Hypersequents and Systems of Rules: Embeddings and Applications

AGATA CIABATTONI, TU Wien
FRANCESCO A. GENCO, TU Wien

We define a bi-directional embedding between hypersequent calculi and a subclass of systems of rules (2-systems). In addition to showing that the two proof frameworks have the same expressive power, the embedding allows for the recovery of the benefits of locality for 2-systems, analyticity results for a large class of such systems, and a rewriting of hypersequent rules as natural deduction rules.

CCS Concepts: • **Theory of computation** → *Proof theory*;

Additional Key Words and Phrases: Hypersequents, Systems of Rules, Natural Deduction, Intermediate Logics, Embedding between formalisms

ACM Reference format:
Agata Ciabattoni and Francesco A. Genco. 000. Hypersequents and Systems of Rules: Embeddings and Applications. *ACM Trans. Comput. Logic* 000, 000, Article 000 (000), 27 pages.
https://doi.org/0000001.0000001

1 INTRODUCTION

The multitude and diversity of formalisms introduced to define analytic calculi has made it increasingly important to identify their interrelationships and relative expressive power. *Embeddings* between formalisms, i.e. functions that take any calculus in some formalism and yield a calculus for the same logic in another formalism, are useful tools to prove that a formalism subsumes another one in terms of expressiveness – or, when bi-directional, that two formalisms are equi-expressive. Such embeddings can also provide useful reformulations of known calculi and allow the transfer of certain proof-theoretic results, thus alleviating the need for independent proofs in each system and avoiding duplicating work. Various embeddings between formalisms have appeared in the literature, see, e.g., [11, 13, 20, 21, 23, 24, 26] (and the bibliography thereof).

In this paper we introduce a bi-directional embedding between the hypersequent formalism [3] and a fragment of the system of rules formalism [19]. Hypersequents are a well-studied generalisation of sequents successfully employed to introduce analytic proof systems for large classes of non-classical logics, see, e.g., [4, 5, 8, 15, 16]. Systems of rules have been recently introduced in [19] as a very expressive but complex formalism capable, for example, of capturing all normal modal logics formalised by Sahlqvist formulae. A system of rules consists of different (labelled) sequent rules connected by conditions on the order of their applicability. Hence, derivations containing instances of such systems are non-local objects, unlike hypersequent derivations.

Supported by projects FWF START Y544-N23, WWTF MA16-028 and FWF W1255-N23.

1529-3785/000/000-ART000 $15.00
https://doi.org/0000001.0000001

Non-locality here has two different but closely related roles: (i) to avoid as much bureaucracy as possible in the representation of proofs, and (ii) to capture more logics.

Ad (i): Natural deduction [12] is a traditional example of a formalism relying exclusively on formulae and non-local effects, such as hypotheses discharge, to construct proofs. This is particularly useful when investigating, e.g., the computational content of proofs via a Curry–Howard correspondence [14], but might complicate the search for and manipulation of proofs. Sequent [12] and Hypersequent calculi, by contrast, have been designed precisely with the aim to avoid any form of non-locality. Locality guarantees indeed a tighter control over proofs, thus making local proof-systems easier to implement and to use for proving properties of the formalised logics. The price to pay is to deal with more complex basic objects, e.g., derivability assertions (sequents) and their parallel composition (hypersequents).

Ad (ii): The role of non-locality to increase the expressive power of formalisms is demonstrated in [19], where the use of systems of labelled rules allows the definition of modular analytic calculi for (modal and intermediate) logics whose frame conditions are beyond the geometric fragment [18].

The system of rules formalism combines the bookkeeping machinery of (labelled) sequent calculus with a generalised version of the discharging mechanism of natural deduction. More precisely, a system of rules is a set of rules that can only be applied in a certain order and possibly share metavariables for formulae or sets of formulae. The word "system" is used in the same sense as in linear algebra, where there are systems of equations with variables in common, and each equation is meaningful and can be solved only if considered together with the other equations of the system. Consider for example the following system of sequent rules:

$$\dfrac{\begin{array}{cc} \dfrac{\Sigma, \Gamma_1 \Rightarrow \Pi_1}{\Gamma_1 \Rightarrow \Pi_1}\ (s) & \dfrac{}{\Sigma \Rightarrow}\ (t) \\ \vdots & \vdots \\ \Gamma \Rightarrow \Pi & \Gamma \Rightarrow \Pi \end{array}}{\Gamma \Rightarrow \Pi}\ (r)$$

Here (s) and (t) can only be applied above the premisses of the rule (r) and must share the metavariable Σ. Hence the application of (r) discharges the occurrences of (s) and (t).

The non-locality of systems of rules is twofold: it is horizontal, because of the dependency between rules occurring in disjoint branches; and vertical, because of rules that can only be applied above other rules.

A possible connection between hypersequents and systems of rules is hinted in [19]. Following [9] this paper formalises and proves this intuition. Focusing on propositional logics intermediate between intuitionistic and classical logic, we define a bi-directional *embedding* between hypersequents and a subclass of systems of sequent rules (*2-systems*) in which the vertical non-locality is restricted to at most two (non labelled) sequent rules. Our embeddings show that these two seemingly different extensions of the sequent calculus have the same expressive power, arising from non-local conditions for 2-systems and from bookkeeping mechanisms for hypersequents.

From the embedding into hypersequents, 2-systems have the practical gain of very general analyticity results. Recall indeed that analyticity (i.e. the subformula property) is shown in [19] for systems of rules sharing only variables or atomic formulae; while this restriction does not yield any loss of generality in the context of labelled sequents, it does for systems of rules operating on non-labelled sequents, e.g., defined with the aim of directly capturing Hilbert axioms [8]. Moreover the embedding enables the introduction of new cut-free 2-systems.

The bonds unveiled by the embeddings between hypersequents and 2-systems extend further, leading to a rewriting of the former as natural deduction systems. As observed, e.g. in [7], this rewriting is a crucial step to formalise and prove the intuition in [4] that the intermediate logics possessing analytic hypersequent calculi might give rise to correponding parallel λ-calculi. The close relation beween systems of rules and natural deduction enables us to define simple and modular natural deduction calculi for a large class of intermediate logics. The calculi are obtained by extending Gentzen natural deduction calculus NJ by new rules. Similarly to sequent rules belonging to systems, these rules can discharge other rule applications, i.e. they are *higher-level rules* (see e.g., [25]). The results in [2] for the natural deduction calculus in Example 5.3 for Gödel logic – one of the best known intermediate logics – demonstrate the usefulness of our approach for Curry–Howard correspondences.

The article is structured as follows: Section 2 recalls the notions of hypersequent and system of rule; the translations between systems of rules and hypersequent rules are presented in Section 3; Section 4 contains the embeddings between derivations: Section 4.1 the direction from system of rules to hypersequent derivations and Section 4.2 the inverse direction. Sections 4.1.1 and 4.2.1 introduce normal forms for derivations containing systems of rules and hypersequents, respectively. The final section describes the applications of the embedding, which include the definition of new natural deduction calculi for a large class of intermediate logics.

The present paper extends [9] in several ways: it shows how to use the embedding to obtain natural deduction calculi, it contains full proofs with improved techniques (e.g., the new Section 4.1.1) as well as examples and explanations that were not included in the previous version.

2 PRELIMINARIES

A *hypersequent* [3, 4] is a $|$-separated multiset of ordinary sequents, called *components*. The sequents we consider in this paper have the form $\Gamma \Rightarrow \Pi$ where Γ is a (possibly empty) multiset of formulae in the language of intuitionistic logic and Π contains at most one formula.

Notation. Unless stated otherwise we use upper-case Greek letters for multisets of formulae (where Π contains at most one element), lower-case Greek letters for formulae, and G, H for (possibly empty) hypersequents.

As with sequent calculi, the inference rules of hypersequent calculi consist of initial hypersequents (i.e., axioms), the cut-rule as well as logical and structural rules. The logical and structural rules are divided into *internal* and *external rules*. The internal rules deal with formulae within one component of the conclusion. Examples of external structural rules include external weakening (EW) and external contraction (EC), see Fig. 1.

Rules are usually presented as rule schemata. Concrete instances of a rule are obtained by substituting formulae for schematic variables. Following standard practice, we do not explicitly distinguish between a rule and a rule schema.

Fig. 1 displays the hypersequent version HLJ of the propositional sequent calculus LJ for intuitionistic logic. Note that the *hyperlevel* of HLJ is in fact redundant since a hypersequent $\Gamma_1 \Rightarrow \Pi_1 \,|\, \dots \,|\, \Gamma_k \Rightarrow \Pi_k$ is derivable in HLJ if and only if $\Gamma_i \Rightarrow \Pi_i$ is derivable in LJ for some $i \in \{1, \dots, k\}$. Indeed, any sequent calculus can be trivially viewed as a hypersequent calculus. The added expressive power of the latter is due to the possibility of defining new rules which act simultaneously on several components of one or more hypersequents.

Example 2.1. By adding to HLJ the following version of the structural rule introduced in [4]

$$\frac{G \,|\, \Phi, \Gamma_1 \Rightarrow \Pi_1 \quad G \,|\, \Psi, \Gamma_2 \Rightarrow \Pi_2}{G \,|\, \Psi, \Gamma_1 \Rightarrow \Pi_1 \,|\, \Phi, \Gamma_2 \Rightarrow \Pi_2}\ (com)$$

we obtain a cut-free calculus for Gödel logic, which is (axiomatised by) intuitionistic logic plus the linearity axiom $(\varphi \to \psi) \lor (\psi \to \varphi)$.

$$\varphi \Rightarrow \varphi \quad \bot \Rightarrow \Pi \quad \frac{G\,|\,\Gamma, \varphi \Rightarrow \Pi \quad G\,|\,\Gamma, \psi \Rightarrow \Pi}{G\,|\,\Gamma, \varphi \vee \psi \Rightarrow \Pi}\ (\vee l) \quad \frac{G\,|\,\Gamma \Rightarrow \varphi_i}{G\,|\,\Gamma \Rightarrow \varphi_1 \vee \varphi_2}\ (\vee r)$$

$$\frac{G\,|\,\Gamma, \varphi, \psi \Rightarrow \Pi}{G\,|\,\Gamma, \varphi \wedge \psi \Rightarrow \Pi}\ (\wedge l) \quad \frac{G\,|\,\Gamma \Rightarrow \varphi \quad G\,|\,\Gamma \Rightarrow \psi}{G\,|\,\Gamma \Rightarrow \varphi \wedge \psi}\ (\wedge r) \quad \frac{G\,|\,\Gamma \Rightarrow \Pi}{G\,|\,\varphi, \Gamma \Rightarrow \Pi}\ (IW)$$

$$\frac{G\,|\,\Gamma \Rightarrow \varphi \quad G\,|\,\Gamma, \psi \Rightarrow \Pi}{G\,|\,\Gamma, \varphi \rightarrow \psi \Rightarrow \Pi}\ (\rightarrow l) \quad \frac{G\,|\,\Gamma, \varphi \Rightarrow \psi}{G\,|\,\Gamma \Rightarrow \varphi \rightarrow \psi}\ (\rightarrow r) \quad \frac{G\,|\,\varphi, \varphi, \Gamma \Rightarrow \Pi}{G\,|\,\varphi, \Gamma \Rightarrow \Pi}\ (IC)$$

$$\frac{G\,|\,\Gamma \Rightarrow \varphi \quad G\,|\,\varphi, \Gamma' \Rightarrow \Pi}{G\,|\,\Gamma, \Gamma' \Rightarrow \Pi}\ (cut) \quad \frac{G}{G\,|\,\Gamma \Rightarrow \Pi}\ (EW) \quad \frac{G\,|\,\Gamma \Rightarrow \Pi\,|\,\Gamma \Rightarrow \Pi}{G\,|\,\Gamma \Rightarrow \Pi}\ (EC)$$

Fig. 1. Rules and axioms of HLJ.

In [4] Avron suggested that a hypersequent can be thought of as a multiprocess. Under this interpretation, (com) is intended to model the exchange of information between parallel processes.

As the usual interpretation of the symbol "$|$" is disjunctive, the hypersequent calculus can naturally capture properties (Hilbert axioms, algebraic equations…) that can be expressed in a disjunctive form, see [8].

Notation and Assumptions. Given a hypersequent rule (r) with premisses $G\,|\,H_1 \ \ldots\ G\,|\,H_n$ and conclusion $G\,|\,H$, we call *active* the components in the hypersequents $H_1, \ldots, H_n, H$. We call *context components* the components of G. In this paper we will only consider hypersequent rules that (i) are (external) context sharing, i.e., whose premisses all contain the same hypersequent context G, and (ii) (except for (EC)) they have one active component in each premiss, i.e., in which each H_i is a sequent. Note that (i) is not a restriction and, in absence of eigenvariables acting on more than one component, neither is (ii); indeed, using (EC) and (EW), we can always transform a rule into an equivalent one that satisfies these conditions, that are crucial to prove Lemma 4.17.

Systems of rules were introduced in [19] to define analytic labelled calculi for logics semantically characterised by generalised geometric implications, a class of first-order formulae that goes beyond the geometric fragment [18] and includes all frame properties that correspond to formulae in the Sahlqvist fragment.

In general, a *system of rules* is a set of (possibly labelled) sequent rules that are bound to be applied in a predetermined order and that may share (schematic) variables or labels. Analyticity of systems of rules when added to a sequent or labelled sequent calculus for classical or intuitionistic logic was proved in [19] for systems acting on atomic formulae or relational atoms.

The proper restriction of systems of rules that we consider in the paper is defined below.

