UNIVERSITY OF LODZ
DEPARTMENT OF LOGIC

BULLETIN
OF THE SECTION OF LOGIC

VOLUME 54, NUMBER 4

ŁÓDŹ, DECEMBER 2025

COPE
Member since 2018
JM13707

UNIVERSITY OF LODZ
DEPARTMENT OF LOGIC

BULLETIN
OF THE SECTION OF LOGIC

VOLUME 54, NUMBER 4

ŁÓDŹ, DECEMBER 2025

C O P E
Member since 2018
JM13707

Special issue
Non-Classial Logics. Theory and Applications (Part II)

Special issue editor
Michał Zawidzki

Layout
Michał Zawidzki

Initiating Editor
Katarzyna Smyczek

Printed directly from camera-ready materials provided
to the Lodz University Press

Published by Lodz University Press

First edition. W.11658.25.0.C

Printing sheets 11.25

Lodz University Press
90-237 Łódź, 34A Jana Matejki St.
www.wydawnictwo.uni.lodz.pl
e-mail: ksiegarnia@uni.lodz.pl
+48 42 635 55 77

The **Bulletin of the Section of Logic** (*BSL*) is a quarterly peer-reviewed journal published with the support from the University of Lodz. Its aim is to act as a forum for a wide and timely dissemination of new and significant results in logic through rapid publication of relevant research papers. *BSL* publishes contributions on topics dealing directly with logical calculi, their methodology, and algebraic interpretation.

Papers may be submitted through the *BSL* online editorial platform at https://czasopisma.uni.lodz.pl/bulletin. While preparing the manuscripts for publication please consult the Submission Guidelines.

* * *

Editorial Office:
Department of Logic, University of Lodz
ul. Lindleya 3/5, 90-131 Łódź, Poland
e-mail: bulletin@uni.lodz.pl

Homepage:
https://czasopisma.uni.lodz.pl/bulletin

TABLE OF CONTENTS

Bulletin of the **S**ection of **L**ogic
Volume 54/4 (2025), pp. 471–474
https://doi.org/10.18778/0138-0680.2025.20

PREFACE: NON-CLASSICAL LOGICS. THEORY AND APPLICATIONS (PART II)

The articles in the present and forthcoming issues are revised and extended versions of papers presented at the conference *Non-Classical Logics. Theory and Applications*, held in Łódź on 4–8 September 2024.[1]

Non-Classical Logics. Theory and Applications (NCL) is an international conference series devoted to novel results and survey work in broadly understood non-classical logics and their applications. The first two editions took place in Łódź, Poland, in 2008 and 2009. Subsequently, the conference was held alternately in Toruń (2010, 2012, 2015, 2018) and Łódź (2011, 2013, 2016, 2022). The tenth edition, organised by the University of Lodz in 2022, was the first to publish its proceedings in *Electronic Proceedings in Theoretical Computer Science*. This practice was continued in the most recent, eleventh edition, with all accepted long papers again appearing in an EPTCS volume. The 2024 edition was supported by the European Research Council as part of the project *Coming to Terms: Proof Theory for Definite Descriptions and Other Terms* (ExtenDD), and featured four

---

[1] Due to the high number of accepted post-conference submissions, the editors decided to divide them into two sets that have been published in two separate issues.

invited talks, eighteen contributed talks, and eighteen short presentations accepted through a light reviewing process.

Given that non-classical logics form a broad and diverse area of research within logic, the contributions collected in this issue address a wide range of topics. They include, among others, a non-standard definition of identity, semi-substructural logic, Lambek calculus, or multi-dimensional modal logics.

The paper "Qualified Definiteness" by Bartosz Więckowski revisits Russell's analysis of definite descriptions and targets a well-known tension: natural language permits "loose" uses of "the F" even when strict uniqueness fails. Rather than abandoning Russell's framework or shifting to a purely model-theoretic treatment of incomplete descriptions, the author refines the uniqueness clause itself. The key move is to replace primitive identity in Russell's analysis with a defined notion of qualified identity—identity relative to a specified set of predicates Q. This yields a graded notion of definiteness: maximal (strict) definiteness when Q includes all predicates, and restricted (loose) definiteness when Q is a proper subset. The resulting system formally distinguishes between genuine uniqueness and indiscernibility relative to selected respects.

The main contribution is a proof-theoretic treatment of qualified definiteness within an intuitionistically acceptable framework. Building on a bipredicational language with a primitive notion of predication failure, the paper develops natural deduction systems, establishes normalisation and subformula (more generally, subexpression) properties, and provides a proof-theoretic semantics via canonical derivations. The technical novelty lies in internalising identity conditions through introduction/elimination rules that mirror the definitional structure of qualified identity, avoiding second-order quantification while preserving fine-grained control over discernibility. The framework is then applied systematically to complete, incomplete, generic, nested, predicative, and possessive definite descriptions.

Cheng-Syuan Wan's article "Semi-Substructural Logics à la Lambek with Symmetry" addresses a structural mismatch in the proof theory of skew monoidal closed categories. Earlier sequent calculi (with stoups, à la Girard) successfully captured left skew monoidal closed categories, but did not extend smoothly to right skew or bi-closed variants, especially in the presence of symmetry. Wan identifies the source of the difficulty in the implicit left-associative structure of antecedents and proposes more flexible syntactic frameworks: a tree-based sequent calculus (in the style of non-associative Lambek calculus) and an equivalent axiomatic calculus with single-formula antecedents.

The principal contributions are twofold. First, the paper proves equivalence between the stoup-based and tree-based systems and establishes cut elimination. Second, it provides soundness and completeness results for the axiomatic calculi with respect to ternary relational semantics, and proves a correspondence theorem linking frame conditions to structural laws in the categorical setting. The technically interesting move is the shift from flat antecedents to tree-structured ones, which makes the skew (non-symmetric, non-associative) behaviour explicit and enables a uniform treatment of left, right, and bi-closed skew structures—including symmetric extensions. This yields an algebraically transparent account of how categorical coherence corresponds to relational constraints.

The article "Complexity of Nonassociative Lambek Calculus with Classical and Intuitionistic Logic" by Paweł Płaczek investigates the computational complexity of consequence relations for extensions of the non-associative Lambek calculus (NL). While pure NL has a polynomial-time decidable finitary consequence relation, adding unrestricted additive connectives (as in full NL) leads to undecidability. Interestingly, the distributive version restores decidability at exponential time. The author examines what happens when NL is combined not merely with distributive lattice structure, but with full Boolean (BFNL) or Heyting (HFNL) algebraic structure.

