [\vee L] \\
\frac{\Gamma \longrightarrow A_i}{\Gamma \longrightarrow A_1 \vee A_2} [\vee_i R] \quad \frac{\Gamma \longrightarrow C}{\Gamma, A \longrightarrow C} [W_L] \quad \frac{\Gamma, A, A \longrightarrow C}{\Gamma, A \longrightarrow C} [C_L] \\
\frac{}{A \longrightarrow A} [\text{Init}] \quad \frac{\Gamma_1 \longrightarrow A \quad \Gamma_2, A \longrightarrow C}{\Gamma_1, \Gamma_2 \longrightarrow C} [\text{Cut}]
\end{array}$$

FIG. 4. The sequent calculus system $G1m$ for minimal logic.

$$\begin{array}{ll}
(\supset_L) [A \supset B]^\perp \otimes (!^l ?^r [A] \otimes ?^l [B]) & (\supset_R) [A \supset B]^\perp \otimes !(?^l [A] \wp ?^r [B]) \\
(\wedge_L) [A \wedge B]^\perp \otimes (?^l [A] \wp ?^l [B]) & (\wedge_R) [A \wedge B]^\perp \otimes (!^l ?^r [A] \otimes !^l ?^r [B]) \\
(\vee_L) [A \vee B]^\perp \otimes (?^l [A] \& ?^l [B]) & (\vee_R) [A \vee B]^\perp \otimes (!^l ?^r [A] \oplus !^l ?^r [B]) \\
(\forall_L) [\forall B]^\perp \otimes ?^l [Bx] & (\forall_R) [\forall B]^\perp \otimes !^l \forall x ?^r [Bx] \\
(\exists_L) [\exists B]^\perp \otimes \forall x ?^l [Bx] & (\exists_R) [\exists B]^\perp \otimes !^l ?^r [Bx] \\
(\text{Init}) [B]^\perp \otimes [B]^\perp & (\text{Cut}) !^l ?^r [B] \otimes ?^l [B] \\
(C_L) [B]^\perp \otimes (?^l [B] \wp ?^l [B]) & (W_L) [B]^\perp \otimes \perp
\end{array}$$

FIG. 5. The theory, \mathcal{L}_{G1m} , for $G1m$.

When introducing the tensor, the formula $[C]$ cannot go to the left branch because, in that case, the r context would not be empty and therefore the $!^l$ could not be introduced. Hence, the only way to introduce the formula (Cut) in \mathcal{L}_{G1m} is with a derivation as the one above.

In contrast, it does not seem possible to encode $G1m$ in linear logic (without subexponentials) with such a strong correspondence. The sequents of the dyadic version of linear logic [1] have only two contexts, one for the unbounded formulas and another for the linear formulas. Hence, in linear logic, all linear meta-level atoms would appear in the same context illustrated by the sequent $\vdash \Theta : [\Gamma], [C]$. Furthermore, using the linear logic $!$ enforces that not only $[C]$, but *all* linear formulas in this sequent, namely $[\Gamma]$ and $[C]$, are moved to a different branch. Therefore, one cannot capture, as done by using the subexponential bang $!^l$, that only $[C]$ is necessarily moved to a different branch as specified in the $G1m$ rules \supset_L and Cut.

Finally, as the derivation above illustrates, the $!^l$ s appearing in the specification of $G1m$'s introduction rules specify the structural restriction that $G1m$'s sequents contain exactly one formula on their right-hand-side. This allows us to specify these introduction rules without explicitly mentioning any side-formulas in the sequent, such as, the formula C in the Cut rule. As we show in Section 5, the use of such declarative specifications allow for simple proofs about the object-level systems, such as the proof that it admits cut-elimination.

Repeating this exercise for each inference rule, we establish the following adequacy result in the same lines as done in the encoding of proof systems in our previous work [25].

$$\begin{array}{c}
 \frac{\Gamma, A \supset B \longrightarrow A, \Delta \quad \Gamma, A \supset B, B \longrightarrow \Delta}{\Gamma, A \supset B \longrightarrow \Delta} [\supset_L] \quad \frac{\Gamma, A \longrightarrow B}{\Gamma \longrightarrow A \supset B, \Delta} [\supset_R] \\
 \\
 \frac{\Gamma, A \wedge B, A, B \longrightarrow \Delta}{\Gamma, A \wedge B \longrightarrow \Delta} [\wedge_L] \quad \frac{\Gamma \longrightarrow A \wedge B, A, \Delta \quad \Gamma \longrightarrow A \wedge B, B, \Delta}{\Gamma \longrightarrow A \wedge B, \Delta} [\wedge_R] \\
 \\
 \frac{\Gamma, A \vee B, A, \longrightarrow \Delta \quad \Gamma, A \vee B, B \longrightarrow \Delta}{\Gamma, A \vee B \longrightarrow \Delta} [\vee_L] \quad \frac{\Gamma \longrightarrow A \vee B, A, B, \Delta}{\Gamma \longrightarrow A \vee B, \Delta} [\vee_R] \\
 \\
 \frac{\Gamma, \forall x A, A\{t/x\} \longrightarrow \Delta}{\Gamma, \forall x A \longrightarrow \Delta} [\forall_L] \quad \frac{\Gamma \longrightarrow A\{c/x\}}{\Gamma \longrightarrow \Delta, \forall x A} [\forall_R] \\
 \\
 \frac{\Gamma, \exists x A, A\{c/x\} \longrightarrow \Delta}{\Gamma, \exists x A \longrightarrow \Delta} [\exists_L] \quad \frac{\Gamma \longrightarrow \Delta, \exists x A, A\{t/x\}}{\Gamma \longrightarrow \Delta, \exists x A} [\exists_R] \\
 \\
 \frac{}{\Gamma, A \longrightarrow A, \Delta} [\text{Init}] \quad \frac{\Gamma \longrightarrow B, \Delta \quad \Gamma, B \longrightarrow \Delta}{\Gamma \longrightarrow \Delta} [\text{Cut}] \quad \frac{}{\Gamma, \perp \longrightarrow \Delta} [\perp_L]
 \end{array}$$

 FIG. 6. The multi-conclusion intuitionistic sequent calculus, mLJ , with additive rules.

$$\begin{array}{ll}
 (\supset_L) & [A \supset B]^+ \otimes (?^r[A] \otimes ?^l[B]) \\
 (\wedge_L) & [A \wedge B]^+ \otimes (?^l[A] \wp ?^l[B]) \\
 (\vee_L) & [A \vee B]^+ \otimes (?^l[A] \otimes ?^l[B]) \\
 (\forall_L) & [\forall B]^+ \otimes ?^l[Bx] \\
 (\exists_L) & [\exists B]^+ \otimes \forall x ?^l[Bx] \\
 (\perp_L) & [\perp]^+ \\
 (\text{Init}) & [B]^+ \otimes [B]^+ \\
 (\text{Pos}) & [B]^+ \otimes ?^l[B] \\
 (\supset_R) & [A \supset B]^+ \otimes !^l(?^l[A] \wp ?^r[B]) \\
 (\wedge_R) & [A \wedge B]^+ \otimes (?^r[A] \otimes ?^r[B]) \\
 (\vee_R) & [A \vee B]^+ \otimes (?^r[A] \wp ?^r[B]) \\
 (\forall_R) & [\forall B]^+ \otimes !^l \forall x ?^r[Bx] \\
 (\exists_R) & [\exists B]^+ \otimes ?^r[Bx] \\
 (\text{Cut}) & ?^l[B] \otimes ?^r[B] \\
 (\text{Neg}) & [B]^+ \otimes ?^r[B]
 \end{array}$$

 FIG. 7. Theory \mathcal{L}_{mlj} for the multi-conclusion intuitionistic logic system mLJ .

PROPOSITION 4.1

Let $\Gamma \cup \{C\}$ be a set of object logic formulas, and let the subexponentials, l and r , be specified by the signature $\langle \{\infty, l, r\}, \{r < l < \infty\}, \{\infty\} \rangle$. Then the sequent $\vdash \mathcal{L}_{G1m} \dot{\circ} [\Gamma] \dot{\circ} [C] \dot{\circ} \dot{\circ} \uparrow$ is provable in $SELLF$ if and only if the sequent $\Gamma \longrightarrow C$ is provable in $G1m$.

4.2 mLJ

We now encode in $SELLF$ the multi-conclusion sequent calculus mLJ for intuitionistic logic depicted in Figure 6. Its encoding illustrates the use of subexponentials to specify rules requiring some formulas to be weakened. In particular, the mLJ 's rules \supset_R and \forall_R require that the formulas Δ appearing in their conclusions are weakened in their premises.

Formally, the theory \mathcal{L}_{mlj} is formed by the clauses shown in Figure 7. This theory specifies mLJ 's rules by using the subexponential signature $\langle \{\infty, l, r\}; \{l < \infty, r < \infty\}; \{\infty, l, r\} \rangle$. As before with the encoding of $G1m$, we make use of two subexponentials l and r to store, respectively, meta-level atoms $[\cdot]$ and $[\cdot]$, but now we allow both contraction and weakening to these subexponential indices. As described in Section 3.2, the use of $!$ in the clauses (\supset_R)

$$\begin{array}{c}
\frac{\Gamma, A \supset B \rightarrow A; \cdot \quad \Gamma, A \supset B, B \vdash \Delta}{\Gamma, A \supset B \vdash \Delta} [\supset_L] \quad \frac{\Gamma, A \vdash B}{\Gamma \rightarrow A \supset B; \Delta} [\supset_R] \\
\frac{\Gamma, A \vee B, A \vdash \Delta \quad \Gamma, A \vee B, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta} [\vee_L] \quad \frac{\Gamma \vdash A, B, \Delta}{\Gamma \rightarrow A \vee B; \Delta} [\vee_R] \\
\frac{\Gamma, A \wedge B, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta} [\wedge_L] \quad \frac{\Gamma \rightarrow A; \Delta \quad \Gamma \rightarrow B; \Delta}{\Gamma \rightarrow A \wedge B; \Delta} [\wedge_R] \\
\frac{}{\Gamma, A \rightarrow A; \Delta} [\text{Init}] \quad \frac{\Gamma \rightarrow C; \Delta}{\Gamma \vdash C, \Delta} [D] \quad \frac{}{\Gamma, \perp \vdash \Delta} [\perp_L]
\end{array}$$

FIG. 8: The the cut-free fragment of the focused multi-conclusion system for intuitionistic logic - LJQ^* .

$$\begin{array}{ll}
(\text{Init}) & [A]^\perp \otimes [A]^\perp \quad (\perp_L) \quad [\perp]^\perp \\
(\supset_L) & [A \supset B]^\perp \otimes (!^l ?^f [A] \otimes !^r ?^l [B]) \quad (\supset_R) \quad [A \supset B]^\perp \otimes !^l (?^l [A] \wp ?^r [B]) \\
(\vee_L) & [A \vee B]^\perp \otimes (!^r ?^l [A] \otimes !^r ?^l [B]) \quad (\vee_R) \quad [A \vee B]^\perp \otimes !^r (?^r [A] \wp ?^r [B]) \\
(\wedge_L) & [A \wedge B]^\perp \otimes !^r (?^l [A] \wp ?^l [B]) \quad (\wedge_R) \quad [A \wedge B]^\perp \otimes (!^r ?^f [A] \otimes !^r ?^f [B])
\end{array}$$

FIG. 9. The theory \mathcal{L}_{ljq} encoding the cut-free fragment of the system LJQ^* .

and (\forall_R) specifies that the formulas in the context r should be necessarily weakened. This is illustrated by the following derivation introducing the formula (\forall_R) in \mathcal{L}_{mlj} :

$$\frac{\frac{\frac{}{\vdash \mathcal{L}_{mlj} \dot{\circ} [\Gamma] \dot{i} [\Delta, \forall x A] \dot{i} \cdot \Downarrow [\forall x A]^\perp} [\text{I}_R]}{\vdash \mathcal{L}_{mlj} \dot{\circ} [\Gamma] \dot{i} [\Delta, \forall x A] \dot{i} \cdot \Downarrow [\forall x A]^\perp} [\otimes]}{\vdash \mathcal{L}_{mlj} \dot{\circ} [\Gamma] \dot{i} [\Delta, \forall x A] \dot{i} \cdot \Downarrow [\forall x A]^\perp \otimes !^l \forall x ?^r [Ax]} [D_{\infty, \exists}]}{\vdash \mathcal{L}_{mlj} \dot{\circ} [\Gamma] \dot{i} [\Delta, \forall x A] \dot{i} \cdot \Downarrow [\forall x A]^\perp \otimes !^l \forall x ?^r [Ax]} [\forall, ?^r]}$$

Since $l \not\leq r$, all formulas in the context r should be weakened in the premise of the promotion rule. The derivation above also illustrates how one can specify fresh values with the use of the universal quantifier. As in mLJ , the eigenvariable c cannot appear in Δ nor Γ .