Definition 2.2. A *two-level system of rules* (*2-system* for short) is a set of sequent rules $\{(r_1), \ldots, (r_k), (r_B)\}$ that can only be applied according to the following schema:

$$\frac{\begin{array}{c}\mathcal{D}_1\\ \vdots\\ \Gamma \Rightarrow \Pi\end{array} \quad \ldots \quad \begin{array}{c}\mathcal{D}_k\\ \vdots\\ \Gamma \Rightarrow \Pi\end{array}}{\Gamma \Rightarrow \Pi}\ (r_B)$$

where each derivation $\mathcal{D}_i$, for $1 \leq i \leq k$, may contain several applications of

$$\frac{\Sigma_1, \Gamma' \Rightarrow \Pi' \quad \ldots \quad \Sigma_n, \Gamma' \Rightarrow \Pi'}{\Sigma_0, \Gamma' \Rightarrow \Pi'}\ (r_i)$$

that act on the same multisets of formulae $\Sigma_0, \Sigma_1, \ldots, \Sigma_n$.

The rule (r_B) is called *bottom rule*, while $(r_1), \ldots, (r_k)$ *top rules*.

In this paper we will consider 2-systems that manipulate LJ sequents.

Given a calculus $\mathcal{C}$ and a set of rules $\mathbb{R}$, $\mathcal{C} + \mathbb{R}$ will denote the calculus obtained by adding the elements of $\mathbb{R}$ to $\mathcal{C}$, and $\vdash_{\mathcal{C}+\mathbb{R}}$ its derivability relation.

Example 2.3. The 2-system $Sys_{(com^*)}$ in [19] for the linearity axiom (cf. Example 2.1) is the following (φ and ψ are metavariables for formulae):

$$\dfrac{\begin{array}{c}\dfrac{\varphi,\psi,\Gamma_1 \Rightarrow \Pi_1}{\psi,\Gamma_1 \Rightarrow \Pi_1}\,(com_1^*) \\ \vdots \\ \Gamma \Rightarrow \Pi\end{array} \qquad \begin{array}{c}\dfrac{\varphi,\psi,\Gamma_2 \Rightarrow \Pi_2}{\varphi,\Gamma_2 \Rightarrow \Pi_2}\,(com_2^*) \\ \vdots \\ \Gamma \Rightarrow \Pi\end{array}}{\Gamma \Rightarrow \Pi}\,(com_B^*)$$

The analyticity of LJ + $Sys_{(com^*)}$ is shown in [19] for *atomic* φ and ψ.

Remark. The above definition of 2-system differs from the one in [9] where each rule (r_i) could only be applied once in $\mathcal{D}_i$. The following example motivates the adoption of the more general condition in Definition 2.2.

Example 2.4. A cut-free derivation in LJ + $Sys_{(com^*)}$ (see Example 2.3) of the formula $((\varphi \to \psi) \wedge (\varphi \to \psi)) \vee ((\psi \to \varphi) \wedge (\psi \to \varphi))$ requires two applications of each of the top rules (com_1^*) and (com_2^*):

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\psi \Rightarrow \psi}{\varphi,\psi \Rightarrow \psi}}{\varphi \Rightarrow \psi}\,(com_1^*)}{\Rightarrow \varphi \to \psi} \qquad \dfrac{\dfrac{\dfrac{\psi \Rightarrow \psi}{\varphi,\psi \Rightarrow \psi}}{\varphi \Rightarrow \psi}\,(com_1^*)}{\Rightarrow \varphi \to \psi}}{\Rightarrow (\varphi \to \psi) \wedge (\varphi \to \psi)}}{\Rightarrow ((\varphi \to \psi) \wedge (\varphi \to \psi)) \vee ((\psi \to \varphi) \wedge (\psi \to \varphi))} \quad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\varphi \Rightarrow \varphi}{\psi,\varphi \Rightarrow \varphi}}{\psi \Rightarrow \varphi}\,(com_2^*)}{\Rightarrow \psi \to \varphi} \qquad \dfrac{\dfrac{\dfrac{\varphi \Rightarrow \varphi}{\psi,\varphi \Rightarrow \varphi}}{\psi \Rightarrow \varphi}\,(com_2^*)}{\Rightarrow \psi \to \varphi}}{\Rightarrow (\psi \to \varphi) \wedge (\psi \to \varphi)}}{\Rightarrow ((\varphi \to \psi) \wedge (\varphi \to \psi)) \vee ((\psi \to \varphi) \wedge (\psi \to \varphi))}}{\Rightarrow ((\varphi \to \psi) \wedge (\varphi \to \psi)) \vee ((\psi \to \varphi) \wedge (\psi \to \varphi))}$$

3 FROM 2-SYSTEMS TO HYPERSEQUENT RULES AND BACK

We show how to rewrite a 2-system Sys into the corresponding hypersequent rule Hr_{Sys}; vice versa, from a hypersequent rule Hr we construct the corresponding 2-system Sys_{Hr}. The transformation of derivations from HLJ + Hr into LJ + Sys_{Hr} (and from LJ + Sys into HLJ + Hr_{Sys}) is shown in Section 4.

From 2-systems to hypersequent rules. Given a 2-system Sys of the form

$$\dfrac{\begin{array}{c}\mathcal{D}_1 \\ \vdots \\ \Gamma \Rightarrow \Pi\end{array} \quad \ldots \quad \begin{array}{c}\mathcal{D}_k \\ \vdots \\ \Gamma \Rightarrow \Pi\end{array}}{\Gamma \Rightarrow \Pi}\,(r_B)$$

where each derivation $\mathcal{D}_i$, for $1 \leq i \leq k$, may contain several applications of the rule

$$\dfrac{\varphi_i^1, \ldots, \varphi_i^{l_i}, \Gamma_i \Rightarrow \Pi_i \quad \ldots \quad \psi_i^1, \ldots, \psi_i^{m_i}, \Gamma_i \Rightarrow \Pi_i}{\theta_i^1, \ldots, \theta_i^{n_i}, \Gamma_i \Rightarrow \Pi_i}\,(r_i)$$

the corresponding hypersequent rule Hr_{Sys} is as follows:

$$\dfrac{M_1 \qquad \ldots \qquad M_k}{G \,|\, \theta_1^1, \ldots, \theta_1^{n_1}, \Gamma_1 \Rightarrow \Pi_1 \,|\, \ldots \,|\, \theta_k^1, \ldots, \theta_k^{n_k}, \Gamma_k \Rightarrow \Pi_k}$$

where M_i, for $1 \leq i \leq k$, is the multiset of premisses

$$G \,|\, \varphi_i^1, \dots, \varphi_i^{l_i}, \Gamma_i \Rightarrow \Pi_i \quad \dots \quad G \,|\, \psi_i^1, \dots, \psi_i^{m_i}, \Gamma_i \Rightarrow \Pi_i$$

Example 3.1. From Negri's 2-system in Example 2.3 we obtain the rule acting on formulae φ, ψ

$$\frac{G \,|\, \varphi, \psi, \Gamma_1 \Rightarrow \Pi_1 \quad G \,|\, \varphi, \psi, \Gamma_2 \Rightarrow \Pi_2}{G \,|\, \psi, \Gamma_1 \Rightarrow \Pi_1 \,|\, \varphi, \Gamma_2 \Rightarrow \Pi_2}\ (com^*)$$

From hypersequent rules to 2-systems. Given any hypersequent rule *Hr* of the form

$$\frac{M_1 \qquad \dots \qquad M_k}{G \,|\, \Theta_1^1, \dots, \Theta_1^{n_1}, \Gamma_1 \Rightarrow \Pi_1 \,|\, \dots \,|\, \Theta_k^1, \dots, \Theta_k^{n_k}, \Gamma_k \Rightarrow \Pi_k}$$

where the sets M_i, for $1 \leq i \leq k$, constitute a partition of the set of premisses of *Hr* and each M_i contains the premisses

$$G \,|\, C_i^1 \quad \dots \quad G \,|\, C_i^{m_i}$$

where $C_i^1, \dots, C_i^{m_i}$ are sequents. The corresponding 2-system Sys_{Hr} is

$$\frac{\begin{array}{c}\mathcal{D}_1\\ \vdots\\ \Gamma \Rightarrow \Pi\end{array} \quad \dots \quad \begin{array}{c}\mathcal{D}_k\\ \vdots\\ \Gamma \Rightarrow \Pi\end{array}}{\Gamma \Rightarrow \Pi}\ (r_B)$$

where the derivation $\mathcal{D}_i$, for $1 \leq i \leq k$, may contain several applications of the rule

$$\frac{C_i^1 \quad \dots \quad C_i^{m_i}}{\Theta_i^1, \dots, \Theta_i^{n_i}, \Gamma_i \Rightarrow \Pi_i}\ (r_i)$$

Definition 3.2. We say that the premisses of *Hr* contained in M_i, for $1 \leq i < k$, are *linked* to the component $\Theta_i^1, \dots, \Theta_i^{n_1}, \Gamma_i \Rightarrow \Pi_i$ of the conclusion.

Example 3.3. The rewriting $Sys_{(com)}$ of the rule (com) in Example 2.1 is

$$\frac{\begin{array}{c}\dfrac{\Phi, \Gamma_1 \Rightarrow \Pi_1}{\Psi, \Gamma_1 \Rightarrow \Pi_1}\ (com_1)\\ \vdots\\ \Gamma \Rightarrow \Pi\end{array} \qquad \begin{array}{c}\dfrac{\Psi, \Gamma_2 \Rightarrow \Pi_2}{\Phi, \Gamma_2 \Rightarrow \Pi_2}\ (com_2)\\ \vdots\\ \Gamma \Rightarrow \Pi\end{array}}{\Gamma \Rightarrow \Pi}\ (com_B)$$

4 EMBEDDING THE TWO FORMALISMS

We introduce algorithms for transforming 2-system derivations into hypersequent derivations and vice versa.

4.1 From 2-systems to hypersequent derivations

Given any set $\mathbb{S}$ of 2-systems and set $\mathbb{H}$ of hypersequent rules s.t. if $Sys \in \mathbb{S}$ then $Hr_{Sys} \in \mathbb{H}$, starting from a derivation $\mathcal{D}$ in LJ + $\mathbb{S}$ we construct a derivation $\mathcal{D}'$ in HLJ + $\mathbb{H}$ of the same end-sequent. The construction proceeds by a stepwise translation of the rules in $\mathcal{D}$: the rules of LJ are translated into rules of HLJ – possibly using (EW) – and, for the 2-systems in $\mathbb{S}$, the top rules are translated into applications of the corresponding rules in $\mathbb{H}$ – and additional (EW), if needed – and the bottom rules are translated into applications of (EC). To keep track of the various translation steps, we mark the derivation $\mathcal{D}$. We start by marking and translating the leaves of $\mathcal{D}$. The rules with marked premisses are then translated one by one and the marks are moved to the conclusions of the rules.

The process is repeated until we reach and translate the root of $\mathcal{D}$. The correct termination of the procedure is guaranteed when $\mathcal{D}$ satisfies the following conditions

(1) two applications of a top rule belonging to the same 2-system instance never occur on the same path of the derivation,
(2) for each pair of 2-system instances, no top rule of one of the two instances occurs below any top rule of the other instance (see Definition 4.4 as used in Lemma 4.6)

Section 4.1.1 shows that each 2-system derivation can be transformed into one satisfying them.

The algorithm. Input: a derivation $\mathcal{D}$ in $\mathrm{LJ} + \mathbb{S}$. Output: a derivation $\mathcal{D}'$ of the same sequent in $\mathrm{HLJ} + \mathbb{H}$.
Translating axioms. The leaves of $\mathcal{D}$ are marked and copied as leaves of $\mathcal{D}'$.
Translating rules. Rules are translated one by one in the following order: first the one-premiss logical and structural rules applied to marked sequents, then the two-premiss logical rules and bottom rules with all premisses marked, and finally all the top rules of one 2-system instance[1]. After having translated each rule – or all top rules of a 2-system instance – we remove the marks from the premisses of the translated rules and mark their conclusions.

When we translate the top rules of a 2-system we apply the corresponding hypersequent rule once for each possible combination of different top rules of such system. For instance, if a 2-system contains two applications $(r_1)'$ and $(r_1)''$ of one top rule, and one application (r_2) of another top rule, we will have one hypersequent rule application translating the pair $\langle (r_1)', (r_2)\rangle$, and one hypersequent rule application translating the pair $\langle (r_1)'', (r_2)\rangle$.

Since the LJ rules are particular instances of HLJ rules, we only show how to translate 2-systems. Hence, consider a 2-system $Sys \in \mathbb{S}$ applied in $\mathcal{D}$ with the following instances of

(1) top rules:

$$\dfrac{\begin{matrix}\vdots & & \vdots\\ C_1^1 & \dots & C_1^{m_1}\end{matrix}}{\Delta_1, \Gamma_1 \Rightarrow \Pi_1}\ (r_1) \qquad \dots \qquad \dfrac{\begin{matrix}\vdots & & \vdots\\ C_k^1 & \dots & C_k^{m_k}\end{matrix}}{\Delta_k, \Gamma_k \Rightarrow \Pi_k}\ (r_k)$$

where $C_1^1, \dots, C_1^{m_1}, \dots, C_k^1, \dots, C_k^{m_k}$ are marked sequents and each top rule $(r_1), \dots, (r_k)$ is possibly applied more than once.
By the definition of the algorithm, we have hypersequent derivations of

$$G \mid C_1^1 \quad \dots \quad G \mid C_1^{m_1} \qquad \dots \qquad G \mid C_k^1 \quad \dots \quad G \mid C_k^{m_k}$$

for each application of the top rules. We apply Hr_{Sys} as follows

$$\dfrac{M_1 \qquad\qquad \dots \qquad\qquad M_k}{G \mid \Delta_1, \Gamma_1 \Rightarrow \Pi_1 \mid \dots \mid \Delta_k, \Gamma_k \Rightarrow \Pi_k}$$

for each possible combination of k applications of the top rules $(r_1), \dots, (r_k)$ – possibly duplicating the hypersequent derivations previously obtained. We move the marks to the conclusions of $(r_1), \dots, (r_k)$.
Notice that we always have hypersequents containing suitable active components and matching context components. Indeed, given that we translate into a hypersequent rule application each possible combination of top rules, at each translation step (above the bottom rule) we have exactly one hypersequent for each possible combination of marked sequents.

