Graphical Proof Theory I: Sequent Systems on Undirected Graphs

Matteo Acclavio

Abstract

In this paper we explore the design of sequent calculi operating on graphs. For this purpose, we introduce a set of logical connectives extending the well-known correspondence between classical propositional formulas and cographs, and we define sequent systems operating on formulas over these connectives.

We prove, using an analyticity argument based on cut-elimination, that our systems provide conservative extensions of multiplicative linear logic (without and with mix) and classical propositional logics. We conclude by showing that one of our systems captures graph isomorphism as logical equivalence, and that this system is also sound and complete for the graphical logic GS*.*

Contents

1 Introduction

In theoretical computer science, *formulas* play a crucial role in describing complex abstract objects. At the syntactical level, the formulas of a logic describe complex structures by means of unary and binary operators, usually thought of as *connectives* and *modalities* respectively. On the other hand, graph-based syntaxes are often favored in formal representation, as they provide an intuitive and canonical description of properties, relations and systems. By means of example, consider the two graphs below:

$$a \leftarrow b \rightarrow c \leftarrow d \qquad \text{or} \qquad a — b — c — d$$

It follows from results in [62, 21] that describing any of the above graphs by means of formulas only employing binary connectives would require repeating at least one vertex. As a consequence, formulas describing complex graphs are usually long and convoluted, and a specific *encodings* are needed to standardize such formulas.

Since graphs are ubiquitous in theoretical computer science and its applications, a natural question to ask is whether it is possible to define formalisms having graphs, instead of formulas, as first-class terms of the syntax. Such a paradigm shift would allow to design efficient automated tools free from the bureaucracy introduced to handle the encoding required to represent graphs. At the same time, a graphical syntax would provide a useful tool for investigations such as the ones in [63] or [27, 25], where the authors restrain their framework to sequential-parallel orders, as these can be represented by means of formulas with at most binary connectives.

Two recent lines of works have generalized proof theoretical methodologies to graphs, extending the correspondence between classical propositional formulas and cographs. In these works, systems operating on graphs are defined via local and context-free rewriting rules, similarly as what done in *deep inference* systems [33, 34, 9]. The first line of research, carried out by Calk, Das, Rice and Waring in various works [18, 17, 64, 23, 24], explores the use of maximal stable sets/cliques-preserving homomorphisms to define notions of entailment[1], and study the resulting proof theory. Here, the choice of the using of a deep inference formalism is natural, since the rules of the calculus are local rewritings. The second line of research, investigated by the author, Horne, Mauw and Straßburger in several contributions [4, 5, 2], studies the (sub-)structural proof theory of arbitrary graphs, with an approach inspired by linear logic [29] and deep inference [33]. The main goal of this line of research, partially achieved with the system $\mathsf{GV}^{\mathsf{sl}}$ operating on mixed graphs [2], is to obtain a generalization of the completeness result of the logic BV with respect to pomset inclusion. The logic BV contains a non-commutative binary connective $\triangleleft$ allowing to represent series-parallel partial order multisets as formulas in the syntax (as in Retoré's Pomset logic [57]), and to capture order inclusion as logical implication. However, as shown in [60], no cut-free sequent system for BV can exists – therefore neither for Pomset logic, which strictly contains it [54, 53]. For this reason the aforementioned line of work focused on deep inference systems, and the question about the existence of a cut-free sequent calculus for GS (the restriction of $\mathsf{GV}^{\mathsf{sl}}$ on undirected graphs originally defined in [4]) was left open.

1.1 Main contributions

In this paper we focus on the definition of sequent calculi for *graphical logics*, and we positively answer the above question by providing, among other results, a cut-free sound and complete sequent calculus for GS. By using standard techniques in sequent calculus, we thus obtain a proof of analiticity for this logic which is simpler and more concise with respect to the one in [5].

To achieve these results, we introduce *graphical connectives*, which are operators that can be naturally interpreted as graphs. We then define the sequent calculi MGL, MGL° and GLK, containing rules to handle these connectives. After showing that cut-elimination holds for these systems, we prove that MGL, MGL° and GLK define conservative extensions of *multiplicative linear logic*, *multiplicative linear logic with mix* and *classical propositional logic* respectively. We then prove that formulas interpreted as the same graph are logically equivalent, thus justifying the fact that we consider these systems as operating on graphs rather than formulas. We conclude by showing that MGL° is sound and complete with respect to the logic GS, thus providing a simple sequent calculus for the logic.

1.2 Outline of the paper

In Section 2 we recall definitions and results in graph theory and the notion of modular decomposition. In Section 3 we use these notions to extend the correspondence between classical propositional formulas and

[1] A similar approach was proposed in [56] for studying pomsets.

cographs to any graph. We define linear sequent calculi and we prove their properties. In Section 4 we show that one of these calculi is sound and complete with respect to the set of non-empty graphs provable in the deep inference system GS studied in [4, 5]. In Section 5 we define a proof system which is a conservative extension of classical logic. To conclude, we summarize in Section 6 some of the possible the research directions opened by this work.

2 From Formulas To Graphs

In this section we recall standard results from the literature on graphs such as *modular decomposition* and *cographs*. We then introduce the notion of *graphical connectives* allowing us to extend the correspondence between cographs and classical propositional formulas to general graphs.

2.1 Graphs and Modular Decomposition

In this work are interested in using graphs to represent patterns of interactions by means of the binary relations (edges) between their components (vertices). For this reason we recall the definition of *labeled graph* (the mathematical structure we use to encode these patterns) together with the definition of *isomorphism* (the standard notion of identity on labeled graphs) and the rougher notion of *similarity* (equivalence up-to labels over vertices).

Definition 1. A ***$\mathcal{L}$-labeled graph*** (or simply ***graph***) $G = \langle V_G, \ell_G, \overset{G}{\frown} \rangle$ is given by a finite set of ***vertices*** V_G, a partial ***labeling function*** $\ell_G \colon V_G \to \mathcal{L}$ associating a label $\ell(v)$ from a given set of labels $\mathcal{L}$ to each vertex $v \in V_G$ (we may represent ℓ_G as a set of equations of the form $\ell(v) = \ell_v$ and denote by $\varnothing$ the empty function), and a non-reflexive symmetric edge relation $\overset{G}{\frown} \subset V_G \times V_G$ whose elements, called ***edges***, may be denoted vw instead of (v, w). The ***empty*** graph $\langle \varnothing, \varnothing, \varnothing \rangle$ is denoted $\varnothing$.

A ***similarity*** between two graphs G and G' is a bijection $f \colon V_G \to V_{G'}$ such that $x \overset{G}{\frown} y$ iff $f(x) \overset{G'}{\frown} f(y)$ for any $x, y \in V_G$. An ***isomorphism*** is a similarity f such that $\ell(v) = \ell(f(v))$ for any $x, y \in V_G$. Two graphs G and G' are ***similar*** (denoted $G \sim G'$) if there is an similarity between G and G'. A ***symmetry*** is a similarity of a graph with itself. They are ***isomorphic*** (denoted $G = G'$) if there is a isomorphism between G and G'. From now on, we consider two isomorphic graphs to be ***the same*** graph.

Two vertices v and w in G are ***connected*** if there is a sequence $v = u_0, \ldots, u_n = w$ of vertices in G (called ***path***) such that $u_{i-1} \overset{G}{\frown} u_i$ for all $i \in \{1, \ldots, n\}$. A ***connected component*** of G is a maximal set of connected vertices in G. A graph G is a ***clique*** (resp. a ***stable set***) iff $\overset{G}{\not\frown} = \varnothing$ (resp. $\overset{G}{\frown} = \varnothing$).

Notation 2. When drawing a graph or an unlabeled graph we draw v—w whenever $v \frown w$, we draw no edge at all whenever $v \not\frown w$. We may represent a vertex of a graph by using its label instead of its name. For example, the single-vertex graph $G = \langle \{v\}, \ell_G, \varnothing \rangle$ may be represented either by a the vertex name v or by the vertex label $\ell(v)$ (or $\bullet$ if $\ell(v)$ is not defined). Note that, since we are considering isomorphic graphs to be the same, as soon as there is no ambiguity due to vertices represented by the same symbol, we can assume that the representation of a graph to provide us one of the possible triple (set of vertices, label function, and set of edges) defining it.

Example 3. Consider the following graphs:

$$
\begin{aligned}
F &= \langle \{u_1, u_2, u_3, u_4\}, \{\ell(u_1) = a, \ell(u_2) = b, \ell(u_3) = c, \ell(u_4) = d\}, \{u_1u_2, u_2u_3, u_3u_4\} \rangle \\
G &= \langle \{v_1, v_2, v_3, v_4\}, \{\ell(v_1) = b, \ell(v_2) = a, \ell(v_3) = c, \ell(v_4) = d\}, \{v_1v_2, v_1v_3, v_3v_4\} \rangle \\
H &= \langle \{w_1, w_2, w_3, w_4\}, \{\ell(w_1) = a, \ell(w_2) = b, \ell(w_3) = c, \ell(w_4) = d\}, \{w_1w_2, w_1w_3, w_3w_4\} \rangle
\end{aligned}
$$

They are all symmetric, that is $F \sim G \sim H$, but $F = G \neq H$ as can easily be verified using their representations:

$$F = a\text{—}b\text{—}c\text{—}d = G \qquad \text{and} \qquad H = b\text{—}a\text{—}c\text{—}d$$

Observation. The problem of graph isomorphism is a standard **NP**-problem (to be more precise, its complexity is quasi-polynomial [12]). That is, verify that a given bijection between the sets of vertices of two graphs is an isomorphism can be checked in polynomial time, while there is no known polynomial time algorithm to find such an isomorphism. For this reason, whenever we say that two graphs are the same, either we assume they share the same set of vertices, therefore implicitly assuming the isomorphism f to be defined by the identity function over the set of vertices, or we assume an isomorphism to be given. This allows us to verify whether two graphs are the same in polynomial time.

In order to use proof theoretical methodologies on graphs, we need a suitable notion of subgraphs to be used in the same way sub-formulas are used in proof systems, that is, to state properties of the calculus or to define the behavior of rules. For this purpose, we use for a notion of *module* to identify subgraph allowing us to decompose a graph using abstract syntax trees similar to the ones underlying formulas [28, 42, 37, 46, 49, 26]. A module is a subset of vertices of a graph having the same edge-relation with any vertex outside the subset. This definition generalizes the interaction we usually be observed in formulas, where, in the formula tree, any literal in a subformula has the same relation (the one given by the least common ancestor) with a given literal not occurring in the subformula itself.

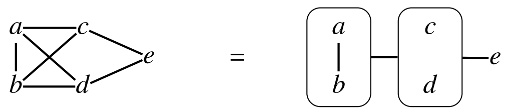
Definition 4. Let $G = \langle V_G, \ell_G, E_G \rangle$ be a graph and $W \subseteq V_G$. The ***graph induced*** by W is the graph $G|_W := \left\langle W, \ell_G|_W, \stackrel{G}{\frown} \cap (W \times W) \right\rangle$ where $\ell_G|_W(v) := \ell_G(v)$ for all $v \in W$.

A ***module*** of a graph G is a subset M of V_G such that $x \frown z$ iff $y \frown z$ for any $x, y \in M$, $z \in V_G \setminus M$. A module M is ***trivial*** if $M = \emptyset$, $M = V_G$, or $M = \{x\}$ for some $x \in V_G$. From now on, we identify a module M of a graph G with the induced subgraph $G|_M$.

Remark 5. A connected component of a graph G is a module of G.

Using modules we can optimize the way we represent graphs reducing the number of edges drawn without losing information, relying on the fact that all vertices of a module has the same edge-relation with any vertex outside the module.

Notation 6. In representing graphs we may border vertices of a same module by a closed line. An edges connected to such a closed line denotes the existence of an edge to each vertex inside it. By means of example, consider the following graph and its more compact modular representation.



(1)

The notion of module is related to a notion of context, which can be intuitively formulated as a graph with a special vertex playing the role of a hole in which we can plug in a module.

Definition 7. A ***context*** $C[\square]$ is a (non-empty) graph containing a single occurrence of a special vertex $\square$ (such that $\ell(\square)$ is undefined). It is ***trivial*** if $C[\square] = \square$. If $C[\square]$ is a context and G a graph, we define $C[G]$ as the graph obtained by replacing $\square$ by G. Formally,

$$C[G] := \left\langle (V_{C[\square]} \setminus \{\square\}) \uplus V_G \ ,\ \ell_C \cup \ell_G \ ,\ \left\{ vw \mid v, w \in V_{C[\square]} \setminus \{\square\}, v \stackrel{C[\square]}{\frown} w \right\} \cup \left\{ vw \mid v \in V_{C[\square]} \setminus \{\square\}, w \in V_G, v \stackrel{C[\square]}{\frown} \square \right\} \right\rangle$$

Remark 8. A set of vertices M is a module of a graph G iff there is a context $C[\square]$ such that $G = C[M]$.

We generalize this idea of replacing a vertex of a graph with a module by defining the operations of *composition-via* a graph, where all vertices of a graph are replaced in a "modular way" by modules.

Definition 9. Let G be a graph with $V_G = \{v_1, \ldots, v_n\}$ and let $H_1, \ldots, H_n$ be graphs. We define the ***composition of $H_1, \ldots, H_n$ via G*** as the graph $G(\!|H_1, \ldots, H_n|\!)$ obtained by replacing each vertex v_i of G with a module H_i for all $i \in \{1, \ldots, n\}$. Formally,

$$G(\!|H_1, \ldots, H_n|\!) = \left\langle \biguplus_{i=1}^{n} V_{H_i} \ ,\ \bigcup_{i=1}^{n} \ell_{H_i} \ ,\ \left(\bigcup_{i=1}^{n} \stackrel{H_i}{\frown} \right) \cup \left\{ (x, y) \;\middle|\; x \in V_{H_i}, y \in V_{H_j}, v_i \stackrel{G}{\frown} v_j \right\} \right\rangle \tag{2}$$

The subgraphs $H_1, \ldots, H_n$ are called ***factors*** of $G(\!|H_1, \ldots, H_n|\!)$ and, by definition, are (possibly not maximal) modules of $G(\!|H_1, \ldots, H_n|\!)$.

Remark 10. The information about the labels of the graph G used to define the composition-via operation is lost. Moreover, if G is a graph with $V_G = \{v_1, \ldots, v_n\}$ and σ a permutation over the set $\{1, \ldots, n\}$ such that the map $f_\sigma : V_G \to V_G$ mapping v_i in $f_\sigma(v_i) = v_{\sigma(i)}$ for all $i \in \{1, \ldots, n\}$ is an similarity between G and G, then $G(\!|H_1, \ldots, H_n|\!) = G'(\!|H_1, \ldots, H_n|\!)$.

In order to establish a connection between graphs and formulas, from now on we only consider graphs whose set of labels belong to the set $\mathcal{L} = \{a, a^\perp \mid a \in \mathcal{A}\}$ where $\mathcal{A}$ is a fixed set of propositional variables. We then define the *dual* of a graphs.

Definition 11. Let $G = \langle V_G, \ell_G, E_G \rangle$ be a graph. We define the edge relation $\overset{G}{\not\frown} := \left\{(v, w) \mid v \neq w \text{ and } vw \notin \overset{G}{\frown}\right\}$ and we define the ***dual*** graph of G as the graph $G^\perp := \left\langle V_G, \overset{G}{\not\frown}, \ell_{G^\perp} \right\rangle$ with $\ell_{G^\perp}(v) = (\ell_G(v))^\perp$ (assuming $a^{\perp\perp} = a$ for all $a \in \mathcal{A}$).

Remark 12. By definition, each module of a graph corresponds to a module of its dual graph. It follows that a connected component of $G^\perp$ is a module of G.

Notation 13. If $\mathcal{G}$ is the representation of a graph G, then we may represent the graph $G^\perp$ by bordering the representation of G with a closed line with the negation symbol on the upper-right corner, that is, $\boxed{\mathcal{G}}^{\perp}$.

2.2 Classical Propositional Formulas and Cographs

The set of ***classical (propositional) formulas*** is generated from a set of propositional variable $\mathcal{A}$ using the ***negation*** $(\cdot)^\perp$, the ***disjunction*** $\vee$ and the ***conjunction*** $\wedge$ using the following grammar:

$$\phi, \psi := a \mid \phi \vee \psi \mid \phi \wedge \psi \mid \phi^\perp \qquad \text{with } a \in \mathcal{A}. \tag{3}$$

We denote by $\equiv$ the equivalence relation over formulas generated by the following laws:

$$\begin{array}{rl} \textbf{\textit{Equivalence laws}} & \left\{ \begin{array}{ll} \phi \vee \psi \equiv \psi \vee \phi & \qquad \phi \vee (\psi \vee \chi) \equiv (\phi \vee \psi) \vee \chi \\ \phi \wedge \psi \equiv \psi \wedge \phi & \qquad \phi \wedge (\psi \wedge \chi) \equiv (\phi \wedge \psi) \wedge \chi \end{array} \right. \\ \textbf{\textit{De-Morgan laws}} & \left\{ \begin{array}{ll} (\phi^\perp)^\perp \equiv \phi & \qquad (\phi \wedge \psi)^\perp \equiv \phi^\perp \vee \psi^\perp \end{array} \right. \end{array} \tag{4}$$

We define a map from literals to single-vertex graphs, which extends to formulas via the composition-via the unlabeled two-vertices stable set S_2 and two-vertices clique K_2.