The following result is proved by induction on the height of focused proofs.

PROPOSITION 4.2

Let $\Gamma \cup \Delta$ be a set of object-logic formulas, and let the subexponentials l and r be specified by the signature $\langle \{\infty, l, r\}; \{l < \infty, r < \infty\}; \{\infty, l, r\} \rangle$. Then the sequent $\vdash \mathcal{L}_{mlj} \dot{\circ} [\Gamma] \dot{i} [\Delta] \dot{i} \cdot \uparrow$ is provable in $SELLF$ if and only if the sequent $\Gamma \longrightarrow \Delta$ is provable in mLJ .

4.3 LJQ^*

The systems in the previous sections always required two contexts. There are systems, however, that require more than two contexts to be specified, such as the focused multi-conclusion system for intuitionistic logic LJQ^* depicted in Figure 8. This system is a variant of the system proposed by Herbelin [14, page 78] and it was used by Dyckhoff & Lengrand in [8].

formula (\Box_R) has necessarily the following shape:

$$\frac{\frac{\frac{\frac{\vdash \mathcal{L}_{S4} \dot{\infty} [\Box\Gamma] \dot{\square}_L \cdot \dot{i} [\diamond\Delta] \dot{\diamond}_R [A] \dot{i} \cdot \dot{e} \cdot \dot{\uparrow} \cdot}{\vdash \mathcal{L}_{S4} \dot{\infty} [\Box\Gamma] \dot{\square}_L \cdot \dot{i} [\diamond\Delta] \dot{\diamond}_R \cdot \dot{i} \cdot \dot{e} \cdot \dot{\uparrow} \cdot ?^r [A]} \quad [!^e]}{\vdash \mathcal{K} \Downarrow [\Box A]^\perp} \quad [I_R] \quad \frac{\vdash \mathcal{L}_{S4} \dot{\infty} [\Box\Gamma] \dot{\square}_L [\Gamma', \diamond\Gamma''] \dot{i} [\diamond\Delta] \dot{\diamond}_R [\Delta', \Box\Delta''] \dot{i} \cdot \dot{e} \cdot \dot{\Downarrow} !^{e?^r} [A]}{\vdash \mathcal{L}_{S4} \dot{\infty} [\Box\Gamma] \dot{\square}_L [\Gamma', \diamond\Gamma''] \dot{i} [\diamond\Delta] \dot{\diamond}_R [\Delta', \Box\Delta''] \dot{i} \cdot \dot{e} \cdot \dot{\Downarrow} [\Box A]^\perp \otimes !^{e?^r} [A]} \quad [\otimes]}{\vdash \mathcal{L}_{S4} \dot{\infty} [\Box\Gamma] \dot{\square}_L [\Gamma', \diamond\Gamma''] \dot{i} [\diamond\Delta] \dot{\diamond}_R [\Delta', \Box\Delta''] \dot{i} \cdot \dot{e} \cdot \dot{\uparrow} \cdot} \quad [D_\infty, \exists]}$$

where \mathcal{K} is $\vdash \mathcal{L}_{S4} \dot{\infty} [\Box\Gamma] \dot{\square}_L [\Gamma', \diamond\Gamma''] \dot{i} [\diamond\Delta] \dot{\diamond}_R [\Delta', \Box\Delta''] \dot{i} \cdot \dot{e} \cdot \dot{\cdot}$. As one can easily check, the derivation above corresponds exactly to S4's rule \Box_R .

The following proposition can be proved by induction on the height of focused proofs.

PROPOSITION 4.4

Let $\Gamma \cup \Gamma' \cup \Gamma'' \cup \Delta \cup \Delta' \cup \Delta''$ be a set of object logic formulas, and let the subexponentials $l, r, \Box_L, \diamond_R, e$, and ∞ be specified by the signature

$$\langle \{l, r, \Box_L, \diamond_R, e, \infty\}, \{r < \diamond_R < \infty, l < \Box_L < \infty, e < \diamond_R, e < \Box_L\}, \{l, r, \Box_L, \diamond_R, e, \infty\} \rangle.$$

Then the sequent $\vdash \mathcal{L}_{S4} \dot{\infty} [\Box\Gamma] \dot{\square}_L [\Gamma', \diamond\Gamma''] \dot{i} [\diamond\Delta] \dot{\diamond}_R [\Delta', \Box\Delta''] \dot{i} \cdot \dot{e} \cdot \dot{\uparrow} \cdot$ is provable in *SELLF* if and only if the sequent $\Box\Gamma, \Gamma', \diamond\Gamma'' \vdash \diamond\Delta, \Delta', \Box\Delta''$ is provable in S4.

As a final remark, it is also possible to encode the proof system for intuitionistic S4, which only allows for at most one formula to be at the right-hand-side of sequents. The encoding is similar to the the encoding above for classical logic with the difference that it contains extra subexponential bangs for specifying this restriction on sequents, similar to what was done in our encoding of *G1m*. Formally, the encoding is based on the following subexponential signature with two dummy subexponentials e_l and e_r , where the former behaves as the one used in the encoding of classical logic, while the latter additionally checks that the context to the right-hand-side of sequents is empty:

$$\langle \{l, r, \Box_L, \diamond_R, e_l, e_r, \infty\}, \{r < \diamond_R < \infty, l < \Box_L < \infty, e_l < \diamond_R, e_l < \Box_L, e_r < \Box_L\}, \{l, \Box_L, \infty\} \rangle.$$

For instance, the introduction rule \Box_R shown below is specified by the clause $\exists A. [\Box A]^\perp \otimes !^{e_r} ?^r [A]$.

$$\frac{\Box\Gamma \longrightarrow A}{\Box\Gamma, \Gamma' \longrightarrow \Box A}$$

4.5 Lax Logic

Our next example is the encoding of the proof system for minimal *Lax* logic depicted in Figure 12. Its encoding illustrates the use of subexponentials to specify that a formula can only be introduced if a side-formula is present in the premise. An example of such a rule is the introduction rule for \bigcirc on the left. To introduce it on the left, the main connective of the formula on the right-hand-side must also be a \bigcirc . As we detailed next, we use subexponentials to perform such a check, without mentioning the formula on the right-hand-side, as described at the end of Section 3.2.

Consider the following signature $\langle \{l, r, \circ_r, \infty\}; \{r < \circ_r < l < \infty\}; \{l, \infty\} \rangle$. We will interpret an object-logic sequent of the forms $\Gamma \longrightarrow H$ and $\Gamma \longrightarrow \bigcirc G$ as the meta-level sequents,

$$\begin{array}{c}
\frac{\Gamma, A \wedge B, A, B \longrightarrow C}{\Gamma, A \wedge B \longrightarrow C} [\wedge L] \quad \frac{\Gamma \longrightarrow A \quad \Gamma \longrightarrow B}{\Gamma \longrightarrow A \wedge B} [\wedge R] \\
\frac{\Gamma, A \vee B, A \longrightarrow C \quad \Gamma, A \vee B, B \longrightarrow C}{\Gamma, A \vee B \longrightarrow C} [\vee L] \quad \frac{\Gamma \longrightarrow A_i}{\Gamma \longrightarrow A_1 \vee A_2} [\vee R_i] \\
\frac{\Gamma, A \supset B \longrightarrow A \quad \Gamma, A \supset B, B \longrightarrow C}{\Gamma, A \supset B \longrightarrow C} [\supset L] \quad \frac{\Gamma, A \longrightarrow B}{\Gamma \longrightarrow A \supset B} [\supset R] \\
\frac{\Gamma, \circ A, A \longrightarrow \circ B}{\Gamma, \circ A \longrightarrow \circ B} [\circ L] \quad \frac{\Gamma \longrightarrow A}{\Gamma \longrightarrow \circ A} [\circ R] \\
\frac{}{\Gamma, A \longrightarrow A} [\text{Init}] \quad \frac{\Gamma \longrightarrow A \quad \Gamma, A \longrightarrow C}{\Gamma \longrightarrow C} [\text{Cut}]
\end{array}$$

FIG. 12. The additive version of the proof system for minimal lax logics – Lax .

$$\begin{array}{ll}
(\wedge_l) \quad [A \wedge B]^\perp \otimes (?^l[A] \wp ?^l[B]) & (\wedge_r) \quad [A \wedge B]^\perp \otimes (!^l ?^r[A] \otimes !^l ?^r[B]) \\
(\vee_l) \quad [A \vee B]^\perp \otimes (?^l[A] \& ?^l[B]) & (\vee_r) \quad [A \vee B]^\perp \otimes (!^l ?^r[A] \oplus !^l ?^r[B]) \\
(\supset_l) \quad [A \supset B]^\perp \otimes (!^l ?^r[A] \otimes ?^l[B]) & (\supset_r) \quad [A \supset B]^\perp \otimes !^l (?^l[A] \wp !^l ?^r[B]) \\
(\circ_L) \quad [\circ A]^\perp \otimes !^{\circ_r} ?^l[A] & (\circ_R) \quad [\circ A]^\perp \otimes !^l ?^r[A] \\
(I) \quad [A]^\perp \otimes [A]^\perp & (\text{Cut}) \quad ?^l[A] \otimes !^l ?^r[A] \\
(\circ_S) \quad [\circ A]^\perp \otimes ?^{\circ_r}[A]
\end{array}$$

FIG. 13. The theory \mathcal{L}_{Lax} encoding the system Lax

respectively, $\vdash \mathcal{L}_{Lax} \overset{\circ}{\circ} [\Gamma] \overset{\circ}{\circ} \overset{\circ}{\circ} [H] \overset{\circ}{\circ} \overset{\circ}{\circ} \uparrow \cdot$ and $\vdash \mathcal{L}_{Lax} \overset{\circ}{\circ} [\Gamma] \overset{\circ}{\circ} \overset{\circ}{\circ} [\circ G] \overset{\circ}{\circ} \overset{\circ}{\circ} \uparrow \cdot$. That is, the context of the index l will contain all the formulas on the left-hand-side, while the formula to the right-hand side will either be in the context of r or the context of \circ_r . However, only object-level formulas whose main connective is \circ can be in the context of \circ_r . The encoding of the proof system Lax is given in Figure 13. As in the specification of S4, this is accomplished by using the following equivalence derived from the structural clause (\circ_S) :

$$\forall A. [\circ A] \equiv ?^{\circ_r}[A].$$

That is, one can move whenever needed a meta-level formula $[\circ A]$ to the context of \circ_r .