The main result is that both BFNL and HFNL retain decidability with an ExpTime upper bound (in the unital case), despite incorporating classical or intuitionistic logic into a non-associative, non-commutative setting. The proof strategy is algebraic: the author develops machinery for partial residuated Boolean and Heyting algebras, analyses embeddability conditions, and uses filter-extension techniques (adapted to partial structures) to control model construction. A key technical insight is that, in the Boolean case, working with the full power set rather than families of upsets simplifies the treatment of negation while preserving the necessary structural properties. The complexity bounds are thus obtained by careful algebraic model analysis rather than purely syntactic proof-theoretic methods.

Finally, Yuki Nishimura's paper "Agent-Knowledge Logic for Alternative Epistemic Logic" introduces *agent-knowledge logic*, a two-dimensional modal-hybrid system designed as an alternative to standard epistemic logic. Inspired by Facebook logic and the Logic of Hide and Seek Game, the framework separates the dimensions of agents and epistemic alternatives. It incorporates two modal operators (over agents and knowledge), two kinds of propositional variables (agent-dependent and agent-independent), and two kinds of nominals, enabling explicit reference to particular agents and epistemic states.

The first main contribution is a formal embedding of standard epistemic logic into agent-knowledge logic, demonstrating that the new system properly generalizes the classical one. The second is the construction of a tableau calculus with a termination property, establishing decidability via finite proof search. Conceptually, the innovative aspect lies in decoupling agent-dependent and agent-independent propositions while preserving hybrid reference mechanisms. This yields expressive power beyond standard epistemic logic (e.g., formalising "one of an agent's friends knows p") while maintaining a well-behaved proof-theoretic framework.

Michał Zawidzki
University of Lodz
Department of Logic
Lindleya 3/5
91-131 Łódź, Poland
e-mail: michal.zawidzki@filhist.uni.lodz.pl

Bulletin of the **S**ection of **L**ogic
Volume 54/4 (2025), pp. 475–518
https://doi.org/10.18778/0138-0680.2025.16

C O P E
Member since 2018
JM13707

Bartosz Więckowski

QUALIFIED DEFINITENESS

Abstract

According to Russell, the definite article ‘the’ in a definite description ‘the F’ is used strictly in case there is a unique F and it is used loosely in case there is more than one F. Russell’s analysis of constructions of the form ‘the F is G’ is concerned only with the strict use. We modify this analysis so as to allow also for the loose use. This is achieved essentially by replacing the usual undefined notion of identity in Russell’s uniqueness clause with the defined notion of qualified identity (i.e., ‘a is the same as b in all $\mathcal{Q}$-respects’, where $\mathcal{Q}$ is a subset of the set of predicate constants $\mathcal{P}$) proposed in earlier work. This modification gives us qualified notions of uniqueness and definiteness. A qualified definiteness statement ‘the $\mathcal{Q}$-unique F is G’ is strict in case $\mathcal{Q} = \mathcal{P}$ and loose in case $\mathcal{Q}$ is a proper subset of $\mathcal{P}$. The account is made formally precise in terms of proof theory and proof-theoretic semantics. The framework is intended to be acceptable from a foundational intuitionistic point of view. It is applied to

Presented by: Michał Zawidzki
Received: December 16, 2024, **Received in revised form**: July 11, 2025,
Accepted: October 28, 2025, **Published online**: March 13, 2026

natural language constructions with complete, incomplete, and generic definite descriptions. Also constructions with nested and with predicatively used definite descriptions are considered as well as constructions involving possessives. This work incorporates and extends my *NCL'24*-paper 'Incomplete descriptions and qualified definiteness'.

Keywords: definite descriptions (generic, incomplete, nested, Haddock, predicative), identity, intuitionistic logic, proof-theoretic semantics, uniqueness.

2020 Mathematical Subject Classification: 03A05, 03B65, 03F03.

1. Introduction

Sometimes we use the definite description 'the F' in cases in which there is a unique F. According to Russell, the definite article 'the' is used strictly in such cases.[1] For example, speaking about Leo XIV, we use 'the pope' in (1) in this way.

(1) The pope is bald.

Sometimes, as Russell notes, we use 'the F' also in cases, in which there is more than one F. For example, 'the bishop' in (2) is used in this loose way (as would be 'the pope' during a schism).

(2) The pope blesses the bishop.

According to Russell, such loose uses of 'the F' should be avoided in favour of the indefinite description 'an F'. It seems, though, that Russell's advice is not entirely adequate. For example, in case multiple Fs are present and one intends to say something about a particular F (e.g., [20]).

In this paper, we propose a formal account of both uses of 'the F' in terms of qualified definiteness. On a Russellian analysis, a construction of

---

[1] Russell ([16]: 481): "Now *the*, when it is strictly used, involves uniqueness; we do, it is true, speak of "*the* son of So-and-so" even when So-and-so has several sons, but it would be more correct to say "*a* son of So-and-so". Thus for our purposes we take *the* as involving uniqueness."

the form 'the F is G' is explained in terms of an existence, a uniqueness, and a predication clause:

(E) There is at least one F.

(U) There is at most one F.

(P) Every F is G.

We modify this analysis mainly by replacing the usual undefined notion of identity in the definition of uniqueness with the defined notion of *qualified identity* proposed in [24], i.e., 'a is the same as b in all $\mathcal{Q}$-respects', where $\mathcal{Q}$ is a subset of the set of predicate constants $\mathcal{P}$. The notion of *qualified uniqueness* that results from this replacement says:

(QU) For every x and y, if they are F, then they are identical with respect to every predicate in $\mathcal{Q}$.

Finally, a statement of *qualified definiteness* says, combining the three Russellian components:

(QD) The $\mathcal{Q}$-unique x which is F is G.

Qualified definiteness, unlike standard definiteness, allows for fine-tuning. Let $\mathcal{Q}'$ be a proper subset of $\mathcal{P}$ (i.e., $\mathcal{Q}' \subset \mathcal{P}$). If $\mathcal{Q} = \mathcal{P}$ in (QD), then we get the reading 'the *only* x which is F is G'. We may use this reading only in case there is a single x that is F. This is definiteness proper. If, on the other hand, we put $\mathcal{Q} = \mathcal{Q}'$, then we get: 'the x which is F is G'. We may use this reading only in case there are at least two things which are F that are indiscernible with respect to $\mathcal{Q}'$, but discernible with respect to $\mathcal{P} \setminus \mathcal{Q}'$. This is restricted definiteness. What is subject to restriction, on this account, is thus the set of $\mathcal{Q}$-respects (rather than, e.g., a domain of quantifiers [18]).

Below, we provide the details of this proposal. It will differ from competing semantic analyses of incomplete descriptions also in that it will be couched in a framework of proof-theoretic semantics (see [17] for an overview) rather than in some version of model-theoretic semantics. (For an overview of the literature on incomplete descriptions see, e.g., [1, ch. 9], [12, sect. 5.3]. An elaborate model-theoretic account is [2].)