[1]Condition 1 guarantees that all top rules of a 2-system instance can be translated by one hypersequent rule.

(2) bottom rule:

$$\dfrac{\begin{array}{c}\vdots \\ \Gamma \Rightarrow \Pi\end{array} \quad \begin{array}{c}\vdots \\ \Gamma \Rightarrow \Pi\end{array}}{\Gamma \Rightarrow \Pi}\,(r_B)$$

Without loss of generality we can assume that the top rules of the considered 2-system have been applied above the premisses of (r_B) – as otherwise the application of the 2-system is redundant. Hence we have a derivation in HLJ + $\mathbb{H}$ of $G\,|\,\Gamma \Rightarrow \Pi\,|\,\ldots\,|\,\Gamma \Rightarrow \Pi$. The desired derivation of $G\,|\,\Gamma \Rightarrow \Pi$ is obtained by repeatedly applying (EC). We move the marks to the conclusion of (r_B).

Theorem 4.1. *For any set $\mathbb{H}$ of hypersequent rules and set $\mathbb{S}$ of 2-systems s.t. if $Sys \in \mathbb{S}$ then $Hr_{Sys} \in \mathbb{H}$, if $\vdash_{\mathrm{LJ}+\mathbb{S}} \Gamma \Rightarrow \Pi$ then $\vdash_{\mathrm{HLJ}+\mathbb{H}} \Gamma \Rightarrow \Pi$.*

Proof. Apply the above algorithm to the LJ + $\mathbb{S}$ derivation $\mathcal{D}$ of $\Gamma \Rightarrow \Pi$ to obtain $\mathcal{D}'$. The algorithm terminates because the number of rule applications in a derivation is finite. We show that the algorithm does not stop before translating the root of $\mathcal{D}$. The proof is by induction on the number u of 2-system instances whose top rules are still to be translated. If $u = 0$ all remaining rules can be translated as soon as the premisses are marked. Assume $u = n + 1$. Lemma 4.6 assures that there is at least a 2-system instance S whose top rules are still untranslated and do not occur below any untranslated top rule. Hence the rule applications that have to be translated before the top rules of S do not belong to any 2-system and can be translated as soon as their premisses are marked. After translating these rules, we can translate the top rules of S and obtain $u = n$. □

Example 4.2. The following derivation in the calculus LJ + $Sys_{(com)}$ for Gödel logic (see Example 3.3)

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\psi \Rightarrow \psi}{\varphi \Rightarrow \psi}\,(com_1)'}{\Rightarrow \varphi \to \psi} \quad \dfrac{\dfrac{\psi \Rightarrow \psi}{\varphi \Rightarrow \psi}\,(com_1)''}{\Rightarrow \varphi \to \psi}}{\Rightarrow (\varphi \to \psi) \wedge (\varphi \to \psi)}}{\Rightarrow ((\varphi \to \psi) \wedge (\varphi \to \psi)) \vee (\psi \to \varphi)} \quad \dfrac{\dfrac{\dfrac{\varphi \Rightarrow \varphi}{\psi \Rightarrow \varphi}\,(com_2)}{\Rightarrow \psi \to \varphi}}{\Rightarrow ((\varphi \to \psi) \wedge (\varphi \to \psi)) \vee (\psi \to \varphi)}}{\Rightarrow ((\varphi \to \psi) \wedge (\varphi \to \psi)) \vee (\psi \to \varphi)}\,(com_B)$$

is translated into the HLJ + (com) derivation (see Example 2.1)

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\psi \Rightarrow \psi \quad \varphi \Rightarrow \varphi}{\varphi \Rightarrow \psi\,|\,\psi \Rightarrow \varphi}\,(com)'}{\varphi \Rightarrow \psi\,|\,\Rightarrow \psi \to \varphi}}{\Rightarrow \varphi \to \psi\,|\,\Rightarrow \psi \to \varphi} \quad \dfrac{\dfrac{\dfrac{\psi \Rightarrow \psi \quad \varphi \Rightarrow \varphi}{\varphi \Rightarrow \psi\,|\,\psi \Rightarrow \varphi}\,(com)''}{\varphi \Rightarrow \psi\,|\,\Rightarrow \psi \to \varphi}}{\Rightarrow \varphi \to \psi\,|\,\Rightarrow \psi \to \varphi}}{\Rightarrow (\varphi \to \psi) \wedge (\varphi \to \psi)\,|\,\Rightarrow \psi \to \varphi}}{\Rightarrow (\varphi \to \psi) \wedge (\varphi \to \psi)\,|\,\Rightarrow ((\varphi \to \psi) \wedge (\varphi \to \psi)) \vee (\psi \to \varphi)}}{\Rightarrow ((\varphi \to \psi) \wedge (\varphi \to \psi)) \vee (\psi \to \varphi)\,|\,\Rightarrow ((\varphi \to \psi) \wedge (\varphi \to \psi)) \vee (\psi \to \varphi)}}{\Rightarrow ((\varphi \to \psi) \wedge (\varphi \to \psi)) \vee (\psi \to \varphi)}\,(EC)$$

where $(com)'$ translates the pair of top rule applications $\langle (com_1)', (com_2) \rangle$, while $(com)''$ translates the pair $\langle (com_1)'', (com_2) \rangle$.

4.1.1 Normal forms of 2-systems derivations. We introduce the normal forms of 2-system derivations needed by the algorithm of Section 4.1 and we show how to obtain them. The definition of 2-systems (Def. 2.2) is indeed decidedly liberal. It allows unrestricted nesting of 2-systems and does not limit the application of the top rule (r_i) inside $\mathcal{D}_i$. Such freedom matches naturally the general idea of a system of rules, but complicates the structure of derivations and the algorithm for transforming 2-system derivations into hypersequent derivations. We show below that w.l.o.g. we can consider derivations of a simplified form.

Lemma 4.3. *Any 2-system derivation can be transformed into one with the following property: two applications of a top rule (t) belonging to the same 2-system instance never occur on the same path of the derivation.*

Proof. Let $\mathcal{P}$ be a 2-system derivation in which two applications of (t) occur along the same path, as, e.g., in

$$\dfrac{\begin{array}{c}\vdots\\ \Sigma_1,\Gamma\Rightarrow\Pi\end{array} \quad\dots\quad \begin{array}{c}\dfrac{\begin{array}{c}\vdots\\ \Sigma_1,\Gamma'\Rightarrow\Pi'\end{array}\quad\dots\quad\begin{array}{c}\vdots\\ \Sigma_n,\Gamma'\Rightarrow\Pi'\end{array}}{\Delta,\Gamma'\Rightarrow\Pi'}\,(t)\\ \vdots\;\mathcal{D}\\ \Sigma_i,\Gamma\Rightarrow\Pi\end{array}\quad\dots\quad \begin{array}{c}\vdots\\ \Sigma_n,\Gamma\Rightarrow\Pi\end{array}}{\Delta,\Gamma\Rightarrow\Pi}\,(t)$$

We use (IW) and (IC) to transform it into

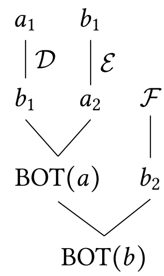
$$\dfrac{\begin{array}{c}\vdots\\ \Sigma_1,\Gamma\Rightarrow\Pi\end{array} \quad\dots\quad \begin{array}{c}\dfrac{\begin{array}{c}\dfrac{\begin{array}{c}\vdots\\ \Sigma_i,\Gamma'\Rightarrow\Pi'\end{array}}{\Sigma_i,\Delta,\Gamma'\Rightarrow\Pi'}\,(IW)\\ \vdots\;\mathcal{D}'\\ \Sigma_i,\Sigma_i,\Gamma\Rightarrow\Pi\end{array}}{\Sigma_i,\Gamma\Rightarrow\Pi}\,(IC)\end{array}\quad\dots\quad \begin{array}{c}\vdots\\ \Sigma_n,\Gamma\Rightarrow\Pi\end{array}}{\Delta,\Gamma\Rightarrow\Pi}\,(t)$$

where for each sequent $\Gamma''\Rightarrow\Pi''$ in $\mathcal{D}$ there is a sequent $\Sigma_i,\Gamma''\Rightarrow\Pi''$ in $\mathcal{D}'$. □

Derivations using 2-systems can be further simplified. Indeed the lemma below shows that we can restrict our attention to derivations with a limited nesting of 2-systems. We use the notion of entanglement to formalise a violation of this limitation.

Definition 4.4. Two 2-system instances S_1 and S_2 are *entangled* if some top rules of S_1 occur above some top rules of S_2 and some of the former occur below some of the latter.

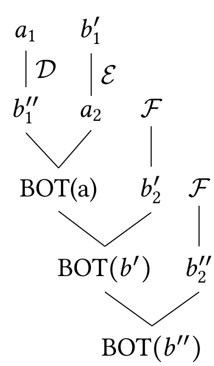
Consider, for instance, the following derivation schema containing two 2-system instances a and b with bottom rules BOT(a) and BOT(b) and top rules a_1, a_2 and b_1, b_2, respectively:



We use $\mathcal{D}$, $\mathcal{E}$ and $\mathcal{F}$ to denote derivations. The entanglement here occurs because b_1 is applied once below a_1 and once above a_2.

Remark. If two 2-system instances are entangled, then all rules of one of them occur necessarily above exactly one premiss of the bottom rule of the other.

Example 4.5. To disentangle a and b, we make two copies b' and b'' of b that are going to contain the rules formerly belonging to b:



The 2-system instances are now disentangled: no top rule of b' occurs below any top rule of a and no top rule of b'' occurs above any top rule of a.

The above transformation is the basic step employed in the following lemma.

Lemma 4.6. *Any 2-system derivation $\mathcal{P}$ can be transformed into a 2-system derivation $\mathcal{P}'$ of the same end-sequent in which no entanglement occurs.*

Proof. First we introduce the transformation of derivations (*e-reduction*) that decreases the number of top rule applications involved in entanglements. We then provide a strategy to obtain the desired derivation $\mathcal{P}'$ using such transformation, and we prove termination.
E-reduction: given a 2-system instance S (with bottom rule (B_S)) entangled with 2-system instances $S_1, \ldots, S_n$:

$$\dfrac{\begin{matrix}\mathcal{D}_1 \\ \vdots \\ \Gamma \Rightarrow \Pi\end{matrix} \quad \ldots \quad \begin{matrix}\mathcal{D}_n \\ \vdots \\ \Gamma \Rightarrow \Pi\end{matrix}}{\Gamma \Rightarrow \Pi}\ (B_S)$$

we make two copies S' and S'' of S with bottom rules $(B_{S'})$ respectively $(B_{S''})$:

$$\dfrac{\dfrac{\begin{array}{c}\mathcal{D}_1'\\ \vdots\\ \Gamma \Rightarrow \Pi\end{array} \quad \begin{array}{c}\mathcal{D}_2\\ \vdots\\ \Gamma \Rightarrow \Pi\end{array} \quad \dots \quad \begin{array}{c}\mathcal{D}_m\\ \vdots\\ \Gamma \Rightarrow \Pi\end{array}}{\Gamma \Rightarrow \Pi}\,(B_{S'}) \quad \begin{array}{c}\mathcal{D}_2\\ \vdots\\ \Gamma \Rightarrow \Pi\end{array} \quad \dots \quad \begin{array}{c}\mathcal{D}_n\\ \vdots\\ \Gamma \Rightarrow \Pi\end{array}}{\Gamma \Rightarrow \Pi}\,(B_{S''})$$

in such a way that:

- if a top rule in $\mathcal{D}_1$ belonging to S occurs above a top rule of one among $S_1, \dots, S_n$, then its copy in $\mathcal{D}_1'$ belongs to S',
- if a top rule in $\mathcal{D}_1$ belonging to S occurs below a top rule of one among $S_1, \dots, S_n$, then its copy in $\mathcal{D}_1'$ belongs to S''.

Notice that in the obtained derivation no top rule of S' occurs below any top rule of $S_1, \dots, S_n$, and no top rule of S'' occurs above any top rule of $S_1, \dots, S_n$. Moreover, also due to Lemma 4.3: $(*)$ neither S' and S'' nor two copies of the same 2-system instance in $\mathcal{D}_2, \dots, \mathcal{D}_n$ can be entangled or have top rules along the same path of the derivation.

A strategy to apply e-reductions that leads to the required derivation $\mathcal{P}'$ is the following. We start reducing one of the 2-system instances with lowermost bottom rule. Whenever we apply an e-reduction we collect all entangled copies of the same 2-system instance in the same class. We continue the disentanglement focusing on a single class and reducing all its elements before we move on to another class. Notice that the number of classes never increases and is bounded by the number of 2-system instances in the original derivation. Fixed a class, the strategy guarantees that its elements are disentangled one by one without duplicating other maximally entangled elements of the same class. See the appendix for the formalisation of the strategy and termination proof. □

4.2 From hypersequent to 2-system derivations

Given any set $\mathbb{H}$ of hypersequent rules and set $\mathbb{S}$ of 2-systems s.t. if $Hr \in \mathbb{H}$ then $Sys_{Hr} \in \mathbb{S}$. Starting from a derivation in HLJ + $\mathbb{H}$ we construct a derivation in LJ + $\mathbb{S}$ of the same end-sequent.

The algorithm. Input: a derivation $\mathcal{D}$ of a sequent $\Gamma \Rightarrow \Pi$ in HLJ + $\mathbb{H}$. Output: a derivation $\mathcal{D}'$ of $\Gamma \Rightarrow \Pi$ in LJ + $\mathbb{S}$.