In the specification \mathcal{L}_{Lax} , the clause (\circ_L) is the most interesting one specifying the corresponding rule of the proof system. The $!^{\circ_r}$ specifies the restriction that the formula on the right must be marked with a \circ . This is illustrated by the following derivation:

$$\frac{\frac{\frac{\vdash \mathcal{L}_{Lax} \overset{\circ}{\circ} [\Gamma, \circ A, A] \overset{\circ}{\circ} \overset{\circ}{\circ} [\circ B] \overset{\circ}{\circ} \overset{\circ}{\circ} \uparrow \cdot}{\vdash \mathcal{L}_{Lax} \overset{\circ}{\circ} [\Gamma, \circ A] \overset{\circ}{\circ} \overset{\circ}{\circ} [\circ B] \overset{\circ}{\circ} \overset{\circ}{\circ} \uparrow ?^l[A]} [?^l]}{\vdash \mathcal{L}_{Lax} \overset{\circ}{\circ} [\Gamma, \circ A] \overset{\circ}{\circ} \overset{\circ}{\circ} \uparrow ?^{\circ_r}[A]} [!^{\circ_r}]}{\frac{\frac{\vdash \mathcal{L}_{Lax} \overset{\circ}{\circ} [\Gamma, \circ A] \overset{\circ}{\circ} \overset{\circ}{\circ} \downarrow [\circ A]^\perp}{\vdash \mathcal{L}_{Lax} \overset{\circ}{\circ} [\Gamma, \circ A] \overset{\circ}{\circ} \overset{\circ}{\circ} \downarrow !^{\circ_r} ?^l[A]} [\otimes]}{\vdash \mathcal{L}_{Lax} \overset{\circ}{\circ} [\Gamma, \circ A] \overset{\circ}{\circ} \overset{\circ}{\circ} \downarrow [\circ A]^\perp \otimes !^{\circ_r} ?^l[A]} [D_\infty, \exists]} [I]}{\vdash \mathcal{L}_{Lax} \overset{\circ}{\circ} [\Gamma, \circ A] \overset{\circ}{\circ} \overset{\circ}{\circ} \uparrow \cdot}$$

Notice that due to the $!^{\circ_r}$, the context of r must be empty. That is, the formula $[\circ B]$ must be in the context of \circ_r , or in other words the main connective of the object-logic formula to the right-hand-side is necessarily a \circ .

Notice as well that since $r < \circ_r$, the clause (\bigcirc_R) is admissible in the theory. That is, a formula can move from the context of \circ_r to the context of r . With respect to the proof system Lax this formula specifies exactly the rule \bigcirc_R , introducing the connective \bigcirc on the right. Therefore, in order to obtain a stronger level of adequacy, namely on the level of derivations [25], we mention it explicitly in the encoding.

The following proposition is proved by induction on the height of derivations.

PROPOSITION 4.5

Let $\Gamma \cup \{C\}$ be a set of object logic formulas, and let the subexponentials l, r and \circ_r be specified by the signature $\langle \{l, r, \circ_r, \infty\}; \{r < \circ_r < l < \infty\}; \{l, \infty\} \rangle$. Then the sequent $\vdash \mathcal{L}_{Lax} \dot{\circ} [\Gamma] \dot{i} \cdot \dot{\circ}_r [C] \dot{i} \cdot \dot{\uparrow} \cdot$ is provable in $SELLF$ if and only if the sequent $\Gamma \longrightarrow C$ is provable in Lax and the sequent $\vdash \mathcal{L}_{Lax} \dot{\circ} [\Gamma] \dot{i} [\bigcirc C] \dot{\circ}_r \cdot \dot{i} \cdot \dot{\uparrow} \cdot$ is provable in $SELLF$ if and only if the sequent $\Gamma \longrightarrow \bigcirc C$ is provable in Lax .

4.6 System G3K

For our last example, we illustrate the use of relations, $R(\cdot, \cdot)$, in the meta-level. We encode, in particular, the labelled proof system G3K proposed by Negri in [21], which is depicted in Figure 14. This is a powerful proof system that can be used for different modal logics, namely T, 4, B, S4, TB, S5, by simply specifying the properties of the relation R .

In order to encode this proof system, we specify one relation predicate $R(\cdot, \cdot)$ and use the two unbounded subexponential indexes l and r , whose contexts will store $[\cdot]$ and $[\cdot]$ atoms, respectively. As explained in Section 3.3, the relation predicate R also comes with its own unbounded subexponential index R . Thus, a G3K sequent of the form $a_1 R b_1, \dots, a_n R b_n, \Gamma \Rightarrow \Delta$ is encoded as the $SELLF$ sequent

$$\vdash \mathcal{L}_{G3k} \dot{\circ} \mathcal{R}, R(a_1, b_1), \dots, R(a_n, b_n) \dot{R} [\Gamma] \dot{i} [\Delta] \dot{i} \cdot \dot{\uparrow} \cdot$$

where $\mathcal{R} \subseteq \{(\text{Ref}), (\text{Sym}), (\text{Trans}), (\text{Eucl})\}$ is the theory, described in Section 3.3, specifying the relation R and \mathcal{L}_{G3k} is the theory specifying G3K's inference rules. The theory for G3K, \mathcal{L}_{G3k} , is depicted in Figure 15.

It is easy to check that the theory \mathcal{L}_{G3k} encodes adequately the proof system G3K in the level of derivations. For instance, the introduction rule for \Box_I is specified by the corresponding clause in \mathcal{L}_{G3k} , as illustrated by the derivation below:

$$\frac{\frac{\vdash \dot{\circ} \mathcal{R} \dot{R} \cdot \dot{i} \cdot \dot{\uparrow} \cdot R(a, b)^\perp}{\vdash \mathcal{L}_{G3k} \dot{\circ} \mathcal{R} \dot{R} [a : \Box A, \Gamma] \dot{i} [\Delta] \dot{i} \cdot \dot{\downarrow} !^R R(a, b)^\perp} [!^R] \quad \frac{\vdash \mathcal{L}_{G3k} \dot{\circ} \mathcal{R} \dot{R} [a : \Box A, b : A, \Gamma] \dot{i} [\Delta] \dot{i} \cdot \dot{\uparrow} \cdot}{\vdash \mathcal{L}_{G3k} \dot{\circ} \mathcal{R} \dot{R} [a : \Box A, \Gamma] \dot{i} [\Delta] \dot{i} \cdot \dot{\downarrow} ?^I [b : A]} [R \downarrow, ?^I]}{\Xi \quad \frac{\vdash \mathcal{L}_{G3k} \dot{\circ} \mathcal{R} \dot{R} [a : \Box A, \Gamma] \dot{i} [\Delta] \dot{i} \cdot \dot{\downarrow} (!^R R(a, b)^\perp \otimes ?^I [b : A])}{\vdash \mathcal{L}_{G3k} \dot{\circ} \mathcal{R} \dot{R} [a : \Box A, \Gamma] \dot{i} [\Delta] \dot{i} \cdot \dot{\downarrow} [a : \Box A]^\perp \otimes \exists y. (!^R R(a, y)^\perp \otimes ?^I [y : A])} [\otimes, \exists]}{\vdash \mathcal{L}_{G3k} \dot{\circ} \mathcal{R} \dot{R} [a : \Box A, \Gamma] \dot{i} [\Delta] \dot{i} \cdot \dot{\uparrow} \cdot} [D_\infty, \exists]}$$

where Ξ is a derivation containing only an initial rule. Notice that in the left-premise, the formula $R(a, b)^\perp$ is provable only if it is provable using the formulas in \mathcal{R} .

Repeating this exercise with the other introduction rules, we can easily prove the following proposition:

PROPOSITION 4.6

Let $\Gamma \cup \{\Delta\}$ be a set of object logic formulas, let $\mathcal{R} \subseteq \{(\text{Ref}), (\text{Sym}), (\text{Trans}), (\text{Eucl})\}$ be a set of relation clauses, let $\mathcal{R}' = \{R(a_1, b_1), \dots, R(a_n, b_n)\}$ be a set of relations on labels, and let

$$\begin{array}{c}
\frac{}{x : A, \Gamma \Rightarrow \Delta, x : A} [I] \quad \frac{\Gamma \Rightarrow \Delta, x : A \quad x : A, \Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta} [Cut] \\
\frac{x : A, x : B, \Gamma \Rightarrow \Delta}{x : A \& B, \Gamma \Rightarrow \Delta} [\&_l] \quad \frac{\Gamma \Rightarrow \Delta, x : A \quad \Gamma \Rightarrow \Delta, x : B}{\Gamma \Rightarrow \Delta, x : A \& B} [\&_r] \\
\frac{x : A, \Gamma \Rightarrow \Delta \quad x : B, \Gamma \Rightarrow \Delta}{x : A \vee B, \Gamma \Rightarrow \Delta} [\vee_l] \quad \frac{\Gamma \Rightarrow \Delta, x : A, x : B}{\Gamma \Rightarrow \Delta, x : A \vee B} [\vee_r] \\
\frac{\Gamma \Rightarrow \Delta, x : A \quad x : B, \Gamma \Rightarrow \Delta}{x : A \supset B, \Gamma \Rightarrow \Delta} [\supset_l] \quad \frac{x : A, \Gamma \Rightarrow \Delta, x : B}{\Gamma \Rightarrow \Delta, x : A \supset B} [\supset_r] \\
\frac{}{x : \perp, \Gamma \Rightarrow \Delta} [\perp_l] \\
\frac{y : A, x : \Box A, xRy, \Gamma \Rightarrow \Delta}{x : \Box A, xRy, \Gamma \Rightarrow \Delta} [\Box_l] \quad \frac{xRy, \Gamma \Rightarrow \Delta, y : A}{\Gamma \Rightarrow \Delta, x : \Box A} [\Box_r] \\
\frac{xRy, y : A, \Gamma \Rightarrow \Delta}{x : \Diamond A, \Gamma \Rightarrow \Delta} [\Diamond_l] \quad \frac{xRy, \Gamma \Rightarrow \Delta, x : \Diamond A, y : A}{xRy, \Gamma \Rightarrow \Delta, x : \Diamond A} [\Diamond_r]
\end{array}$$

FIG. 14. The system G3K.

$$\begin{array}{ll}
(\&_l) \quad [x : A \& B]^\perp \otimes (?^l[x : A] \wp ?^l[x : B]) & (\&_r) \quad [x : A \& B]^\perp \otimes (?^r[x : A] \otimes ?^r[x : B]) \\
(\vee_l) \quad [x : A \vee B]^\perp \otimes (?^l[x : A] \otimes ?^l[x : B]) & (\vee_r) \quad [x : A \vee B]^\perp \otimes (?^r[x : A] \wp ?^r[x : B]) \\
(\supset_l) \quad [x : A \supset B]^\perp \otimes (?^r[x : A] \otimes ?^l[x : B]) & (\supset_r) \quad [x : A \supset B]^\perp \otimes (?^l[x : A] \wp ?^r[x : B]) \\
(\Box_l) \quad [x : \Box A]^\perp \otimes \exists y. (!^R R(x, y)^\perp \otimes ?^l[y : A]) & (\Box_r) \quad [x : \Box A]^\perp \otimes \forall y. (?^R R(x, y) \wp ?^r[y : A]) \\
(\Diamond_l) \quad [x : \Diamond A]^\perp \otimes \forall y. (?^R R(x, y) \wp ?^l[y : A]) & (\Diamond_r) \quad [x : \Diamond A]^\perp \otimes \exists y. (!^R R(x, y)^\perp \otimes ?^r[y : A]) \\
(I) \quad [x : A]^\perp \otimes [x : A]^\perp & (Cut) \quad ?^r[x : A] \otimes ?^l[x : A] \\
(\perp_l) \quad [x : \perp]^\perp &
\end{array}$$

FIG. 15. Specification \mathcal{L}_{G3k} in *SELLF* for G3K.

the subexponentials l, r and R be specified by the signature $\langle \{l, r, R, \infty\}; \{r, l < \infty\}; \{l, r, R, \infty\} \rangle$. Then the sequent $\vdash \mathcal{L}_{G3k} \dot{\circ} \mathcal{R}, R(a_1, b_1), \dots, R(a_n, b_n) \dot{\circ} \dot{R} [\Gamma] \dot{\circ} [\Delta] \dot{\circ} \uparrow \cdot$ is provable in *SELLF* if and only if the sequent $a_1 R b_1, \dots, a_n R b_n, \Gamma \Rightarrow \Delta$ is provable in G3K using the corresponding rules for the relation R .