Sect. 2 defines the formal language. Sect. 3 recapitulates the relevant fragment of the intuitionistic bipredicational natural deduction systems defined in [24] and combines it with the rules for definiteness proposed in [3, 4] into proof systems for qualified definiteness, establishing normalization and the subexpression (incl. subformula) property for them. Sect. 4 defines an intuitionistic proof-theoretic semantics for qualified definiteness in terms of canonical derivations, and Sect. 5 applies this semantics to complete, incomplete, and generic definite descriptions in the manner suggested above. In Sect. 6 this account of definiteness is generalized so as to allow also for the analysis of natural language constructions with nested and with predicative uses of definite descriptions. Also constructions involving possessives are considered. This work incorporates [25] and extends that paper essentially with the generalizations described in Sect. 6.

2. The language

We extend the bipredicational language $\mathcal{L}$ motivated and defined in [24] with contextually defined operators for qualified definiteness and call the extended language $\mathcal{L}\iota$.

$\mathcal{L}$ is a first-order language. It is bipredicational, since it allows for both predication and predication failure. We first recapitulate those parts of its definition which are relevant for present purposes.

Definition 2.1. $\mathcal{C}$ is the set of individual (or nominal) constants (form: α_i) and $\mathcal{P}$ is the set of n-ary predicate constants (form: φ^n_i) of $\mathcal{L}$. Moreover, Atm is the set of atomic sentences (form: $\varphi^n\alpha_1...\alpha_n$) of $\mathcal{L}$. $Atm(\alpha) =_{def} \{A \in Atm : A \text{ contains at least one occurrence of } \alpha \in \mathcal{C}\}$ and $Atm(\varphi^n) =_{def} \{A \in Atm : A \text{ contains an occurrence of } \varphi^n \in \mathcal{P}\}$. A nominal term o_i is either a nominal constant or a nominal variable x_i. Atomic formulae have the form $\varphi^n o_1...o_n$ and are used for predication. Negative predications (or predication failures) take the form $-\varphi^n o_1...o_n$ (reading: ‘the ascriptive combination of φ^n with $o_1, ..., o_n$ fails’).

Definition 2.2. Defined symbols of $\mathcal{L}$:

1. $\neg A =_{def} A \supset \bot$ (negation)
2. $A \leftrightarrow B =_{def} (A \supset B)\&(B \supset A)$ (equivalence)
3. Let φ^n be an n-ary predicate constant.

$$\begin{aligned}
&P^n_{\varphi^n}(o_1, o_2) =_{def}\\
&\quad \forall x_1 ... \forall x_{n-1} \forall x_n\ ((\varphi^n o_1 x_2 ... x_n \leftrightarrow \varphi^n o_2 x_2 ... x_n)\\
&\quad \& \ (\varphi^n x_1 o_1 ... x_n \leftrightarrow \varphi^n x_1 o_2 ... x_n)\\
&\quad \& \ ... \ \& \ (\varphi^n x_1 ... x_{n-1} o_1 \leftrightarrow \varphi^n x_1 ... x_{n-1} o_2))
\end{aligned}$$

$$\begin{aligned}
&N^n_{\varphi^n}(o_1, o_2) =_{def}\\
&\quad \forall x_1 ... \forall x_{n-1} \forall x_n\ ((-\varphi^n o_1 x_2 ... x_n \leftrightarrow -\varphi^n o_2 x_2 ... x_n)\\
&\quad \& \ (-\varphi^n x_1 o_1 ... x_n \leftrightarrow -\varphi^n x_1 o_2 ... x_n)\\
&\quad \& \ ... \ \& \ (-\varphi^n x_1 ... x_{n-1} o_1 \leftrightarrow -\varphi^n x_1 ... x_{n-1} o_2))
\end{aligned}$$

Let $\varphi^{k_1}_1, ..., \varphi^{k_m}_m$ be all the predicate constants in $\mathcal{Q}$, where φ_i is k_i-ary and $\mathcal{Q} \subseteq \mathcal{P}$.

Positive qualified identity:

$$o_1 \overset{+}{=}_{\mathcal{Q}} o_2 =_{def} P^{k_1}_{\varphi_1}(o_1, o_2) \ \& \ ... \ \& \ P^{k_m}_{\varphi_m}(o_1, o_2)$$

(‘o_1 is the same as o_2 in all $\mathcal{Q}$-respects’)

Negative qualified identity:

$$o_1 \overset{-}{=}_{\mathcal{Q}} o_2 =_{def} N^{k_1}_{\varphi_1}(o_1, o_2) \ \& \ ... \ \& \ N^{k_m}_{\varphi_m}(o_1, o_2)$$

(‘o_1 is the same as o_2 in no $\mathcal{Q}$-respect’)

Remark 2.3. Note that, in contrast to $\neg$, the operator for predication failure $-$ is primitive. Moreover, unlike the former, it is sensitive to the internal structure of the formula to which it is prefixed. See [24] for the motivation of $-$.

Remark 2.4. Being a defined notion, qualified identity differs not only from standard identity, but also from the notion of relative identity introduced by Geach in [5]. For a more detailed comparison see [24].

Remark 2.5. If the stroke for predication failure were not present in the language, $\mathcal{L}$ would be the language of plain first-order logic without standard identity. Importantly, the definition of qualified identity does not appeal to second-order quantification. In this respect our notion differs from the notion of identity considered, for example, by Read in [14]: $a = b =_{def} \forall F(Fa \leftrightarrow Fb)$.

$\mathcal{L}\iota$ extends $\mathcal{L}$ with operators for qualified definiteness by adapting the definitions from [3, 4].

Definition 2.6. We write $\varphi(x)$, suppressing the arity of φ, for atomic formulae $\varphi^n o_1 ... o_n$ containing (possibly multiple occurrences of) x. Let $\mathcal{Q} \subseteq \mathcal{P}$.

1. *Positive qualified definiteness*:

   $$\psi(\iota_{\mathcal{Q}} x \varphi(x)) =_{def} \exists x \varphi(x) \mathbin{\&} \underbrace{\forall u \forall v((\varphi(u) \mathbin{\&} \varphi(v)) \supset u \overset{+}{=}_{\mathcal{Q}} v)}_{Positive\ qualified\ uniqueness}$$
   $$\mathbin{\&} \forall w(\varphi(w) \supset \psi(w))$$

   (‘the $\mathcal{Q}$-unique x which is φ is ψ’; simpler: ‘the $\mathcal{Q}$-unique φ is ψ’)

2. *Negative qualified definiteness*:

   $$\psi(\iota_{\mathcal{Q}} x - \varphi(x)) =_{def} \exists x - \varphi(x) \mathbin{\&} \underbrace{\forall u \forall v((-\varphi(u) \mathbin{\&} - \varphi(v)) \supset u \overset{-}{=}_{\mathcal{Q}} v)}_{Negative\ qualified\ uniqueness}$$
   $$\mathbin{\&} \forall w(-\varphi(w) \supset \psi(w))$$

   (‘the $\mathcal{Q}$-unique x which fails to be φ is ψ’; simpler: ‘the $\mathcal{Q}$-unique $-\varphi$ is ψ’)

Remark 2.7. The definition of positive qualified definiteness differs from the definition of definiteness proposed in [3, 4] in that it does not make use of the familiar primitive notion of identity in the uniqueness part. In this respect, it significantly departs also from the tradition.