Definition 14. Let ϕ be a classical formula, then $[\![\phi]\!]$ is the graph inductively defined as follows:

$$[\![a]\!] = a \qquad \left[\!\!\left[\phi^\perp\right]\!\!\right] = [\![\phi]\!]^\perp \qquad [\![\phi \vee \psi]\!] = \mathsf{S}_2 \left(\!\left([\![\phi]\!], [\![\psi]\!]\right)\!\right) \qquad [\![\phi \wedge \psi]\!] = \mathsf{K}_2 \left(\!\left([\![\phi]\!], [\![\psi]\!]\right)\!\right)$$

where S_2 and K_2 are respectively a stable set and a clique with 2 vertices, and where we denote by a the single-vertex graph, whose vertex is labeled by a.

We can easily observe that the map $[\![\cdot]\!]$ well-behaves with respect to the equivalence over formulas $\equiv$, that is, equivalent formulas are mapped to the symmetric graphs.

Proposition 15. *Let ϕ and ψ be classical formulas. Then $\phi \equiv \psi$ iff $[\![\phi]\!] = [\![\psi]\!]$.*

We finally recall the definition of *cographs*, and the theorem establishing the relation between cographs and classical formulas, i.e., providing an alternative definition of cographs as graphs generated by single-vertex graphs using the composition-via a two-vertices no-edge graph and a two-vertices one-edge graph.

Definition 16. A ***cograph*** is a graph G such that there are no four vertices v_1, v_2, v_3, v_4 in G such that the induced subgraph $G|_{\{v_1,v_2,v_3,v_4\}}$ is similar to the graph $\langle \{a, b, c, d\}, \emptyset, \{ab, bc, cd\} \rangle = a — b — c — d$.

Theorem 17 ([28])**.** *A graph G is a cograph iff there is a formula ϕ such that $G \sim [\![\phi]\!]$.*

2.3 Modular Decomposition of Graphs

We recall the notion of *prime graph*, allowing us to provide canonical representatives of graphs via modular decomposition. (see e.g., [28, 42, 37, 46, 49, 26]).

Definition 18. A graph G is ***prime*** if $|V_G| > 1$ and all its modules are trivial.

We recall the following standard result from the literature.

Theorem 19 ([42])**.** *Let G be a graph with at least two vertices. Then there are non-empty modules $M_1, \ldots, M_n$ of G and a prime graph P such that $G = P(\!|M_1, \ldots, M_n|\!)$.*

This result implies the possibility of describing graphs using single-vertex graphs and the operation of composition-via prime graphs. More precisely, we can define the notion of *modular decomposition* of a graph composition-via prime graphs to provide a more canonical representation.

Definition 20. Let G be a non-empty graph. A ***modular decomposition*** of G is a way to write G using single-vertex graphs and the operation of composition-via prime graphs:

- if G is a graph with a single vertex x labeled by a, then $G = a$ (i.e., $G = \langle \{x\}, \ell(x) = a, \emptyset \rangle$);
- if $H_1, \ldots, H_n$ are maximal modules of G such that $V_G = \biguplus_{i=1}^{n} V_{H_i}$, then there is a unique prime graph P such that $G = P(\!|H_1, \ldots, H_n|\!)$.

Remark 21. There are various reasons why modular decomposition is not unique.

The first is due to the possible presence of cliques and stable sets. By means of example, consider a clique with three vertices u, v and w can be represented as $(u \otimes v) \otimes w$ or $u \otimes (v \otimes w)$.

We already observed the second reason in Remark 10, since graph symmetries allow us to represent the same graph by different decompositions, as shown in top-most modular decomposition below on the left.

$$\begin{array}{l} P(\!|u,v,w,t|\!) = u\text{—}v\text{—}w\text{—}t = P(\!|t,w,v,u|\!) \\ P(\!|u,v,w,t|\!) = u\text{—}v\text{—}w\text{—}t = P'(\!|u,w,v,t|\!) \end{array} \quad \text{where} \quad P = a\text{—}b\text{—}c\text{—}d \quad \text{and} \quad P' = a\text{—}c\text{—}b\text{—}d\ .$$

Finally, two symmetric prime graphs could provide distinct modular decompositions of the same graph, as shown above with symmetric prime graphs P and P'.

The first problem could be addressed by considering in the modular decomposition not only prime graphs, but also cliques and stable sets, that is, including n-ary versions of the operations $⅋$ and $\otimes$. We show later in this paper that this problem is irrelevant due to the associativity of $⅋$ and $\otimes$. The second problem cannot be addressed without enforcing a cumbersome order over graphs taking into account vertex labels and factor positions. However, we can address the latter source of ambiguity by introducing the notion of *base* of *graphical connectives*, allowing us to provide a single canonical prime graph for each class of symmetric prime graphs.

Definition 22. A ***graphical connective*** $C = \langle V_C, \overset{C}{\frown} \rangle$ (with ***arity*** $n = |V_C|$) is given by a finite list of vertices $V_C = \langle v_1, \ldots, v_n \rangle$ and a non-reflexive symmetric edge relation $\overset{C}{\frown}$ over the set of vertices occurring in V_C. We denote by G_C the graph corresponding to C, that is, the graph $G_C = \langle \{v \mid v \text{ in } V_C\}, \emptyset, \overset{C}{\frown} \rangle$. The ***composition-via*** a graphical connective is defined as the composition-via the graph G_C.

A graphical connective is ***prime*** if G_C is a prime graph. A set $\mathcal{P}$ of prime graphical connectives is a ***base*** if for each prime graph P there is a unique connective $C \in \mathcal{P}$ such that $P \sim G_C$.

Given an n-ary connective C, we define the following sets of permutations over the set $\{1, \ldots, n\}$:

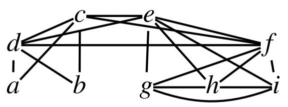
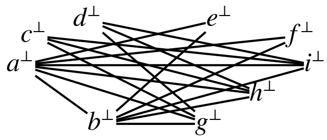
$$\begin{array}{llcl} \text{the } \textbf{\textit{group}}^2 \textbf{\textit{ of symmetries}} \text{ of } C & : \mathfrak{S}(C) & := & \left\{ \sigma \mid C(\!|a_1, \ldots, a_n|\!) = C(\!|a_{\sigma(1)}, \ldots, a_{\sigma(n)}|\!) \right\} \\ \text{the } \textbf{\textit{set of dualizing symmetries}} \text{ of } C & : \mathfrak{S}^\perp(C) & := & \left\{ \sigma \mid (C(\!|a_1, \ldots, a_n|\!))^\perp = C(\!|a^\perp_{\sigma(1)}, \ldots, a^\perp_{\sigma(n)}|\!) \right\} \end{array} \tag{5}$$

for any $a_1, \ldots, a_n$ single-vertex graphs.

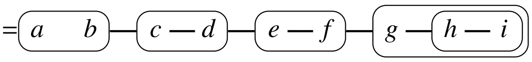
Notation 23. We define the following graphical connectives (with $n > 1$):

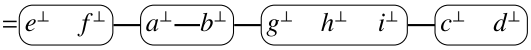
$$\begin{array}{rcl} ⅋(\!|v_1, v_2|\!) := \langle \langle v_1, v_2 \rangle, \emptyset \rangle & = & \boxed{v_1 \quad v_2} \\ \otimes(\!|v_1, v_2|\!) := \langle \langle v_1, v_2 \rangle, \{v_1 v_2\} \rangle & = & \boxed{v_1 \text{—} v_2} \\ \mathsf{P_n}(\!|v_1, \ldots, v_n|\!) := \langle \langle v_1, \ldots, v_n \rangle, \{v_i v_{i+1} \mid i \in \{1, \ldots, n-1\}\} \rangle & = & \boxed{v_1 \text{—} v_2 \text{—} \cdots \text{—} v_{n-1} \text{—} v_n} \\ \mathsf{Bull}(\!|v_1, \ldots, v_5|\!) := \langle \langle v_1, \ldots, v_5 \rangle, \{(v_1 v_2, v_2 v_3, v_3 v_4, v_5 v_2, v_5 v_3)\} \rangle & = & \boxed{v_1 \text{—} v_2 \text{—} v_3 \text{—} v_4 \quad (v_2 \text{—} v_5 \text{—} v_3)} \end{array} \tag{6}$$

Example 24. Consider the following graph G and its dual $G^\perp$:

$G =$  and $G^\perp =$ 

We can write them as

$$G \; = \; \mathsf{P}_4 \left(\!\left| a \parr b, c \otimes d, e \otimes f, g \otimes (h \otimes i) \right|\!\right) =$$ 

$$G^\perp = \mathsf{P}_4^\perp \left(\!\left| a^\perp \otimes b^\perp, c^\perp \parr d^\perp, e^\perp \parr f^\perp, g^\perp \parr (h^\perp \parr i^\perp) \right|\!\right) =$$
$$= \mathsf{P}_4 \left(\!\left| c^\perp \parr d^\perp, a^\perp \otimes b^\perp, g^\perp \parr (h^\perp \parr i^\perp), e^\perp \parr f^\perp \right|\!\right) =$$ 

We can reformulate the standard result on modular decomposition as follows.

Theorem 25. *Let G be a non-empty graph and $\mathcal{P}$ a base. Then then there is a unique way (up to symmetries of graphical connectives and associativity of $\parr$ and $\otimes$) to write G using single-vertex graphs and the graphical connectives in $\mathcal{P}$.*

Corollary 26. *Two graphs are isomorphic iff they admit a same modular decomposition.*

2.4 Graphs as Formulas

In order to represent graphs as formulas, we define new connectives beyond conjunction and disjunction to represent graphical connectives in a base $\mathcal{P}$. From now on, we assume bases $\mathcal{P}$ containing the graphical connectives in Equation (6) to be fixed.

Definition 27. The set of ***formulas*** is generated by the set of propositional atoms $\mathcal{A}$, a ***unit*** $\circ$, using the following syntax:

$$\phi_1, \dots, \phi_n \coloneqq \circ \mid a \mid a^\perp \mid \kappa_P(\!|\phi_1, \dots, \phi_{n_P}|\!) \qquad \text{with } a \in \mathcal{A} \text{ and } P \in \mathcal{P} \tag{7}$$

We simply denote $\parr$ (resp. $\otimes$) the binary connective $\kappa_{\parr}$ (resp. $\kappa_\otimes$) and we write $\phi \parr \psi$ instead of $\kappa_{\parr}(\!|\phi, \psi|\!)$ (resp. $\phi \otimes \psi$ instead of $\kappa_\otimes(\!|\phi, \psi|\!)$). The ***arity*** of the connective κ_P is the arity n_P of P.

A ***literal*** is a formula of the form a or $a^\perp$ for an atom $a \in \mathcal{A}$. The set of literals is denoted $\mathcal{L}$. A ***κ-formula*** is a formula with ***main connective*** κ, that is, a formula of the form $\kappa(\!|\phi_1, \dots, \phi_n|\!)$. A formula is ***unit-free*** if it contains no occurrences of $\circ$ and ***vacuous*** if it contains no atoms. A formula is ***pure*** if non-vacuous and such that its vacuous subformulas are $\circ$. A **MLL-*formula*** is a formula containing only occurrences of $\parr$ and $\otimes$ connectives.

A ***context formula*** (or simply ***context***) $\zeta[\square]$ is a formula containing an ***hole*** $\square$ taking the place of an atom. Given a context $\zeta[\square]$, the formula $\zeta[\phi]$ is defined by simply replacing the atom $\square$ with the formula ϕ. For example, if $\zeta[\square] = \psi \parr (\square \otimes \chi)$, then $\zeta[\phi] = \psi \parr (\phi \otimes \chi)$.

For each ϕ formula (or context), the graph $[\![\phi]\!]$ is defined as follows:

$$[\![\square]\!] = \square \qquad [\![\circ]\!] = \varnothing \qquad [\![a]\!] = a \qquad \left[\!\left[\phi^\perp\right]\!\right] = [\![\phi]\!]^\perp \qquad [\![\kappa_P(\!|\phi_1, \dots, \phi_n|\!)]\!] = P\left(\!\left|[\![\phi_1]\!], \dots, [\![\phi_n]\!]\right|\!\right) \tag{8}$$

Notation 28. We could consider a formula ϕ over the set of occurrences of literals $\{x_1, \dots, x_n\}$ as a ***synthetic connective***. That is, we may denote by $\phi(\!|\psi_1, \dots, \psi_n|\!)$ the formula obtained by replacing each literal x_i with a corresponding ψ_i for all $i \in \{1, \dots, n\}$. The set of ***symmetries*** of ϕ (denoted $\mathfrak{S}(\phi)$) is the set of permutations σ over $\{1, \dots, n\}$ such that $[\![\phi(\!|x_1, \dots, x_n|\!)]\!] = [\![\phi(\!|x_{\sigma(1)}, \dots, x_{\sigma(n)}|\!)]\!]$.

Definition 29. The equivalence relation $\equiv$ over formulas is generated by the following equations:

$$\textbf{\textit{Equivalence laws}} \begin{cases} \kappa_P(\!|\phi_1, \dots, \phi_{|P|}|\!) \equiv \kappa_P(\!|\phi_{\sigma(1)}, \dots, \phi_{\sigma(|V_P|)}|\!) & \text{for each } \sigma \in \mathfrak{S}(P) \\ \phi \otimes (\psi \otimes \chi) \equiv (\phi \otimes \psi) \otimes \chi & \\ \phi \parr (\psi \parr \chi) \equiv (\phi \parr \psi) \parr \chi & \end{cases}$$

$$\textbf{\textit{De-Morgan laws}} \begin{cases} & \circ^\perp \equiv \circ \qquad \phi^{\perp\perp} \equiv \phi \\ \text{only if } \mathfrak{S}^\perp(P) = \varnothing : & (\kappa_P(\!|\phi_1, \dots \phi_{n_P}|\!))^\perp \equiv \kappa_{P^\perp}(\!|\phi^\perp_{\sigma(1)}, \dots, \phi^\perp_{\sigma(n_P)}|\!) \\ \text{only if } \mathfrak{S}^\perp(P) \neq \varnothing : & (\kappa_P(\!|\phi_1, \dots \phi_{n_P}|\!))^\perp \equiv \kappa_P(\!|\phi^\perp_{\rho(1)}, \dots, \phi^\perp_{\rho(n_P)}|\!) \quad \text{for each } \rho \in \mathfrak{S}^\perp(P) \end{cases}$$

for each $P \in \mathcal{P}$ (with arity n_P).

The ***(linear) negation*** over formulas is defined by letting

$$\circ^{\perp} = \circ \qquad \phi^{\perp\perp} = \phi \qquad (\kappa_P(\!|\phi_1, \ldots, \phi_n|\!))^{\perp} = \kappa_Q(\!|\phi^{\perp}_{\sigma_P(1)}, \ldots, \phi^{\perp}_{\sigma_P(n)}|\!)$$

where Q is the unique graphical connective in $\mathcal{P}$ such that $\left[\!\left[\kappa_P\left(\!\left|a_1, \ldots, a_n\right|\!\right)\right]\!\right] = Q\left(\!\left|a^{\perp}_{\sigma(1)}, \ldots, a^{\perp}_{\sigma_n}\right|\!\right)$ for any single-vertex graphs $a_1^{\perp}, \ldots, a_n^{\perp}$ (with vertex labeled by $a_1^{\perp}, \ldots, a_n^{\perp}$ respectively) and a permutation σ_P over the set $\{1, \ldots, n\}$. [3]

The ***linear implication*** $\phi \multimap \psi$ is defined as $\phi^{\perp} \parr \psi$, while the ***logical equivalence*** $\phi \multimapboth \psi$ is defined as $(\phi \multimap \psi) \otimes (\psi \multimap \phi)$.

Remark 30. As explained in [5] (Section 9), the graphical connectives we discuss in this paper are *multiplicative connectives* (in the sense of [22, 32, 48, 6]) but they are not the same as the *connectives-as-partitions* discussed in these works. In fact, there is a unique 4-ary graphical connective P_4 has symmetry group $\{\mathsf{id}, (1,4)(2,3)\}$, while, as shown in [48, 6], there is a unique pair of dual "primitive" 4-ary multiplicative connectives-as-partitions G_4 and $\mathsf{G}_4^{\perp}$, and $\mathfrak{S}(\mathsf{P}_4) \subsetneq \mathfrak{S}(\mathsf{G}_4) = \mathfrak{S}(\mathsf{G}_4^{\perp})$.

The following result is consequence of Theorem 19.

Proposition 31. *Let ϕ and ψ be formulas. If $\phi \equiv \psi$, then $[\![\phi]\!] = [\![\psi]\!]$. Moreover, if ϕ and ψ are unit-free, then $\phi \equiv \psi$ iff $[\![\phi]\!] = [\![\psi]\!]$.*

Note that the the stronger result does not hold in presence of units. For an example consider the formulas $\circ \otimes \circ$ and $\circ \parr \circ$.