5 Reasoning about Sequent Calculus

This section presents general and effective criteria for checking whether a proof system encoded in *SELLF* has important proof-theoretic properties, namely, cut-elimination, invertibility of rules, and the completeness of atomic identity rules. Instead of proving each one of these properties from scratch, we just need to check whether the specification of a proof system satisfies the corresponding criteria. Moreover, we show that checking such criteria can be easily automated.

5.1 Cut-elimination for cut-coherent systems

The rule *Cut* is often presented as the rule below

$$\frac{\Gamma_1 \vdash \Delta_1, A \quad A, \Gamma_2 \vdash \Delta_2}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2} \text{ [Cut]}$$

where $\Gamma_1, \Gamma_2, \Delta_1, \Delta_2$ may be sets or multisets of formulas. The formula A is called the cut-formula. A proof system is said to have the cut-elimination property when the cut rule is admissible on this system, *i.e.*, every proof that uses cuts can be transformed into a cut-free proof. There are at least two important consequences of the cut-elimination theorem, namely the *sub-formula property* and the *consistency* of the proof system. Cut-elimination was first proved by Gentzen [12] for proof systems for classical (LK) and intuitionistic logic (LJ). Gentzen's proof strategy has been re-used to prove the cut-elimination of a number of proof systems. The proof is quite elaborated and it involves a number of cases, thus being exhaustive and error prone. The strategy can be summarized by the following steps:

1. (Reduction to Principal Cuts) Transforming a proof with cuts into a proof with *principal cuts*, that is, a cut whose premises are derived by introducing the cut-formula itself. This is normally shown by permuting inference rules, *e.g.*, permuting the cut-rule over other introduction rules.
2. (Reduction to Atomic Cuts) Transforming a proof with principal cuts into a proof with atomic cuts. This is normally shown by reducing a cut with a complex cut-formula into (possible many) cuts with simpler cut-formulas.
3. (Elimination of Atomic Cuts) Transforming a proof with atomic cuts into a cut-free proof. This is normally shown by permuting atomic cuts over other introduction rules until it reaches the leaves and it is erased.

We provide a criterion for each one of the steps above. The step two is not problematic. In particular, a criterion for reducing principal cuts to atomic cuts was given by Pimentel and Miller in [19] when encoding systems in linear logic. This criterion easily extends to the use of *SELLF* (see Definition 5.6 and Theorem 5.8).

While for specifications in linear logic steps one and three did not cause any problems [19], for specifications in *SELLF* they do not work as smoothly. For the step three of eliminating atomic cuts, however, we could still find a simple criterion for when this step can be performed (see Definition 5.9 and Theorem 5.10). But determining criteria for when it is possible to transform arbitrary cuts into principal cuts (step one) turned out to be a real challenge. This is expected as *SELLF* allows for much more complicated proof systems to be encoded, such as *mLJ* and *LJQ**, with the strongest level of adequacy. There are at least three possible strategies or reductions one can use to perform this transformation:

- (Permute Cut Rules Upwards) As done by Gentzen, one can try to permute cuts over other introduction rules. The following is an example of such a transformation in *G1m*:

$$\frac{\Gamma \multimap A \quad \frac{\Gamma', A, F \multimap G}{\Gamma', A \multimap F \supset G} [\supset_R]}{\Gamma, \Gamma' \multimap F \supset G} [\text{Cut}] \quad \rightsquigarrow \quad \frac{\Gamma \multimap A \quad \frac{\Gamma', A, F \multimap G}{\Gamma, \Gamma', F \multimap G} [\text{Cut}]}{\Gamma, \Gamma' \multimap F \supset G} [\supset_R]$$

We identify a criterion for when such permutations are always possible (see Lemma 5.2).

- (Permute Introduction Rules Downwards) In some cases, it is not possible to permute the cut over an introduction rule. For instance, in the *mLJ* derivation to the left, it is not always possible to permute a cut over an \supset_R , because such a permutation would weaken the formulas in Δ , which may be needed in the proof of left premise of the cut rule.

$$\frac{\frac{\frac{\Gamma, A, B, F \longrightarrow G}{\Gamma, A \wedge B, F \longrightarrow G} [\wedge_L]}{\Gamma, A \wedge B \longrightarrow F \supset G, \Delta} [\supset_R]}{\Gamma \longrightarrow F \supset G, \Delta} [\text{Cut}] \rightsquigarrow \frac{\frac{\frac{\Gamma, A, B, F \longrightarrow G}{\Gamma, A, B \longrightarrow F \supset G, \Delta} [\supset_R]}{\Gamma, A \wedge B \longrightarrow F \supset G, \Delta} [\wedge_L]}{\Gamma \longrightarrow F \supset G, \Delta} [\text{Cut}]$$

The strategy then is to permute downwards the rule introducing the cut-formula ($A \wedge B$) on the Cut's right premise, as illustrated by the derivation to the right. In some cases, however, the cut-formula might need to be introduced multiple times. For instance, in the following S4 derivation, the cut cannot permute upwards, but one can still introduce the cut-formula $\Box A$ on the right before introducing the formula $\Box F$. Only, in this case, the cut-formula is introduced twice, as illustrated by the derivation to the right.¹⁰

$$\frac{\frac{\frac{\frac{\Box\Gamma, \Box A, A \vdash \diamond\Delta, F}{\Box\Gamma, \Box A \vdash \diamond\Delta, F} [\Box_L]}{\Box\Gamma, \Gamma', \Box A \vdash \diamond\Delta, \Delta', \Box F} [\Box_R]}{\Box\Gamma, \Gamma' \vdash \diamond\Delta, \Delta', \Box F} [\text{Cut}]}{\Box\Gamma, \Gamma' \vdash \diamond\Delta, \Delta', \Box F} \rightsquigarrow \frac{\frac{\frac{\frac{\Box\Gamma, \Box A, A \vdash \diamond\Delta, F}{\Box\Gamma, \Box A \vdash \diamond\Delta, F} [\Box_L]}{\Box\Gamma, \Gamma', \Box A, A \vdash \diamond\Delta, \Delta', \Box F} [\Box_R]}{\Box\Gamma, \Gamma', \Box A \vdash \diamond\Delta, \Delta', \Box F} [\Box_L]}{\Box\Gamma, \Gamma' \vdash \diamond\Delta, \Delta', \Box F} [\text{Cut}]$$

A similar case also appears in *mLJ*, *e.g.*, when the cut formula is $A \supset B$. We identify criteria for when an introduction rule can permute over another introduction rule (see Lemma 5.4), which handles the cases for *mLJ* and S4 illustrated above.

- (Transform one Cut into Another Cut) There are systems, such as *LJQ**, which have more than one cut rule. For instance, *LJQ** has eight different cut rules, three of them shown in Example 5.3. In these cases, for permuting a cut of one type over an introduction rule might involve transforming this cut into another type of cut. As these permutations involve more elaborated proof transformations, finding criteria that is not ad-hoc to one system is much more challenging (if not impossible) and we will not provide one here.

We start our discussion of cut-elimination on specified sequent systems by the permutability step (step one). For this purpose, we define the notion *permutation of clauses* and then establish criteria for permutation of cut and introduction clauses.

DEFINITION 5.1

Given C_1 and C_2 clauses in a canonical proof system theory \mathcal{X} , we say that C_1 *permutes over* C_2 if, given an arbitrary focused proof π of a sequent S ending with a bipole derivation introducing C_2 followed by a bipole derivation introducing C_1 , then there exists a focused proof π' of S ending with a bipole derivation introducing C_1 followed by a bipole derivation introducing C_2 .

LEMMA 5.2 (Criteria cut permutation)

Let \mathcal{X} be a canonical proof system theory. A cut clause permutes over an introduction or structural clause $C \in \mathcal{X}$ if, for each $s, t \in I$ such that $!^s B$ appears in C and $?^t B'$ is a subformula of the monopole B , one of the following holds:¹¹

¹⁰This problem of permuting cuts in the system S4 was emphasized by Stewart and Stouppa in [32] and the complete proof can be found in [17].

¹¹Of course, if the subexponential $!^s$ is not present in C , then the restrictions on s don't apply.

1. $Cut = \exists A. !^a ?^b [A] \otimes !^c ?^d [A]$ and either:
 - i. permutation by vacuously: $s \not\leq b$ and b is bounded; or $s \not\leq d$ and d is bounded;
 - ii. permutation to the right: $s \leq a, d$ and $c \leq t$;
 - iii. permutation to the left: $s \leq b, c$ and $a \leq t$;
2. $Cut = \exists A. !^a ?^b [A] \otimes ?^d [A]$ and either:
 - i. permutation by vacuously: $s \not\leq b$ and b is bounded; or $s \not\leq d$ and d is bounded;
 - ii. permutation to the right: $s \leq a, d$;
3. $Cut = \exists A. ?^b [A] \otimes !^c ?^d [A]$ and either:
 - i. permutation by vacuously: $s \not\leq b$ and b is bounded; or $s \not\leq d$ and d is bounded;
 - ii. permutation to the left: $s \leq b, c$;
4. $Cut = \exists A. ?^b [A] \otimes ?^d [A]$ and either:
 - i. permutation by vacuously: $s \not\leq b$ and b is bounded; or $s \not\leq d$ and d is bounded;
 - ii. permutation to the right or left: s is the least element of $\langle I, \leq \rangle$.

PROOF. Suppose that C is a formula of the shape $!^s ?^t B^{12}$.

- Case $Cut = \exists A. !^a ?^b [A] \otimes !^c ?^d [A]$. Consider the proof:

$$\frac{\frac{\frac{\Xi_1}{\vdash \mathcal{K}_1 \leq_a +_b [A] : \cdot \uparrow \cdot} \quad \frac{!^a, ?^b}{\vdash \mathcal{K}_1 : \cdot \downarrow !^a ?^b [A]}}{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 : \cdot \downarrow !^a ?^b [A] \otimes !^c ?^d [A]} \quad \frac{\frac{\frac{\Xi'_2}{\vdash \mathcal{K}_2 \leq_{c,s} +_d [A] +_t B : \cdot \uparrow \cdot} \quad \frac{!^s, ?^t}{\vdash \mathcal{K}_2 \leq_c +_d [A] : \cdot \downarrow !^s ?^t B}}{\vdash \mathcal{K}_2 \leq_c +_d [A] : \cdot \uparrow \cdot} \quad \frac{!^c, ?^d}{\vdash \mathcal{K}_2 : \cdot \downarrow !^c ?^d [A]}}{\vdash \mathcal{K}_2 \otimes \mathcal{K}_1 : \cdot \downarrow !^c ?^d [A] \otimes !^a ?^b [A]} \quad \frac{[D_\infty]}{[D_\infty, \exists]}}{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 : \cdot \uparrow} \quad [D_\infty, \exists]$$

If $s \not\leq d$ and d is bounded this case will not happen and the permutation is by vacuously. Otherwise, if $s \leq d$, $s \leq a$ and $c \leq t$, the proof above can be replaced by

$$\frac{\frac{\frac{\Xi_1}{\vdash \mathcal{K}_1 \leq_{s,a} +_b [A] : \cdot \uparrow \cdot} \quad \frac{!^a, ?^b}{\vdash \mathcal{K}_1 \leq_s : \cdot \downarrow !^a ?^b [A]}}{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 \leq_s +_t B : \cdot \downarrow !^a ?^b [A] \otimes !^c ?^d [A]} \quad \frac{\frac{\frac{\Xi'_2}{\vdash \mathcal{K}_2 \leq_{s,c} +_t B +_d [A] : \cdot \uparrow \cdot} \quad \frac{!^c, ?^d}{\vdash \mathcal{K}_2 \leq_s +_t B : \cdot \downarrow !^c ?^d [A]}}{\vdash \mathcal{K}_2 \leq_s +_t B : \cdot \uparrow \cdot} \quad \frac{[D_\infty]}{[D_\infty, \exists]}}{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 \leq_s +_t B : \cdot \uparrow \cdot} \quad \frac{!^s, ?^t}{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 : \cdot \downarrow !^s ?^t B} \quad \frac{[D_\infty]}{[D_\infty]}$$

Notice that, since $s \leq a$, $\mathcal{K}_1 \leq_{s,a} = \mathcal{K}_1 \leq_a$. Hence, in this case, the permutation is to the right. The same reasoning can be done for the left premise.