Remark 2.8. The reading of the positive qualified uniqueness formula is ‘there is at most one $\mathcal{Q}$-qualified φ’, that of the negative qualified uniqueness formula is ‘there is at most one $\mathcal{Q}$-qualified $-\varphi$’.

Qualified definiteness allows for degrees.

Definition 2.9. Let $\mathcal{Q}' \subset \mathcal{P}$. Qualified definiteness has (i) the highest degree of definiteness, in case $\mathcal{Q} = \mathcal{P}$, and (ii) a lower degree, in case $\mathcal{Q} = \mathcal{Q}'$. Given $\mathcal{Q}' \subset \mathcal{P}$, we can make the following distinction:

1. *Maximal definiteness*:
    (a) $\psi(\iota_{\mathcal{P}} x \varphi(x))$: 'the only x which is φ is ψ';
    (b) $\psi(\iota_{\mathcal{P}} x - \varphi(x))$: 'the only x which fails to be φ is ψ'.
2. *Restricted definiteness*:
    (a) $\psi(\iota_{\mathcal{Q}'} x \varphi(x))$: 'the x which is φ is ψ';
    (b) $\psi(\iota_{\mathcal{Q}'} x - \varphi(x))$: 'the x which fails to be φ is ψ'.

A loosely used definite description 'the F' is, thus, construed as a restriction of a strictly used 'the F' (i.e., the maximally definite description 'the only F').

Definition 2.10. Negative predications with qualified definite descriptions take the following forms:

1. $-\psi(\iota_{\mathcal{Q}} x \varphi(x))$: 'the $\mathcal{Q}$-unique x which is φ fails to be ψ';
2. $-\psi(\iota_{\mathcal{Q}} x - \varphi(x))$: 'the $\mathcal{Q}$-unique x which fails to be φ fails to be ψ'.

Remark 2.11. $\mathcal{L}\iota$ is an extension of $\mathcal{L}$ only in the sense that it uses abbreviations which are not present in $\mathcal{L}$.

3. Proof systems

In order to obtain a proof system for reasoning with qualified definiteness, we enrich the intuitionistic bipredicational $\mathbf{I0}(\mathcal{S}_b^{=})$-systems defined in [24] with rules for qualified definiteness, by adapting the rules for definiteness presented in [3, 4]. We call the resulting systems $\mathbf{I0}(\mathcal{S}_b^{=})\iota$*-systems.*

3.1. Bipredicational natural deduction

We first repeat the parts of the definition of $\mathbf{I0}(\mathcal{S}_b^{=})$-systems from [24] which are relevant for present purposes.

3.1.1. Bipredicational subatomic systems

Definition 3.1. A *bipredicational subatomic system* $\mathcal{S}_b$ is a pair $\langle \mathcal{I}, \mathcal{R}_b \rangle$, where $\mathcal{I}$ is a *subatomic base* and $\mathcal{R}_b$ is a set of *introduction and elimination rules for atomic sentences and negative predications.* $\mathcal{I}$ is a 3-tuple $\langle \mathcal{C}, \mathcal{P}, v \rangle$, where v is such that:

1. For any $\alpha \in \mathcal{C}$, $v : \mathcal{C} \rightarrow \wp(Atm)$, where $v(\alpha) \subseteq Atm(\alpha)$.
2. For any $\varphi^n \in \mathcal{P}$, $v : \mathcal{P} \rightarrow \wp(Atm)$, where $v(\varphi^n) \subseteq Atm(\varphi^n)$.

We let $\tau\Gamma =_{def} v(\tau)$ for any $\tau \in \mathcal{C} \cup \mathcal{P}$, and call $\tau\Gamma$ the set of *term assumptions* for τ. $\mathcal{R}_b$ contains I/E-rules of the following form:

$$\dfrac{\begin{matrix}\mathcal{D}_0 \\ \varphi_0^n\Gamma\end{matrix} \qquad \begin{matrix}\mathcal{D}_1 \\ \alpha_1\Gamma\end{matrix} \quad \ldots \quad \begin{matrix}\mathcal{D}_n \\ \alpha_n\Gamma\end{matrix}}{\varphi_0^n\alpha_1\ldots\alpha_n}\,(as\mathrm{I}) \qquad \dfrac{\begin{matrix}\mathcal{D}_1 \\ \varphi_0^n\alpha_1\ldots\alpha_n\end{matrix}}{\tau_i\Gamma}\,(as\mathrm{E}_i)$$

$$\dfrac{\begin{matrix}\mathcal{D}_0 \\ \varphi_0^n\Gamma\end{matrix} \qquad \begin{matrix}\mathcal{D}_1 \\ \alpha_1\Gamma\end{matrix} \quad \ldots \quad \begin{matrix}\mathcal{D}_n \\ \alpha_n\Gamma\end{matrix}}{-\varphi_0^n\alpha_1\ldots\alpha_n}\,(-as\mathrm{I}) \qquad \dfrac{\begin{matrix}\mathcal{D}_1 \\ -\varphi_0^n\alpha_1\ldots\alpha_n\end{matrix}}{\tau_i\Gamma}\,(-as\mathrm{E}_i)$$

Side conditions:

1. asI: $\varphi_0^n\alpha_1\ldots\alpha_n \in \varphi_0^n\Gamma \cap \alpha_1\Gamma \cap \ldots \cap \alpha_n\Gamma$.
2. $-as$I: $\varphi_0^n\alpha_1\ldots\alpha_n \notin \varphi_0^n\Gamma \cap \alpha_1\Gamma \cap \ldots \cap \alpha_n\Gamma$.
3. asE$_i$ and $-as$E$_i$: $i \in \{0, \ldots, n\}$ and $\tau_i \in \{\varphi_0^n, \alpha_1, \ldots, \alpha_n\}$.

Terminology: We say that $-\varphi_0^n\alpha_1\ldots\alpha_n$ is *negatively contained* in $\varphi_0^n\Gamma \cap \alpha_1\Gamma \cap \ldots \cap \alpha_n\Gamma$, in case the side condition on $-as$I is satisfied.