3 Sequent calculi over operating on graphs-as-formulas

We assume the reader to be familiar with the definition of sequent calculus derivations as trees of sequents (see, e.g., [61]) but we recall here some definitions.

Definition 32. We define a ***sequent*** is a set of occurrences of formulas. A ***sequent system*** S is a set of ***sequent rules*** as the ones in Figure 1. In a sequent rule ρ, we say that a formula is ***active*** if it occurs in one of its premises but not in its conclusion, and ***principal*** if it occurs in its conclusion but in none of its premises.

A ***proof*** of a sequent Γ is a derivation with no open premises, denoted $\overset{\pi \|\mathsf{S}}{\Gamma}$. We denote by $\begin{array}{c}\Gamma' \\ \pi' \|\mathsf{S} \\ \Gamma\end{array}$ an ***(open) derivation*** of Γ from Γ', that is, is a proof tree having exactly one open premise Γ'.

A rule is ***admissible*** in S if there is a derivation of the conclusion of the rule whenever all premises of the rule are derivable. A rule is ***derivable*** in S, if there is a derivation in S from the premises to the conclusion of the rule.

Notation 33. In this paper we use the same notation to denote a sequent system S and the set of formulas admitting a proof in S.

Definition 34. We define the following sequent systems using the rules in Figure 1.

$$\begin{array}{ll} \textit{Multiplicative Graphical Logic}: & \mathsf{MGL} = \{\mathsf{ax}, \parr, \otimes, \mathsf{d}\text{-}P \mid P \in \mathfrak{P}\} \\ \textit{Multiplicative Graphical Logic with mix}: & \mathsf{MGL}^{\circ} = \mathsf{MGL} \cup \{\mathsf{mix}, \mathsf{wd}_{\otimes}, \mathsf{unitor}_{\kappa}\} \end{array} \tag{9}$$

Observation (Rules Exegesis)**.** The rules ***axiom*** (ax), ***par*** ($\parr$), ***tensor*** ($\otimes$), ***cut*** (cut), and ***mix*** (mix) are the standard as in multiplicative linear logic with mix. Note that ax is restricted to atomic formulas.

The ***dual connectives*** rule ($\mathsf{d}\text{-}\kappa$) handles a pair of dual connectives at the same time.[4] To get an intuition of this rule, consider the right-conjunction rule ($\wedge_R$) used in two-sided sequent calculi for classical logic

[3] Note that the permutation σ_P may be not unique. This is not a problem if we consider formulas up-to the equivalence relation $\equiv$. Otherwise, in order to properly define the linear negation, we should fix a permutation σ_P for each graphical connective $P \in \mathfrak{P}$ in such a way either σ_P is an involution (in case $G_P \sim (G_P)^{\perp}$), or $\sigma_P \sigma_Q$ is the identity (in case $G_P \not\sim (G_P)^{\perp} \sim G_Q$ for a $Q \in \mathfrak{P} \setminus \{P\}$).

[4] Rules handling multiple operators at the same time are not a novelty in structural proof theory: in focused proof systems (see, e.g.[10, 51, 50]) rules can handle multiple connectives of a same formula, while in modal logic and linear logic (see, e.g., [31, 13, 15, 45]) is quite standard to have rules handling modalities occurring in different formulas of a same sequent.

$$\text{ax}\,\frac{}{\vdash a, a^\perp} \qquad \parr\,\frac{\vdash \Gamma, \phi, \psi}{\vdash \Gamma, \phi \parr \psi} \qquad \otimes\,\frac{\vdash \Gamma, \phi \quad \vdash \psi, \Delta}{\vdash \Gamma, \phi \otimes \psi, \Delta}$$

$$\text{d-}\kappa\,\frac{\vdash \Gamma_1, \phi_{\sigma(1)}, \psi_{\tau(1)} \quad \cdots \quad \vdash \Gamma_n, \phi_{\sigma(n)}, \psi_{\tau(n)}}{\vdash \Gamma_1, \dots, \Gamma_n, \kappa(\!|\phi_1, \dots, \phi_n|\!), \kappa^\perp(\!|\psi_1, \dots \psi_n|\!)} \begin{cases}\sigma \in \mathfrak{S}(\kappa)\\ \tau \in \mathfrak{S}(\kappa^\perp)\end{cases}$$

$$\text{mix}\,\frac{\vdash \Gamma_1 \quad \vdash \Gamma_2}{\vdash \Gamma_1, \Gamma_2} \qquad \text{wd}_\otimes\,\frac{\vdash \Gamma, \phi_k \quad \vdash \Delta, \kappa(\!|\phi_1, \dots, \phi_{k-1}, \circ, \phi_{k+1}, \dots, \phi_n|\!)}{\vdash \Gamma, \Delta, \kappa(\!|\phi_1, \dots, \phi_n|\!)}$$

$$\text{unitor}_\kappa\,\frac{\vdash \Gamma, \chi(\!|\phi_{\sigma(1)}, \dots, \phi_{\sigma(n)}|\!)}{\vdash \Gamma, \kappa(\!|\phi_1, \dots, \phi_k, \circ, \phi_{k+1}, \dots, \phi_n|\!)}\,\dagger$$

$$\dagger := \sigma \in \mathfrak{S}(\chi) \quad \text{and} \quad [\![\kappa(\!|\phi_1, \dots, \phi_k, \circ, \phi_{k+1}, \dots, \phi_n|\!)]\!] = [\![\chi(\!|\phi_{\sigma(1)}, \dots, \phi_{\sigma(n)}|\!)]\!] \neq \emptyset$$

Figure 1: Linear sequent calculus rules for MGL and MGL$^\circ$.

shown below on the left. The interpretation of this rule is that if the left premise *and* the right premise are derivable, then the conclusion is. Note that, even if the rule does not introduce a conjunction on the lefthand-side of the $\vdash$, the interpretation of the conclusion sequent is the same of the interpretation of the sequent in which ϕ_1 and ϕ_2 are in conjunction because the standard interpretation of a two-sides sequent $\Gamma \vdash \Delta$ is defined as $\left(\bigwedge_{\phi\in\Gamma} \phi^\perp\right) \vee \left(\bigvee_{\psi\in\Delta} \psi\right)$.

$$\wedge_R\,\frac{\boxed{\Gamma_1, \phi_1 \vdash \psi_1, \Delta_1} \quad \text{“and”} \quad \boxed{\Gamma_2, \phi_2 \vdash \psi_2, \Delta_2}}{\Gamma_1, \Gamma_2, \phi_1, \phi_2 \vdash \psi_1 \wedge \psi_2, \Delta_1, \Delta_2} \quad \Bigg| \quad \frac{\mathsf{P}_4 (\!\!|\boxed{\Gamma_1, \phi_1 \vdash \psi_1, \Delta_1}, \boxed{\Gamma_2, \phi_2 \vdash \psi_2, \Delta_2}, \boxed{\Gamma_3, \phi_3 \vdash \psi_3, \Delta_3}, \boxed{\Gamma_4, \phi_4 \vdash \psi_4, \Delta_4}|\!\!)}{\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4, \kappa_{\mathsf{P}_4}(\!|\phi_1, \phi_2, \phi_3, \phi_4|\!) \vdash \kappa_{\mathsf{P}_4}(\!|\psi_1, \psi_2, \psi_3, \psi_4|\!), \Delta_1, \Delta_2, \Delta_3, \Delta_4}$$

In a two-sided setting the rule d-κ could have been reformulated by introducing the same connective in both sides. Intuitively, such a rule would internalize in the logic a meta-connective establishing a relation between the premises of the rule, as intuitively shown above on the right for the connective P_4.

The names of the rules ***unitor*** (unitor$_\kappa$) and ***weak-distributivity*** (wd$_\otimes$) are inspired by the literature of *monoidal categories* [47] and *weakly distributive categories* [59, 20, 19]. The rule unitor$_\kappa$ internalize the fact that the unit $\circ$ is the neutral element for all connectives (its side condition prevents the creation of non-pure formulas). Under the assumption of the existence of a $\circ$ which is the unit of both $\otimes$ and $\parr$, the rule wd$_\otimes$ generalizes the *weak-distribution law* (shown below on the left) of the $\otimes$ over the $\parr$ to the weak-distributivity of $\otimes$ over any connective (see below on the top-right)

$$\phi \otimes (\psi \parr \chi) \longrightarrow (\phi \otimes \psi) \parr \chi \quad \Bigg| \quad \begin{array}{l} \chi \otimes \kappa(\!|\phi_1, \dots, \phi_k, \psi, \phi_{k+1}, \dots, \phi_n|\!) \longrightarrow \kappa(\!|\phi_1, \dots, \phi_k, \psi \otimes \chi, \phi_{k+1}, \dots, \phi_n|\!) \\ \kappa(\!|\phi_1, \dots, \phi_k, \psi \parr \chi, \phi_{k+1}, \dots, \phi_n|\!) \longrightarrow \kappa(\!|\phi_1, \dots, \phi_k, \psi, \phi_{k+1}, \dots, \phi_n|\!) \parr \chi \end{array} \tag{10}$$

Note that an additional law is required to formalize the weak-distributivity law of all connectives over $\parr$ (see above on the bottom-right). This law corresponds to the rule wd$_\parr$ in Figure 2.

Notation 35. Unless strictly needed for sake of clarity, we omit to the permutations over the indices of the subformulas in rules.

3.1 Properties of the systems MGL and MGL$^\circ$

We start by observing that these systems are *initial coherent* [11, 50], that is, we can derive the implication $\phi \multimap \phi$ for any formula ϕ only using atomic axioms. To prove this result we observe that the generalized version of d-κ (that is, the rule d-χ) is derivable by induction on the structure of χ using the rule d-κ. Therefore, we can prove that the generalized non-atomic axiom rule (AX) is derivable, and that both MGL and MGL$^\circ$ are initial coherent

Lemma 36. *Let χ be a pure formula. Then rule d-χ is derivable.*

$$\mathsf{AX}\,\frac{}{\vdash \phi, \phi^\perp}\ \phi \text{ pure} \qquad \mathsf{cut}\,\frac{\vdash \Gamma_1, \phi \quad \vdash \Gamma_2, \phi^\perp}{\vdash \Gamma_1, \Gamma_2} \qquad \mathsf{wd}_{\parr}\,\frac{\vdash \Gamma, \kappa(\!|\phi, \psi_1, \ldots, \psi_n|\!)}{\vdash \Gamma, \kappa(\!|\circ, \psi_1, \ldots, \psi_n|\!), \phi}$$

$$\mathsf{deep}\,\frac{\vdash \Gamma, \phi \quad \vdash \Delta, \psi}{\vdash \Gamma, \Delta, \zeta[\phi]}\ [\![\zeta[\circ]]\!] = [\![\psi]\!] \qquad \mathsf{d}\text{-}\chi\,\frac{\vdash \Gamma_1, \phi_{\sigma(1)}, \psi_{\tau(1)} \quad \cdots \quad \vdash \Gamma_n, \phi_{\sigma(n)}, \psi_{\tau(n)}}{\vdash \Gamma_1, \ldots, \Gamma_n, \chi(\!|\phi_1, \ldots, \phi_n|\!), \chi^\perp(\!|\psi_1, \ldots \psi_n|\!)} \begin{cases} \sigma \in \mathfrak{S}(\chi) \\ \tau \in \mathfrak{S}(\chi^\perp) \end{cases}$$

Figure 2: Admissible rules in MGL°.

$$\mathsf{cut}\,\frac{\mathsf{ax}\,\frac{}{\vdash a, a^\perp} \quad \vdash a, \Gamma}{\vdash a, \Gamma} \rightsquigarrow \vdash a^\perp, \Gamma \qquad \mathsf{cut}\,\frac{\otimes\,\frac{\vdash \Gamma, \phi \quad \vdash \Delta, \psi}{\vdash \Gamma, \Delta, \phi \otimes \psi} \quad \parr\,\frac{\vdash \Sigma, \phi^\perp \parr \psi^\perp}{\vdash \Sigma, \phi^\perp \parr \psi^\perp}}{\vdash \Gamma, \Delta, \Sigma} \rightsquigarrow \mathsf{cut}\,\frac{\vdash \Gamma, \phi \quad \mathsf{cut}\,\frac{\vdash \Delta, \psi \quad \vdash \Sigma, \phi^\perp, \psi^\perp}{\vdash \Delta, \Sigma, \phi^\perp}}{\vdash \Gamma, \Delta, \Sigma}$$

$$\mathsf{cut}\,\frac{\mathsf{d}\text{-}\kappa\,\frac{\vdash \Gamma_1, \phi_1, \psi_1 \quad \cdots \quad \vdash \Gamma_n, \phi_n, \psi_n}{\vdash \Gamma_1, \ldots, \Gamma_n, \kappa(\!|\phi_1, \ldots, \phi_n|\!), \kappa^\perp(\!|\psi_1, \ldots, \psi_n|\!)} \quad \mathsf{d}\text{-}\kappa\,\frac{\vdash \Delta_1, \psi_1^\perp, \chi_1 \quad \cdots \quad \vdash \Delta_n, \psi_n^\perp, \chi_n}{\vdash \Delta_1, \ldots, \Delta_n, \kappa(\!|\psi_1^\perp, \ldots, \psi_n^\perp|\!), \kappa^\perp(\!|\chi_1, \ldots, \chi_n|\!)}}{\vdash \Gamma_1, \ldots, \Gamma_n, \Delta_1, \ldots, \Delta_n, \kappa(\!|\phi_1, \ldots, \phi_n|\!), \kappa^\perp(\!|\chi_1, \ldots, \chi_n|\!)}$$

$$\wr$$

$$\mathsf{d}\text{-}\kappa\,\frac{\mathsf{cut}\,\frac{\vdash \Gamma_1, \phi_1, \psi_1 \quad \vdash \Delta_1, \psi_1^\perp, \chi_1}{\vdash \Gamma_1, \Delta_1, \phi_1, \chi_1} \quad \cdots \quad \mathsf{cut}\,\frac{\vdash \Gamma_n, \phi_n, \psi_n \quad \vdash \Delta_n, \psi_n^\perp, \chi_n}{\vdash \Gamma_n, \phi_n, \chi_n}}{\vdash \Gamma_1, \ldots, \Gamma_n, \Delta_1, \ldots, \Delta_n, \kappa^\perp(\!|\psi_1, \ldots, \psi_n|\!), \kappa(\!|\chi_1, \ldots, \chi_n|\!)}$$

Figure 3: Cut-elimination steps for MGL.

Proof. By induction on the structure of χ:

- if $\phi = a$ is a literal, then AX is an instance of ax;
- if $\phi = \kappa(\!|\psi_1, \ldots, \psi_k, \circ, \psi_{k+1}, \ldots, \psi_n|\!)$, then apply twice unitor_κ to the sequent $\vdash \phi, \phi^\perp$ to obtain the sequent of pure formulas $\vdash \kappa_\chi(\!|\psi_1, \ldots, \psi_n|\!), \kappa_{\chi^\perp}(\!|\psi_1^\perp, \ldots, \psi_n^\perp|\!)$. We conclude by inductive hypothesis;
- if $\phi = \kappa(\!|\psi_1, \ldots, \psi_n|\!)$ and $\psi_i \neq \circ$ for all $i \in \{1, \ldots, n\}$, then apply the rule $\mathsf{d}\text{-}\kappa$ to obtain sequents of pure formulas the form $\psi_i, \psi_i^\perp$ for all $i \in \{1, \ldots, |\kappa|\}$. We conclude by inductive hypothesis.

□

Corollary 37. *The rule* AX *is derivable in* MGL *and in* MGL°.

Theorem 38. *The systems* MGL *and* MGL° *are initial coherent (with respect to pure formulas).*

We then prove the admissibility of cut via *cut-elimination* by providing a cut-elimination procedure.

Theorem 39 (Cut-elimination)**.** *Let* $\mathsf{X} \in \{\mathsf{MGL}, \mathsf{MGL}^\circ\}$. *The rule* cut *is admissible in* X.

Proof. We define the *size* of a formula as sum of the number of $\circ$, connectives and twice the number of literals in it. The *size* of a derivation is the sum of the sizes of the active formulas in all cut-rules. The result follows by the fact that each *cut-elimination step* from Figures 3 and 4 reduces the size of a derivation.

Note that in order to ensure that both active formulas of a cut are principal with respect to the rule immediately above it we also need to consider the *commutative* cut-elimination steps from Figure 5. The treatment of these rule, as well as the definition of a size taking into account them, is not covered in the detail here because it is standard in the literature (see, e.g., [61]). □

Corollary 40. *Let* $\mathsf{X} \in \{\mathsf{MGL}, \mathsf{MGL}^\circ\}$. *If* $\vdash_{\mathsf{X}} \phi \multimap \psi$ *and* $\vdash_{\mathsf{X}} \psi \multimap \chi$, *then* $\vdash_{\mathsf{X}} \phi \multimap \chi$.