¹²In fact, we should consider bipoles D containing subformulas of the form $!^s C$ with C a monopole, but we will present only the case where $D = !^s ?^t B$ for readability purposes.

- Case $Cut = !^a\gamma^b[B] \otimes ?^d[B]$. If $s \leq d$ and $s \leq a$, then the derivation

$$\frac{\frac{\frac{\Xi_1}{\vdash \mathcal{K}_1 \leq_a +_b[A] : \cdot \uparrow \cdot} \quad \vdash \mathcal{K}_1 : \cdot \downarrow !^a\gamma^b[A]}{[!^a, \gamma^b]} \quad \frac{\frac{\frac{\Xi'_2}{\vdash \mathcal{K}_2 \leq_s +_d[A] +_t B : \cdot \uparrow \cdot} \quad \vdash \mathcal{K}_2 +_d[A] : \cdot \downarrow !^s\gamma^t B} {D_\infty} \quad \frac{\vdash \mathcal{K}_2 +_d[A] : \cdot \uparrow \cdot} {[\gamma^d]} \quad \frac{\vdash \mathcal{K}_2 : \cdot \downarrow ?^d[A]}{[\otimes]}}{[\otimes]} \quad \frac{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 : \cdot \downarrow !^a\gamma^b[A] \otimes ?^d[A]}{[D_\infty, \exists]}}{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 : \cdot \uparrow} [D_\infty, \exists]$$

can be replaced by

$$\frac{\frac{\frac{\Xi_1}{\vdash \mathcal{K}_1 \leq_{s,a} +_b[A] : \cdot \uparrow \cdot} \quad \vdash \mathcal{K}_1 \leq_s : \cdot \downarrow !^a\gamma^b[A]}{[!^a, \gamma^b]} \quad \frac{\frac{\Xi'_2}{\vdash \mathcal{K}_2 \leq_s +_t B +_d[A] : \cdot \uparrow \cdot} \quad \vdash \mathcal{K}_2 \leq_s +_t B : \cdot \downarrow ?^d[A]}{[\gamma^d]} \quad \frac{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 \leq_s +_t B : \cdot \downarrow !^a\gamma^b[A] \otimes ?^d[A]}{[D_\infty, \exists]}}{[\otimes]} \quad \frac{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 \leq_s +_t B : \cdot \uparrow \cdot} {[\otimes]} \quad \frac{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 : \cdot \downarrow !^s\gamma^t B} {[\otimes]} \quad \frac{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 : \cdot \uparrow} {[D_\infty]} [D_\infty]$$

There is a very interesting observation in this case: the restrictions for permuting the cut clause over the left premise form a superset of the restrictions for the right premise. In fact, other the fact that s should be the least element of I , it should also be the case that $a \leq t$. That is, if the permutation is possible at all, it can be always done over the right premise. Finally, if $s \not\leq d$ and d is bounded, then focusing over $!^s\gamma^t B$ is not possible at all (the same for the left premise).

- Case $Cut = ?^b[B] \otimes !^c\gamma^d[B]$. Analogous to the last case.
- Case $Cut = ?^b[B] \otimes ?^d[B]$. If s is the least element of I , then the derivation

$$\frac{\frac{\frac{\Xi_1}{\vdash \mathcal{K}_1 +_b[A] : \cdot \uparrow \cdot} \quad \vdash \mathcal{K}_1 : \cdot \downarrow ?^b[A]}{[\gamma^b]} \quad \frac{\frac{\Xi'_2}{\vdash \mathcal{K}_2 \leq_s +_d[A] +_t B : \cdot \uparrow \cdot} \quad \vdash \mathcal{K}_2 +_d[A] : \cdot \downarrow !^s\gamma^t B} {D_\infty} \quad \frac{\vdash \mathcal{K}_2 +_d[A] : \cdot \uparrow \cdot} {[\gamma^d]} \quad \frac{\vdash \mathcal{K}_2 : \cdot \downarrow ?^d[A]}{[\otimes]}}{[\otimes]} \quad \frac{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 : \cdot \downarrow ?^b[A] \otimes ?^d[A]}{[D_\infty, \exists]}}{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 : \cdot \uparrow} [D_\infty, \exists]$$

can be replaced by¹³

$$\frac{\frac{\frac{\Xi_1}{\vdash \mathcal{K}_1 \leq_s +_b[A] : \cdot \uparrow \cdot} \quad \vdash \mathcal{K}_1 \leq_s : \cdot \downarrow ?^b[A]}{[\gamma^b]} \quad \frac{\frac{\Xi'_2}{\vdash \mathcal{K}_2 \leq_s +_t B +_d[A] : \cdot \uparrow \cdot} \quad \vdash \mathcal{K}_2 \leq_s +_t B : \cdot \downarrow ?^d[A]}{[\gamma^d]} \quad \frac{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 \leq_s +_t B : \cdot \downarrow ?^b[A] \otimes ?^d[A]}{[D_\infty, \exists]}}{[\otimes]} \quad \frac{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 \leq_s +_t B : \cdot \uparrow \cdot} {[\otimes]} \quad \frac{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 : \cdot \downarrow !^s\gamma^t B} {[\otimes]} \quad \frac{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 : \cdot \uparrow} {[D_\infty]} [D_\infty]$$

¹³Observe that the permutation could be done also on the left premise.

EXAMPLE 5.3

Note that, from the systems presented in Section 4, the cuts defined in systems $G1m$ and Lax permutes over any introduction or structural clause. This means that, for these systems, the classical argument of permuting cuts up the proof until getting principal cuts works fine.

The cut clause for the mLJ system ($Cut_{mLJ} = \exists A. ?^l[A] \otimes ?^r[A]$), on the other hand, does not permute over clauses (\supset_R) and (\forall_R) , since $!^l$ is present in both clauses but neither r is bounded (while $l \not\leq r$) nor l is the least element in the signature $\langle \{\infty, l, r\}; \{l \leq \infty, r \leq \infty\}; \{\infty, l, r\} \rangle$. This captures well, at the meta-level, the fact that the cut rule does not permute over the rules (\supset_R) and (\forall_R) at the object-level.

In the same way, in $S4$, the cut clause $Cut_{S4} = \exists A. ?^l[A] \otimes ?^r[A]$ does not permute over the clauses (\square_R) and (\diamond_L) since l, r are unbounded and e is not the least element of the signature

$$\langle \{l, r, \square_L, \diamond_R, e, \infty\}, \{r \leq \diamond_R \leq \infty, l \leq \square_L \leq \infty, e \leq \diamond_R, e \leq \square_L\}, \{l, r, \square_R, \diamond_R, \infty\} \rangle.$$

In LJQ^* , three cut rules are admissible¹⁴:

$$\frac{\Gamma_1 \rightarrow A; \Delta_1 \quad A, \Gamma_2 \rightarrow B; \Delta_2}{\Gamma_1, \Gamma_2 \rightarrow B; \Delta_1, \Delta_2} [Cut_1] \quad \frac{\Gamma_1 \rightarrow A; \Delta_1 \quad A, \Gamma_2 \vdash \Delta_2}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2} [Cut_2]$$

$$\frac{\Gamma_1 \vdash \Delta_1, A \quad A, \Gamma_2 \vdash \Delta_2}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2} [Cut_3]$$

The first rule cannot be encoded in $SELLF$ using only bipoles with the signature presented in this paper. In fact, we would need to add “dummy” subexponentials for guaranteeing the presence of focused formulas on the context, more or less the same way done for the Lax logic. The other cut rules can be specified, respectively, by the clauses

$$(Cut_2) \quad !^r ?^l[A] \otimes !^r ?^f[A] \quad (Cut_3) \quad !^r ?^l[A] \otimes !^r ?^r[A].$$

It is interesting to note that, in Cut_2 , the permutation to the right is by vacuously with *every* clause in the system. And it should be so since, at the object level, the left premise of the Cut_2 rule has a focused right cut formula, which *must be* principal. Hence the cut rule cannot permute up in the object level, as the cut clause does not permute over any other clause of the system. For the permutation to the left, the conditions $s \leq l$ and $s \leq r$ and $r \leq t$ for any clause of the form $!^s B(\cdots ?^t B')$ appearing in \mathcal{L}_{ljq} , implies that: $s = r$ and $t = r, l$. Hence the Cut_2 clause permutes to the left over (\supset_L) , (\forall_L) , (\wedge_L) and (\forall_R) and it does not permute over (\supset_R) and (\wedge_R) . As it should be since, at the object level, the premises of the rule (\wedge_R) are focused and, as already discussed for the system mLJ , the rule (\supset_R) erases formulas of the premises, hence not permuting with the cut rule.

For the Cut_3 clause, the argument is similar to the one just presented and Cut_3 does not permute to the right or to the left with (\supset_R) and (\wedge_R) , permuting over the other introduction clauses of the system. As said before, the cut-elimination process for LJQ^* is more involving, making use of exchange between cuts, and it will not be discussed in more details here.

The following lemma establishes criteria for checking when a clause permutes over another clause. It captures all the non-trivial permutations for the systems mLJ and $S4$, that is, all the cases that are not true by vacuously.

¹⁴In fact, there are five admissible cut rules in LJQ^* , but the other two are derived from those presented here. And it is also worthy to note that there are three non-admissible cut rules in LJQ^* .

bottom¹⁵.

$$\begin{array}{c}
 \Xi \\
 \frac{\dots \frac{\frac{\vdash \mathcal{K} \leq_s +_t A +_u B : \cdot \uparrow \cdot}{\vdash \mathcal{K} \leq_s +_t A : \cdot \uparrow ?^u B} [\text{?}^u] \dots}{\vdash \mathcal{K} \leq_s +_t A : \cdot \uparrow B_1} [\text{!}^s] \dots}{\vdash \mathcal{K} +_t A : \cdot \downarrow !^s B_1} [\text{!}^s] \dots}{\vdash \mathcal{K} +_t A : \cdot \downarrow C_1} [\text{?}^t] \dots}{\vdash \mathcal{K} : \cdot \uparrow ?^t A} [\text{?}^t] \dots} \\
 \frac{\vdash \mathcal{K} : \cdot \downarrow C_2}{\vdash \mathcal{K} : \cdot \uparrow} [D_\infty]
 \end{array}$$

For when $s \not\leq t$, then the introduction of $!^s$ will cause the weakening of the formula A . However, if $q(\diamond(x_1, \dots, x_n)) \equiv ?^v q(\diamond(x_1, \dots, x_n))$, where $q(\diamond(x_1, \dots, x_n))$ is the formula used by C_2 and $s \leq v$, then there is a derivation where $q(\diamond(x_1, \dots, x_n))$ is not weakened by the introduction of $!^s$. Hence, it is possible to focus on C_2 again after focusing on C_1 and recover the formula A . ■

Observe that this last lemma is much more involved than Lemma 5.2. In fact, the cut clause is a formula with no negated atomic formulas, and what it roughly does is to split the context into two and add a left formula in one part and a right formula in the other. When permuting two introduction clauses, on the other hand, one has to be careful not erasing contexts that will be necessary for the application of the next clause. For instance, the atomic formulas needed for introducing the clause C_1 can be in a context that will be eventually erased by the clause C_2 , hence the permutation cannot happen.