Intuitively, each application of a HLJ rule in $\mathcal{D}$ is rewritten as an application of an LJ rule in $\mathcal{D}'$. Some care is needed to handle the external structural rules in $\mathbb{H}$ as well as (EW) and (EC). To deal with the latter rules, which have no direct translation in LJ + $\mathbb{S}$, we consider only derivations $\mathcal{D}$ in which (i) all applications of (EC) occur immediately above the root, and (ii) all applications of (EW) occur where immediately needed, that is where they introduce components of the context of rules with more than one premiss. As shown in Section 4.2.1 each hypersequent derivation (of a sequent) can be transformed into an equivalent one of this form.

The rules in $\mathbb{H}$ are translated in two steps. First for each component of the premiss of the uppermost application of (EC) in $\mathcal{D}$ we find a *partial derivation*, that is a derivation in LJ extended by the top rules of the 2-systems in $\mathbb{S}$ without any applicability condition (Lemma 4.13). The desired derivation $\mathcal{D}'$ is then obtained by suitably applying to these partial derivations the corresponding bottom rules (Theorem 4.15).

Definition 4.7. A partial derivation in LJ + $\mathbb{S}$ is a derivation in LJ extended with the top rules of $\mathbb{S}$ (without their applicability conditions relative to a bottom rule application).

We show an example of the first part of the translation to guide the reader's intuition through the proofs that follow.

Example 4.8. Consider the HLJ + (com) derivation

$$
\begin{prooftree}
\AxiomC{$\theta \Rightarrow \theta$}
\UnaryInfC{$\varphi, \theta \Rightarrow \theta$}
\AxiomC{$\psi \Rightarrow \psi$}
\RightLabel{$(com)'$}
\BinaryInfC{$\varphi, \psi \Rightarrow \theta \,|\, \theta \Rightarrow \psi$}
\AxiomC{$\theta \Rightarrow \theta$}
\UnaryInfC{$\theta, \psi \Rightarrow \theta$}
\AxiomC{$\varphi \Rightarrow \varphi$}
\RightLabel{$(com)''$}
\BinaryInfC{$\varphi, \psi \Rightarrow \theta \,|\, \theta \Rightarrow \varphi$}
\RightLabel{$(\wedge r)$}
\BinaryInfC{$\varphi, \psi \Rightarrow \theta \,|\, \theta \Rightarrow \psi \wedge \varphi$}
\UnaryInfC{$\varphi \wedge \psi \Rightarrow \theta \,|\, \theta \Rightarrow \psi \wedge \varphi$}
\UnaryInfC{$\varphi \wedge \psi \Rightarrow \theta \,|\, \Rightarrow \theta \to \psi \wedge \varphi$}
\UnaryInfC{$\Rightarrow \varphi \wedge \psi \to \theta \,|\, \Rightarrow \theta \to \psi \wedge \varphi$}
\UnaryInfC{$\Rightarrow \varphi \wedge \psi \to \theta \,|\, \Rightarrow (\varphi \wedge \psi \to \theta) \vee (\theta \to \psi \wedge \varphi)$}
\UnaryInfC{$\Rightarrow (\varphi \wedge \psi \to \theta) \vee (\theta \to \psi \wedge \varphi) \,|\, \Rightarrow (\varphi \wedge \psi \to \theta) \vee (\theta \to \psi \wedge \varphi)$}
\RightLabel{(EC)}
\UnaryInfC{$\Rightarrow (\varphi \wedge \psi \to \theta) \vee (\theta \to \psi \wedge \varphi)$}
\end{prooftree}
$$

and observe that it satisfies property (i) and, trivially, property (ii). The partial derivations in LJ + $Sys_{(com)}$ (see Ex. 3.3) of the components of the uppermost application of (EC) in the above proof are:

$$
\begin{prooftree}
\AxiomC{$\theta \Rightarrow \theta$}
\UnaryInfC{$\varphi, \theta \Rightarrow \theta$}
\RightLabel{$(com_1)'$}
\UnaryInfC{$\varphi, \psi \Rightarrow \theta$}
\AxiomC{$\theta \Rightarrow \theta$}
\UnaryInfC{$\theta, \psi \Rightarrow \theta$}
\RightLabel{$(com_1)''$}
\UnaryInfC{$\varphi, \psi \Rightarrow \theta$}
\RightLabel{$dummy$}
\BinaryInfC{$\varphi, \psi \Rightarrow \theta$}
\UnaryInfC{$\varphi \wedge \psi \Rightarrow \theta$}
\UnaryInfC{$\Rightarrow \varphi \wedge \psi \to \theta$}
\UnaryInfC{$\Rightarrow (\varphi \wedge \psi \to \theta) \vee (\theta \to \psi \wedge \varphi)$}
\end{prooftree}
\qquad
\begin{prooftree}
\AxiomC{$\varphi \Rightarrow \varphi$}
\RightLabel{$(com_2)''$}
\UnaryInfC{$\theta \Rightarrow \varphi$}
\AxiomC{$\psi \Rightarrow \psi$}
\RightLabel{$(com_2)'$}
\UnaryInfC{$\theta \Rightarrow \psi$}
\RightLabel{$(\wedge r)$}
\BinaryInfC{$\theta \Rightarrow \psi \wedge \varphi$}
\UnaryInfC{$\Rightarrow \theta \to \psi \wedge \varphi$}
\UnaryInfC{$\Rightarrow (\varphi \wedge \psi \to \theta) \vee (\theta \to \psi \wedge \varphi)$}
\end{prooftree}
$$

where $(com_1)'$ and $(com_2)'$ translate $(com)'$ while $(com_1)''$ and $(com_2)''$ translate $(com)''$. Notice that in order to handle the context component duplication relative to $(\wedge r)$, we apply a *dummy* bottom rule.

The partial derivations obtained have the same structure as the hypersequent derivations of the corresponding components (see *ancestor tree* in Def. 4.12).

We use Definitions 4.9 and 4.10 to formalise and achieve properties (i) and (ii).

Definition 4.9. For any one-premiss rule (r) we call a *queue of* (r) any sequence of consecutive applications of (r) that is neither immediately preceded nor immediately followed by applications of (r).

Definition 4.10. We say that an HLJ + $\mathbb{H}$ derivation is in *structured form iff* all (EC) applications occur in a queue immediately above the root, and all (EW) applications occur in subderivations of the form

$$
\begin{prooftree}
\AxiomC{$G_1 \,|\, C_1$}
\RightLabel{(EW)}
\UnaryInfC{$\vdots$}
\RightLabel{(EW)}
\UnaryInfC{$G \,|\, C_1$}
\AxiomC{$\dots$}
\AxiomC{$G_n \,|\, C_n$}
\RightLabel{(EW)}
\UnaryInfC{$\vdots$}
\RightLabel{(EW)}
\UnaryInfC{$G \,|\, C_n$}
\RightLabel{(r)}
\TrinaryInfC{$G \,|\, C_0$}
\end{prooftree}
$$

where (r) is any rule with more than one premiss and each component of G is contained in at least one of the hypersequents $G_1, \dots, G_n$.

A derivation in structured form can be divided into a part containing only (EC) applications and a part containing the applications of any other rule. We introduce a notation for the hypersequent separating the two parts.

Definition 4.11. If $\mathcal{D}$ is a derivation in structured form, we denote by $\widehat{H}_{\mathcal{D}}$ the premiss of the uppermost application of (EC) in $\mathcal{D}$.

Definition 4.12. Given a HLJ + $\mathbb{H}$ derivation. A sequent (hypersequent component) C' is a *parent* of a sequent C, denoted as $p(C, C')$, if one of the following conditions holds:

- C is active in the conclusion of an application of some $Hr \in \mathbb{H}$, and C' is the active component of a premiss linked to C (see Definition 3.2);
- C is active in the conclusion of an application of a rule of HLJ, and C' is the active component of a premiss of such application;
- C is a context component in the conclusion of any rule application, and C' is the corresponding context component in a premiss of such application.

We say that a sequent C' is an *ancestor* of a sequent C, and we write $a(C, C')$, if the pair $\langle C, C' \rangle$ is in the transitive closure of the relation $p(\cdot, \cdot)$. The *ancestor tree* of a sequent C is the tree whose nodes are all sequents related to C by $a(\cdot, \cdot)$ and whose edges are defined by the relation $p(\cdot, \cdot)$ between such nodes.

We prove below that from any HLJ + $\mathbb{H}$ derivation $\mathcal{D}$ of a sequent we can construct a partial derivation for each component of $\widehat{H}_{\mathcal{D}}$ having the *same structure* as the ancestor tree of that component, i.e., consisting of the translation of the rules in the ancestor tree, with the exception of (EW).

Remark.

- In an HLJ + $\mathbb{H}$ derivation that does not use (EC), the ancestor tree of each hypersequent is a sequent derivation.
- If C is the active component of an application of (EW), then there is no C' such that $p(C, C')$.

As usual, the *length* of a derivation is the maximal number of rule applications occurring on any branch plus 1.

Lemma 4.13. *Let $\mathbb{H}$ be a set of hypersequent rules and $\mathbb{S}$ of 2-systems s.t. if $Hr \in \mathbb{H}$ then $Sys_{Hr} \in \mathbb{S}$. Given any* HLJ + $\mathbb{H}$ *derivation $\mathcal{D}$ in structured form, for each component C of $\widehat{H}_{\mathcal{D}}$ we can construct a partial derivation in* LJ + $\mathbb{S}$ *having the same structure as the ancestor tree of C in $\mathcal{D}$.*

Proof. Let H be a hypersequent in $\mathcal{D}$ derived without using (EC). We construct a partial derivation in LJ + $\mathbb{S}$ with the required property for each of its components. The proof proceeds by induction on the length l of the derivation of H by translating each rule of HLJ + $\mathbb{H}$, with the exception of (EW), into the corresponding sequent rule in LJ + $\mathbb{S}$.

Base case. If $l = 1$ (i.e. H is an axiom) the partial derivation in LJ + $\mathbb{S}$ simply contains H.

Inductive step. We consider the last rule $(r) \neq (EW)$ applied in the subderivation $\mathcal{D}'$ of H, and we distinguish the two cases: (i) (r) is a one-premiss rule and (ii) (r) has more premisses; for the latter case, since $\mathcal{D}'$ is in structured form, we deal also with possible queues of (EW) above its premisses.

(1) Assume that the derivation ending in a one-premiss rule $(r) \in$ HLJ is

$$\dfrac{\begin{array}{c}\mathcal{D}\\ \vdots\\ G \mid C\end{array}}{G \mid C'}\ (r)$$

By induction hypothesis there is a partial derivation of C (and of each component of G) having the same structure as the ancestor tree of C. The partial derivation of C' is simply obtained by applying (r).
The case in which (r) is a one-premiss rule belonging to $\mathbb{H}$ is a special case of (ii) for which there is no need to consider queues of (EW).

(2) Assume that $(r) = (Hr) \in \mathbb{H}$ has more than one premiss, the remaining cases – $(r) \in$ HLJ, and $(r) \in \mathbb{H}$ and has only one premiss – being simpler. Assume that the derivation $\mathcal{D}'$, of length n, is the following

$$\dfrac{\begin{array}{c}\mathcal{D}_1^1\\ \vdots\\ G\,|\,C_1^1\end{array}\quad \ldots \quad \begin{array}{c}\mathcal{D}_1^{m_1}\\ \vdots\\ G\,|\,C_1^{m_1}\end{array}\quad \ldots \quad \begin{array}{c}\mathcal{D}_k^1\\ \vdots\\ G\,|\,C_k^1\end{array}\quad \ldots \quad \begin{array}{c}\mathcal{D}_k^{m_k}\\ \vdots\\ G\,|\,C_k^{m_k}\end{array}}{G\,|\,\Delta_1, \Gamma_1 \Rightarrow \Pi_1\,|\,\ldots\,|\,\Delta_k, \Gamma_k \Rightarrow \Pi_k}\;(Hr)$$

where the premisses $G\,|\,C_j^i$ of (Hr) are possibly inferred by a queue of (EW). When this is the case, we consider the uppermost hypersequents in the queues. More precisely, we consider the following derivations (each of which has length strictly less than n)

$$\begin{array}{c}\mathcal{D}_1^1\\ \vdots\\ G_1^1\,|\,C_1^1\end{array}\quad \ldots \quad \begin{array}{c}\mathcal{D}_1^{m_1}\\ \vdots\\ G_1^{m_1}\,|\,C_1^{m_1}\end{array}\quad \ldots \quad \begin{array}{c}\mathcal{D}_k^1\\ \vdots\\ G_k^1\,|\,C_k^1\end{array}\quad \ldots \quad \begin{array}{c}\mathcal{D}_k^{m_k}\\ \vdots\\ G_k^{m_k}\,|\,C_k^{m_k}\end{array}$$

where, for $1 \leq y \leq k$ and $1 \leq x \leq m_y$, the hypersequent G_y^x is G if there is no (EW) application immediately above $G\,|\,C_y^x$; otherwise, $G_y^x\,|\,C_y^x$ is the premiss of the uppermost (EW) application in the queue immediately above $G\,|\,C_y^x$.
Since $\mathcal{D}$ (and hence $\mathcal{D}'$) is in structured form, each component of G must occur in at least one of the hypersequents $G_1^1, \ldots, G_1^{m_1}, \ldots, G_k^1, \ldots, G_k^{m_k}$. We obtain partial derivations for $\Delta_1, \Gamma_1 \Rightarrow \Pi_1, \ldots, \Delta_k, \Gamma_k \Rightarrow \Pi_k$ applying the top rules of the 2-system Sys_{Hr} as follows

$$\dfrac{C_1^1 \quad \ldots \quad C_1^{m_1}}{\Delta_1, \Gamma_1 \Rightarrow \Pi_1}\;(r_1) \qquad \ldots \qquad \dfrac{C_k^1 \quad \ldots \quad C_k^{m_k}}{\Delta_k, \Gamma_k \Rightarrow \Pi_k}\;(r_k)$$

Indeed, by induction hypothesis, we have a partial derivation for each C_y^x. In case a component C of G occurs in more than one premiss, we have different partial derivations. Hence we apply a dummy bottom rule

$$\dfrac{C \quad \ldots \quad C}{C}$$

and obtain one partial derivation.