The admissibility of the cut-rule implies analyticity of MGL via the standard *sub-formula property*, that is, all (occurrences of) formulas occurring in the premises of a rule are subformulas of the ones in the conclusion.

$$\mathsf{cut}\,\dfrac{\mathsf{unitor}_\kappa\,\dfrac{\vdash \Gamma, \chi(\!|\phi_2,\dots,\phi_n|\!)}{\vdash \Gamma, \kappa_P(\!|\circ,\phi_2,\dots,\phi_n|\!)} \qquad \mathsf{unitor}_\kappa\,\dfrac{\vdash \Delta, \chi^\perp(\!|\phi_2^\perp,\dots,\phi_n^\perp|\!)}{\vdash \Delta, \kappa_{P^\perp}(\!|\circ,\phi_2^\perp,\dots,\phi_n^\perp|\!)}}{\vdash \Gamma,\Delta} \rightsquigarrow \mathsf{cut}\,\dfrac{\vdash \Gamma, \chi(\!|\phi_2,\dots,\phi_n|\!) \qquad \vdash \Delta, \chi^\perp(\!|\phi_2^\perp,\dots,\phi_n^\perp|\!)}{\vdash \Gamma,\Delta}$$

$$\mathsf{cut}\,\dfrac{\mathsf{wd}_\otimes\,\dfrac{\vdash \Gamma_1,\phi_1 \quad \vdash \Gamma_2, \kappa_P(\!|\circ,\phi_2,\dots,\phi_n|\!)}{\vdash \Gamma_1,\Gamma_2,\kappa_P(\!|\phi_1,\dots,\phi_n|\!)} \qquad \mathsf{wd}_\otimes\,\dfrac{\vdash \Delta_1,\phi_1^\perp \quad \vdash \Delta_2, \kappa_{P^\perp}(\!|\circ,\phi_2^\perp,\dots,\phi_n^\perp|\!)}{\vdash \Delta,\kappa_{P^\perp}(\!|\phi_1^\perp,\dots,\phi_n^\perp|\!)}}{\vdash \Gamma_1,\Gamma_2,\Delta_1,\Delta_2}$$

$$\rightsquigarrow$$

$$\mathsf{mix}\,\dfrac{\mathsf{cut}\,\dfrac{\vdash \Gamma_1,\phi_1 \quad \vdash \Delta_1,\phi_1^\perp}{\vdash \Gamma_1,\Delta_1} \qquad \mathsf{cut}\,\dfrac{\vdash, \Gamma_2, \kappa_P(\!|\circ,\phi_2,\dots\phi_n|\!) \quad \vdash \Delta_2, \kappa_{P^\perp}(\!|\circ,\phi_2^\perp,\dots\phi_n^\perp|\!)}{\vdash \Gamma_2,\Delta_2}}{\vdash \Gamma_1,\Gamma_2,\Delta_1,\Delta_2}$$

$$\mathsf{cut}\,\dfrac{\mathsf{d}\text{-}\kappa\,\dfrac{\vdash \Gamma_1,\phi_1,\psi_1 \qquad \cdots \qquad \vdash \Gamma_n,\phi_n,\psi_n}{\vdash \Gamma_1,\dots,\Gamma_n,\kappa_P(\!|\phi_1,\dots,\phi_n|\!),\kappa_{P^\perp}(\!|\psi_1,\dots,\psi_n|\!)} \qquad \mathsf{wd}_\otimes\,\dfrac{\vdash \Delta,\psi_1^\perp \quad \vdash \Sigma,\kappa_P(\!|\circ,\psi_2^\perp,\dots,\psi_n^\perp|\!)}{\vdash \Delta,\Sigma,\kappa_P(\!|\psi_1^\perp,\dots,\psi_n^\perp|\!)}}{\vdash \Gamma_1,\dots,\Gamma_n,\Delta,\Sigma,\kappa_P(\!|\phi_1,\dots,\phi_n|\!)}$$

$$\rightsquigarrow$$

$$\mathsf{wd}_\otimes\,\dfrac{\mathsf{cut}\,\dfrac{\vdash \Gamma_1,\phi_1,\psi_1 \quad \vdash \Delta,\psi_1^\perp}{\vdash \Gamma_1,\Delta,\phi_1} \qquad \mathsf{cut}\,\dfrac{2{\times}\mathsf{unitor}_\kappa\,\dfrac{\mathsf{d}\text{-}\chi\,\dfrac{\vdash \Gamma_2,\phi_2,\psi_2 \qquad \cdots \qquad \vdash \Gamma_n,\phi_n,\psi_n}{\vdash \Gamma_2,\dots,\Gamma_n,\kappa_\chi(\!|\phi_1,\dots,\phi_n|\!),\kappa_\chi^\perp(\!|\psi_1,\dots,\psi_n|\!)}}{\vdash \Gamma_2,\dots,\Gamma_n,\kappa_P(\!|\circ,\phi_1,\dots,\phi_n|\!),\kappa_{P^\perp}(\!|\circ,\psi_1,\dots,\psi_n|\!)} \quad \vdash \Sigma,\kappa_P(\!|\circ,\psi_2^\perp,\dots,\psi_n^\perp|\!)}{\vdash \Gamma_2,\dots,\Gamma_n,\Sigma,\kappa_P(\!|\circ,\phi_2,\dots,\phi_n|\!)}}{\vdash \Gamma_1,\dots,\Gamma_n,\Delta,\Sigma,\kappa_P(\!|\phi_1,\dots,\phi_n|\!)}$$

Figure 4: Additional cut-elimination steps in GML°.

Corollary 41 (Analyticity of MGL)**.** *Let* Γ *be a sequent. If* $\vdash_{\mathsf{MGL}} \Gamma$*, then there is a proof of* Γ *in* MGL *only containing occurrences of sub-formulas of formulas* Γ.

However, the same result does not hold for MGL° because of the rule unitor_κ. In fact, the presence of more-than-binary connectives and their units (in this case, a unique unit $\circ$) implies, as observed in the previous works on graphical logic [4, 5, 2], the possibility of having *sub-connectives*, that is, connectives with smaller arity behaving as if certain entries of the connective are fixed to be units.

Definition 42. Let P and Q be prime graphs. If $P(\!|\circ,\dots,\circ,v_{i_1},\circ,\dots,\circ,v_{i_k},\circ,\dots,\circ|\!) \sim Q(\!|v_1,\dots,v_n|\!)$ for single-vertex graphs $v_1,\dots,v_n$ and for some distinct $i_1,\dots,i_k \in \{1,\dots,n\}$, then we may write $\kappa_{P|_{i_1,\dots,i_k}} = \kappa_Q$ and we say that the connective κ_Q is a ***sub-connective*** of if κ_P.

A ***quasi-subformula*** of a formula $\phi = \kappa_P(\!|\psi_1,\dots,\psi_n|\!)$ is a formula of the form $\kappa_{P'|_{i_1,\dots,i_k}}(\!|\psi'_{i_1},\dots,\psi'_{i_k}|\!)$ with ψ'_{i_j} a quasi-subformula of ψ_{i_j} for all $i_j \in \{i_1,\dots,i_k\}$.

Corollary 43 (Analyticity of MGL°)**.** *Let* Γ *be a sequent. If* $\vdash_{\mathsf{MGL}^\circ} \Gamma$ *then there is a proof of* Γ *in* MGL° *only containing occurrences of quasi-subformula of formulas in* Γ.

Corollary 44 (Conservativity)**.** *The logic* MGL *is a conservative extension of* MLL*. The logic* MGL° *is a conservative extension of* MLL°.

Proof. For MGL it is consequence of the subformula property. For MGL° it suffices to remark that $⅋$ and $\otimes$ have no sub-connectives, therefore quasi-subformula are simply sub-formulas. □

For both MGL and MGL° we have the following result which takes the name of *splitting* in the deep inference literature (see, e.g, [8, 35, 36]). This result states that is always possible, during proof search, to

$$\text{cut}\,\dfrac{\rho\,\dfrac{\vdash \Gamma_1, \Delta', \phi}{\vdash \Gamma_1, \Delta, \phi} \qquad \vdash \phi^{\perp}, \Gamma_2}{\vdash \Gamma_1, \Gamma_2, \Delta} \quad\rightsquigarrow\quad \rho\,\dfrac{\text{cut}\,\dfrac{\vdash \Gamma_1, \Delta', \phi \qquad \vdash \phi^{\perp}, \Gamma_2}{\vdash \Gamma_1, \Gamma_2, \Delta}}{\vdash \Gamma_1, \Gamma_2, \Delta'}$$

$$\text{cut}\,\dfrac{\rho\,\dfrac{\vdash \Gamma_1, \Delta'_1 \quad \cdots \quad \vdash \Gamma_n, \Delta'_n, \phi}{\vdash \Gamma_1, \ldots, \Gamma_n, \Delta, \phi} \qquad \vdash \phi^{\perp}, \Gamma_{n+1}}{\vdash \Gamma_1, \ldots, \Gamma_{n+1}, \Delta} \quad\rightsquigarrow\quad \rho\,\dfrac{\vdash \Gamma_1, \Delta'_1 \quad \cdots \quad \vdash \Gamma_{n-1}, \Delta'_{n-1} \qquad \text{cut}\,\dfrac{\vdash \Gamma_n, \Delta'_n, \phi \quad \vdash \Gamma_{n+1}, \phi^{\perp}}{\vdash \Gamma_n, \Gamma_{n+1}, \Delta'_n}}{\vdash \Gamma_1, \ldots, \Gamma_{n+1}, \Delta}$$

$$\text{unitor}_{\kappa}\,\dfrac{\text{unitor}_{\kappa}\,\dfrac{\vdash \Gamma, \chi(\!|\phi_1, \ldots, \phi_{i-1}, \phi_{i+1}, \ldots, \phi_{j-1}, \phi_{j+1}, \ldots, \phi_n|\!)}{\vdash \Gamma, \kappa_P(\!|\phi_1, \ldots, \phi_{i-1}, \circ, \phi_{i+1}, \ldots, \phi_{j-1}, \phi_{j+1}, \ldots, \phi_n|\!)}}{\vdash \Gamma \kappa_P(\!|\phi_1, \ldots, \phi_{i-1}, \circ, \phi_{i+1}, \ldots, \phi_{j-1}, \circ, \phi_{j+1}, \ldots, \phi_n|\!)} \quad\rightsquigarrow\quad \text{unitor}_{\kappa}\,\dfrac{\text{unitor}_{\kappa}\,\dfrac{\vdash \Gamma, \chi(\!|\phi_1, \ldots, \phi_{i-1}, \phi_{i+1}, \ldots, \phi_{j-1}, \phi_{j+1}, \ldots, \phi_n|\!)}{\vdash \Gamma, \kappa_{P'}(\!|\phi_1, \ldots, \phi_{i-1}, \phi_{i+1}, \ldots, \phi_{j-1}, \circ, \phi_{j+1}, \ldots, \phi_n|\!)}}{\vdash \Gamma, \kappa_P(\!|\phi_1, \ldots, \phi_{i-1}, \circ, \phi_{i+1}, \ldots, \phi_{j-1}, \circ, \phi_{j+1}, \ldots, \phi_n|\!)}$$

Figure 5: Commutative cut-elimination steps.

apply a rule removing a connective after having applied certain rules in the context.[5]

Lemma 45 (Splitting). *Let* $\Gamma, \kappa(\!|\phi_1, \ldots, \phi_n|\!)$ *be a sequent and let* $\mathsf{X} \in \{\mathsf{MGL}, \mathsf{MGL}^{\circ}\}$. *If* $\vdash_{\mathsf{X}} \Gamma, \kappa(\!|\phi_1, \ldots, \phi_n|\!)$, *then there is a derivation of the following shape*

$$\begin{array}{c} \pi_1 \,\Vert \\ \text{unitor}_{\kappa}\,\dfrac{\vdash \Gamma', \chi(\!|\phi_1, \ldots, \phi_{k-1}, \phi_{k+1}, \phi_n|\!)}{\vdash \Gamma', \kappa(\!|\phi_1, \ldots, \phi_{k-1}, \circ, \phi_{k+1}, \phi_n|\!)} \\ \pi_0 \,\Vert \\ \vdash \Gamma, \kappa(\!|\phi_1, \ldots, \phi_{k-1}, \circ, \phi_{k+1}, \phi_n|\!) \end{array} \quad \textit{or} \quad \begin{array}{c} \pi_1 \,\Vert \qquad\qquad \pi_n \,\Vert \\ \rho\,\dfrac{\vdash \Delta_1, \phi_1 \quad \cdots \quad \vdash \Delta_n, \phi_n}{\vdash \Gamma', \kappa(\!|\phi_1, \ldots, \phi_n|\!)} \\ \pi_0 \,\Vert \\ \vdash \Gamma, \kappa(\!|\phi_1, \ldots, \phi_n|\!) \end{array} \quad \textit{with } \rho \in \{⅋, \otimes, \mathsf{d}\text{-}\kappa\}$$

Proof. By case analysis of the last rule occurring in a proof π of $\Gamma, \kappa(\!|\phi_1, \ldots, \phi_n|\!)$:

- the last rule cannot be a ax since $\kappa(\!|\phi_1, \ldots, \phi_n|\!)$ contains at least one connective;
- if the last rule is a ⅋ or a unitor_{κ}, then either this is the desired rule, or we conclude by inductive hypothesis on its premise;
- if the last rule is a mix, then we conclude by inductive hypothesis on the premise containing the formula $\kappa(\!|\phi_1, \ldots, \phi_n|\!)$;
- if the last rule is in $\{\otimes, \mathsf{d}\text{-}\kappa, \mathsf{wd}_{\otimes}, \mathsf{unitor}_{\kappa}\}$ then either this is the desired rule or one of the (provable) premises of this rule is of the shape $\Gamma', \kappa(\!|\phi_1, \ldots, \phi_n|\!)$, allowing us to conclude by inductive hypothesis. □

We conclude this section proving the admissibility of the rule $\mathsf{wd}_{⅋}$ which we use to simplify proofs in the next section.

Lemma 46. *The rule* $\mathsf{wd}_{⅋}$ *is admissible in* MGL°.

Proof. In Figure 6 we providde a procedure to remove (top-down) all occurrences of $\mathsf{wd}_{⅋}$. Similar to cut-elimination, we use the commutative steps from Figure 5 to ensure that the active formula of the $\mathsf{wd}_{⅋}$ we want to remove is principal with respect to the rule immediately above it. □

Lemma 47. *The rule* deep *is admissible in* MGL°.

Proof. Since $\zeta[\circ] \neq \circ$, then w.l.o.g., $\zeta[\Box] = \kappa(\!|\zeta'[\Box], \psi'_1, \ldots, \psi'_n|\!)$. If $\zeta'[\Box] = \Box$, then w.l.o.g., $\psi = \chi(\!|\psi'_1, \ldots, \psi'_n|\!)$ and we conclude since we have

$$\mathsf{wd}_{⅋}\,\dfrac{\vdash \Gamma, \phi \qquad \text{unitor}_{\kappa}\,\dfrac{\vdash \Delta, \chi(\!|\psi'_1, \ldots, \psi'_n|\!)}{\vdash \Delta, \kappa(\!|\circ, \psi'_1, \ldots, \psi'_n|\!)}}{\vdash \Gamma, \Delta, \kappa(\!|\phi, \psi'_1, \ldots, \psi'_n|\!)}$$

[5]Note that in the linear logic literature, the splitting lemma is usually formulated as a special case of the lemma here, where a $\otimes$ is removed without requiring the application of rules to the context.