As said before, our main interest on permuting clauses is to be able to consider only object-level *principal cuts*. We clarify this concept better now. Let \mathcal{X} be a canonical system and Ξ be a *SELLF* proof of the sequent $\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 : \cdot \uparrow \cdot$ ending with an introduction of the *Cut* clause. The premise of that decide rule is the conclusion of an $[\exists]$ infer rule. Let A be the substitution term used to instantiate the existential quantifier. We say that this occurrence of the $[D_\infty]$ inference rule is an *object-level cut* with *cut formula* A . Suppose $A = \diamond(\bar{B})$ is a non-atomic object level formula with left and right introduction rules

$$\exists \bar{x}[\diamond(\bar{x})]^\perp \otimes B_l \quad \text{and} \quad \exists \bar{x}[\diamond(\bar{x})]^\perp \otimes B_r$$

We say that this introduction of the *Cut* clause is *principal* if Ξ has the form

$$\frac{\frac{\frac{\Xi_1}{\vdash \mathcal{K}_1 \leq_a +_b[\diamond(\bar{B})] : \cdot \downarrow B_l[\bar{B}/\bar{x}]} [D_\infty, \exists, \otimes, I]}{\vdash \mathcal{K}_1 \leq_a +_b[\diamond(\bar{B})] : \cdot \uparrow \cdot} [\text{!}^a, ?^b]}{\vdash \mathcal{K}_1 : \cdot \downarrow !^a ?^b[\diamond(\bar{B})]} [\otimes] \quad \frac{\frac{\Xi_2}{\vdash \mathcal{K}_2 \leq_c +_d[\diamond(\bar{B})] : \cdot \downarrow B_r[\bar{B}/\bar{x}]} [D_\infty, \exists, \otimes, I]}{\vdash \mathcal{K}_2 \leq_c +_d[\diamond(\bar{B})] : \cdot \uparrow \cdot} [\text{!}^c, ?^d]}{\vdash \mathcal{K}_2 : \cdot \downarrow !^c ?^d[\diamond(\bar{B})]} [\otimes]} \\
 \frac{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 : \cdot \downarrow !^a ?^b[\diamond(\bar{B})] \otimes !^c ?^d[\diamond(\bar{B})]}{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 : \cdot \uparrow} [D_\infty, \exists]$$

DEFINITION 5.5

Let \mathcal{X} be a canonical proof system theory. We say that \mathcal{X} is *cut-principal* if every proof Ξ of a sequent S of the form $\vdash \mathcal{K} : \Delta \uparrow \cdot$, with $\mathcal{K}[\infty] = \mathcal{X}$, having an introduction of a *Cut*

¹⁵Since $s \leq t$, the introduction of $!^s$ does not cause the weakening of the formula A .

clause, can be transformed, using permutations over clauses, into a proof Ξ' of S where that introduction of the *Cut* clause is principal.

Hence, for example, the systems *G1m* and *Lax* are cut-principal, since their cut clauses permutes over any other clause of the system. A straightforward case analysis shows that *mLJ* and *S4* also have this property: when cuts cannot permute up, rules can permute down, making the cuts principal.

Once we can transform an introduction of a cut into a principal one, the proof of cut elimination for logical systems continues by showing how to transform a principal cut into cuts with “simpler” formulas. This transformation is often based on the fact that systems have “dual” introduction rules for each connective. In [19], Pimentel and Miller introduced the concept of cut-coherence for linear logic specifications that captures this notion of duality. We extend this definition to our setting with subexponentials.

DEFINITION 5.6

Let \mathcal{X} be a canonical proof system theory and \diamond an object-level connective of arity $n \geq 0$. Furthermore, let the formulas

$$\exists \bar{x}([\diamond(\bar{x})]^\perp \otimes B_l) \quad \text{and} \quad \exists \bar{x}([\diamond(\bar{x})]^\perp \otimes B_r)$$

be the left and right introduction rules for \diamond , where the free variables of B_l and B_r are in the list of variables \bar{x} . The object-level connective \diamond has *cut-coherent introduction rules* if the sequent $\vdash \mathcal{K}_\infty : \cdot \uparrow \forall \bar{x}(B_l^\perp \wp B_r^\perp)$ is provable in *SELLF*, where $\mathcal{K}_\infty[\infty] = \{\text{Cut}\}$, $\{\text{Cut}\}$ is the set of all cut clauses in \mathcal{X} and $\mathcal{K}_\infty[i] = \emptyset$ for any other $i \in I$. A canonical proof system theory is called *cut-coherent* if all object-level connectives have cut-coherent introduction rules.

EXAMPLE 5.7

The cut-coherence of the *G1m* specification is established by proving the following sequents.

$$\begin{aligned} (\supset) & \vdash \text{Cut}_{G1m} \overset{\cdot}{\circlearrowleft} \cdot \overset{\cdot}{\dot{\cdot}} \cdot \overset{\cdot}{\dot{\cdot}} \cdot \overset{\cdot}{\dot{\cdot}} \cdot \uparrow \text{?!}^r[A]^\perp \wp \text{!}^l[B]^\perp, \text{?!}^l(\text{!}^l[A]^\perp \otimes \text{!}^r[B]^\perp) \\ (\wedge) & \vdash \text{Cut}_{G1m} \overset{\cdot}{\circlearrowleft} \cdot \overset{\cdot}{\dot{\cdot}} \cdot \overset{\cdot}{\dot{\cdot}} \cdot \overset{\cdot}{\dot{\cdot}} \cdot \uparrow \text{!}^l[A]^\perp \otimes \text{!}^l[B]^\perp, \text{?!}^r[A]^\perp \wp \text{?!}^r[B]^\perp \\ (\vee) & \vdash \text{Cut}_{G1m} \overset{\cdot}{\circlearrowleft} \cdot \overset{\cdot}{\dot{\cdot}} \cdot \overset{\cdot}{\dot{\cdot}} \cdot \overset{\cdot}{\dot{\cdot}} \cdot \uparrow \text{!}^l[A]^\perp \oplus \text{!}^l[B]^\perp, \text{?!}^r[A]^\perp \& \text{?!}^r[B]^\perp \\ (\forall) & \vdash \text{Cut}_{G1m} \overset{\cdot}{\circlearrowleft} \cdot \overset{\cdot}{\dot{\cdot}} \cdot \overset{\cdot}{\dot{\cdot}} \cdot \overset{\cdot}{\dot{\cdot}} \cdot \uparrow \text{!}^l[Bx]^\perp, \text{?!}^r \exists x. \text{!}^r[Bx]^\perp \\ (\exists) & \vdash \text{Cut}_{G1m} \overset{\cdot}{\circlearrowleft} \cdot \overset{\cdot}{\dot{\cdot}} \cdot \overset{\cdot}{\dot{\cdot}} \cdot \overset{\cdot}{\dot{\cdot}} \cdot \uparrow \exists x. \text{!}^l[Bx]^\perp, \text{?!}^r[Bx]^\perp \end{aligned}$$

All these sequents have simple proofs. In general, deciding whether or not canonical systems are cut-coherent involves a simple algorithm (see Theorem 5.11).

Intuitively, the notion of cut-coherence on the meta-level corresponds to the property of reducing the complexity of a cut on the object-level. If a connective \diamond is proven to have cut-coherent introduction rules, then a cut with formula $\diamond(\bar{x})$ can be replaced by simpler cuts using the operations of reductive cut-elimination, until atomic cuts are reached. This is proved by Theorem 5.8.

We need the following definition specifying cuts with atomic cut formulas only.

$$ACut = \exists A. \text{Cut}(A) \otimes \text{atomic}(A).$$

The following theorem establishes the condition for when there are reductions from proof with principal cuts to proofs with atomic cuts.

THEOREM 5.8

Let the disjoint union $\mathcal{X} \cup \{Cut\}$ be a principal, cut-coherent proof system whose structural clauses are all contextual. If $\vdash \mathcal{K} : \cdot \uparrow \cdot$ is provable, then $\vdash A\mathcal{K} : \cdot \uparrow \cdot$ is provable where $\mathcal{K}[\infty] = \mathcal{X} \cup \{Cut\}$ and $A\mathcal{K}[\infty] = \mathcal{X} \cup \{ACut\}$.

PROOF. (Sketch – see [19] for the detailed proof.) The proof of this theorem follows the usual line of replacing cuts on general formulas for cuts on atomic formulas for first-order logic, being careful about the subexponentials. Let Ξ be a proof of the sequent $\vdash \mathcal{K} : \cdot \uparrow \cdot$ ending with an object-level cut over a cut formula $\diamond(\bar{B})$ with left and right introduction rules

$$\exists \bar{x}([\diamond(\bar{x})]^\perp \otimes B_l) \quad \text{and} \quad \exists \bar{x}([\diamond(\bar{x})]^\perp \otimes B_r)$$

Since \mathcal{X} is cut-principal, there exist proofs of $\vdash \mathcal{K}_1 : \cdot \uparrow B_l$ and $\vdash \mathcal{K}_2 : \cdot \uparrow B_r$, where $\mathcal{K} = \mathcal{K}_1 \otimes \mathcal{K}_2$. Since \mathcal{X} is a cut-coherent proof system theory the sequent $\vdash \mathcal{K}_\infty : \cdot \uparrow \forall \bar{x}(B_l^\perp \wp B_r^\perp)$ is provable. Thus, the following three sequents all have cut-free proofs in $SELL$ ¹⁶:

$$\vdash \mathcal{K}_1, B_l[\bar{B}/\bar{x}] \quad \vdash \mathcal{K}_2, B_r[\bar{B}/\bar{x}] \quad \vdash \text{?}^\infty Cut, B_l[\bar{B}/\bar{x}]^\perp, B_r[\bar{B}/\bar{x}]^\perp$$

By using two instances of $SELL$ cut, we can conclude that¹⁷

$$\vdash \mathcal{K}_1, \mathcal{K}_2$$

has a proof with cut. Applying the cut-elimination process for $SELL$ will yield a cut-free $SELL$ proof of the same sequent. Observe that the elimination process can only instantiate eigenvariables of the proof with “simpler” formulas, hence the sizes of object-level cut formulas in the resulting cut-free meta-level proof does not increase. Using the completeness of $SELL$ in $SELLF$ we know that

$$\vdash \mathcal{K} : \cdot \uparrow \cdot$$

has a proof of smaller object-level cuts and the result follows by induction. ■

The condition that all structural rules are contextual (see Definition 3.2) is necessary as otherwise known non-trivial cases involving weakening and contraction would arise. Since these are not solved by the duality of connectives, it is out of the scope of the theorem above. Despite this limitation, this step of cut-elimination can be proved for wide of range of proof system. In fact, the only system shown above that has structural rules that are not contextual is $G1m$.

The last step in Gentzen’s cut-elimination strategy is to eliminate atomic cuts by permuting them upwards. However, as in the transformation of proofs with cuts into proofs with principal cuts only, the subexponential bangs may disallow that atomic cuts can be eliminated. A further restriction on cut clauses is needed.

DEFINITION 5.9

Let \mathcal{X} be a principal, cut-coherent proof system theory. We say that a cut clause $Cut = \exists A. !^a ?^b [A] \otimes !^c ?^d [A]$ is *weak* if for all $s, t \in I$ such that $?^s[\cdot], ?^t[\cdot]$ appears in \mathcal{X} , $b \leq s$ and $d \leq t$.