The obtained partial derivations clearly satisfy the following property: with the exception of (EW) and of dummy bottom rules, a rule application occurs in the ancestor tree of a hypersequent component in $\mathcal{D}$ *iff* its translation occurs in the partial derivation of such component. □

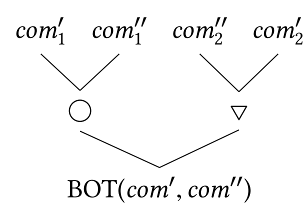
The next step of the translation consists in applying a bottom rule for each group of top rules translating one hypersequent rule application. If we applied dummy bottom rules inside the partial derivations, we might be forced to apply a single bottom rule for more than one of such groups – thus creating what will be called a *mixed system*. In Theorem 4.15 we prove that we can always restructure the derivation and obtain the desired exact match between groups of top rules and bottom rules. We first show an example that clarifies the main ideas exploited in the following proof.

Example 4.14. Consider the partial derivations obtained in Ex. 4.8, if we apply a bottom rule to them we obtain the following derivation:

$$
\dfrac{
\dfrac{\dfrac{\dfrac{
\dfrac{\dfrac{\theta \Rightarrow \theta}{\varphi, \theta \Rightarrow \theta}}{\varphi, \psi \Rightarrow \theta}\,(com_1)'
\quad
\dfrac{\dfrac{\theta \Rightarrow \theta}{\theta, \psi \Rightarrow \theta}}{\varphi, \psi \Rightarrow \theta}\,(com_1)''
}{\varphi, \psi \Rightarrow \theta}\,dummy}{\varphi \wedge \psi \Rightarrow \theta}}{\Rightarrow \varphi \wedge \psi \to \theta}}{\Rightarrow (\varphi \wedge \psi \to \theta) \vee (\theta \to \psi \wedge \varphi)}
\quad
\dfrac{\dfrac{\dfrac{
\dfrac{\varphi \Rightarrow \varphi}{\theta \Rightarrow \varphi}\,(com_2)''
\quad
\dfrac{\psi \Rightarrow \psi}{\theta \Rightarrow \psi}\,(com_2)'
}{\theta \Rightarrow \psi \wedge \varphi}\,(\to r)}{\Rightarrow \theta \to \psi \wedge \varphi}}{\Rightarrow (\varphi \wedge \psi \to \theta) \vee (\theta \to \psi \wedge \varphi)}
}{\Rightarrow (\varphi \wedge \psi \to \theta) \vee (\theta \to \psi \wedge \varphi)}\,(com_B)
$$

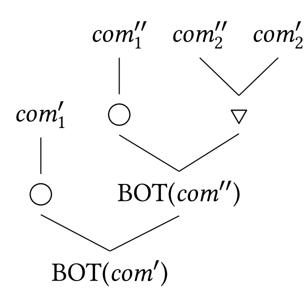
where (com_B) is the bottom rule both for $(com_1)'$ and $(com_2)'$ and for $(com_1)''$ and $(com_2)''$. We call this a mixed system.

We can abstract this derivation as



where we represent by BOT(com', com'') the bottom rule of com' and com'', by ◯ the forks in the derivation tree corresponding to dummy bottom rules, and by ▽ the forks corresponding to non-dummy rules.

Given that the removal of premisses from the ◯ forks is a logically sound operation, we transform the structure of the derivation as follows:



Now the group of top rules translating com' and the one translating com'' have different bottom rules. The derivation resulting from this is the following

$$
\dfrac{
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\theta \Rightarrow \theta}{\varphi, \theta \Rightarrow \theta}}{\varphi, \psi \Rightarrow \theta}\,(com_1)'}{\varphi \wedge \psi \Rightarrow \theta}}{\Rightarrow \varphi \wedge \psi \to \theta}}{\Rightarrow \alpha}
\quad
\dfrac{
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\theta \Rightarrow \theta}{\theta, \psi \Rightarrow \theta}}{\varphi, \psi \Rightarrow \theta}\,(com_1)''}{\varphi \wedge \psi \Rightarrow \theta}}{\Rightarrow \varphi \wedge \psi \to \theta}}{\Rightarrow \alpha}
\quad
\dfrac{\dfrac{\dfrac{
\dfrac{\varphi \Rightarrow \varphi}{\theta \Rightarrow \varphi}\,(com_2)''
\quad
\dfrac{\psi \Rightarrow \psi}{\theta \Rightarrow \psi}\,(com_2)'
}{\theta \Rightarrow \psi \wedge \varphi}}{\Rightarrow \theta \to \psi \wedge \varphi}}{\Rightarrow \alpha}
}{\Rightarrow \alpha}\,(com_B)''
}{\Rightarrow \alpha}\,(com_B)'
$$

where α is the formula $(\varphi \land \psi \to \theta) \lor (\theta \to \psi \land \varphi)$.

Theorem 4.15. *For any set $\mathbb{H}$ of hypersequent rules and set $\mathbb{S}$ of 2-systems s.t. if $Hr \in \mathbb{H}$ then $Sys_{Hr} \in \mathbb{S}$, if $\vdash_{\mathrm{HLJ}+\mathbb{H}} \Gamma \Rightarrow \Pi$ then $\vdash_{\mathrm{LJ}+\mathbb{S}} \Gamma \Rightarrow \Pi$.*

Proof. Let $\mathcal{D}$ be a HLJ + $\mathbb{H}$ derivation of $\Gamma \Rightarrow \Pi$. By the results in Section 4.2.1 we can assume that $\mathcal{D}$ is in structured form. By applying the procedure of Lemma 4.13 to the premiss $\widehat{H}_{\mathcal{D}}$ of the uppermost application of (EC) in $\mathcal{D}$ we obtain a set of partial derivations $\{\mathcal{D}_i\}_{i\in I}$ whose rules translate those occurring in the ancestor trees of each component of $\widehat{H}_{\mathcal{D}}$.

We show that we can suitably apply the bottom rules of 2-systems in $\mathbb{S}$ to the roots of $\{\mathcal{D}_i\}_{i\in I}$ in order to obtain the required LJ + $\mathbb{S}$ derivation of $\Gamma \Rightarrow \Pi$. First, we group all top rule applications in $\{\mathcal{D}_i\}_{i\in I}$ according to the application of $Hr \in \mathbb{H}$ that these rules translate. For each such group we apply one bottom rule below the partial derivations in which the top rules of the group occur. As shown in Example 4.14, due to the duplication of context sequents in hypersequent rules (that we handle using dummy bottom rules), we may need to apply a single bottom rule below groups of top rules translating different hypersequent rules. In particular, this happens when a hypersequent rule application (r) with more than one premiss has an active component C_0 and some context components $C_1, \dots, C_n$ in the conclusion, and two hypersequent rule applications (h') and (h'') have active components including different ancestors of some C_i with $0 \le i \le n$. In this case, the top rules translating (h') and (h'') occur above different premisses of a non-dummy rule with conclusion C_0 (just like the two applications of (com_2) in Example 4.14) and of some dummy bottom rules with conclusions $C_1, \dots, C_n$ (just like the two applications of (com_1) in Example 4.14). When we apply a bottom rule for such a group of top rules we obtain a *mixed 2-system*, i.e. a 2-system that contains more than one group of top rules translating different hypersequent rule applications.

We show that we can replace each mixed 2-system by regular 2-systems. First notice that

(1) two top rule applications belonging to the same mixed 2-system cannot occur on the same path of the derivation tree,
(2) if we remove all premisses but one from a dummy bottom rule in a partial derivation we still obtain a partial derivation,
(3) every time a pair of top rules translating different hypersequent rule applications occur in the same mixed 2-system above different premisses of a non-dummy rule, all other pairs of top rules translating these two hypersequent rule applications occur above different premisses of dummy bottom rules.

From (1) and (2) it follows that if two top rules occur above different premisses of a dummy bottom rule, we can remove one of them from the partial derivation containing the other. If we do so, we say that we *split* the dummy bottom rule.

Consider now a mixed 2-system

$$\dfrac{\begin{array}{c}\mathcal{D}_1\\ \vdots\\ \Gamma \Rightarrow \Delta\end{array} \quad \dots \quad \begin{array}{c}\mathcal{D}_k\\ \vdots\\ \Gamma \Rightarrow \Delta\end{array}}{\Gamma \Rightarrow \Delta}$$

where the derivation $\mathcal{D}_i$, for $1 \le i \le k$, contains the rule applications $(r_i^1), \dots, (r_i^n)$. We adopt the convention that the rules with same superscript index translate the same hypersequent rule.

To replace such mixed 2-system with regular 2-systems we proceed as follows. First we replace the mixed 2-system with a 2-system for the group of top rules with superscript 1:

$$\dfrac{\begin{matrix}\mathcal{D}'_1 \\ \vdots \\ \Gamma \Rightarrow \Delta\end{matrix} \quad \dots \quad \begin{matrix}\mathcal{D}'_k \\ \vdots \\ \Gamma \Rightarrow \Delta\end{matrix}}{\Gamma \Rightarrow \Delta}\ (b^1)$$

where $\mathcal{D}'_1, \dots, \mathcal{D}'_k$ only contain the rules $(r^1_1), \dots, (r^1_k)$ and those top rules that cannot be removed from the partial derivations by splitting dummy bottom rules (if we need to choose, we pick the top rules with minimum superscript index). After this, we introduce further bottom rules as follows

$$\dfrac{\dfrac{\begin{matrix}\mathcal{D}'_1 \\ \vdots \\ \Gamma \Rightarrow \Delta\end{matrix} \quad \begin{matrix}\mathcal{D}''_2 \\ \vdots \\ \Gamma \Rightarrow \Delta\end{matrix} \quad \dots \quad \begin{matrix}\mathcal{D}''_k \\ \vdots \\ \Gamma \Rightarrow \Delta\end{matrix}}{\Gamma \Rightarrow \Delta}\ (b^2) \quad \dots \quad \dfrac{\begin{matrix}\mathcal{D}''_1 \\ \vdots \\ \Gamma \Rightarrow \Delta\end{matrix} \quad \dots \quad \begin{matrix}\mathcal{D}''_{k-1} \\ \vdots \\ \Gamma \Rightarrow \Delta\end{matrix} \quad \begin{matrix}\mathcal{D}'_k \\ \vdots \\ \Gamma \Rightarrow \Delta\end{matrix}}{\Gamma \Rightarrow \Delta}\ (b^2)}{\Gamma \Rightarrow \Delta}\ (b^1)$$

where the bottom rules (b^2) are only introduced below the branches $\mathcal{D}'_1, \dots, \mathcal{D}'_k$ containing some of the rules $(r^2_1), \dots, (r^2_k)$, and the derivations $\mathcal{D}''_1, \dots, \mathcal{D}''_k$ are copies of $\mathcal{D}_1, \dots, \mathcal{D}_k$ only containing $(r^2_1), \dots, (r^2_k)$ and those top rules that cannot be removed by splitting dummy bottom rules. We keep duplicating the derivation in such way until either we do not need any more bottom rules or we introduced bottom rules for all superscript indices $1, \dots, n$. Given that we can add bottom rules for all groups of top rules in the mixed 2-system, in order to be sure that the result does not contain any mixed 2-system we only need to show that we never add a top rule application above the wrong premiss of its bottom rule. For the sake of contradiction suppose that we do. We add a top rule application (r^i_p) above a wrong premiss of its bottom rule only if we just introduced a new bottom rule (b^j), for $i < j \leq n$, and we cannot remove (r^i_p) – by splitting a dummy bottom rule – from the derivation containing a top rule (r^j_p) that we need in the branch that we are considering. But if we cannot remove (r^i_p) from the partial derivation containing (r^j_p), by (3) we can remove any (r^j_q) from any partial derivation containing any (r^i_q), as long as $q \neq p$. Given that the bottom rule (b^i) occurs below (b^j), it follows that there is no top rule (r^j_q) on this branch of the bottom rule (b^i). By (1) we can rule out the involvement of 2-system instances different from i and j, and hence we can infer that (r^j_p) is not needed and we do not need to add (r^i_p) in the first place, contrarily to the assumptions.

Notice that the procedure does not require all groups of top rules to have exactly k elements. If, for example, the group with superscript index i contains l top rule applications for $l < k$, then the bottom rules for i will have l premisses. This does not influence any other group of top rules.

Thus, we eventually obtain an LJ + $\mathbb{S}$ derivation of $\Gamma \Rightarrow \Pi$. □

4.2.1 Normal forms of hypersequent derivations. In the previous algorithm we only considered hypersequent derivations in structured form, i.e. in which (EC) applications occur immediately above the root and (EW) applications occur where needed. Here we show how to transform each hypersequent derivation into a derivation in structured form.

Definition 4.16. The *external contraction rank (ec-rank)* of an application E of (EC) in a derivation is the number of applications of rules other than (EC) between E and the root of the derivation.

LEMMA 4.17. *Each* HLJ + $\mathbb{H}$ *derivation* $\mathcal{D}$ *can be transformed into a derivation of the same end-hypersequent in which all* (EC) *applications have ec-rank* 0.

Proof. Proceed by double induction on the lexicographically ordered pair $\langle \mu, \nu \rangle$, where μ is the maximum ec-rank of any (EC) application in $\mathcal{D}$, and ν is the number of (EC) applications in $\mathcal{D}$ with maximum ec-rank.

Base case. If $\mu = 0$ the claim trivially holds.