$$\text{wd}_{\parr}\dfrac{\parr\dfrac{\overset{\pi_1}{\vdots}\ \vdash \Gamma, \phi, \psi}{\vdash \Gamma, \phi \parr \psi}}{\vdash \Gamma, \phi, \circ \parr \psi} \rightsquigarrow \text{unitor}_\kappa \dfrac{\overset{\pi_1}{\vdots}\ \vdash \Gamma, \phi, \psi}{\vdash \Gamma, \phi, \circ \parr \psi} \qquad \text{wd}_{\otimes}\dfrac{\otimes\dfrac{\overset{\pi_1}{\vdots}\ \vdash \Gamma, \phi \quad \overset{\pi_2}{\vdots}\ \vdash \Delta, \psi}{\vdash \Gamma, \Delta, \phi \otimes \psi}}{\vdash \Gamma, \Delta, \phi, \circ \otimes \psi} \rightsquigarrow \text{unitor}_\kappa\dfrac{\text{mix}\dfrac{\overset{\pi_1}{\vdots}\ \vdash \Gamma, \phi \quad \overset{\pi_2}{\vdots}\ \vdash \Delta, \psi}{\vdash \Gamma, \Delta, \phi, \psi}}{\vdash \Gamma, \Delta, \phi, \circ \otimes \psi}$$

$$\text{wd}_{\parr}\dfrac{\text{d-}\kappa\dfrac{\overset{\pi_1}{\vdots}\ \vdash \Gamma_1, \phi, \psi_1 \quad \overset{\pi_2}{\vdots}\ \vdash \Gamma_2, \phi_2, \psi_2 \quad \cdots \quad \overset{\pi_n}{\vdots}\ \vdash \Gamma_n, \phi_n, \psi_n}{\vdash \Gamma_1, \ldots, \Gamma_n, \kappa^\perp(\!|\psi_1, \ldots, \psi_n|\!), \kappa(\!|\phi, \phi_2, \ldots, \phi_n|\!)}}{\vdash \Gamma_1, \ldots, \Gamma_n, \kappa^\perp(\!|\psi_1, \ldots, \psi_n|\!), \kappa(\!|\circ, \phi_2, \ldots, \phi_n|\!), \phi} \rightsquigarrow \text{wd}_{\otimes}\dfrac{\overset{\pi_1}{\vdots}\ \vdash \Gamma_1, \phi, \psi_1 \quad 2\times\text{unitor}_\kappa\dfrac{\text{d-}\chi\dfrac{\overset{\pi_2}{\vdots}\ \vdash \Gamma_2, \psi_2, \chi_2 \quad \cdots \quad \overset{\pi_n}{\vdots}\ \vdash \Gamma_n, \psi_n, \chi_n}{\vdash \Gamma_2, \ldots, \Gamma_n, \chi^\perp(\!|\psi_2, \ldots, \psi_n|\!), \chi(\!|\phi_2, \ldots, \phi_n|\!)}}{\vdash \Gamma_2, \ldots, \Gamma_n, \kappa^\perp(\!|\circ, \psi_1, \ldots, \psi_n|\!), \kappa(\!|\circ, \phi_2, \ldots, \phi_n|\!)}}{\vdash \Gamma_1, \ldots, \Gamma_n, \kappa^\perp(\!|\psi_1, \ldots, \psi_n|\!), \kappa(\!|\circ, \phi_2, \ldots, \phi_n|\!), \phi}$$

$$\text{wd}_{\parr}\dfrac{\text{wd}_{\otimes}\dfrac{\overset{\pi_1}{\vdots}\ \vdash \Gamma_1, \psi_k \quad \overset{\pi_2}{\vdots}\ \vdash \Gamma_2, \kappa(\!|\phi, \psi_2, \ldots, \psi_{k-1}, \circ, \psi_{k+1}, \ldots, \psi_n|\!)}{\vdash \Gamma_1, \Gamma_2, \kappa(\!|\phi, \psi_2, \ldots, \psi_n|\!)}}{\vdash \Gamma_1, \Gamma_2, \kappa(\!|\circ, \psi_2, \ldots, \psi_n|\!), \phi} \rightsquigarrow \text{wd}_{\otimes}\dfrac{\overset{\pi_1}{\vdots}\ \vdash \Gamma_1, \psi' \quad \text{wd}_{\parr}\dfrac{\overset{\pi_2}{\vdots}\ \vdash \Gamma_2, \kappa(\!|\phi, \psi_2, \ldots, \psi_{k-1}, \circ, \psi_{k+1}, \ldots, \psi_n|\!)}{\vdash \Gamma_2, \kappa(\!|\circ, \psi_2, \ldots, \psi_{k-1}, \circ, \psi_{k+1}, \ldots, \psi_n|\!), \phi}}{\vdash \Gamma, \kappa(\!|\circ, \psi_2, \ldots, \psi_k, \psi', \psi_{k+1}, \ldots, \psi_n|\!), \phi}$$

$$\text{wd}_{\parr}\dfrac{\text{unitor}_\kappa\dfrac{\overset{\pi_1}{\vdots}\ \vdash \Gamma, \chi(\!|\phi, \psi_2, \ldots, \psi_{n-1}|\!)}{\vdash \Gamma, \kappa(\!|\phi, \psi_2, \ldots, \psi_{n-1}, \circ|\!)}}{\vdash \Gamma, \kappa(\!|\circ, \psi_2, \ldots, \psi_{n-1}, \circ|\!), \phi} \rightsquigarrow \text{unitor}_\kappa\dfrac{\text{wd}_{\parr}\dfrac{\overset{\pi_1}{\vdots}\ \vdash \Gamma, \chi(\!|\phi, \psi_2, \ldots, \psi_{n-1}|\!)}{\vdash \Gamma, \chi(\!|\circ, \psi_2, \ldots, \psi_{n-1}|\!), \phi}}{\vdash \Gamma, \kappa(\!|\circ, \psi_2, \ldots, \psi_{n-1}, \circ|\!), \phi}$$

Figure 6: Steps to eliminate $\text{wd}_{\parr}$ rules.

Otherwise we conclude by inductive hypothesis on the size of $\zeta[\square]$ since by Lemma 45 we can define a derivation of the form

$$\begin{array}{c}\text{unitor}_\kappa\dfrac{\overset{}{\vdots}\text{IH}\ \ \vdash \Gamma', \Delta', \chi(\!|\zeta[\phi], \psi_1, \ldots, \psi_n|\!)}{\vdash \Gamma', \Delta', \kappa(\!|\zeta[\phi], \psi_1, \ldots, \psi_{k-1}, \circ, \psi_{k+1} \ldots \psi_n|\!)} \\ \pi_0 \| \\ \vdash \Gamma, \Delta, \kappa(\!|\zeta[\phi], \psi_1, \ldots, \psi_{k-1}, \circ, \psi_{k+1} \ldots \psi_n|\!)\end{array} \quad \text{or} \quad \begin{array}{c}\rho\dfrac{\overset{}{\vdots}\text{IH}\ \ \vdash \Gamma', \Delta_0, \zeta[\phi] \quad \overset{\pi_1}{\vdots}\ \vdash \Delta_1, \psi'_1 \quad \cdots \quad \overset{\pi_n}{\vdots}\ \vdash \Delta_n, \psi'_n}{\vdash \Gamma', \Delta', \kappa(\!|\zeta[\phi], \psi'_1, \ldots, \psi'_n|\!)} \\ \pi_0 \| \\ \vdash \Gamma, \Delta, \kappa(\!|\zeta[\phi], \psi'_1, \ldots, \psi'_n|\!)\end{array}$$

with $\rho \in \{\parr, \otimes, \text{d-}\kappa\}$. □

3.2 Graph Isomorphism as Logical Equivalence

In this sub-section we prove that two formulas ϕ and ψ are interpreted by a same graph (i.e., $[\![\phi]\!] = [\![\phi]\!]$) iff they are logically equivalent (i.e., $\phi \multimapboth \psi$). For this purpose, we show that all equivalence and De Morgan laws from Definition 29 can be reformulated as logical equivalences.

We first prove that connectives symmetries are derivable in MGL.

Lemma 48. *The following rules are admissible in* MGL.

$$\text{sym-}\kappa \dfrac{\vdash \Gamma, \kappa(\!|\phi_1, \ldots, \phi_n|\!)}{\vdash \Gamma, \kappa(\!|\phi_{\sigma(1)}, \ldots, \phi_{\sigma(n)}|\!)}\ \sigma \in \mathfrak{S}(Q) \qquad \text{dsym-}\kappa \dfrac{\vdash \Gamma, \kappa^\perp(\!|\phi_1, \ldots, \phi_n|\!)}{\vdash \Gamma, \kappa(\!|\phi_{\rho(1)}, \ldots, \phi_{\rho(n)}|\!)}\ \rho \in \mathfrak{S}^\perp(Q) \tag{11}$$

Proof. By Theorem 39, it suffices to prove that the following implications are derivable.

$$\underbrace{\begin{array}{c}\kappa(\!|\phi_1, \ldots, \phi_n|\!) \multimap \kappa(\!|\phi_{\sigma(1)}, \ldots, \phi_{\sigma(n)}|\!) \\ \kappa(\!|\phi_{\sigma(1)}, \ldots, \phi_{\sigma(n)}|\!) \multimap \kappa(\!|\phi_1, \ldots, \phi_n|\!)\end{array}}_{\text{for all } \sigma\in\mathfrak{S}(Q)} \quad \text{and} \quad \underbrace{\begin{array}{c}\kappa(\!|\phi_1, \ldots, \phi_n|\!) \multimap \kappa^\perp(\!|\phi_{\rho(1)}, \ldots, \phi_{\rho(n)}|\!) \\ \kappa^\perp(\!|\phi_{\rho(1)}, \ldots, \phi_{\rho(n)}|\!) \multimap \kappa(\!|\phi_1, \ldots, \phi_n|\!)\end{array}}_{\text{for all } \tau\in\mathfrak{S}^\perp(Q)}$$

These are easily derivable using an instance of d-κ and AX-rules. □

Remark 49. The rule sym-$\parr$ is derivable directly because sequents are sets if occurrences of formulas, therefore the order of the occurrences of the formulas in a sequent is not relevant, and we can permute this order before applying the rule $\parr$. This because the interpretation of the meta-connective comma we use to separate formulas in a sequent is the same of $\parr$.

Similarly, the rule sym-$\otimes$ is derivable because in our sequent system, as in standard sequent calculus, the order of the premises of the rules is not relevant. Said differently, the space between branches in a derivation is a commutative meta-connective which is internalized by the $\otimes$.

Similarly we can prove that the associativity of $\parr$ and $\otimes$ is derivable.

Lemma 50. *The following rules are admissible.*

$$\parr\text{-asso}\,\frac{\vdash \Gamma, (\phi_1 \parr \phi_2) \parr \phi_3}{\vdash \Gamma, \phi_1 \parr (\phi_2 \parr \phi_3)} \qquad \otimes\text{-asso}\,\frac{\vdash \Gamma, (\phi_1 \otimes \phi_2) \otimes \phi_3}{\vdash \Gamma, \phi_1 \otimes (\phi_2 \otimes \phi_3)} \tag{12}$$

Proof. The result follows by Theorem 39 after showing that following implications hold:

$$\begin{array}{cc} \phi_1 \parr (\phi_2 \parr \phi_3) \multimap (\phi_1 \parr \phi_2) \parr \phi_3 & (\phi_1 \parr \phi_2) \parr \phi_3 \multimap \phi_1 \parr (\phi_2 \parr \phi_3) \\ \phi_1 \otimes (\phi_2 \otimes \phi_3) \multimap (\phi_1 \otimes \phi_2) \otimes \phi_3 & (\phi_1 \otimes \phi_2) \otimes \phi_3 \multimap \phi_1 \otimes (\phi_2 \otimes \phi_3) \end{array}$$

□

We can therefore immediately conclude that MGL is sound and complete with respect to graph isomorphism if we consider unit-free formulas.

Proposition 51. *Let ϕ and ψ be unit-free formulas. Then $\phi \equiv \psi$ iff $\vdash_{\mathsf{MGL}} \phi \multimapboth \psi$.*

Proof. By induction on the formulas ϕ and ψ using Lemmas 48 and 50. □

For a stronger result on general formulas, we need to show that for any two formulas ϕ and ψ are interpreted (via $[\![\cdot]\!]$) by the same non-empty graph, both these formulas are equivalent to a unit-free formula χ representing the modular decomposition of this graph via graphical connectives.

Theorem 52. *Let ϕ and ψ be pure formulas. Then $\phi \equiv \psi$ iff $\vdash_{\mathsf{MGL}^\circ} \phi \multimapboth \psi$.*

Proof. Given any formula ϕ, we can define by induction on the number of units $\circ$ occurring in a unit-free formula ϕ' such that $\phi \multimapboth \phi'$.

- if ϕ is a literal, then $\phi' = \phi$;
- if $\phi = \kappa(\!|\phi_1, \ldots, \phi_n|\!)$ and $\phi_i \neq \circ$ for all $i \in \{1, \ldots, n\}$, then $\phi' = \kappa(\!|\phi'_1, \ldots, \phi'_n|\!)$. Otherwise, w.l.o.g., we assume $\phi_i = \circ$ and we let $\phi' = \chi(\!|\phi'_2, \ldots, \phi'_n|\!)$ for a compact formula χ such that $[\![\kappa(\!|\circ, \phi_2, \ldots, \phi_n|\!)]\!] = [\![\chi(\!|\phi_2, \ldots, \phi_n|\!)]\!]$ and we conclude by inductive hypothesis since we the following derivations:

$$\parr\,\frac{\mathsf{unitor}_\kappa\,\dfrac{\mathsf{d}\text{-}\chi\,\dfrac{\overset{\mathsf{IH}\,\Vert}{\vdash \phi_2^\perp, \phi'_2} \quad \cdots \quad \overset{\mathsf{IH}\,\Vert}{\vdash \phi_n^\perp, \phi'_n}}{\vdash \chi^\perp(\!|\phi_2^\perp, \ldots, \phi_n^\perp|\!), \chi(\!|\phi'_2, \ldots, \phi'_n|\!)}}{\vdash \kappa^\perp(\!|\circ, \phi_2^\perp, \ldots, \phi_n^\perp|\!), \chi(\!|\phi'_2, \ldots, \phi'_n|\!)}}{\vdash \kappa^\perp(\!|\circ, \phi_2^\perp, \ldots, \phi_n^\perp|\!) \parr \chi(\!|\phi'_2, \ldots, \phi'_n|\!)} \quad \text{and} \quad \parr\,\frac{\mathsf{unitor}_\kappa\,\dfrac{\mathsf{d}\text{-}\chi\,\dfrac{\overset{\mathsf{IH}\,\Vert}{\vdash \phi_2'^\perp, \phi_2} \quad \cdots \quad \overset{\mathsf{IH}\,\Vert}{\vdash \phi_n'^\perp, \phi_n}}{\vdash \chi^\perp(\!|\circ, \phi_2'^\perp, \ldots, \phi_n'^\perp|\!), \chi(\!|\circ, \phi_2, \ldots, \phi_n|\!)}}{\vdash \chi^\perp(\!|\phi_2'^\perp, \ldots, \phi_n'^\perp|\!), \kappa(\!|\circ, \phi_2, \ldots, \phi_n|\!)}}{\vdash \chi^\perp(\!|\phi'_2, \ldots, \phi_n'^\perp|\!) \parr \kappa(\!|\circ, \phi_2, \ldots, \phi_n|\!)}$$

Therefore we can find unit-free formulas ϕ' and ψ' such that $\phi \multimapboth \phi'$ and $\psi \multimapboth \psi'$. Moreover, by definition of $[\![\cdot]\!]$ and the rule unitor_κ we have $[\![\phi]\!]' = [\![\phi]\!] = [\![\psi]\!] = [\![\psi]\!]'$. We conclude by Proposition 51 and the transitivity of $\multimapboth$. □

4 Soundness and Completeness of MGL° with respect to GS

In this section we prove that set of graphs which are derivable in the graphical logic GS from [4, 5] is the same set of graph corresponding to formulas which are provable in MGL°.

In Figure 7 we recall the definition of the rules of the deep inference system[6] $\mathsf{GS} = \{\mathsf{ai}\!\downarrow, \mathsf{s}_\parr, \mathsf{s}_\otimes, \mathsf{p}\!\downarrow\}$.

Remark 53. At the syntactical level, the system GS operates on graphs by manipulating their modular decompositions trees. Therefore, for any graph occurring in a derivation in GS we assume a unique formula $[\![G]\!]^{-1}$ to be given. Note that in GS the authors allow themselves to consider modular decomposition trees in which leaves may be empty graphs, corrisponding to formulas with unit.

[6] The definition of deep inference systems operating on graphs can be found in [5] or in Appendix A.

$$\text{ai}\!\downarrow \frac{\varnothing}{a^\perp \parr a} \qquad \text{p}\!\downarrow \frac{(M_1 \parr N_1) \otimes \cdots \otimes (M_n \parr M'_n)}{P^\perp(\!| M_1, \ldots, M_n |\!) \parr P(\!| M'_1, \ldots, M'_n |\!)}$$

$$\text{s}_\parr \frac{P(\!| M_1, \ldots, M_{i-1}, M_i \parr N, M_{i+1}, \ldots M_n |\!)}{M_i \parr P(\!| M_1, \ldots, M_{i-1}, N, M_{i+1}, \ldots, M_n |\!)} \qquad \text{s}_\otimes \frac{M_i \otimes P(\!| M_1, \ldots, M_{i-1}, N, M_{i+1}, \ldots, M_n |\!)}{P(\!| M_1, \ldots, M_{i-1}, M_i \otimes N, M_{i+1}, \ldots, M_n |\!)}$$

Figure 7: Inference rules for the system GS, where P is a prime graph and $M_i \neq \varnothing \neq M'_i$ for all $i \in \{1, \ldots, n\}$.

Remark 54. The set of rules we consider here is a slightly different formulation of with respect to [4] and [5]: we consider a p-rules with a stronger side condition (all factors to be non-empty) which is balanced by the presence of $\text{s}_\otimes$ in the system. The proof that the formulation we consider in this paper is equivalent to the ones in the literature is provided in Appendix A.1.

We can easily prove that each sequent provable in MGL° is interpreted by $[\![\cdot]\!]$ as a graph which is admitting a proof in GS.

Lemma 55. *Let Γ be a sequent. If $\vdash_{\text{MGL}^\circ} \Gamma$, then $\vdash_{\text{GS}} [\![\Gamma]\!]$.*

Proof. We define a derivation $[\![\pi]\!]$ of $[\![\Gamma]\!]$ in GS by induction by induction on the last rule r in a derivation π of Γ in MGL° according to Figure 8. □

To prove the converse, we use the admissibility of $\text{wd}_\parr$ to prove that every time there is a rule in GS with premise H and conclusion G, then there are formulas ϕ and ψ such that $[\![\phi]\!]$ and $[\![\psi]\!]$, and such that $\psi \multimapdotboth \phi$.