\mathcal{X} is called *weak cut-coherent* if, for all $Cut \in \mathcal{X}$, Cut is weak.

¹⁶By abuse of notation, we will represent the contexts in $SELLF$ and its translation in $SELL$ using the same symbol.

¹⁷Reminding that $Cut \in \mathcal{K}[\infty]$.

THEOREM 5.10

Let the disjoint union $\mathcal{X} \cup \{ACut\}$ be a weak cut-coherent proof system. Let $\Gamma_o \longrightarrow \Delta_o$ be an object-level sequent and $\vdash \mathcal{K} : \cdot \uparrow \cdot$ be its *SELLF* encoding, where $\mathcal{K}[\infty] = \mathcal{X} \cup \{ACut\}$. If $\vdash \mathcal{K} : \cdot \uparrow \cdot$ is provable, then $\vdash \mathcal{K}' : \cdot \uparrow \cdot$ is provable where $\mathcal{K}'[\infty] = \mathcal{X}$ and $\mathcal{K}'[i] = \mathcal{K}'[i]$ for any other $i \in I$.

PROOF. The usual proof that permutes an atomic cut up in a proof can be applied here (since the system is principal). Any occurrence of an instance of $[D_\infty]$ on the *ACut* formula can be moved up in a proof until it can either be dropped entirely or until one of the premises is proved by an instance of $[D_\infty]$ on the *Init*.¹⁸

$$\frac{\frac{\frac{\frac{\frac{\vdash \mathcal{K}_1 \leq_a +_b [A] : \cdot \uparrow \cdot}{\vdash \mathcal{K}_1 : \cdot \downarrow !^a \gamma^b [A]} \quad [!^a, \gamma^b]}{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 : \cdot \downarrow !^a \gamma^b [A] \otimes !^c \gamma^d [A]} \quad [\otimes]}{\vdash \mathcal{K} : \cdot \uparrow \cdot} \quad [D_\infty, \exists]}{\frac{\frac{\frac{\frac{\frac{\frac{\vdash \mathcal{K}_2^1 : \cdot \downarrow [A]^\perp \quad \vdash \mathcal{K}_2^2 : \cdot \downarrow [A]^\perp}{\vdash \mathcal{K}_2 \leq_c +_d : \cdot \downarrow [A]^\perp \otimes [A]^\perp} \quad [\otimes]}{\vdash \mathcal{K}_2 \leq_c +_d [A] : \cdot \uparrow \cdot} \quad [D_\infty, \exists]}{\vdash \mathcal{K}_2 : \cdot \downarrow !^c \gamma^d [A]} \quad [!^c, \gamma^d]}{\vdash \mathcal{K}_1 \otimes \mathcal{K}_2 : \cdot \downarrow !^a \gamma^b [A] \otimes !^c \gamma^d [A]} \quad [\otimes]}{\vdash \mathcal{K} : \cdot \uparrow \cdot} \quad [D_\infty, \exists]}$$

In that case, there must exist an index s such that $[A] \in \mathcal{K}_2^2[s]$. If $b \leq s$, then we can substitute the proof of the conclusion of the cut inference above by the proof Ξ (similar to the right case). Hence the result holds for weak cut-coherent systems. \blacksquare

The next result states that to check whether or not a proof system encoding is weak cut-coherent is decidable. See [19] for a similar proof.

THEOREM 5.11

Determining whether or not a canonical proof system is weak cut-coherent is decidable. In particular, determining if the cut clause proves the duality of the introduction rules for a given connective can be achieved by proof search in *SELLF* bounded by the depth $v + 2$ where v is the maximum number of premise atoms in the bodies of the introduction clauses.

We can develop a general method for checking whether a proof system encoded in *SELLF* admits cut-elimination by putting all these results together. The first step is to use Lemma 5.2 to check for which clauses the cut permutes over. Then for each remaining clause, C , check using Lemma 5.4, the introduction/structural clauses of the system permutes over C . After this step one is reduced with the non-trivial cases for when the transformation of a proof with cuts into a proof with atomic cuts only is not straightforward and must be proved individually. We then check whether the theory is cut-coherent, which from Theorem 5.8, implies that principal cuts can be reduced to atomic cuts. This check requires bounded proof search as described in Theorem 5.11. Finally, we check whether atomic cuts can be eliminated by checking whether the theory is weak cut-coherent. We have implemented this method, as well as the checking for atomic identities, as detailed in Section 6.

5.2 Atomic Identities

The notion of cut-coherence implies that non-atomic principle cuts can be replaced by simpler ones. We now consider the dual problem of replacing initial axioms with its atomic version. The discussion bellow is pretty much similar to the ideas presented in [19].

¹⁸Here A is an atomic object level formula.

DEFINITION 5.12

Let \mathcal{X} be a canonical proof system theory and \diamond an object-level connective of arity $n \geq 0$. Furthermore, let the formulas

$$\exists \bar{x}(\lfloor \diamond(\bar{x}) \rfloor^\perp \otimes B_l) \quad \text{and} \quad \exists \bar{x}(\lceil \diamond(\bar{x}) \rceil^\perp \otimes B_r)$$

be the left and right introduction rules for \diamond , where the free variables of B_l and B_r are in the list of variables \bar{x} . The object-level connective \diamond has *initial-coherent introduction rules* if the sequent $\vdash \mathcal{K}_\infty : \cdot \uparrow \forall \bar{x} (?^\infty B_l \wp ?^\infty B_r)$ is provable in *SELLF*, where $\mathcal{K}_\infty[\infty] = \{\text{Init}\}$ and $\mathcal{K}_\infty[i] = \emptyset$ for any other $i \in I$. A canonical proof system theory is called *initial-coherent* if all object-level connectives have initial-coherent introduction rules.

It is easy to see that determining initial-coherency is simple and that initial coherency does not imply cut-coherency (and vice-versa). In general, we take both of these coherence properties together.

DEFINITION 5.13

A cut-coherent theory that is also initial-coherent is called a *coherent theory*.

PROPOSITION 5.14

Let \mathcal{X} be a coherent theory and \diamond an object-level connective of arity $n \geq 0$. Furthermore, let the formulas

$$\exists \bar{x}(\lfloor \diamond(\bar{x}) \rfloor^\perp \otimes B_l) \quad \text{and} \quad \exists \bar{x}(\lceil \diamond(\bar{x}) \rceil^\perp \otimes B_r)$$

be the left and right introduction rules for \diamond . Then B_r and B_l are dual formulas in *SELLF*.

PROOF. From the definition of cut-coherent, B_l entails B_r in a theory containing $\{\text{Cut}\}$. Similarly, from the definition of initial-coherence, B_r entails B_l in a theory containing Init . Thus, the equivalence $B_r \equiv B_l$ is provable in a theory containing $\{\text{Cut}\}$ and Init . Hence B_r and B_l are duals. \blacksquare

Finally, the next theorem states that, in coherent systems, the initial rule can be restricted to its atomic version. For this theorem, we need to axiomatize the meta-level predicate *atomic*(\cdot). This axiomatization can be achieved by collecting into the theory Δ all formulas of the form $\exists \bar{x} : (\text{atomic}(p(x_1; \dots; x_n)))^\perp$ for every predicate of the object logic.

For the next theorem, we also need the following definition

$$A\text{Init} = \exists A. \text{Init}(A) \otimes \text{atomic}(A).$$

THEOREM 5.15

Given an object level formula B , let $\text{Init}(B)$ denote the formula $\lfloor B \rfloor^\perp \otimes \lceil B \rceil^\perp$, let Δ be the theory that axiomatizes the meta-level predicate *atomic*(\cdot), $\mathcal{X} \cup \text{Init}$ be a coherent proof theory and $\mathcal{K}_\infty = \{\mathcal{X}, A\text{Init}, \Delta\}$. Then the sequent $\vdash \mathcal{K}_\infty : \cdot \uparrow \text{Init}(B)$ is provable.

5.3 Invertibility of rules

Another property that has been studied in the sequent calculus setting is the invertibility of rules. We say that a rule is invertible if the provability of the conclusion sequent implies the provability of all the premises.

This property is of interest to proof search since invertible rules permute down with the other rules of a proof, reducing hence proof-search non-determinism. In particular, in systems

$$\begin{array}{ll}
(\Rightarrow L) & [A \Rightarrow B]^\perp \otimes ?^r[A] \& ?^l[B] & (\Rightarrow R) & [A \Rightarrow B]^\perp \otimes ?^l[A] \wp ?^r[B] \\
(\wedge L) & [A \wedge B]^\perp \otimes ?^l[A] \wp ?^l[B] & (\wedge R) & [A \wedge B]^\perp \otimes ?^r[A] \& ?^r[B] \\
(\vee R) & [A \vee B]^\perp \otimes ?^r[A] \wp ?^r[B] & (\vee L) & [A \vee B]^\perp \otimes ?^l[A] \& ?^l[B]
\end{array}$$

FIG. 16. Specification of G3c.

with only invertible rules, the bottom-up search for a proof can stop as soon as a non provable sequent is reached.

For example, it is well known that all rules in G3c (see [33]) are invertible. This system is specified in Figure 16. Observe that the meta level connectives in the bodies are negative. Therefore, its introduction rule is specified using only invertible focused rules. The following is a straightforward result, as all the connectives appearing in a monopole are negative.

THEOREM 5.16

A monopole introduction clause corresponds to an invertible object level rule.

6 Implementation

We have implemented a tool that takes a *SELLF* specification of a proof system and checks automatically whether the proof system admits cut-elimination and whether the system with atomic initials is complete. Note that this checking ensures sufficient but not necessary conditions, thus a negative answer does not mean that the system doesn't enjoy the cut-elimination property. Our tool is implemented in OCaml and there is an online version with some examples at <http://www.logic.at/people/giselle/tatu>. The specification of proof systems is done as described in Section 3. In particular, the clauses specifying a proof system are separated into four parts: introduction clauses, structural clauses, cut clauses, and the identity clauses.

The tool also contains the machinery necessary for checking the conditions described in Section 5. It implements the static analysis described in Lemmas 5.2 and 5.4. As detailed at the end of Section 5.1, the tool determines cases for when the cut rule can permute over other introduction rules and for when an introduction rule permutes over another introduction rule. Whenever some clauses of the encoding does not satisfy such criteria, then it outputs an error message. Detecting corner cases can be useful for detecting design flaws in the specification of a proof system. For the systems *G1m* and *Lax*, our tool was able to check that indeed a proof with cuts can be transformed into a proof with principal cuts only. For the other systems, it identified some permutations by vacuously that it could not prove automatically. However, these can be easily checked manually.

For checking whether an encoding is cut-coherent, our tool performs bounded proof search, where the bound is determined as described in Theorem 5.11. In order to handle the problem of context splitting during proof search, our tool implements the lazy splitting detailed in [4] for linear logic. The method easily extends to *SELLF*. Another difference, however, is that our system is one-sided classical logic. Therefore, we do not implement the back-chaining style proof search used in [4], but rather proof search based on the focused discipline described in Section 2. Furthermore, as previously mentioned, proof search is bounded by the height of derivations, measured by the number of decide rules. This is enough for checking whether an encoding is cut-coherent. In a similar fashion, the tool also checks by using bounded proof search whether the encoded proof system is complete when using atomic initial rules by checking whether the system is initial coherent (see Definition 5.12). For all the examples

that we have implemented, our tool checks all the conditions described above in less than a second.

7 Related Work

The present work has its foundations on the works [25, 19] by Miller, Nigam, and Pimentel, where plain linear logic was used as the framework for specifying sequent systems, and reasoning about them. The motivation for the generalization proposed here was based initially on the fact that there are a number of proof systems that can be encoded *SELLF* but cannot be encoded in the same declarative fashion (such as without mentioning side-formulas) in linear logic without subexponentials. Moreover, the encodings in [19] are only on the level of proofs and not on the level of derivations [25]. Therefore, proving adequacy in [19] involves more complicated techniques than the simple proofs by induction on the height of focused derivations used here. Finally, when trying to deal with the verification using *SELLF*, we ended up being able to propose more general conditions for permutation of clauses, which enabled more general criteria for proving cut-elimination of systems.