Inductive step. Assume that $\mathcal{D}$ has maximum ec-rank μ and that there are ν applications of the rule (EC) with ec-rank μ. We show how to transform $\mathcal{D}$ into a derivation $\mathcal{D}'$ having either maximum ec-rank $\mu' < \mu$ or ec-rank μ and number of (EC) applications with maximum ec-rank $\nu' < \nu$.

Consider an (EC) application with ec-rank μ in $\mathcal{D}$ and the queue of (EC) containing it. There cannot be any applications of (EC) above this queue because the ec-rank of its elements is maximal. We distinguish cases according to the rule (r) applied to the conclusion of the last element of such queue.

Assume that (r) has one premiss. If $(r) = (EW)$, we apply (EW) (with the same active component) before the queue. If $(r) \neq (EW)$, we apply (r) immediately before the queue, possibly followed by applications of (EC).

Notation. Given a hypersequent H we denote by $(H)^u$ the hypersequent $H \mid \dots \mid H$ containing u copies of H ($u \geq 0$).

Let (r) be a(ny external) context-sharing rule with more than one premiss and consider any subderivation of $\mathcal{D}$ of the form

$$\dfrac{\dfrac{\dfrac{\begin{matrix}\mathcal{D}_1\\ \vdots\\ G \mid G'_1 \mid (C_1)^{m_1}\end{matrix}}{\vdots}\,(EC)}{G \mid C_1}\,(EC) \qquad \dots \qquad \dfrac{\dfrac{\begin{matrix}\mathcal{D}_n\\ \vdots\\ G \mid G'_n \mid (C_n)^{m_n}\end{matrix}}{\vdots}\,(EC)}{G \mid C_n}\,(EC)}{G \mid H}\,(r)$$

where G'_i, for $1 \leq i \leq n$, only contains components in G and the derivations $\mathcal{D}_1, \dots, \mathcal{D}_n$ contain no application of (EC). We can transform $\mathcal{D}$ into a derivation $\mathcal{D}'$ in which all applications of (EC) occurring above the hypersequent $G \mid H$ are either immediately above it or immediately above another application of (EC); their ec-rank is reduced by 1 because (r) does not occur below them anymore.

We first prove that $(\star)$ the hypersequent $G \mid G'' \mid (H)^q$, where $G'' = G'_1 \mid \dots \mid G'_n$ and $q = (\sum_{i=1}^{n}(m_i - 1)) + 1$ is derivable from

$$G \mid G'_1 \mid (C_1)^{m_1} \;,\; \dots \;,\; G \mid G'_n \mid (C_n)^{m_n}$$

using only (EW) and (r). The hypersequent $G \mid H$ then follows from $G \mid G'' \mid (H)^q$ by (EC) as all the components of G'' occur also in G. The obtained derivation $\mathcal{D}'$ has maximum ec-rank $\mu' < \mu$, or the occurrences of (EC) with ec-rank μ occurring in it are $\nu' < \nu$.

It remains to prove claim $(\star)$. We have a derivation of any element of the set

$$\mathbb{Q} = \{G \mid G'' \mid (H)^0 \mid (C_1)^{x_1} \mid \dots \mid (C_n)^{x_n} \;:\; \sum_{i=1}^{n} x_i \;=\; (\sum_{i=1}^{n}(m_i - 1)) + 1\}$$

from the hypersequents $G \mid G'_1 \mid (C_1)^{m_1} \;,\; \dots \;,\; G \mid G'_n \mid (C_n)^{m_n}$ using only (EW). Indeed for any hypersequent in $\mathbb{Q}$ and for $1 \leq i \leq n$, there is at least one $x_i \geq m_i$, because otherwise $\sum_{i=1}^{n} x_i < (\sum_{i=1}^{n}(m_i - 1)) + 1$. The claim $(\star)$ therefore follows by Lemma 4.18 below being $G \mid G'' \mid (H)^q$ the only element of the set

$$\mathbb{Q}' = \{G \mid G'' \mid (H)^q \mid (C_1)^{x_1} \mid \dots \mid (C_n)^{x_n} \;:\; \sum_{i=1}^{n} x_i \;=\; 0\}$$

for $q = \left(\sum_{i=1}^{n}(m_i - 1)\right) + 1$. □

The following is the central lemma of the previous proof.

LEMMA 4.18. *For any application of a hypersequent rule*

$$\frac{G\,|\,C_1 \qquad \dots \qquad G\,|\,C_n}{G\,|\,H}\;(r)$$

and natural number $d \geq 0$, consider the set of hypersequents

$$\mathbb{L}_d = \{G\,|\,(H)^c\,|\,(C_1)^{x_1}\,|\,\dots\,|\,(C_n)^{x_n} \;:\; \sum_{i=1}^{n} x_i = d\}$$

where G, H are hypersequents, $C_1, \dots, C_n$ sequents, and c is a natural number. For any natural number e, s.t. $0 \leq e \leq d$, each element of the set

$$\mathbb{L}_{(d-e)} = \{G\,|\,(H)^{c+e}\,|\,(C_1)^{x'_1}\,|\,\dots\,|\,(C_n)^{x'_n} \;:\; \sum_{i=1}^{n} x'_i = d-e\}$$

is derivable from hypersequents in $\mathbb{L}_d$ by repeatedly applying the rule (r).

PROOF. By induction on e.
Base case: If $e = 0$, then $\mathbb{L}_d = \mathbb{L}_{d-e}$.
Inductive step: Assume that $e > 0$ and that the claim holds for all $e' < e$. By induction hypothesis there exists a derivation from the hypersequents in $\mathbb{L}_d$ for each element of the set

$$\mathbb{L}_{(d-(e-1))} = \{G\,|\,(H)^{c+(e-1)}\,|\,(C_1)^{x''_1}\,|\,\dots\,|\,(C_n)^{x''_n} \;:\; \sum_{i=1}^{n} x''_i = d-(e-1)\}$$

that only consists of applications of (r). Any hypersequent

$$G\,|\,(H)^{c+e}\,|\,(C_1)^{x'_1}\,|\,\dots\,|\,(C_n)^{x'_n}$$

in $\mathbb{L}_{(d-e)}$ can be derived from elements of $\mathbb{L}_{(d-(e-1))}$ as follows:

$$\frac{G\,|\,(H)^{c+(e-1)}\,|\,H'_1 \qquad \dots \qquad G\,|\,(H)^{c+(e-1)}\,|\,H'_n}{G\,|\,(H)^{c+e}\,|\,(C_1)^{x'_1}\,|\,\dots\,|\,(C_n)^{x'_n}}\;(r)$$

where, for $1 \leq i \leq n$, $H'_i \;=\; (C_1)^{y_1}\,|\,\dots\,|\,(C_n)^{y_n}$ is such that if $j \neq i$ then $y_j = x'_j$ and if $j = i$ then $x'_j + 1$; i.e., the components $C_1, \dots, C_n \notin G$ occur in the i^{th} premiss as many times as in the conclusion, except for C_i which occurs one more time.

All premisses of this rule application are hypersequents in $\mathbb{L}_{(d-(e-1))}$, indeed

$$(x'_1 + 1) + x'_2 + \dots + x'_n \;=\; \dots \;=\; x'_1 + \dots + x'_{n-1} + (x'_n + 1) \;=\; \left(\sum_{i=1}^{n} x'_i\right) + 1$$

and

$$\left(\sum_{i=1}^{n} x'_i\right) + 1 \;=\; (d-e) + 1 \;=\; d-(e-1)$$

Given that only the rule (r) is used to derive the elements of $\mathbb{L}_{d-(e-1)}$ from the elements of $\mathbb{L}_d$, also the elements of $\mathbb{L}_{(d-e)}$ can be derived from those of $\mathbb{L}_d$ by applying only (r). □

LEMMA 4.19. *Any* HLJ + $\mathbb{H}$ *derivation of a sequent can be transformed into a derivation in structured form.*

Proof. Let $\mathcal{D}$ be a hypersequent derivation of a sequent S in HLJ + $\mathbb{H}$. By Lemma 4.17 we can assume that all applications of (EC) in $\mathcal{D}$ occur in a queue immediately above S. Consider an application of (EW), with premiss G and conclusion $G\,|\,C$, which is not as in Definition 4.10. First notice that $G\,|\,C$ cannot be the root of $\mathcal{D}$. We show how to shift this application of (EW) below other rule applications until the statement is satisfied for such application. Three cases can arise:

(1) C is the active component in the premiss of an application of a rule (r). The conclusion of (r) is simply obtained by applying (EW) (possibly multiple times) to G.
(2) C is a context component in the premiss of an application of a one-premiss rule (r). The (EW) is simply shifted below (r).
(3) C occurs actively inside the queues of (EW) above all the premisses of an application of a rule (r). We remove all the applications of (EW) with active component C in the queues and apply (r) with one context component less, followed by (EW).

The termination of the procedure follows from the fact that $\mathcal{D}$ is finite and that (1)–(3) always reduce the number of rules different from (EW) occurring below the (EW) applications. □

5 APPLICATIONS OF THE EMBEDDINGS

We provided constructive transformations from hypersequent derivations to 2-system derivations and back. These transformations show that the two seemingly different proof frameworks have the same expressive power. The embeddings are not only interesting for their conceptual outcomes, they also have applications that are concretely beneficial to both 2-systems and hypersequents.

5.1 For 2-systems

The benefits of the embeddings with respect to 2-systems include: (i) new cut-free 2-systems, (ii) analyticity proofs, and (iii) locality of derivations using the hypersequent notation.

(i) and (ii) rely on the method in [8] to transform propositional Hilbert axioms in the language of Full Lambek calculus into suitable hypersequent rules. In a nutshell, the method – below described for the case of intermediate logics – is based on the following classification of intuitionistic formulae: $\mathcal{N}_0$ and $\mathcal{P}_0$ are the set of atomic formulae

$$\begin{array}{ll} \mathcal{P}_{n+1} & ::= \bot \,|\, \top \,|\, \mathcal{N}_n \,|\, \mathcal{P}_{n+1} \wedge \mathcal{P}_{n+1} \,|\, \mathcal{P}_{n+1} \vee \mathcal{P}_{n+1} \\ \mathcal{N}_{n+1} & ::= \bot \,|\, \top \,|\, \mathcal{P}_n \,|\, \mathcal{N}_{n+1} \wedge \mathcal{N}_{n+1} \,|\, \mathcal{P}_{n+1} \rightarrow \mathcal{N}_{n+1} \end{array}$$

Remark. The classes $\mathcal{P}_n$ and $\mathcal{N}_n$ contain axioms with leading positive and negative connective, respectively. Recall that a connective is positive (negative) if its left (right) logical rule is invertible [1]; note that in HLJ, $\vee$ is positive, $\rightarrow$ is negative and $\wedge$ is both positive and negative.

As shown in [8] all axioms within the class $\mathcal{P}_3$ can be algorithmically transformed into equivalent *structural* hypersequent rules that are analytic, i.e. that preserve cut-elimination when added to the calculus HLJ. For instance the rule (com) in Example 2.1 can be (automatedly[2]) extracted from the linearity axiom. Furthermore [8] shows how to transform any structural hypersequent rule into an equivalent analytic rule.

Ad (i): the method in [19] rewrites generalised geometric formulae in the class GA_1 into analytic 2-systems. Such formulae follow the schema

$$GA_1 \;\equiv\; \forall \overline{x}(\bigwedge P \rightarrow \exists \overline{y}_1 \bigwedge GA_0 \vee \ldots \vee \exists \overline{y}_m \bigwedge GA_0)$$

Here $\overline{x}, \overline{y}_1, \ldots, \overline{y}_m$ are tuples of first order variables, $\bigwedge P$ is a finite conjunction of atomic formulae, the variables in $\overline{y}_i$ for any i do not occur free in $\bigwedge P$, and $\bigwedge GA_0$ is a finite conjunction of formulae of the form $\forall \overline{x}(\bigwedge P \rightarrow \exists \overline{y}_1 \bigwedge P_1 \vee \ldots \vee \exists \overline{y}_m \bigwedge P_m)$ where the same conditions apply, and $\bigwedge P_j$ is a

[2]Program at https://www.logic.at/tinc/webaxiomcalc/

conjunction of atomic formulae for any j. As observed in [19], formulas in GA_1 need not contain quantifier alternations; indeed there are purely propositional axioms that are in GA_1 but not in GA_0. Notice that the propositional axioms in GA_1 are strictly contained in the class $\mathcal{P}_3$ of [8]. For the strictness of the inclusion, consider the axiom $\neg\alpha \vee \neg\neg\alpha$. If we write, as usual, $\neg\varphi$ as $\varphi \to \bot$, this axiom belongs to $\mathcal{P}_3$ but *not* to GA_1. Hence when applied to $\neg\alpha \vee \neg\neg\alpha$ the method in [19] does not lead to a 2-system, which can instead be defined by translating the hypersequent rule equivalent to the axiom (below left) into the equivalent 2-system (below right):

$$\frac{G \mid \Sigma, \Sigma' \Rightarrow}{G \mid \Sigma \Rightarrow \;\mid \Sigma' \Rightarrow}\ (lq) \qquad\qquad \dfrac{\begin{array}{c} \dfrac{}{\Sigma \Rightarrow} \\ \vdots \\ \Gamma \Rightarrow \Pi \end{array} \qquad \begin{array}{c} \dfrac{\Sigma, \Sigma' \Rightarrow}{\Sigma' \Rightarrow} \\ \vdots \\ \Gamma \Rightarrow \Pi \end{array}}{\Gamma \Rightarrow \Pi}$$

Ad (ii): The analiticity proof in [19] relies on the fact that the obtained 2-systems manipulate atomic formulae only; this is the case for labelled 2-systems arising from frame conditions, but it does not hold anymore when translating axiom schemata, e.g. the axiom $(\varphi \to \psi) \vee (\psi \to \varphi)$ for Gödel logic (cf. Example 2.1). In this case, and for all propositional Hilbert axioms within the class GA_1, analyticity for the 2-systems obtained by the method in [19] can be recovered by (a) first translating them into hypersequent rules, (b) applying the *completion* procedure in [8] to the latter, and (c) translating them back.