Lemma 56. *Let $\text{r} \in \{\text{s}_\parr, \text{s}_\otimes, \text{p}\!\downarrow\}$. If $\text{r}\frac{H}{G}$, then there are formulas ϕ and ψ with $[\![\phi]\!] = G$ and $[\![\psi]\!] = H$ such that $\vdash_{\text{MGL}^\circ} \psi^\perp, \phi$.*

Proof. We first discuss the case if $C[\square] = \square$:

- if $\text{r} = \text{s}_\parr$, then $\phi = \mu_i \parr \kappa(\!| \mu_1, \ldots, \mu_{i-1}, \circ \parr \nu, \mu_{i+1}, \ldots \mu_n |\!)$ and $\psi = \kappa(\!| \mu_1, \ldots, \mu_{i-1}, \mu_i \parr \nu, \mu_{i+1}, \ldots \mu_n |\!)$ for some formulas $\mu_1, \ldots, \mu_n, \nu$ such that $[\![\mu_i]\!] = M_i$ for all $i \in \{1, \ldots, n\}$ and $[\![\nu]\!] = N$. We conclude by Corollary 37 and lemma 46 since we have the following derivation

$$\parr\frac{\text{wd}_\parr\dfrac{\text{AX}\dfrac{}{\vdash \psi^\perp, \kappa(\!| \mu_1, \ldots, \mu_{i-1}, \mu_i \parr \nu, \mu_{i+1}, \ldots, \mu_n |\!)}}{\vdash \psi^\perp, \mu_i, \kappa(\!| \mu_1, \ldots, \mu_{i-1}, \circ \parr \nu, \mu_{i+1}, \ldots, \mu_n |\!)}}{\vdash \psi^\perp, \phi}$$

- if $\text{r} = \text{s}_\otimes$ then $\phi = \kappa(\!| \mu_1, \ldots, \mu_{i-1}, \mu_i \otimes \nu, \mu_{i+1}, \ldots \mu_n |\!)$ and $\psi = \mu_i \otimes \kappa(\!| \mu_1, \ldots, \mu_{i-1}, \circ \otimes \nu, \mu_{i+1}, \ldots \mu_n |\!)$ for some formulas $\mu_1, \ldots, \mu_n, \nu$ such that $[\![\mu_i]\!] = M_i$ for all $i \in \{1, \ldots, n\}$ and $[\![\nu]\!] = N$. We conclude by Corollary 37 and lemma 46 since we have the following derivation

$$\parr\frac{\text{cxt-}\parr\dfrac{\text{AX}\dfrac{}{\vdash \kappa^\perp(\!| \mu_1^\perp, \ldots, \mu_{i-1}^\perp, \mu_i^\perp \parr \nu^\perp, \mu_{i+1}^\perp, \ldots \mu_n^\perp |\!), \phi}}{\vdash \mu_i^\perp, \kappa^\perp(\!| \mu_1^\perp, \ldots, \mu_{i-1}^\perp, \circ \parr \nu^\perp, \mu_{i+1}^\perp, \ldots \mu_n^\perp |\!), \phi}}{\vdash \psi^\perp, \phi}$$

- if $\text{r} = \text{p}\!\downarrow$ then $\phi = \kappa_{P^\perp}(\!| \mu_1, \ldots, \mu_n |\!) \parr \kappa_P(\!| \nu_1, \ldots, \nu_n |\!)$ and $\psi^\perp = (\mu_1^\perp \otimes \nu_1^\perp) \parr \cdots \parr (\mu_n^\perp \otimes \nu_n^\perp)$ for some formulas $\mu_1, \ldots, \mu_n, \nu_1, \ldots, \nu_n$ such that $[\![\mu_i]\!] = M_i \neq \varnothing$ and $[\![\nu_i]\!] = N_i \neq \varnothing$ for all $i \in \{1, \ldots, n\}$. We conclude since we have the following derivation

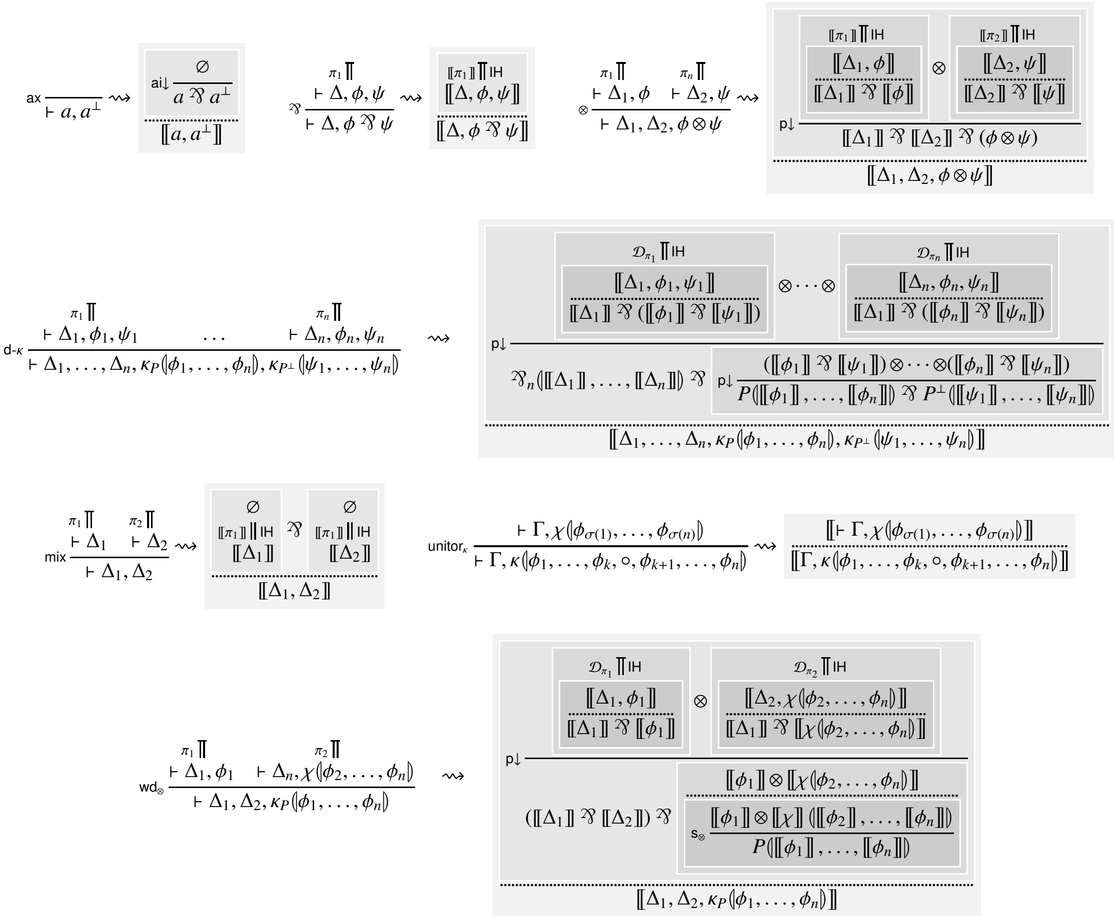


Figure 8: Rules to translate derivations in MGL$^\circ$ into derivations in GS.

$$\parr\,\dfrac{\mathsf{d}\text{-}\kappa\,\dfrac{\otimes\,\dfrac{\mathsf{AX}\,\dfrac{}{\vdash \mu_1,\mu_1^\perp}\quad \mathsf{AX}\,\dfrac{}{\vdash \nu_1,\nu_1^\perp}}{\vdash \mu_1^\perp\otimes\nu_1^\perp,\mu_1,\nu_1}\quad\cdots\quad \otimes\,\dfrac{\mathsf{AX}\,\dfrac{}{\vdash \mu_n,\mu_n^\perp}\quad \mathsf{AX}\,\dfrac{}{\vdash \nu_n,\nu_n^\perp}}{\vdash \mu_n^\perp\otimes\nu_n^\perp,\mu_n,\nu_n}}{\vdash (\mu_1^\perp\otimes\nu_1^\perp),\dots,(\mu_n^\perp\otimes\nu_n^\perp),\phi}}{(\mu_1^\perp\otimes\nu_1^\perp)\parr\cdots\parr(\mu_n^\perp\otimes\nu_n^\perp),\phi}$$

If $C[\square]=\kappa_P(\!|C'[\square],M_1,\dots,M_n|\!)\neq\square$, then we assume w.l.o.g., there is a context formula $\zeta[\square]=\kappa_P(\!|\zeta'[\square],\mu_1,\dots,\mu_n|\!)$ such that $[\![\zeta[\square]]\!]=C[\square]$ and $[\![\zeta'[\square]]\!]=C'[\square]$. We conclude since, by inductive hypothesis on the structure of $C[\square]$, there is a derivation of the following form:

$$\mathsf{d}\text{-}\kappa\,\dfrac{\dfrac{\Vert\,\mathsf{IH}}{\vdash(\zeta'[\psi'])^\perp,\zeta'[\phi']}\quad \mathsf{AX}\,\dfrac{}{\vdash \mu_1^\perp,\mu_1}\quad\cdots\quad \mathsf{AX}\,\dfrac{}{\vdash \mu_n^\perp,\mu_n}}{\vdash \kappa_{P^\perp}\left(\!\left|(\zeta'[\psi'])^\perp,\mu_1^\perp,\dots,\mu_n^\perp\right|\!\right),\kappa_P\left(\!\left|\zeta'[\phi'],\mu_1,\dots,\mu_n\right|\!\right)}\quad.$$

□

We are now able to prove the main result of this section, that is, establishing a correspondence between graphs provable in GS and graphs which are interpretation via $[\![\cdot]\!]$ of formulas provable in MGL$^\circ$.

Theorem 57. *Let $G\neq\emptyset$ be a graph and ϕ a pure formula such that $[\![\phi]\!]=G$. Then $\vdash_{\mathsf{GS}} G$ iff $\vdash_{\mathsf{MGL}^\circ}\phi$.*

$$\mathsf{w}\,\frac{\vdash \Gamma}{\vdash \Gamma, \phi} \qquad \mathsf{c}\,\frac{\vdash \Gamma, \phi, \phi}{\vdash \Gamma, \phi} \quad \Big| \quad \mathsf{w}\!\downarrow\,\frac{\psi}{\psi \parr \phi} \qquad \mathsf{c}\!\downarrow\,\frac{\phi \parr \phi}{\phi} \quad \Big| \quad \mathsf{ac}\!\downarrow\,\frac{a \parr a}{a} \qquad \mathsf{m}\,\frac{P(\!|\phi_1, \dots, \phi_n|\!) \parr P(\!|\psi_1, \dots, \psi_n|\!)}{P(\!|\phi_1 \parr \psi_1, \dots, \phi_n \parr \psi_n|\!)}\ {\scriptstyle \parr \neq P \text{ prime}}$$

Figure 9: Structural rules for sequent calculi, and the corresponding rules in deep inference together with the atomic contraction and the generalized medial rule.

Proof. By Lemma 55, if $\vdash_{\mathsf{MGL}^\circ} \phi$, then by there is a proof of $[\![\phi]\!]$ in MGL°.

To prove the converse, let $\mathcal{D}$ be a proof of $G \neq \emptyset$ in GS. We define a proof $\pi_{\mathcal{D}}$ of ϕ by induction on the number n of rules in $\mathcal{D}$.

- We cannot have $n = 0$ since we are assuming $G \neq \emptyset$.
- If $n = 1$, then $G = a \parr a^\perp$ and $\pi_{\mathcal{D}} = \parr\,\dfrac{\mathsf{ax}\,\dfrac{}{\vdash a, a^\perp}}{\vdash a \parr a^\perp}$.
- If $n > 1$, then $\mathcal{D} = \mathsf{r}\,\dfrac{\mathcal{D}' \,\| \; H}{G}$ then by inductive hypothesis we have a proof $\pi_{\mathcal{D}'}$ of a formula ψ

  such that $[\![\psi]\!] = H$. If $\mathsf{r} \in \{\mathsf{s}_\parr, \mathsf{s}_\otimes, \mathsf{p}\!\downarrow\}$, then by Lemma 56 we have a derivation with cut as the one below on the left of a formula ϕ such that $[\![\phi]\!] = G$. We then conclude by Theorem 39.

  $$\mathsf{cut}\,\frac{\overset{\|\,\mathsf{IH}}{\psi} \qquad \overset{\|\,\text{Lemma 56}}{\vdash \psi^\perp, \phi}}{\vdash \phi} \quad \overset{\textit{Theorem } 39}{\rightsquigarrow^*} \quad \overset{\|\,\mathsf{MGL}^\circ}{\phi} \qquad \Bigg| \qquad \mathsf{deep}\,\frac{\parr\,\dfrac{\mathsf{ax}\,\dfrac{}{\vdash a, a^\perp}}{\vdash a \parr a^\perp} \qquad \overset{\pi_{\mathcal{D}'}\,\|\,\mathsf{IH}}{\psi}}{\dfrac{\vdash \zeta[a \parr a^\perp]}{\vdash \phi}\,=}$$

  Otherwise $\mathsf{r} = \mathsf{ai}\!\downarrow$, then it must have been applied deep inside a context $C[\square] = [\![\zeta[\square]]\!] \neq \square$ such that $C[\emptyset] = H = [\![\psi]\!]$. Therefore $\phi = \zeta[a \parr a^\perp]$. We conclude by applying Lemma 47 to the derivation above on the right. □

5 Classical Logic Beyond Cographs

We conclude this paper by providing an extension of MGL with standard *contraction* and *weakening* structural rules, showing that it provides a conservative extension of propositional classical logic. We then show decomposition results allowing us to factorize any proof into a linear proof (i.e., a proof in MGL) and a *resource management* proof (i.e., a derivation only using weakening and contraction rules).

Definition 58. We define the following sequent system:

$$\textbf{\textit{Classical Graphical Logic}}: \ \mathsf{GLK} = \mathsf{MGL} \cup \{\mathsf{w}, \mathsf{c}\} \tag{13}$$

For GLK we can prove the admissibility of the cut-rule via cut-elimination.

Theorem 59 (Cut-elimination)**.** *The rule* cut *is admissible in* GLK.

Proof. Consider the *cut-elimination steps* from Figure 3 and Figure 10 and the definition of weight from the proof of Theorem 39. A proof of weak normalization of the cut-elimination procedure can be given using the same measure used in the proof of Theorem 39 [a] by restraining the application of the cut-elimination steps only to top-most cut-rules in the derivation. □

We consider the deep inference rules in Figure 9, that is, rules which can be applied on subformulas in a sequent. Using the deep inference version of the structural rules (weakening and contraction) and the ***generalized medial*** rule proposed in [18] we can define a inference system where structural rules can pushed down in a derivation obtaining a decomposition result extending the one in [14, 16] for classical logic.

[a]For the sake of determining if a cut-formula is principal, in a contraction rule (c) we assume both occurrences of ϕ in the premise to be active and the occurrence of ϕ in the conclusion to be principal.

$$\mathsf{cut}\dfrac{\mathsf{w}\dfrac{\vdash \Gamma}{\vdash \Gamma, \phi} \quad \vdash \phi^\perp, \Delta}{\vdash \Gamma, \Delta} \rightsquigarrow \mathsf{w}\dfrac{\vdash \Gamma}{\vdash \Gamma, \Delta} \qquad \mathsf{cut}\dfrac{\mathsf{c}\dfrac{\vdash \Gamma, \phi, \phi}{\vdash \Gamma, \phi} \quad \vdash \phi^\perp, \Delta}{\vdash \Gamma, \Delta} \rightsquigarrow \mathsf{c}\dfrac{\mathsf{cut}\dfrac{\mathsf{cut}\dfrac{\vdash \Gamma, \phi, \phi \quad \vdash \phi^\perp, \Delta}{\vdash \Gamma, \Delta, \phi} \quad \vdash \phi^\perp, \Delta}{\vdash \Gamma, \Delta, \Delta}}{\vdash \Gamma, \Delta}$$

Figure 10: The cut-elimination steps for the structural rules.