It turns out that specification and verification of proof systems is a very important branch of the proof theory field. In fact, there exists a number of works willing to provide adequate tools for dealing with systems in a general and yet natural way, making it possible then to use the rich meta-theory proposed in order to reason about the specifications. For example, Pfenning proposed a method of proving cut-elimination [28] from specifications in intuitionistic logic. This method has been applied to a number of proof systems and implemented by using the theorem prover Twelf [29]. For instance, the encoding of *Lax* logic and its cut-elimination proof can be found at http://twelf.org/wiki/Lax_logic. It happens that this procedure is only semi-automated, in the sense that, for any given proof system, one has to prove all the permutation lemmas and reductions needed in the cut-elimination from scratch.

In the present paper, we adopted a more uniform approach, establishing general criteria to the specification for proving properties of the specified systems. Since we are dealing with classical linear logic (where negation is involutive), our encodings never mention side-formulas, only the principal formulas of the rules. Such declarative specifications produce not only clean and natural encodings, but it also allows for easy meta-level reasoning.

Ciabattoni and Terui in [6] have proposed a general method for extracting cut-free sequent calculus proof systems from Hilbert style proof systems. Their method can be used for a number of non-trivial logics, including intuitionistic linear logic extended with knotted structural rules. The main difference to our work is that they do not provide a decision criteria for when a system falls into their framework. On the other hand, we do not provide means to encode Hilbert style proof systems. It seems that our methods are complementary and can be combined, so to enable the specification of Hilbert style proof systems as well as reason over them. Thus, the challenges of integrating these methods have still to be investigated.

Checking whether a rule permutes over another was also topic of the recent work [15]. As in our approach, Lutovac and Harland investigate syntactic conditions which allow to check the validity of such permutations. A number of cases of permutations and examples are provided. A main difference to our approach is that we fixed the specification language, namely *SELLF*, to specify inference rules and proof systems, whereas [15] does not make such commitment. On one hand, we can only reason about systems “specifiable” in *SELLF*, but on the other hand, the use a logical framework allows for the construction of a general tool that can check for permutations automatically. It is not yet clear how one could construct

a similar tool using the approach in [15]. On the other hand, Nigam, Reis and Lima used [27] logic programming techniques to check which rule of a proof system specified in *SELLF* permute over another rule. This study refines the permutation lemmas we established in this paper.

8 Conclusions and Future Work

In this paper, we showed that it is possible to specify a number of non-trivial structural properties by using subexponential connectives. In particular, we demonstrated that it is possible to specify proof systems whose sequents have multiple contexts that are treated as multisets or sets. Moreover, it is possible to specify inference rules that require some formulas to be weakened and inference rules that require some side-formula to be present in its conclusion. We have also introduced the machinery for checking whether encoded proof systems have three important properties, namely the admissibility of the cut rule, the completeness of atomic identity rules, and the invertibility of rules. Finally, we have also build an implementation that automatically checks some of these criteria.

There are a number of directions to follow from this work. One direction would be investigating the role of the polarity of atomic meta-level formulas in the specification of proof systems using *SELLF*. Nigam and Miller [25] showed in a linear logic setting that a number of proof systems can be faithfully encoded by playing with the polarity of atomic formulas. Here, we assigned to all atomic formulas a negative polarity, but this choice is not enforced by the completeness of the focusing strategy (see [20]). In fact, a different (global) assignment for atoms could be chosen. However, to use such a technique here would imply a change on the definition of bipoles, as with the current definition polarities would play a very limited role because all atomic formulas are in the scope of a subexponential question-mark. We are investigating alternative definitions, so that we can still use subexponentials in a sensible way and at the same time play with the polarity of atomic formulas.

Another direction to be pursued would be analyzing different classes of systems other than the canonical ones, such as to handle quasi-canonical systems, *e.g.*, paraconsistent C-systems [2] or Dyckhoff and Negri's contraction-free *G4ip* [9]. It seems possible to encode the paraconsistent systems in *SELLF* with the strongest level of adequacy (adequacy of derivations), while it seems only possible to encode *G4ip* with weaker levels of adequacy. In both cases, however, the encodings are not-canonical and therefore the methods developed here are not applicable. It would be interesting to develop more flexible conditions for checking whether such systems that are not canonical admit cut-elimination. This task is left to future work.

Finally, we could apply the notion of coherency in order to build the so called "harmonical systems" [11]. In fact, harmonicity comes for free in coherent systems, due to Theorems 5.10 and 5.15. Furthermore, since we can also handle labelled systems, one could use all the machinery developed here in order to prove nice properties on harmonical mathematical systems such as those containing geometric theories [22].

Acknowledgments We thank Dale Miller, Agata Ciabattoni, Roy Dyckhoff and the anonymous reviewers for their comments. We are also grateful to Kaustuv Chaudhuri for pointing out an error in a previous version of this paper. Finally, Nigam was funded by the Alexander von Humboldt foundation and Nigam and Pimentel were funded by CNPq, the Brazilian National Counsel of Technological and Scientific Development.

References

- [1] Jean-Marc Andreoli. Logic programming with focusing proofs in linear logic. *J. of Logic and Computation*, 2(3):297–347, 1992.
- [2] Arnon Avron, Beata Konikowska, and Anna Zamansky. Cut-free sequent calculi for c-systems with generalized finite-valued semantics. *J. Log. Comput.*, 23(3):517–540, 2013.
- [3] Arnon Avron and Iddo Lev. Canonical propositional Gentzen-type systems. In R. Goré, A. Leitsch, and T. Nipkow, editors, *IJCAR 2001*, volume 2083 of *LNAI*, pages 529–544. Springer, 2001.
- [4] Iliano Cervesato, Joshua S. Hodas, and Frank Pfenning. Efficient resource management for linear logic proof search. *Theoretical Computer Science*, 232(1-2):133–163, 2000.
- [5] Alonzo Church. A formulation of the simple theory of types. *J. of Symbolic Logic*, 5:56–68, 1940.
- [6] Agata Ciabattoni, Nikolaos Galatos, and Kazushige Terui. From axioms to analytic rules in nonclassical logics. In *23th Symp. on Logic in Computer Science*, pages 229–240. IEEE Computer Society Press, 2008.
- [7] Vincent Danos, Jean-Baptiste Joinet, and Harold Schellinx. The structure of exponentials: Uncovering the dynamics of linear logic proofs. In Georg Gottlob, Alexander Leitsch, and Daniele Mundici, editors, *Kurt Gödel Colloquium*, volume 713 of *LNCS*, pages 159–171. Springer, 1993.
- [8] R. Dyckhoff and S. Lengrand. LJQ: a strongly focused calculus for intuitionistic logic. In A. Beckmann and *et al.*, editors, *Computability in Europe 2006*, volume 3988 of *LNCS*, pages 173–185. Springer, 2006.
- [9] Roy Dyckhoff and Sara Negri. Admissibility of structural rules for contraction-free systems of intuitionistic logic. *J. of Symbolic Logic*, 65(4):1499–1518, 2000.
- [10] M. Fairtlough and M.V. Mendler. Propositional lax logic. *Information and Computation*, 137(1):1–33, August 1997.
- [11] Nissim Francez and Roy Dyckhoff. A note on harmony. *J. Philosophical Logic*, 41(3):613–628, 2012.
- [12] Gerhard Gentzen. Investigations into logical deduction. In M. E. Szabo, editor, *The Collected Papers of Gerhard Gentzen*, pages 68–131. North-Holland, Amsterdam, 1969. Translation of articles that appeared in 1934–35.
- [13] Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.
- [14] Hugo Herbelin. *Séquents qu'on calcule: de l'interprétation du calcul des séquents comme calcul de lambda-termes et comme calcul de stratégies gagnantes*. PhD thesis, Université Paris 7, 1995.
- [15] Tatjana Lutovac and James Harland. A contribution to automated-oriented reasoning about permutability of sequent calculi rules. *Computer Science and Information Systems*, 10, 2013.
- [16] S. Maehara. Eine darstellung der intuitionistischen logik in der klassischen. *Nagoya Mathematical Journal*, pages 45–64, 1954.
- [17] Simone Martini and Andrea Masini. A modal view of linear logic. *Journal of Symbolic Logic*, 59(3):888–899, 1994.
- [18] Dale Miller and Elaine Pimentel. Using linear logic to reason about sequent systems. In Uwe Egly and Christian G. Fermüller, editors, *International Conference on Automated Reasoning with Analytic Tableaux and Related Methods*, volume 2381 of *LNCS*, pages 2–23. Springer, 2002.
- [19] Dale Miller and Elaine Pimentel. A formal framework for specifying sequent calculus proof systems. *Theor. Comput. Sci.*, 474:98–116, 2013.
- [20] Dale Miller and Alexis Saurin. From proofs to focused proofs: a modular proof of focalization in linear logic. In J. Duparc and T. A. Henzinger, editors, *CSL 2007: Computer Science Logic*, volume 4646 of *LNCS*, pages 405–419. Springer, 2007.
- [21] Sara Negri. Proof analysis in modal logic. *Journal of Philosophical Logic*, 34(5-6):507–544, 2005.
- [22] Sara Negri and Jan von Plato. *Proof Analysis: A Contribution to Hilbert's Last Problem*. Cambridge University Press, 2011.
- [23] Vivek Nigam. *Exploiting non-canonicity in the sequent calculus*. PhD thesis, Ecole Polytechnique, September 2009.
- [24] Vivek Nigam and Dale Miller. Algorithmic specifications in linear logic with subexponentials. In *ACM SIG-PLAN Conference on Principles and Practice of Declarative Programming (PPDP)*, pages 129–140, 2009.
- [25] Vivek Nigam and Dale Miller. A framework for proof systems. *J. of Automated Reasoning*, 45(2):157–188, 2010.
- [26] Vivek Nigam, Elaine Pimentel, and Giselle Reis. Specifying proof systems in linear logic with subexponentials. In *Proceedings of the Fifth Logical and Semantic Frameworks, with Applications Workshop (LSFA 2010)*, volume 269 of *ENTCS*, pages 109–123, 2011.

- [27] Vivek Nigam, Giselle Reis, and Leonardo Lima. Checking proof transformations with asp. *TPLP*, 13(4-5-Online-Supplement), 2013.
- [28] Frank Pfenning. Structural cut elimination. In *10th Symp. on Logic in Computer Science*, pages 156–166, San Diego, California, 1995. IEEE Computer Society Press.
- [29] Frank Pfenning and Carsten Schürmann. System description: Twelf — A meta-logical framework for deductive systems. In H. Ganzinger, editor, *16th Conf. on Automated Deduction (CADE)*, number 1632 in LNAI, pages 202–206, Trento, 1999. Springer.
- [30] Elaine Pimentel and Dale Miller. On the specification of sequent systems. In *LPAR 2005: 12th International Conference on Logic for Programming, Artificial Intelligence and Reasoning*, number 3835 in LNAI, pages 352–366, 2005.
- [31] Elaine Gouvêa Pimentel. *Lógica linear e a especificação de sistemas computacionais*. PhD thesis, Universidade Federal de Minas Gerais, Belo Horizonte, M.G., Brasil, December 2001. Written in English.
- [32] Charles Stewart and Phiniki Stouppa. A systematic proof theory for several modal logics. In *Advances in Modal Logic*, pages 309–333, 2004.
- [33] Anne S. Troelstra and Helmut Schwichtenberg. *Basic Proof Theory*. Cambridge University Press, 1996.