Definition 3.2 (Derivations in $\mathcal{S}_b$-systems).

Basic step. Any term assumption $\tau\Gamma$, any atomic sentence (resp. negative predication), i.e., a derivation from the open assumption of $\varphi_0^n\alpha_1 ... \alpha_n$ (resp. $-\varphi_0^n\alpha_1 ... \alpha_n$) is an $\mathcal{S}_b$-derivation.

Induction step. If $\mathcal{D}_i$, for $i \in \{0, ..., n\}$, are $\mathcal{S}_b$-derivations, then an $\mathcal{S}_b$-derivation can be constructed by means of the I/E-rules for as and $-as$ displayed above.

Remark 3.3. The term assumptions are, so to speak, proof-theoretic semantic values of the non-logical constants. Applications of the subatomic introduction rules asI and $-as$I serve to establish, on the basis of these values, the truth of atomic sentences and negative predications, respectively. Negative predication (or predication failure) is understood as subatomic derivation failure (cf. [24]).

3.1.2. Bipredicational subatomic identity systems

Definition 3.4. Atomic sentences $\varphi(\alpha_1)$ and $\varphi(\alpha_2)$ are *mirror atomic sentences* if and only if they are exactly alike except that the former contains occurrences of α_1 at all the places at which the latter contains occurrences of α_2, and vice versa.

Definition 3.5. A *bipredicational subatomic identity system* $\mathcal{S}_b^{=}$ is a 3-tuple $\langle \mathcal{I}, \mathcal{R}_b, \mathcal{R}_b^{=} \rangle$, which extends a bipredicational subatomic system with a set $\mathcal{R}_b^{=}$ of *I/E-rules for (positive/negative) qualified identity sentences*, where $\mathcal{Q} \subseteq \mathcal{P}$.

1. $\overset{\pm}{=}_{\mathcal{Q}}$:

$$\dfrac{\begin{array}{ccccc} [\varphi_1(\alpha_1)]^{(1_1)} & [\varphi_1(\alpha_2)]^{(1_2)} & & [\varphi_k(\alpha_1)]^{(k_1)} & [\varphi_k(\alpha_2)]^{(k_2)} \\ \mathcal{D}_{1_1} & \mathcal{D}_{1_2} & & \mathcal{D}_{k_1} & \mathcal{D}_{k_2} \\ \varphi_1(\alpha_2) & \varphi_1(\alpha_1) & \dots & \varphi_k(\alpha_2) & \varphi_k(\alpha_1) \end{array}}{\alpha_1 \overset{\pm}{=}_{\mathcal{Q}} \alpha_2} \; {\scriptstyle (\overset{\pm}{=}_{\mathcal{Q}}\mathrm{I}),\ 1_1, \dots, k_2}$$

$$\dfrac{\begin{array}{cc}\mathcal{D}_1 & \mathcal{D}_{i_1}\\ \alpha_1 \overset{\pm}{=}_{\mathcal{Q}} \alpha_2 & \varphi_i(\alpha_1)\end{array}}{\varphi_i(\alpha_2)}\,(\overset{\pm}{=}_{\mathcal{Q}}\mathrm{E}_i1) \qquad \dfrac{\begin{array}{cc}\mathcal{D}_1 & \mathcal{D}_{i_2}\\ \alpha_1 \overset{\pm}{=}_{\mathcal{Q}} \alpha_2 & \varphi_i(\alpha_2)\end{array}}{\varphi_i(\alpha_1)}\,(\overset{\pm}{=}_{\mathcal{Q}}\mathrm{E}_i2)$$

where $\varphi_i \in \mathcal{Q}$, $i \in \{1, ..., k\}$, and $\varphi_i(\alpha_1)$ and $\varphi_i(\alpha_2)$ are mirror atomic sentences.

2. $\overset{-}{=}_{\mathcal{Q}}$:

$$\dfrac{\begin{array}{ccccc}[-\varphi_1(\alpha_1)]^{(1_1)} & [-\varphi_1(\alpha_2)]^{(1_2)} & & [-\varphi_k(\alpha_1)]^{(k_1)} & [-\varphi_k(\alpha_2)]^{(k_2)}\\ \mathcal{D}_{1_1} & \mathcal{D}_{1_2} & & \mathcal{D}_{k_1} & \mathcal{D}_{k_2}\\ -\varphi_1(\alpha_2) & -\varphi_1(\alpha_1) & \dots & -\varphi_k(\alpha_2) & -\varphi_k(\alpha_1)\end{array}}{\alpha_1 \overset{-}{=}_{\mathcal{Q}} \alpha_2}\,{\scriptstyle (\overset{-}{=}_{\mathcal{Q}}\mathrm{I}),\ 1_1, ..., k_2}$$

$$\dfrac{\begin{array}{cc}\mathcal{D}_1 & \mathcal{D}_{i_1}\\ \alpha_1 \overset{-}{=}_{\mathcal{Q}} \alpha_2 & -\varphi_i(\alpha_1)\end{array}}{-\varphi_i(\alpha_2)}\,(\overset{-}{=}_{\mathcal{Q}}\mathrm{E}_i1) \qquad \dfrac{\begin{array}{cc}\mathcal{D}_1 & \mathcal{D}_{i_2}\\ \alpha_1 \overset{-}{=}_{\mathcal{Q}} \alpha_2 & -\varphi_i(\alpha_2)\end{array}}{-\varphi_i(\alpha_1)}\,(\overset{-}{=}_{\mathcal{Q}}\mathrm{E}_i2)$$

where $\varphi_i \in \mathcal{Q}$, $i \in \{1, ..., k\}$, and $\varphi_i(\alpha_1)$ and $\varphi_i(\alpha_2)$ are mirror atomic sentences.

Remark 3.6. In contrast to the standard I-rules for identity, the I-rules for qualified identity allow one to introduce formulae in which the identity predicate is not necessarily flanked by two occurrences of the same constant.

DEFINITION 3.7. It will sometimes be convenient to use the notation $\{\mathcal{D}\}$ for the set of the subderivations $\mathcal{D}_{2_1}, \mathcal{D}_{2_2}, ..., \mathcal{D}_{k_1}, \mathcal{D}_{k_2}$ in applications of I-rules for qualified identity.