Example 5.1. We show the transformation of a 2-system into an analytic 2-system. Consider the law of excluded middle $\varphi \vee \neg\varphi \in GA_1$. The method in [19] transforms it into the 2-system (below left), which is translated into the hypersequent rule (below right) following the procedure in Section 3:

$$\dfrac{\begin{array}{c} \dfrac{\varphi, \Gamma_1 \Rightarrow \Delta_1}{\Gamma_1 \Rightarrow \Delta_1} \\ \vdots \\ \Gamma \Rightarrow \Delta \end{array} \qquad \begin{array}{c} \dfrac{\bot, \Gamma_2 \Rightarrow \Delta_2}{\varphi, \Gamma_2 \Rightarrow \Delta_2} \\ \vdots \\ \Gamma \Rightarrow \Delta \end{array}}{\Gamma \Rightarrow \Delta} \qquad\qquad \frac{G \mid \varphi, \Gamma_1 \Rightarrow \Delta_1 \quad G \mid \bot, \Gamma_2 \Rightarrow \Delta_2}{G \mid \Gamma_1 \Rightarrow \Delta_1 \mid \varphi, \Gamma_2 \Rightarrow \Delta_2}$$

Using the results in [8] we complete the latter rule and obtain the analytic hypersequent rule (below left), whose translation leads to the 2-system below right:

$$\frac{G \mid \Sigma, \Gamma_1 \Rightarrow \Pi_1}{G \mid \Gamma_1 \Rightarrow \Pi_1 \mid \Sigma, \Gamma_2 \Rightarrow \Pi_2} \qquad\qquad \dfrac{\begin{array}{c} \dfrac{\Sigma, \Gamma_1 \Rightarrow \Pi_1}{\Gamma_1 \Rightarrow \Pi_1} \\ \vdots \\ \Gamma \Rightarrow \Pi \end{array} \qquad \begin{array}{c} \dfrac{}{\Sigma, \Gamma_2 \Rightarrow \Pi_2} \\ \vdots \\ \Gamma \Rightarrow \Pi \end{array}}{\Gamma \Rightarrow \Pi}$$

The analiticity of LJ extended with the obtained system of rules follows from Theorem 4.15.

5.2 For hypersequent calculi

We show below how to use the embeddings to reformulate hypersequent calculi as natural deduction systems inheriting the simplicity of Gentzen's natural deduction calculus NJ for intuitionistic logic (see, e.g., [22]).

Such reformulation is a step forward to prove the connection, suggested in [4], between intermediate logics formalised as cut-free hypersequent systems and parallel λ-calculi. An attempt to reveal this connection is the natural deduction calculus introduced in [7] for Gödel logic, one of the main intermediate logics. Following [6], this calculus deals with parallel intuitionistic derivations connected by a symbol $*$; this new deduction structure mirroring the hypersequent separator hinders however the definition of a corresponding λ-calculus by Curry–Howard isomorphism.

Our reformulation of hypersequent calculi as natural deduction systems is modular, and simply obtained by adding to Gentzen's NJ *higher-level rules* simulating hypersequent rules acting on several components. The transformation from hypersequent derivations into 2-systems allows us to reformulate the former without using $|$-separated components and without the need of (EC), which is internalised by the bottom rules of the 2-systems. The resulting derivations are close to natural deduction.

To present the transformation in a simple way, henceforth we consider hypersequent rules of the following form:

$$\frac{M_1 \qquad \dots \qquad M_k}{G \mid \Sigma^1_1, \dots, \Sigma^1_{n_1}, \Gamma_1 \Rightarrow \Pi_1 \mid \dots \mid \Sigma^k_1, \dots, \Sigma^k_{n_k}, \Gamma_k \Rightarrow \Pi_k} \ (Hr)$$

where, for any $1 \leq i \leq k$, M_i is a (possibly empty) set of hypersequents of the form $G \mid \Delta^i_j, \Gamma_i \Rightarrow \Pi_i$, for some j, with $\Delta^i_j = \Sigma^p_q$ for some $1 \leq p \leq k$ and $1 \leq q \leq n_p$, and with Γ_i and Π_i non-empty.

These rules arise by applying the algorithm in [8] to $\mathcal{P}_3$ formulae (cf. the grammar in Section 5.1) of the following form[3]:

$$((\sigma^1_1 \wedge \dots \wedge \sigma^1_{n_1}) \rightarrow (\delta^1_1 \vee \dots \vee \delta^1_{m_1})) \vee \dots \vee ((\sigma^k_1 \wedge \dots \wedge \sigma^k_{n_k}) \rightarrow (\delta^k_1 \vee \dots \vee \delta^k_{m_k}))$$

where σ^i_j and δ^i_j are schematic variables and $(\delta^i_1 \vee \dots \vee \delta^i_{m_i})$ is $\bot$, if $m_i = 0$. Henceforth we will refer to this formula as the axiom *associated* to the rule (Hr). As shown in [8], HLJ extended with (Hr) is *equivalent* to HLJ extended with its associated axiom – that is, their derivability relations coincide.

Example 5.2. $\mathcal{P}_3$ formulae of the above form are, e.g., the linearity axiom $(\varphi \rightarrow \psi) \vee (\psi \rightarrow \varphi)$ (see Example 2.1), the law of excluded middle, and the axioms (Bck) characterizing the intermediate logics with k worlds, $k \geq 1$, $\varphi_0 \vee (\varphi_0 \rightarrow \varphi_1) \vee \dots \vee (\varphi_0 \wedge \dots \wedge \varphi_{k-1} \rightarrow \varphi_k)$. Also the formulae in [17] for implicational logics and the disjunctive tautologies in [10] are of this form; the former paper introduces natural deduction calculi for some intermediate logics with no normalisation procedure while the latter interprets the disjunctive tautologies as synchronisation protocols within the Curry–Howard correspondence framework.

The above hypersequent rule (Hr) is transformed by the embedding in Section 3 into the following 2-system

$$\dfrac{\begin{array}{c}\dfrac{M_1}{\Sigma^1_1, \dots, \Sigma^1_{n_1}, \Gamma_1 \Rightarrow \Pi_1}\ Tr_1 \\ \vdots \\ \Gamma \Rightarrow \Pi\end{array} \qquad \dots \qquad \begin{array}{c}\dfrac{M_k}{\Sigma^k_1, \dots, \Sigma^k_{n_k}, \Gamma_k \Rightarrow \Pi_k}\ Tr_k \\ \vdots \\ \Gamma \Rightarrow \Pi\end{array}}{\Gamma \Rightarrow \Pi}$$

[3]In the general case, $\mathcal{P}_3$ formulae correspond to hypersequent rules with the same form as Hr but with more than one Δ^i_j in each premiss.

which is translated into a natural deduction rule *Nr* of the following form

$$\dfrac{\dfrac{\sigma_1^1 \quad \dots \quad \sigma_{n_1}^1 \quad \begin{array}{c}[\delta_1^1]\\ \vdots\\ \varphi_1\end{array} \quad \dots \quad \begin{array}{c}[\delta_{m_1}^1]\\ \vdots\\ \varphi_1\end{array}}{\begin{array}{c}\varphi_1\\ \vdots\\ \varphi\end{array}} \qquad \dots \qquad \dfrac{\sigma_1^k \quad \dots \quad \sigma_{n_k}^k \quad \begin{array}{c}[\delta_1^k]\\ \vdots\\ \varphi_k\end{array} \quad \dots \quad \begin{array}{c}[\delta_{m_k}^k]\\ \vdots\\ \varphi_k\end{array}}{\begin{array}{c}\varphi_k\\ \vdots\\ \varphi\end{array}}}{\varphi} \tag{1}$$

where σ_j^i corresponds to Σ_j^i and δ_j^i corresponds to $\bot$ if $M_i = \varnothing$ and to Δ_j^i otherwise.

When an upper inference has only one or no δ_j^i we we can simplify the notation as in the following examples.

Remark. These rules are higher-level rules à la Schroeder-Heister [25], indeed they also discharge rule applications rather than only formulae. To make this more evident, we denote them by $\ast$.

Example 5.3. The hypersequent rule for the linearity axiom $(\delta \to \sigma) \vee (\sigma \to \delta)$ below left (see Example 2.3 for the corresponding 2-system) is translated into the natural deduction rule below right:

$$\dfrac{G \mid \sigma, \Gamma_1 \Rightarrow \Pi_1 \qquad G \mid \delta, \Gamma_2 \Rightarrow \Pi_2}{G \mid \delta, \Gamma_1 \Rightarrow \Pi_1 \mid \sigma, \Gamma_2 \Rightarrow \Pi_2} \qquad\qquad \dfrac{\begin{array}{c}\dfrac{\delta}{\sigma}\\ \vdots\\ \varphi\end{array} \quad \begin{array}{c}\dfrac{\sigma}{\delta}\\ \vdots\\ \varphi\end{array}}{\varphi}$$

Using this rule, the linearity axiom can be derived as follows

$$\dfrac{\dfrac{\dfrac{\dfrac{[\delta]^1}{\sigma}\,\ast}{\delta \to \sigma}\,1}{(\delta \to \sigma) \vee (\sigma \to \delta)} \quad \dfrac{\dfrac{\dfrac{[\sigma]^2}{\delta}\,\ast}{\sigma \to \delta}\,2}{(\delta \to \sigma) \vee (\sigma \to \delta)}}{(\delta \to \sigma) \vee (\sigma \to \delta)}\,\ast$$

The addition to NJ of the resulting natural deduction rule yields the calculus NG for Gödel logic, whose normalisation and Curry–Howard correspondence have been shown in [2].

Example 5.4. The hypersequent rule below left for the law of excluded middle $\sigma \vee \neg\sigma$ (see Example 5.1 for the corresponding 2-system) translates into the natural deduction rule below right:

$$\dfrac{G \mid \Sigma, \Gamma_1 \Rightarrow \Pi_1}{G \mid \Gamma_1 \Rightarrow \Pi_1 \mid \Sigma, \Gamma_2 \Rightarrow \Pi_2} \qquad\qquad \dfrac{\begin{array}{c}\dfrac{\sigma}{\bot}\\ \vdots\\ \varphi\end{array} \quad \begin{array}{c}[\sigma]\\ \vdots\\ \varphi\end{array}}{\varphi}$$

We can derive the law of excluded middle using this rule as follows

$$\dfrac{\dfrac{\dfrac{\dfrac{[\sigma]^1}{\bot}\,\ast}{\neg\sigma}\,1}{\sigma \vee \neg\sigma} \quad \dfrac{[\sigma]^\ast}{\sigma \vee \neg\sigma}}{\sigma \vee \neg\sigma}\,\ast$$

We show now that a hypersequent rule (Hr) and the corresponding natural deduction rule *Nr* are equivalent, i.e. that $\vdash_{\mathrm{HLJ}+Hr} \varphi$ if and only if $\vdash_{\mathrm{NJ}+Nr} \varphi$.

Theorem 5.5. HLJ *extended with any hypersequent rule (Hr) is equivalent to* NJ *extended with its translated rule Nr.*

Proof. We show that if $\vdash_{\mathrm{HLJ}+Hr} \varphi$ then $\vdash_{\mathrm{NJ}+Nr} \varphi$. Indeed a derivation of the axiom r_α associated to (Hr) is as follows:

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[\sigma_1^1 \wedge \ldots \wedge \sigma_{n_1}^1]^1}{\sigma_1^1} \quad \ldots \quad \dfrac{[\sigma_1^1 \wedge \ldots \wedge \sigma_{n_1}^1]^1}{\sigma_{n_1}^1} \quad \dfrac{[\delta_1^1]^2}{\delta_1^1 \vee \ldots \vee \delta_{m_1}^1} \quad \ldots \quad \dfrac{[\delta_{m_1}^1]^2}{\delta_1^1 \vee \ldots \vee \delta_{m_1}^1}}{\delta_1^1 \vee \ldots \vee \delta_{m_1}^1}\,2*}{(\sigma_1^1 \wedge \ldots \wedge \sigma_{n_1}^1) \to (\delta_1^1 \vee \ldots \vee \delta_{m_1}^1)}\,1}{r_\alpha} \quad \ldots \quad \begin{matrix}\vdots \\ r_\alpha\end{matrix}}{r_\alpha}\,*
$$

All hypotheses are derived as shown for the leftmost. The rest of the premisses of the bottom-most inference are derived similarly using the implications

$$
(\sigma_1^2 \wedge \ldots \wedge \sigma_{n_2}^2) \to (\delta_1^2 \vee \ldots \vee \delta_{m_2}^2)\,, \ldots, \,(\sigma_1^k \wedge \ldots \wedge \sigma_{n_k}^k) \to (\delta_1^k \vee \ldots \vee \delta_{m_k}^k)
$$

The claim follows by the equivalence between r_α and (Hr) shown in [8].