Lemma 60. *The contraction rule* $\mathsf{c}\downarrow$ *is derivable using atomic contraction (*$\mathsf{ac}\downarrow$*) and medial rule (*m*).*

Proof. By induction on the contracted formula ϕ. If $\phi = a$ is an atom, then $\mathsf{c}\downarrow$ is an instance of $\mathsf{ac}\downarrow$. Otherwise, $\phi = \kappa(\!|\psi_1, \ldots, \psi_n|\!)$ and we conclude since we can apply inductive hypothesis to replace each application of $\mathsf{c}\downarrow$ with a derivation of the following form

$$\mathsf{c}\downarrow\dfrac{\kappa(\!|\psi_1, \ldots, \psi_n|\!) \parr \kappa(\!|\psi_1, \ldots, \psi_n|\!)}{\kappa\left(\!\left|\psi_1, \ldots, \psi_n\right|\!\right)} \quad \rightsquigarrow \quad \mathsf{m}\dfrac{\kappa(\!|\psi_1, \ldots, \psi_n|\!) \parr \kappa(\!|\psi_1, \ldots, \psi_n|\!)}{\kappa\left(\!\left|\begin{array}{c}\psi_1 \parr \psi_1 \\ \mathsf{IH}\,\Vert\,\{\mathsf{m},\mathsf{ac}\downarrow\} \\ \psi_1\end{array}, \ldots, \begin{array}{c}\psi_n \parr \psi_n \\ \mathsf{IH}\,\Vert\,\{\mathsf{m},\mathsf{ac}\downarrow\} \\ \psi_n\end{array}\right|\!\right)}\,.$$

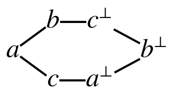
□

Theorem 61 (Decomposition)**.** *Let* Γ *be a sequent. If* $\vdash_{\mathsf{GLK}} \Gamma$*, then:*

1. *there is a sequent* Γ' *such that* $\vdash_{\mathsf{MGL}} \Gamma' \vdash_{\{\mathsf{w}\downarrow,\mathsf{c}\downarrow\}} \Gamma$
2. *there are sequent* Γ'*,* Δ'*, and* Δ *such that* $\vdash_{\mathsf{MGL}} \Gamma' \vdash_{\{\mathsf{m}\}} \Delta' \vdash_{\{\mathsf{ac}\downarrow\}} \Delta \vdash_{\{\mathsf{w}\downarrow\}} \Gamma$

Proof. The proof of Item 1 is immediate by applying rule permutations. For a reference, see [7]. Item 2 is consequence of the previous point since by Lemma 60 we can replace all instances of $\mathsf{c}\downarrow$-rules with derivations containing only m and $\mathsf{ac}\downarrow$, and conclude by applying rule permutations to move all ac-rules below the instances of m-rules, and $\mathsf{w}\downarrow$ to the bottom of a derivation. □

To conclude this section, we recall that classical graphical logic, is not the same logic of the ***boolean graphical logic*** (denoted GBL) defined in [18] (an inference systems on graphs by extending the semantics of read-once boolean relations from cographs to general graphs). In fact, even if both are conservative extensions of classical logic, the following graph from [5] which is expected to be provable in GBL, but is not provable in GS (and there is no formula ϕ provable in GLK such that $[\![\phi]\!]$ is the given graph).



6 Conclusion and Future Works

In this paper we have provided foundations for the design of proof systems operating on graph by defining *graphical connectives*, a class of logical operators generalizing the classical conjunction and disjunction, and whose semantics is solely defined by their interpretation as prime graphs.

We studied two sub-structural sequent calculi operating on formulas defined via graphical connectives (MGL and MGL°), proving that cut-elimination holds in these systems and that they are conservative extensions of the multiplicative linear logic and the multiplicative linear logic with mix respectively. For these calculi, we proved that they capture graph isomorphisms as provable logical equivalences[7]. We were able to prove that the class of graphs representing provable formulas in MGL° coincides with the class of non-empty graphs provable in the proof system GS from [4, 3]. As a consequence, the proofs of cut-elimination in MGL serves as simplified version of the proof of transitivity of implication in GS.

We concluded by providing a conservative extension of both classical propositional logic and MGL, and proving the existence of a decomposition result allowing us to have canonical forms for proofs in which all structural rules can be relegated at the bottom of a derivation.

[7]Note that the sequent calculus is only capable of checking if two graphs sharing the same set of vertices are isomorphic (problem with polynomial complexity), but not to find an correspondence between vertices of two graphs which is an isomorphism (a well-known **NP** problem)

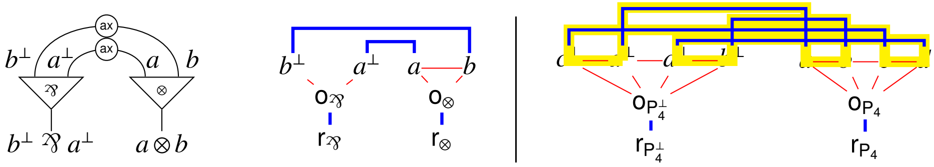


Figure 11: On the left: the same proof net in the original Girard's syntax and Retoré's one. On the right: an RB-proof net of $\kappa_{\mathsf{P}_4}(\!|a,b,c,d|\!) \multimap \kappa_{\mathsf{P}_4}(\!|a,b,c,d|\!)$ containing the chorded æ-cycle $a \cdot b \cdot b^\perp \cdot d^\perp \cdot d \cdot c \cdot c^\perp \cdot a^\perp$.

6.1 Future Works

Categorical Semantics. Unit-free *star-autonomous* and *IsoMix* categories [19, 20] provide categorical models of MLL and MLL° respectively. We conjecture that categorical models for MGL and MGL° can be defined by enriching such structures with additional n-ary monoidal products and natural transformation, reflecting the symmetries observed in the symmetry groups of prime graphs.

Digraphs, Games and Event Structures. In this work we have extended the correspondence between classical propositional and cographs from [21] to the case of general (undirected) graphs using graphical connectives. The same idea can be found in [2] for mixed graphs. A similar generalization of the correspondence between intuitionistic propositional formulas and *arenas* used in Hyland-Ong *game semantics* [41]. Arenas are directed graphs characterized by the absence of specific induced subgraphs. We foresee the possibility of defining conservative extensions of intuitionistic propositional logic beyond arenas, analogously as what done in this paper, where we provided conservative extensions of classical propositional logic beyond cographs. Such proof systems would provide new insights on the proof theory connected to concurrent games [1, 58, 65], and could be used to define automated tools operating on event structures [55].

Proof nets and automated proof search. We plan to design proof nets [29, 22, 30] for MGL and MGL°, as well as combinatorial proofs [40] for GLK. For this purpose, we envisage to extend Retoré's *handsome proof net* syntax, where proof nets are represented by two-colored graphs (see the left of Figure 11). In Retoré's syntax, the graph induced by the vertices corresponding to the inputs of a ⅋-gate (or a ⊗-gate) is similar to the corresponding prime graph ⅋ (resp. ⊗). Thus, gates for graphical connectives could be easily defined by extending this correspondence (see the proof net on the right of Figure 11). The standard correctness condition defined via *acyclicity* would fails for these proof nets, as shown in the right-hand side of Figure 11: the (correct) proof-net of the sequent $\mathsf{P}_4(\!|a,b,c,d|\!) \multimap \mathsf{P}_4(\!|a,b,c,d|\!)$ contains a cycle. We foresee the possibility of using results on the *primeval* decomposition of graphs [43, 38] to isolate those cycles witnessing unsoundness, as proposed in [52]. Such a result would open to the possibility of defining *combinatorial proofs* [40, 39] for GLK relying on the decomposition result (Theorem 61), and may provide a methodology to find machine-learning guided automated theorem provers for MGL and MGL° using the methods in [44].

References

[1] Samson Abramsky and P-A Mellies. Concurrent games and full completeness. In *Proceedings. 14th Symposium on Logic in Computer Science (Cat. No. PR00158)*, pages 431–442. IEEE, 1999.

[2] Matteo Acclavio, Ross Horne, Sjouke Mauw, and Lutz Straßburger. A Graphical Proof Theory of Logical Time. In Amy P. Felty, editor, *7th International Conference on Formal Structures for Computation and Deduction (FSCD 2022)*, volume 228 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 22:1–22:25, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[3] Matteo Acclavio, Ross Horne, and Lutz Straßburger. An Analytic Propositional Proof System On Graphs. This is an extended version of a paper published at LICS 2020 [AHS20]., December 2020.

[4] Matteo Acclavio, Ross Horne, and Lutz Straßburger. Logic beyond formulas: A proof system on graphs. In *Proceedings of the 35th Annual ACM/IEEE Symposium on Logic in Computer Science*, LICS '20, page 38–52, New York, NY, USA, 2020. Association for Computing Machinery.

[5] Matteo Acclavio, Ross Horne, and Lutz Straßburger. An Analytic Propositional Proof System on Graphs. *Logical Methods in Computer Science*, Volume 18, Issue 4, October 2022.

[6] Matteo Acclavio and Roberto Maieli. Generalized connectives for multiplicative linear logic. In Maribel Fernández and Anca Muscholl, editors, *28th EACSL Annual Conference on Computer Science Logic (CSL 2020)*, volume 152 of *LIPIcs*, pages 6:1–6:16, Dagstuhl, Germany, 2020. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[7] Matteo Acclavio and Lutz Straßburger. From syntactic proofs to combinatorial proofs. In Didier Galmiche, Stephan Schulz, and Roberto Sebastiani, editors, *Automated Reasoning - 9th International Joint Conference, IJCAR 2018, Held as Part of the Federated Logic Conference, FloC 2018, Oxford, UK, July 14-17, 2018, Proceedings*, volume 10900, pages 481–497. Springer, 2018.

[8] Andrea Aler Tubella and Alessio Guglielmi. Subatomic proof systems: Splittable systems. *ACM Trans. Comput. Logic*, 19(1), January 2018.

[9] Andrea Aler Tubella and Lutz Straßburger. Introduction to Deep Inference. Lecture, August 2019.

[10] Jean-Marc Andreoli. Logic programming with focusing proofs in linear logic. *Journal of Logic and Computation*, 2(3):297–347, 1992.

[11] Arnon Avron and Iddo Lev. Canonical propositional Gentzen-type systems. In Rajeev Goré, Alexander Leitsch, and Tobias Nipkow, editors, *Automated Reasoning*, pages 529–544, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.

[12] László Babai. Graph isomorphism in quasipolynomial time [extended abstract]. In *Proceedings of the Forty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '16, page 684–697, New York, NY, USA, 2016. Association for Computing Machinery.

[13] Patrick Blackburn, Maarten De Rijke, and Yde Venema. *Modal logic: graph. Darst*, volume 53. Cambridge University Press, 2001.

[14] Kai Brünnler. Locality for classical logic. *Notre Dame Journal of Formal Logic*, 47(4):557–580, 2006.

[15] Kai Brünnler and Lutz Straßburger. Modular sequent systems for modal logic. In Martin Giese and Arild Waaler, editors, *Automated Reasoning with Analytic Tableaux and Related Methods, TABLEAUX'09*, volume 5607 of *Lecture Notes in Computer Science*, pages 152–166. Springer, 2009.

[16] Paola Bruscoli and Lutz Straßburger. On the length of medial-switch-mix derivations. In Juliette Kennedy and Ruy J. G. B. de Queiroz, editors, *Logic, Language, Information, and Computation - 24th International Workshop, WoLLIC 2017, London, UK, July 18-21, 2017, Proceedings*, volume 10388 of *Lecture Notes in Computer Science*, pages 68–79. Springer, 2017.

[17] Cameron Calk. A graph theoretical extension of boolean logic. Bachelor's thesis, 2016.

[18] Cameron Calk, Anupam Das, and Tim Waring. Beyond formulas-as-cographs: an extension of boolean logic to arbitrary graphs, 2020.

[19] J.R.B. Cockett and R.A.G. Seely. Proof theory for full intuitionistic linear logic, bilinear logic, and mix categories. *Theory and Applications of Categories*, 3(5):85–131, 1997.

[20] J.R.B. Cockett and R.A.G. Seely. Weakly distributive categories. *J. of Pure and Applied Algebra*, 114:133–173, 1997.

[21] D.G. Corneil, H. Lerchs, and L.Stewart Burlingham. Complement reducible graphs. *Discrete Applied Mathematics*, 3(3):163–174, 1981.

[22] Vincent Danos and Laurent Regnier. The structure of multiplicatives. *Archive for Mathematical logic*, 28(3):181–203, 1989.

[23] Anupam Das. Complexity of evaluation and entailment in boolean graph logic. preprint, 2019.

[24] Anupam Das and Alex A. Rice. New minimal linear inferences in boolean logic independent of switch and medial. In Naoki Kobayashi, editor, *6th International Conference on Formal Structures for Computation and Deduction, FSCD 2021, July 17-24, 2021, Buenos Aires, Argentina (Virtual Conference)*, volume 195 of *LIPIcs*, pages 14:1–14:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.

[25] Pierre-Malo Deniélou and Nobuko Yoshida. Buffered communication analysis in distributed multiparty sessions. In Paul Gastin and François Laroussinie, editors, *CONCUR 2010 - Concurrency Theory*, pages 343–357, Berlin, Heidelberg, 2010. Springer.

[26] A. Ehrenfeucht, T. Harju, and G Rozenberg. *The Theory of 2-Structures A Framework for Decomposition and Transformation of Graphs*. World Scientific, 1999.

[27] Xiang Fu, Tevfik Bultan, and Jianwen Su. Analysis of interacting BPEL web services. In *Proceedings of the 13th international conference on World Wide Web*, pages 621–630. ACM, 2004.

[28] Tibor Gallai. Transitiv orientierbare Graphen. *Acta Mathematica Academiae Scientiarum Hungarica*, 18(1–2):25–66, 1967.

[29] Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.

[30] Jean-Yves Girard. Proof-nets : the parallel syntax for proof-theory. In Aldo Ursini and Paolo Agliano, editors, *Logic and Algebra*. Marcel Dekker, New York, 1996.

[31] Jean-Yves Girard. Light linear logic. *Information and Computation*, 143:175–204, 1998.

[32] Jean-Yves Girard. On the meaning of logical rules II: multiplicatives and additives. *NATO ASI Series F: Computer and Systems Sciences*, 175:183–212, 2000.

[33] Alessio Guglielmi. A system of interaction and structure. *ACM Transactions on Computational Logic*, 8(1):1–64, 2007.

[34] Alessio Guglielmi, Tom Gundersen, and Michel Parigot. A proof calculus which reduces syntactic bureaucracy. In Christopher Lynch, editor, *Proceedings of the 21st International Conference on Rewriting Techniques and Applications*, volume 6 of *LIPIcs*, pages 135–150, Dagstuhl, Germany, 2010. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[35] Alessio Guglielmi and Lutz Straßburger. Non-commutativity and MELL in the calculus of structures. In Laurent Fribourg, editor, *Computer Science Logic*, pages 54–68, Berlin, Heidelberg, 2001. Springer.

[36] Alessio Guglielmi and Lutz Straßburger. A non-commutative extension of MELL. In Matthias Baaz and Andrei Voronkov, editors, *Logic for Programming, Artificial Intelligence, and Reasoning*, pages 231–246, Berlin, Heidelberg, 2002. Springer.

[37] Michel Habib and Christophe Paul. A survey of the algorithmic aspects of modular decomposition. *Computer Science Review*, 4(1):41–59, 2010.

[38] Stefan Hougardy. *On the P4-structure of perfect graphs*. Citeseer, 1996.

[39] Dominic Hughes. Proofs Without Syntax. *Annals of Mathematics*, 164(3):1065–1076, 2006.

[40] Dominic Hughes. Towards Hilbert's 24^{th} problem: Combinatorial proof invariants: (preliminary version). *Electr. Notes Theor. Comput. Sci.*, 165:37–63, 2006.

[41] J. Martin E. Hyland and Chih-Hao Luke Ong. On full abstraction for PCF: I. Models, observables and the full abstraction problem, II. Dialogue games and innocent strategies, III. A fully abstract and universal game model. *Information and Computation*, 163:285–408, 2000.

[42] Lee O James, Ralph G Stanton, and Donald D Cowan. Graph decomposition for undirected graphs. In *Proceedings of the Third Southeastern Conference on Combinatorics, Graph Theory, and Computing (Florida Atlantic Univ., Boca Raton, Fla., 1972)*, pages 281–290, 1972.

[43] Beverly Jamison and Stephan Olariu. P-components and the homogeneous decomposition of graphs. *SIAM Journal on Discrete Mathematics*, 8(3):448–463, 1995.

[44] Konstantinos Kogkalidis, Michael Moortgat, and Richard Moot. Neural proof nets. In Raquel Fernández and Tal Linzen, editors, *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 26–40, Online, November 2020. Association for Computational Linguistics.

[45] Björn Lellmann and Elaine Pimentel. Modularisation of sequent calculi for normal and non-normal modalities. *ACM Trans. Comput. Logic*, 20(2), feb 2019.

[46] László Lovász and Michael D Plummer. *Matching theory*, volume 367. American Mathematical Soc., 2009.

[47] Saunders Mac Lane. *Categories for the Working Mathematician*. Number 5 in Graduate Texts in Mathematics. Springer, 1971.

[48] Roberto Maieli. Non decomposable connectives of linear logic. *Annals of Pure and Applied Logic*, 170(11):102709, 2019.

[49] Ross M. McConnell and Jeremy P. Spinrad. Linear-time modular decomposition and efficient transitive orientation of comparability graphs. In *Proceedings of the Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '94, pages 536–545, USA, 1994. Society for Industrial and Applied Mathematics.

[50] Dale Miller and Elaine Pimentel. A formal framework for specifying sequent calculus proof systems. *Theoretical Computer Science*, 474:98–116, 2013.

[51] Dale Miller and Alexis Saurin. From proofs to focused proofs: a modular proof of focalization in linear logic. In J. Duparc and T. A. Henzinger, editors, *CSL 2007: Computer Science Logic*, volume 4646 of *LNCS*, pages 405–419. Springer-Verlag, 2007.