Remark 3.8. The I-rules for qualified identity reflect the definitions of the qualified identity predicates stated in Definition 2.2(3). They absorb the logical operators at work in the definientia into the metalanguage. As mentioned, second-order quantifiers are not among these operators. It can be argued that the rule (=I) proposed in [14] and its refined version (=I′) proposed in [15] do indeed reflect the second-order definition of identity mentioned in Remark 2.5, that is, $a = b =_{def} \forall F(Fa \leftrightarrow Fb)$. Read's rules, with F a predicate variable that ranges over monadic (*sic*!) predicate letters ([15, p. 415, fn. 20]), are as follows:

$$\dfrac{\begin{matrix}[Fa]\\ \mathcal{D}_1\\ Fb\end{matrix}}{a = b}\,(=\mathrm{I}) \qquad\qquad \dfrac{\begin{matrix}[Fa]\\ \mathcal{D}_1\\ Fb\end{matrix} \qquad \begin{matrix}[Fb]\\ \mathcal{D}_2\\ Fa\end{matrix}}{a = b}\,(=\mathrm{I}')$$

Side condition on (=I):
"provided 'F' does not occur (as a predicate variable) in any assumption other than Fa" ([14, p. 116]).

Side condition on (=I′):
"where the predicate variable 'F' does not occur in any parametric assumptions" ([15, p. 415]).

Read's rules involve no second-order universal quantifier in their body. However, their side conditions are reminiscent of the usual condition on ∀I, except for dealing with predicate variables. In this way, the second-order universal quantifier of the aforementioned definition appears to be absorbed into the side conditions. In [14], Read argues that (=I′) can be simplified to (=I). We note that, from a foundational intuitionistic perspective, Read's argument is not acceptable as it appeals to classical *reductio ad absurdum* (cf. [14, p. 116]). In [15], Read adopts general elimination rules for = and argues in favour of (=I′).

3.1.3. Bipredicational subatomic natural deduction systems

DEFINITION 3.9 (Derivations in $\mathbf{I0}(\mathcal{S}_b^{=})$-systems).

Basic step. Any derivation in an $\mathcal{S}_b^{=}$-system and any formula A (i.e., a derivation from the open assumption of A) is a derivation in an $\mathbf{I0}(\mathcal{S}_b^{=})$-system.

Induction step. If $\mathcal{D}_1$, $\mathcal{D}_2$, and $\mathcal{D}_3$ are derivations in an $\mathbf{I0}(\mathcal{S}_b^{=})$-system, and C possibly a term assumption, then a derivation in an $\mathbf{I0}(\mathcal{S}_b^{=})$-system can be constructed by means of the rules:

$$\dfrac{\begin{matrix}\mathcal{D}_1 & \mathcal{D}_2\\ A & B\end{matrix}}{A\&B}\,(\&\mathrm{I}) \qquad \dfrac{\begin{matrix}\mathcal{D}_1\\ A\&B\end{matrix}}{A}\,(\&\mathrm{E1}) \qquad \dfrac{\begin{matrix}\mathcal{D}_1\\ A\&B\end{matrix}}{B}\,(\&\mathrm{E2})$$

$$\dfrac{\begin{matrix}\mathcal{D}_1\\ A\end{matrix}}{A\vee B}\,(\vee\mathrm{I1}) \qquad \dfrac{\begin{matrix}\mathcal{D}_1\\ B\end{matrix}}{A\vee B}\,(\vee\mathrm{I2})$$

$$\dfrac{\begin{matrix} & [A]^{(u)} & [B]^{(v)}\\ \mathcal{D}_1 & \mathcal{D}_2 & \mathcal{D}_3\\ A\vee B & C & C\end{matrix}}{C}\,(\vee\mathrm{E}),\, u, v \qquad \dfrac{\begin{matrix}[A]^{(u)}\\ \mathcal{D}_1\\ B\end{matrix}}{A\supset B}\,(\supset\mathrm{I}),\, u \qquad \dfrac{\begin{matrix}\mathcal{D}_1 & \mathcal{D}_2\\ A\supset B & A\end{matrix}}{B}\,(\supset\mathrm{E})$$

$$\dfrac{\begin{matrix}\mathcal{D}_1\\ A(x/o)\end{matrix}}{\forall xA}\,(\forall\mathrm{I}) \qquad \dfrac{\begin{matrix}\mathcal{D}_1\\ \forall xA\end{matrix}}{A(x/o)}\,(\forall\mathrm{E}) \qquad \dfrac{\begin{matrix}\mathcal{D}_1\\ A(x/o)\end{matrix}}{\exists xA}\,(\exists\mathrm{I}) \qquad \dfrac{\begin{matrix} & [A(x/o)]^{(u)}\\ \mathcal{D}_1 & \mathcal{D}_2\\ \exists xA & C\end{matrix}}{C}\,(\exists\mathrm{E}),\, u$$

$$\dfrac{\begin{matrix}\mathcal{D}_1\\ \perp\end{matrix}}{A}\,(\perp\mathrm{i})$$

Side conditions:

1. In $\forall$I: (i) if o is a proper variable y, then $o \equiv x$ or o is not free in A, and o is not free in any assumption of a formula which is open in the derivation of $A(x/o)$; (ii) if o is a nominal constant, then o does neither occur in an undischarged assumption of a formula, nor in $\forall xA$, nor in a term assumption leaf $o\Gamma$; (iii) o is nominal constant and $\begin{matrix}\mathcal{D}_1\\ A(x/o)\end{matrix}$ for all $o \in \mathcal{C}$.
2. In $\forall$E: o is free for x in A.
3. In $\exists$E: (i) if o is a proper variable y, then $o \equiv x$ or o is not free in A, and o is not free in C nor in any assumption of a formula which is open in the derivation of the upper occurrence of C other than

$[A(x/o)]^{(u)}$; (ii) if o is a nominal constant, then o does neither occur in an undischarged assumption of a formula, nor in $\exists x A$, nor in C, nor in a term assumption leaf $o\Gamma$.

4. In $\exists$I: o is free for x in A.

Minimal bipredicational subatomic natural deduction systems, $\mathbf{M0}(\mathcal{S}_b^{=})$-systems, result from $\mathbf{I0}(\mathcal{S}_b^{=})$-systems, in case $\bot$i is removed.

In case we employ the $\forall$I-rule according to the provisos for it given in (i) [(ii), (iii)], we use the labels $\forall$I.i [$\forall$I.ii, $\forall$I.iii]. Similarly, for the $\exists$E-rule and the labels $\exists$E.i and $\exists$E.ii.

We mention the main results obtained for $\mathbf{I0}(\mathcal{S}_b^{=})$-systems in [24] making use of the methods developed in [13]; see also [21].

THEOREM 3.10 (Normalization). *Any derivation $\mathcal{D}$ in an $\mathbf{I0}(\mathcal{S}_b^{=})$-system can be transformed into a normal $\mathbf{I0}(\mathcal{S}_b^{=})$-derivation.*

Importantly, $\mathbf{I0}(\mathcal{S}_b^{=})$-systems enjoy the subformula property as a special case of the subexpression property. The latter property deals with units and expressions. Roughly, a unit is either a formula or a term assumption $\tau\Gamma$, and an expression is either a formula or the non-logical constant τ of $\tau\Gamma$.