To show that if $\vdash_{\mathrm{NJ}+Nr} \varphi$ then $\vdash_{\mathrm{HLJ}+Hr} \varphi$, we derive the rule Nr using the rules of NJ and r_α. We can then easily exploit the equivalence between HLJ and NJ. Intuitively, we use conjunction and implication elimination to simulate the upper inferences of Nr (top left part of the following derivation). Then we nest one disjunction elimination ($\vee E$) for each disjunctive subformula of the axiom in order to discharge the implications used above, discharge the formulae δ_j^i, and derive $\varphi, \varphi_1, \ldots, \varphi_k$:

$$
\dfrac{\alpha \qquad \begin{matrix} \dfrac{\dfrac{[(\sigma_1^1 \wedge \ldots \wedge \sigma_{n_1}^1) \to (\delta_1^1 \vee \ldots \vee \delta_{m_1}^1)]^1 \qquad \dfrac{\sigma_1^1 \qquad \dfrac{\sigma_2^1 \qquad \begin{matrix}\vdots \\ \sigma_3^1 \wedge \ldots \wedge \sigma_{n_1}^1\end{matrix}}{\sigma_2^1 \wedge \ldots \wedge \sigma_{n_1}^1}}{\sigma_1^1 \wedge \ldots \wedge \sigma_{n_1}^1}}{\delta_1^1 \vee \ldots \vee \delta_{m_1}^1} \qquad \begin{matrix}[\delta_1^1]^2 \\ \vdots \\ \varphi_1\end{matrix} \qquad \begin{matrix}\vdots \\ \varphi_1\end{matrix}}{\varphi_1}\,\vee E^2 \\ \vdots \\ \varphi \end{matrix} \qquad \begin{matrix}\vdots \\ \varphi\end{matrix}}{\varphi}\,\vee E^1
$$

The open hypotheses here are the formulae $\sigma_1^1, \ldots, \sigma_{n_1}^1, \ldots, \sigma_1^k, \ldots, \sigma_{n_k}^k$, which are exactly the hypotheses of Nr. The claim follows by the equivalence between r_α and (Hr) shown in [8]. ◻

Final Remark. The analiticity of the introduced natural deduction calculi could be proved by exploiting the connection with the corresponding cut-free hypersequent calculi. A computational interpretation of the former calculi calls however for a direct normalisation procedure and an interpretation of its reduction rules as meaningful operations in suitable λ-calculi.

The case study of Gödel logic (see Example 2.1) has been detailed in [2], where we proved normalisation and the subformula property for its natural deduction calculus NG in Example 5.3. Based on this calculus, [2] introduces indeed an extension of simply-typed λ-calculus with a parallel operator that supports higher-order communications between processes. The resulting functional language is strictly more expressive than simply-typed λ-calculus. Inspired by hypersequent cut-elimination, the key reductions to prove the analiticity of NG model a symmetric message exchange and process migration mechanism handling the bindings between code fragments and their computational environments.

APPENDIX

Lemma 4.6. Any 2-system derivation $\mathcal{P}$ can be transformed into a 2-system derivation $\mathcal{P}'$ of the same end-sequent in which no entanglement occur.

Proof. First we introduce a transformation of derivations (*e-reduction*) that reduces the number of top rule applications involved in entanglements. Then we provide a strategy to obtain the desired derivation $\mathcal{P}'$ using such transformation, and we prove termination.
E-reduction: given a 2-system instance S (with bottom rule (B_S)) entangled with 2-system instances $S_1, \dots, S_n$:

$$\dfrac{\begin{array}{c}\mathcal{D}_1\\ \vdots\\ \Gamma \Rightarrow \Pi\end{array} \quad \dots \quad \begin{array}{c}\mathcal{D}_n\\ \vdots\\ \Gamma \Rightarrow \Pi\end{array}}{\Gamma \Rightarrow \Pi}\,(B_S)$$

we make two copies S' and S'' of S with bottom rules $(B_{S'})$ respectively $(B_{S''})$:

$$\dfrac{\dfrac{\begin{array}{c}\mathcal{D}_1'\\ \vdots\\ \Gamma \Rightarrow \Pi\end{array} \quad \begin{array}{c}\mathcal{D}_2\\ \vdots\\ \Gamma \Rightarrow \Pi\end{array} \quad \dots \quad \begin{array}{c}\mathcal{D}_m\\ \vdots\\ \Gamma \Rightarrow \Pi\end{array}}{\Gamma \Rightarrow \Pi}\,(B_{S'}) \quad \begin{array}{c}\mathcal{D}_2\\ \vdots\\ \Gamma \Rightarrow \Pi\end{array} \quad \dots \quad \begin{array}{c}\mathcal{D}_n\\ \vdots\\ \Gamma \Rightarrow \Pi\end{array}}{\Gamma \Rightarrow \Pi}\,(B_{S''})$$

in such a way that:

- if a top rule in $\mathcal{D}_1$ belonging to S occurs above a top rule of one among $S_1, \dots, S_n$, then its copy in $\mathcal{D}_1'$ belongs to S',
- if a top rule in $\mathcal{D}_1$ belonging to S occurs below a top rule of one among $S_1, \dots, S_n$, then its copy in $\mathcal{D}_1'$ belongs to S''.

Notice that in the obtained derivation no top rule of S' occurs below any top rule of $S_1, \dots, S_n$, and no top rule of S'' occurs above any top rule of $S_1, \dots, S_n$. Moreover, also due to Lemma 4.3:

$(*)$ neither S' and S'' nor two copies of the same 2-system instance in $\mathcal{D}_2, \dots, \mathcal{D}_n$ can be entangled or have top rules along the same path of the derivation.

A strategy to apply e-reductions that leads to the required derivation $\mathcal{P}'$ is the following. We start reducing one of the 2-system instances with lowermost bottom rule. Whenever we apply an e-reduction we collect all entangled copies of the same 2-system instance in the same class. We continue the disentanglement focusing on a single class and reducing all its elements before we move on to another class. Notice that the number of classes never increases and is bounded by the number of 2-system instances in the original derivation. Fixed a class, the strategy guarantees that its elements are disentangled one by one without duplicating other maximally entangled elements of the same class.

To formalise this strategy let us introduce some auxiliary notions. We define the equivalence relation $\sim$ as the transitive and symmetric closure of the binary relation that holds between a 2-system instance and any of its copies generated by an e-reduction – notice that e-reductions do not only copy S but also the 2-system instances in $\mathcal{D}_2, \dots, \mathcal{D}_n$. Given any 2-system derivation $\mathcal{P}$, let us denote by $E^{\mathcal{P}}$ the set of all entangled 2-system instances in $\mathcal{P}$, and by $E^{\mathcal{P}}/_{\sim}$ the quotient set of $E^{\mathcal{P}}$ w.r.t. the equivalence relation $\sim$. Moreover, we denote by S^{low} the 2-system instance in $E^{\mathcal{P}}$ which has the lowest and leftmost bottom rule in $\mathcal{P}$. Finally, we compute the *entanglement number* (*e-number* for short) of a 2-system instance S as follows: for each derivation $\mathcal{D}$ of a premiss of the bottom rule of S we count the number of equivalence classes containing 2-system instances that have top rules in $\mathcal{D}$ and are entangled with S, then we sum all the resulting numbers up to obtain the e-number of S.

We prove now the statement of the lemma by induction on the lexicographically ordered triple $\langle \kappa, \mu, \nu \rangle$ where, fixed the derivation $\mathcal{P}$,

- κ is the cardinality of $E^{\mathcal{P}}/_\sim$, i.e. the number of classes of entangled 2-system instances,
- μ is the maximum e-number of the elements of $[S^{low}]_\sim \in E^{\mathcal{P}}/_\sim$,
- ν is the number of elements of $[S^{low}]_\sim \in E^{\mathcal{P}}/_\sim$ with e-number μ.

Base case. If either κ, μ or ν are equal to 0, then no 2-system instance is entangled. Otherwise, first, $E^{\mathcal{P}}/_\sim$ would contain at least one element, and $e \geq 1$. Second, $[S^{low}]_\sim \in E^{\mathcal{P}}/_\sim$ would not be empty and both μ and ν would be greater than 0.
Inductive step. Given any 2-system derivation $\mathcal{P}$ with complexity $\langle \kappa, \mu, \nu \rangle \geq \langle 1, 1, 1 \rangle$ we transform it into a 2-system derivation $\mathcal{P}'$ with complexity smaller than $\langle \kappa, \mu, \nu \rangle$. We obtain $\mathcal{P}'$ applying an arbitrary e-reduction to an uppermost element $S \in [S^{low}]_\sim \in E^{\mathcal{P}}/_\sim$ with e-number μ.

First notice that we never increase κ. Moreover, if $\nu > 1$ we reduce ν without increasing μ and if $\nu = 1$ and $\mu > 1$ we reduce μ. Indeed, after the e-reduction all top rules of S that were involved in an entanglement with the elements of some class $[S']_\sim \in E^{\mathcal{P}}/_\sim$ above the same premiss of (B_S), are no more involved in such entanglement. This holds because, due to $(*)$ and the definition of $\sim$, the top rules of elements contained in $[S']_\sim \in E^{\mathcal{P}}/_\sim$ cannot occur along the same path of the derivation. In general we never increase neither μ nor ν, because if we duplicate a 2-system instance during an e-reduction, either it did not belong to $[S^{low}]_\sim \in E^{\mathcal{P}}/_\sim$ and hence the copies do not belong to $[S^{low}]_\sim \in E^{\mathcal{P}'}/_\sim$, or it did not have maximal entanglement number w.r.t. the class $[S^{low}]_\sim \in E^{\mathcal{P}}/_\sim$, because we always e-reduce a topmost 2-system instance among those with maximal e-number in $[S^{low}]_\sim$. Finally, we change the considered class $[S^{low}]_\sim$ only when it is empty, because our e-reduction strategy chooses S^{low} only if $[S^{low}]_\sim$ is a singleton. If $\nu = 1$ and $\mu = 1$ we reduce κ. Indeed, we replace the unique element of $[S^{low}]_\sim$ with non-entangled 2-system instances and $[S^{low}]_\sim$ does not belong to $E^{\mathcal{P}'}/_\sim$. □

REFERENCES

[1] Jean-Marc Andreoli. 1992. Logic programming with focusing proofs in linear logic. *J. Logic Comput.* 2(3) (1992), 297–347.
[2] Federico Aschieri, Agata Ciabattoni, and Francesco A. Genco. 2017. Gödel logic: from natural deduction to parallel computation. In *LICS 2017*. IEEE Computer Society.
[3] Arnon Avron. 1987. A constructive analysis of RM. *J. Symbolic Logic* 52, 4 (1987), 939–951.
[4] Arnon Avron. 1991. Hypersequents, logical consequence and intermediate logics for concurrency. *Ann. Math. Artif. Intell.* 4, 3 (1991), 225–248.
[5] Arnon Avron. 1996. The method of hypersequents in the proof theory of propositional non-classical logics. In *Logic: from foundations to applications (Staffordshire, 1993)*. Oxford Univ. Press, New York, 1–32.
[6] Matthias Baaz, Agata Ciabattoni, and Christian Fermüller. 2000. A natural deduction system for intuitionistic fuzzy logic. In *Lectures on Soft Computing and Fuzzy Logic*. 1–18.
[7] Arnold Beckmann and Norbert Preining. 2015. Hyper natural deduction. In *LICS 2015*. IEEE Computer Society, 547–558.
[8] Agata Ciabattoni, Nikolaos Galatos, and Kazushige Terui. 2008. From axioms to analytic rules in nonclassical logics. In *LICS 2008*. IEEE Computer Society, 229–240.
[9] Agata Ciabattoni and Francesco A. Genco. 2016. Embedding formalisms: hypersequents and two-level systems of rules. In *Advances in Modal Logic*. Vol. 11. College Publications, 197–216.
[10] Vincent Danos and Jean-Louis Krivine. 2000. Disjunctive tautologies as synchronisation schemes. In *CSL 2000*, Vol. 1862. 292–301.
[11] Melvin Fitting. 2012. Prefixed tableaus and nested sequents. *Ann. Pure Appl. Logic* 163, 3 (2012), 291–313.
[12] Gerhard Gentzen. 1935. Untersuchungen über das logische Schließen. I. *Mathematische Zeitschrift* 39, 1 (1935), 176–210.
[13] Rajeev Goré and Revantha Ramanayake. 2012. Labelled tree sequents, tree hypersequents and nested (deep) sequents. In *Advances in Modal Logic*. College Publications, 279–299.

[14] William A. Howard. 1980. The formulae-as-types notion of construction. In *To H. B. Curry: Essays on Combinatory Logic, Lambda Calculus, and Formalism*. Academic Press, 479–491.

[15] Ori Lahav. 2013. From frame properties to hypersequent rules in modal logics. In *LICS 2013*. 408–417.

[16] Björn Lellmann. 2016. Hypersequent rules with restricted contexts for propositional modal logics. *Theoret. Comput. Sci.* (2016).

[17] Edgar G.K. López-Escobar. 1982. Implicational logics in natural deduction systems. *J. Symbolic Logic* 47 (1982), 184–186. Issue 1.

[18] Sara Negri. 2005. Proof analysis in modal logic. *J. Philos. Logic* 34 (2005), 507–544.

[19] Sara Negri. 2014. Proof analysis beyond geometric theories: from rule systems to systems of rules. *J. Logic Comput.* (2014).

[20] Francesca Poggiolesi. 2010. Display calculi and other modal calculi: a comparison. *Synthese* 173, 3 (2010), 259–279.

[21] Francesca Poggiolesi. 2010. *Gentzen calculi for modal propositional logic*. Springer.

[22] Dag Prawitz. 1971. Ideas and results in proof theory. In *Proceedings of the Second Scandinavian Logic Symposium*.

[23] Revantha Ramanayake. 2015. Embedding the hypersequent calculus in the display calculus. *J. Logic Comput.* 3 (2015), 921–942. Issue 25.

[24] Revantha Ramanayake. 2016. Inducing syntactic cut-elimination for indexed nested sequents. In *Proceedings of IJCAR*. LNCS, 416–432.

[25] Peter Schroeder-Heister. 2014. The calculus of higher-level rules, propositional quantification, and the foundational approach to proof-theoretic harmony. *Studia Logica* 102, 6 (2014), 1185–1216.

[26] Heinrich Wansing. 1998. Translation of hypersequents into display sequents. *Log. J. IGPL* 6, 5 (1998), 719–733.