[52] Lê Thành Dũng Nguyên and Thomas Seiller. Coherent interaction graphs: A non-deterministic geometry of interaction for mll. 2019.

[53] Lê Thành Dũng Nguyên and Lutz Straßburger. A System of Interaction and Structure III: The Complexity of BV and Pomset Logic. working paper or preprint, 2022.

[54] Lê Thành Dũng Nguyên and Lutz Straßburger. BV and Pomset Logic are not the same. In Florin Manea and Alex Simpson, editors, *30th EACSL Annual Conference on Computer Science Logic (CSL 2022)*, volume 216 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 3:1–3:17, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[55] Mogens Nielsen, Gordon Plotkin, and Glynn Winskel. Petri nets, event structures and domains, part i. *Theoretical Computer Science*, 13(1):85–108, 1981.

[56] Vaughan Pratt. Modeling concurrency with partial orders. *International journal of parallel programming*, 15:33–71, 1986.

[57] Christian Retoré. Pomset logic: The other approach to noncommutativity in logic. *Joachim Lambek: The Interplay of Mathematics, Logic, and Linguistics*, pages 299–345, 2021.

[58] Silvain Rideau and Glynn Winskel. Concurrent strategies. In *2011 IEEE 26th Annual Symposium on Logic in Computer Science*, pages 409–418. IEEE, 2011.

[59] R.A.G. Seely. Linear logic, *-autonomous categories and cofree coalgebras. *Contemporary Mathematics*, 92, 1989.

[60] Alwen Fernanto Tiu. A system of interaction and structure II: The need for deep inference. *Logical Methods in Computer Science*, 2(2):1–24, 2006.

[61] Anne Sjerp Troelstra and Helmut Schwichtenberg. *Basic Proof Theory*. Cambridge University Press, second edition, 2000.

[62] Jacobo Valdes, Robert E Tarjan, and Eugene L Lawler. The recognition of series parallel digraphs. In *Proceedings of the eleventh annual ACM symposium on Theory of computing*, pages 1–12. ACM, 1979.

[63] Gerco van Heerdt, Tobias Kappé, Jurriaan Rot, and Alexandra Silva. Learning pomset automata. In Stefan Kiefer and Christine Tasson, editors, *Foundations of Software Science and Computation Structures*, pages 510–530, Cham, 2021. Springer International Publishing.

[64] Timothy Waring. A graph theoretic extension of boolean logic. Master's thesis, 2019.

[65] Glynn Winskel, Silvain Rideau, Pierre Clairambault, and Simon Castellan. Games and strategies as event structures. *Logical Methods in Computer Science*, 13, 2017.

A Deep Inference and the Open Deduction Formalism

Open deduction [34] is a proof formalism based on deep inference [9]. It has originally been defined for formulas, but it is abstract enough such that it can equally well be used for graphs, as already done in [3].

Definition 62. An ***inference system*** S is a set of inference rules (as for example shown in Figure 1). A ***derivation*** $\mathcal{D}$ in S with premise G and conclusion H is denoted $\begin{array}{c} G \\ \mathcal{D}\|\mathsf{S} \\ H \end{array}$ and is defined inductively as follows:

- Every graph G is a (***trivial***) derivation with premise G and conclusion G (also denoted G).
- An instance of a rule $\mathsf{r}\dfrac{G}{H}$ in S is a derivation with premise G and conclusion H.
- If $\mathcal{D}_1$ is a derivation with premise G_1 and conclusion H_1, and $\mathcal{D}_2$ is a derivation with premise G_2 and conclusion H_2, and $H_1 = G_2$, then the composition of $\mathcal{D}_1$ and $\mathcal{D}_2$ is a derivation $\mathcal{D}_2 \,;\, \mathcal{D}_1$ denoted as below.

$$\begin{array}{c} G_1 \\ \mathcal{D}_1\|\mathsf{S} \\ H_1 \\ \hline G_2 \\ \mathcal{D}_2\|\mathsf{S} \\ H_2 \end{array} \quad\text{or}\quad \begin{array}{c} G_1 \\ \mathcal{D}_1\|\mathsf{S} \\ H_1 \\ \hline G_2 \\ \mathcal{D}_2\|\mathsf{S} \\ H_2 \end{array} \quad\text{or}\quad \begin{array}{c} G_1 \\ \mathcal{D}_1\|\mathsf{S} \\ H_1 \\ \hline G_2 \\ \mathcal{D}_2\|\mathsf{S} \\ H_2 \end{array} \quad\text{or}\quad \begin{array}{c} G_1 \\ \mathcal{D}_1\|\mathsf{S} \\ G_2 \\ \mathcal{D}_2\|\mathsf{S} \\ H_2 \end{array} \quad\text{or}\quad \begin{array}{c} G_1 \\ \mathcal{D}_1\|\mathsf{S} \\ H_1 \\ \mathcal{D}_2\|\mathsf{S} \\ H_2 \end{array}$$

  Note that even if the symmetry between G_2 and H_1 is not written, we always assume it is part of the derivation and explicitly given.
- If G is a graph with n vertices and $\mathcal{D}_1, \dots, \mathcal{D}_n$ are derivations with premise G_i and conclusion H_i for each $i \in \{1, \dots, n\}$, then $G(\!|\mathcal{D}_1, \dots, \mathcal{D}_n|\!)$ is a derivation with premise $G(\!|G_1, \dots, G_n|\!)$ and conclusion $G(\!|H_1, \dots, H_n|\!)$ denoted as below on the left.

$$G\left(\!\!\left|\begin{array}{c} G_1 \\ \mathcal{D}_1\|\mathsf{S} \\ H_1 \end{array}, \dots, \begin{array}{c} G_n \\ \mathcal{D}_n\|\mathsf{S} \\ H_n \end{array}\right|\!\!\right) \qquad \Bigg| \qquad \begin{array}{c} G_1 \\ \mathcal{D}_1\| \\ H_1 \end{array} \star \begin{array}{c} G_1 \\ \mathcal{D}_1\| \\ H_1 \end{array}$$

  If $G = \star \in \{\parr, \otimes\}$ we may write the derivations as above on the right.

Therefore, $C\left[\begin{array}{c} G \\ \mathcal{D}\|\mathsf{S} \\ H \end{array}\right] := \begin{array}{c} C[G] \\ C[\mathcal{D}]\|\mathsf{S} \\ C[H] \end{array}$ is well-defined for any context $C[\square]$ and any derivation $\begin{array}{c} G \\ \mathcal{D}\|\mathsf{S} \\ H \end{array}$.

A ***proof*** in S is a derivation in S whose premise is $\varnothing$.
A graph G is ***provable*** in S (denoted $\vdash_{\mathsf{S}} G$) iff there is a proof in S with conclusion G.

A.1 Equivalent Definitions of GS

We here show that the formulation of the system GS provided in this paper is equivalent to one provided in [4, 5]. In particular, in the previous these papers the rule $\mathsf{s}_\otimes$ was not included in the system. However, as shown in [5] this rule plays a crucial role in the proof that GS is a conservative extension of MLL° and in [2] it is shown that this rule cannot be admissible in the proof systems operating on mixed graphs. Moreover, we here give a weaker side condition on the p-rule with respect to the rules below:

$\mathsf{p}{\downarrow}$ in [5]	$\mathsf{p}{\downarrow}$ in [4]
$\mathsf{p_1}{\downarrow}\dfrac{(M_1 \parr N_1)\otimes\cdots\otimes(M_n \parr N_n)}{P^\perp(\!|M_1,\dots,M_n|\!) \parr P(\!|N_1,\dots,N_n|\!)}\star$	$\mathsf{p_2}{\downarrow}\dfrac{(M_1 \parr N_1)\otimes\cdots\otimes(M_n \parr N_n)}{P^\perp(\!|M_1,\dots,M_n|\!) \parr P(\!|N_1,\dots,N_n|\!)}\dagger$
$\star :=\ P \notin \{\parr, \otimes\}$ prime $M_i \neq \varnothing$ for all $i \in \{1, \dots, n\}$	$\dagger :=\ P \notin \{\parr, \otimes\}$ prime $M_i \parr N_i \neq \varnothing$ for all $i \in \{1, \dots, n\}$

(14)

In order to prove the equivalence between our system and the ones in [4, 5] we recall the following lemma allowing us to prove that in GS we can derive any graph of the shape $G \multimap G$.

Lemma 63. *Let $M_1, \ldots, M_n, N_1, \ldots, N_n$ and G be graphs such that $|V_G| = n$. Then there is a derivation*

$$\begin{array}{c} (M_1 \parr N_1) \otimes \cdots \otimes (M_n \parr N_n) \\ \Vert\, {\scriptstyle \{\mathsf{s}_\otimes, \mathsf{p}\downarrow\}} \\ G^\perp(\!|M_1, \ldots, M_n|\!) \parr G(\!|N_1, \ldots, N_n|\!) \end{array}$$

Proof. By induction on the modular decomposition of G. □

Thanks to this lemma, we can therefore prove the admissibility of the weaker

Proposition 64. *The following version of* $\mathsf{p}\downarrow$ *with weaker side conditions is admissible in* GS

$$\mathsf{p}_1\!\downarrow \frac{(M_1 \parr N_1) \otimes \cdots \otimes (M_n \parr N_n)}{P^\perp(\!|M_1, \ldots, M_n|\!) \parr P(\!|N_1, \ldots, N_n|\!)} \ \textit{P prime, } M_i \neq \varnothing \textit{ for all } i \in \{1, \ldots, n\}$$

Proof. Note that we may have $N_i = \varnothing$ for some $i \in \{1, \ldots, n\}$. Thus, if $N_i \neq \varnothing$ for all $i \in \{1, \ldots, n\}$, then $\mathsf{p}_1\!\downarrow$ is an occurrence of $\mathsf{p}\downarrow$. Otherwise, w.l.o.g., $N_1 = \varnothing$, thus we have a derivation

$$\begin{array}{c} M_1 \otimes \begin{array}{c} (M_2 \parr N_2) \otimes \cdots \otimes (M_n \parr N_n) \\ \Vert\, {\scriptstyle \text{Lemma 63}} \\ H^\perp(\!|M_2, \ldots, M_n|\!) \parr H(\!|N_2, \ldots, N_n|\!) \end{array} \\ \hdashline \mathsf{s}_\otimes \dfrac{M_1 \otimes P^\perp(\!|\varnothing, M_2, \ldots, M_n|\!)}{P^\perp(\!|M_1, M_2, \ldots, M_n|\!)} \parr P(\!|\varnothing, N_2, \ldots, N_n|\!) \end{array}$$

□

Theorem 65. *Let G be a graph. Then*

$$\vdash_{\mathsf{GS}} G \Leftrightarrow \vdash_{\{\mathsf{ai}\downarrow, \mathsf{s}_\parr, \mathsf{s}_\otimes, \mathsf{p}_1\downarrow\}} G \Leftrightarrow \vdash_{\{\mathsf{ai}\downarrow, \mathsf{s}_\parr, \mathsf{p}_1\downarrow\}} G \Leftrightarrow \vdash_{\{\mathsf{ai}\downarrow, \mathsf{s}_\parr, \mathsf{p}_2\downarrow\}} G$$

Proof. The first equivalence follows from Proposition 64. The other has been proved in [5]. □

B On Rules Introducing a Connective at a Time

A rule introducing only one connective (different from $\parr$ and $\otimes$) at a time inevitably leads to the same problem observed in the literature of *generalized multiplicative connectives* [22, 32, 48, 6], where *initial coherence* (i.e. the possibility of having only atomic axioms in a cut-free system, [11]) is ruled out because of the so-called *packaging problem*.

However, in this appendix we discuss the results about the system extending multiplicative linear logic with the rule $\mathsf{s}\text{-}\kappa$, that is, the system.

$$\mathsf{MLL}^{\mathsf{s}\text{-}\kappa} := \{\mathsf{ax}, \parr, \otimes, \mathsf{mix}, \mathsf{s}\text{-}\kappa_P \mid P \in \mathcal{P}\} \quad \text{where} \quad \mathsf{s}\text{-}\kappa_P \frac{\vdash \Gamma_1, \phi_1 \qquad \cdots \qquad \vdash \Gamma_n, \phi_n}{\vdash \Gamma_1, \ldots, \Gamma_n, \kappa_P(\!|\phi_1, \ldots, \phi_n|\!)}$$

We first observe that in the system does not satisfy anymore initial coherence; e.g., the formula $\kappa_{\mathsf{P}_4}(\!|a, b, c, d|\!) \multimap \kappa_{\mathsf{P}_4}(\!|a, b, c, d|\!)$ is not provable anymore. However, the system still satisfies cut-elimination. The proof cut-elimination is straightforward by considering the following additional cut-elimination steps.

$$\mathsf{cut}\frac{\mathsf{s}\text{-}\kappa\dfrac{\vdash \Gamma_1, \phi_1 \quad \cdots \quad \vdash \Gamma_n, \phi_n}{\vdash \Gamma_1, \ldots, \Gamma_n, \kappa_P(\!|\phi_1, \ldots, \phi_n|\!)} \quad \mathsf{s}\text{-}\kappa\dfrac{\vdash \Delta_1, \phi_1^\perp \quad \cdots \quad \vdash \Delta_n, \phi_n^\perp}{\vdash \Delta_1, \ldots, \Delta_n, \kappa_{P^\perp}(\!|\phi_1^\perp, \ldots, \phi_n^\perp|\!)}}{\vdash \Gamma_1, \ldots, \Gamma_n, \Delta_1, \ldots, \Delta_n} \rightsquigarrow \mathsf{mix}\frac{\mathsf{cut}\dfrac{\vdash \Gamma_1, \phi_1 \quad \vdash \Delta_1, \phi_1^\perp}{\vdash \Gamma_1, \Delta_1} \quad \cdots \quad \mathsf{cut}\dfrac{\vdash \Gamma_n, \phi_n^\perp \quad \vdash \Delta_n, \phi_n^\perp}{\vdash \Gamma_n, \Delta_n}}{\vdash \Gamma_1, \ldots, \Gamma_n, \Delta_1, \ldots, \Delta_n}$$

$$\mathsf{cut}\frac{\mathsf{d}\text{-}\kappa\dfrac{\vdash \Gamma_1, \phi_1, \psi_1 \quad \cdots \quad \vdash \Gamma_n, \phi_n, \psi_n}{\vdash \Gamma_1, \ldots, \Gamma_n, \kappa_{P^\perp}(\!|\psi_1, \ldots, \psi_n|\!), \kappa_P(\!|\phi_1, \ldots, \phi_n|\!)} \quad \mathsf{s}\text{-}\kappa\dfrac{\vdash \Delta_1, \phi_1^\perp \quad \cdots \quad \vdash \Delta_n, \phi_n^\perp}{\vdash \Delta_1, \ldots, \Delta_n, \kappa_{P^\perp}(\!|\phi_1^\perp, \ldots, \phi_n^\perp|\!)}}{\vdash \Gamma_1, \ldots, \Gamma_n, \Delta_1, \ldots, \Delta_n, \kappa_{P^\perp}(\!|\psi_1, \ldots, \psi_n|\!)} \rightsquigarrow \mathsf{s}\text{-}\kappa\frac{\mathsf{cut}\dfrac{\vdash \Gamma_1, \phi_1, \psi_1 \quad \vdash \Delta_1, \phi_1^\perp}{\vdash \Gamma_1, \Delta_1, \psi_1} \quad \cdots \quad \mathsf{cut}\dfrac{\vdash \Gamma_n, \phi_n^\perp, \psi_n \quad \vdash \Delta_n, \phi_n^\perp}{\vdash \Gamma_n, \Delta_n, \psi_n}}{\vdash \Gamma_1, \ldots, \Gamma_n, \Delta_1, \ldots, \Delta_n, \kappa^\perp(\!|\psi_1, \ldots, \psi_n|\!)}$$

Note that $\mathsf{s}\text{-}\kappa$ is derivable in MGL°.

Lemma 66. *The rule* $\mathsf{s}\text{-}\kappa$ *is derivable in* MGL°.

Proof. If $\kappa = \parr$, then $\mathsf{s}\text{-}\kappa$ is derivable using $\parr$ and mix. If $\kappa = \otimes$, then $\mathsf{s}\text{-}\kappa = \otimes$. Otherwise, we conclude by induction on the arity of κ since we have a derivation

$$\dfrac{\vdash \Gamma_1, \phi_1 \qquad \dfrac{\overset{\mathcal{D}' \,\Vert\, \mathsf{IH}}{\vdash \Gamma_2, \ldots, \Gamma_n, \zeta(\!|\phi_1, \ldots, \phi_n|\!)}}{\vdash \Gamma_2, \ldots, \Gamma_n, \kappa(\!|\circ, \phi_2, \ldots, \phi_n|\!)}\,\mathsf{unitor}_\kappa}{\vdash \Gamma_1, \ldots, \Gamma_n, \kappa(\!|\phi_1, \ldots, \phi_n|\!)}\,\mathsf{wd}_\otimes$$

where $\mathcal{D}'$ contains instances of $\mathsf{s}\text{-}\kappa$ introducing connectives whose arities are strictly smaller then the arity of κ. □