THEOREM 3.11 (Subexpression property). *If $\mathcal{D}$ is a normal derivation of a unit U from a set of units Γ in an $\mathbf{I0}(\mathcal{S}_b^{=})$-system, then each unit in $\mathcal{D}$ is a subexpression of an expression in $\Gamma \cup \{U\}$.*

COROLLARY 3.12 (Subformula property). If $\mathcal{D}$ is a normal $\mathbf{I0}(\mathcal{S}_b^{=})$-derivation of formula A from a set of formulae Γ, then each formula in $\mathcal{D}$ is a subformula of a formula in $\Gamma \cup \{A\}$.

These results guarantee, e.g., the consistency of the systems and simplify proof search in them.

Remark 3.13. Digression: $\mathbf{I0}(\mathcal{S}_b^{=})$-systems are special cases of the $\mathbf{I}(\mathcal{S}_b^{=})$-systems studied in [24]. The '**0**' indicates the lack of rules which handle predication conflicts (*c*-rules; cf. [24]: 116). The terminology of negative containment (Definition 3.1) introduced in [25] becomes relevant, in case *c*-rules are present in an $\mathbf{I}(\mathcal{S}_b^{=})$-system. For such systems the following formulation of the subexpression property is more adequate than the one used in [24]: '(...) then each unit in $\mathcal{D}$ is a subexpression of an expression in $\Gamma \cup \{U\}$ or negatively contained'. Similarly, for the formulation of the subformula property: '(...) then each formula in $\mathcal{D}$ is a subformula of a formula in $\Gamma \cup \{A\}$ or negatively contained'. The original formulation which treats $-as$-formulae as "subexpressions" of term assumptions for the non-logical constants in U (resp. A) constitutes an abuse of terminology.

3.2. Bipredicational natural deduction for qualified definiteness

We now add rules for the introduction and elimination of qualified definiteness to $\mathbf{I0}(\mathcal{S}_b^{=})$-systems in order to obtain $\mathbf{I0}(\mathcal{S}_b^{=})\iota$-systems which are sufficient to define a proof-theoretic semantics for the simplest possible constructions involving definite descriptions.

Definition 3.14. Let $\mathcal{Q} \subseteq \mathcal{P}$. In the $\iota_{\mathcal{Q}}$I-rule below, the conclusion of $\mathcal{D}_1$ [$\mathcal{D}_2$, $\mathcal{D}_3$] corresponds to the E- [QU-, P-] clause. Likewise for $\iota_{\mathcal{Q}}-$I.

1. *Rules for positive qualified definiteness*:

$$\dfrac{\begin{array}{c}\mathcal{D}_1\\ \exists x\varphi(x)\end{array} \qquad \begin{array}{c}\mathcal{D}_2\\ \forall u\forall v((\varphi(u)\ \&\ \varphi(v)) \supset u \overset{\pm}{=}_{\mathcal{Q}} v)\end{array} \qquad \begin{array}{c}\mathcal{D}_3\\ \forall w(\varphi(w) \supset \psi(w))\end{array}}{\psi(\iota_{\mathcal{Q}} x\varphi(x))}\ (\iota_{\mathcal{Q}}\mathrm{I})$$

$$\dfrac{\begin{array}{c}\mathcal{D}_1\\ \psi(\iota_{\mathcal{Q}} x\varphi(x))\end{array}}{\exists x\varphi(x)}\ (\iota_{\mathcal{Q}}\mathrm{E1}) \qquad \dfrac{\begin{array}{c}\mathcal{D}_1\\ \psi(\iota_{\mathcal{Q}} x\varphi(x))\end{array}}{\forall u\forall v((\varphi(u)\ \&\ \varphi(v)) \supset u \overset{\pm}{=}_{\mathcal{Q}} v)}\ (\iota_{\mathcal{Q}}\mathrm{E2})$$

$$\dfrac{\begin{array}{c}\mathcal{D}_1\\ \psi(\iota_{\mathcal{Q}} x\varphi(x))\end{array}}{\forall w(\varphi(w) \supset \psi(w))}\ (\iota_{\mathcal{Q}}\mathrm{E3})$$

The $\iota_{\mathcal{Q}}$I/E-rules for $-\psi(\iota_{\mathcal{Q}} x\varphi(x))$ are analogous.

2. *Rules for negative qualified definiteness*:

$$\dfrac{\begin{matrix}\mathcal{D}_1\\ \exists x - \varphi(x)\end{matrix} \qquad \begin{matrix}\mathcal{D}_2\\ \forall u\forall v((-\varphi(u)\ \&\ -\varphi(v)) \supset u \overset{-}{=}_{\mathcal{Q}} v)\end{matrix} \qquad \begin{matrix}\mathcal{D}_3\\ \forall w(-\varphi(w) \supset \psi(w))\end{matrix}}{\psi(\iota_{\mathcal{Q}} x - \varphi(x))}\,(\iota_{\mathcal{Q}}\text{–I})$$

$$\dfrac{\begin{matrix}\mathcal{D}_1\\ \psi(\iota_{\mathcal{Q}} x - \varphi(x))\end{matrix}}{\exists x - \varphi(x)}\,(\iota_{\mathcal{Q}}\text{–E1}) \qquad \dfrac{\begin{matrix}\mathcal{D}_1\\ \psi(\iota_{\mathcal{Q}} x - \varphi(x))\end{matrix}}{\forall u\forall v((-\varphi(u)\ \&\ -\varphi(v)) \supset u \overset{-}{=}_{\mathcal{Q}} v)}\,(\iota_{\mathcal{Q}}\text{–E2})$$

$$\dfrac{\begin{matrix}\mathcal{D}_1\\ \psi(\iota_{\mathcal{Q}} x - \varphi(x))\end{matrix}}{\forall w(-\varphi(w) \supset \psi(w))}\,(\iota_{\mathcal{Q}}\text{–E3})$$

The $\iota_{\mathcal{Q}}$–I/E-rules for $-\psi(\iota_{\mathcal{Q}} x - \varphi(x))$ are analogous.

Example 3.15. Let $\mathcal{Q} = \{\varphi_1, ..., \varphi_k\}$, $\mathcal{Q} \subseteq \mathcal{P}$, and $\varphi_i, \varphi_j \in \mathcal{Q}$, where $i, j \in \{1, ..., k\}$ and $i \neq j$.

$$\mathcal{D}_1 = \dfrac{\dfrac{\varphi_i\Gamma \qquad ... \qquad \alpha\Gamma}{\varphi_i(\alpha)}}{\exists x \varphi_i(x)} \tag{3.1}$$

$$\mathcal{D}_2 = \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[\varphi_1(\alpha)]^{(1_1)}}{\varphi_1\Gamma} \quad ... \quad \dfrac{\dfrac{[\varphi_i(\alpha)\&\varphi_i(\beta)]^{(1)}}{\varphi_i(\beta)}}{\beta\Gamma}}{\varphi_1(\beta)} \qquad \dfrac{\dfrac{[\varphi_1(\beta)]^{(1_2)}}{\varphi_1\Gamma} \quad ... \quad \dfrac{\dfrac{[\varphi_i(\alpha)\&\varphi_i(\beta)]^{(1)}}{\varphi_i(\alpha)}}{\alpha\Gamma}}{\varphi_1(\alpha)} \qquad \{\mathcal{D}\}}{\alpha \overset{\pm}{=}_{\mathcal{Q}} \beta}\,1_1, ..., k_2}{(\varphi_i(\alpha)\&\varphi_i(\beta)) \supset \alpha \overset{\pm}{=}_{\mathcal{Q}} \beta}\,1}{\forall v((\varphi_i(\alpha)\&\varphi_i(v)) \supset \alpha \overset{\pm}{=}_{\mathcal{Q}} v)}\,\text{iii}}{\forall u\forall v((\varphi_i(u)\&\varphi_i(v)) \supset u \overset{\pm}{=}_{\mathcal{Q}} v)}\,\text{iii} \tag{3.2}$$

$$\mathcal{D}_3 = \dfrac{\dfrac{\dfrac{\varphi_j\Gamma \qquad \dots \qquad \dfrac{[\varphi_i(\alpha)]^{(2)}}{\alpha\Gamma}}{\varphi_j(\alpha)}}{\varphi_i(\alpha) \supset \varphi_j(\alpha)}\,2}{\forall w(\varphi_i(w) \supset \varphi_j(w))}\,\text{iii} \tag{3.3}$$

$$\dfrac{\begin{matrix}\mathcal{D}_1\\ \exists x\varphi_i(x)\end{matrix} \qquad \begin{matrix}\mathcal{D}_2\\ \forall u\forall v((\varphi_i(u)\&\varphi_i(v)) \supset u \overset{\pm}{=}_{\mathcal{Q}} v)\end{matrix} \qquad \begin{matrix}\mathcal{D}_3\\ \forall w(\varphi_i(w) \supset \varphi_j(w))\end{matrix}}{\varphi_j(\iota_{\mathcal{Q}} x\varphi_i(x))}\,(\iota_{\mathcal{Q}}\mathrm{I}) \tag{3.4}$$

3.3. Normalization and the subformula property

In order to prove normalization for $\mathbf{I0}(\mathcal{S}_b^{=})\iota$-systems, we make use of the following conversions.

Definition 3.16. The *conversions (detour, permutation, simplification)* for $\mathbf{I0}(\mathcal{S}_b^{=})\iota$-systems comprise those for $\mathbf{I0}(\mathcal{S}_b^{=})$-systems (see [24]) and the following detour conversions:

1. $\iota_{\mathcal{Q}}$-*Conversions*:

$$\dfrac{\dfrac{\begin{matrix}\mathcal{D}_1\\ \exists x\varphi(x)\end{matrix} \qquad \begin{matrix}\mathcal{D}_2\\ \forall u\forall v((\varphi(u)\ \&\ \varphi(v)) \supset u \overset{\pm}{=}_{\mathcal{Q}} v)\end{matrix} \qquad \begin{matrix}\mathcal{D}_3\\ \forall w(\varphi(w) \supset \psi(w))\end{matrix}}{\psi(\iota_{\mathcal{Q}} x\varphi(x))}\,(\iota_{\mathcal{Q}}\mathrm{I})}{\exists x\varphi(x)}\,(\iota_{\mathcal{Q}}\mathrm{E1})$$

$$\text{conv}$$

$$\begin{matrix}\mathcal{D}_1\\ \exists x\varphi(x)\end{matrix}$$

$$\dfrac{\dfrac{\begin{matrix}\mathcal{D}_1\\ \exists x\varphi(x)\end{matrix} \qquad \begin{matrix}\mathcal{D}_2\\ \forall u\forall v((\varphi(u)\ \&\ \varphi(v)) \supset u \overset{\pm}{=}_{\mathcal{Q}} v)\end{matrix} \qquad \begin{matrix}\mathcal{D}_3\\ \forall w(\varphi(w) \supset \psi(w))\end{matrix}}{\psi(\iota_{\mathcal{Q}} x\varphi(x))}\,(\iota_{\mathcal{Q}}\mathrm{I})}{\forall u\forall v((\varphi(u)\ \&\ \varphi(v)) \supset u \overset{\pm}{=}_{\mathcal{Q}} v)}\,(\iota_{\mathcal{Q}}\mathrm{E2})$$

conv

$$\begin{array}{c}\mathcal{D}_2\\ \forall u\forall v((\varphi(u)\ \&\ \varphi(v))\supset u \overset{\pm}{=}_{\mathcal{Q}} v)\end{array}$$

$$\dfrac{\dfrac{\begin{array}{c}\mathcal{D}_1\\ \exists x\varphi(x)\end{array}\qquad \begin{array}{c}\mathcal{D}_2\\ \forall u\forall v((\varphi(u)\ \&\ \varphi(v))\supset u \overset{\pm}{=}_{\mathcal{Q}} v)\end{array}\qquad \begin{array}{c}\mathcal{D}_3\\ \forall w(\varphi(w)\supset\psi(w))\end{array}}{\psi(\iota_{\mathcal{Q}}x\varphi(x))}\,(\iota_{\mathcal{Q}}\mathrm{I})}{\forall w(\varphi(w)\supset\psi(w))}\,(\iota_{\mathcal{Q}}\mathrm{E3})$$

conv

$$\begin{array}{c}\mathcal{D}_3\\ \forall w(\varphi(w)\supset\psi(w))\end{array}$$

2. $\iota_{\mathcal{Q}}-$-*Conversions*: analogous.

Remark 3.17. Unlike the ιE2-rules in [3, 4], the above E2-rules have a single premiss and invert directly.

THEOREM 3.18 (Normalization). *Any derivation $\mathcal{D}$ in an* ***IO***$(\mathcal{S}_b^{=})\iota$*-system can be transformed into a normal* ***IO***$(\mathcal{S}_b^{=})\iota$*-derivation.*

PROOF: We repeat the corresponding proof for **IO**$(\mathcal{S}_b^{=})$-systems in [24], taking also the detour conversions for qualified definiteness into account. As a result, all detours can be eliminated from derivations in these systems. □

THEOREM 3.19 (Subexpression property). *If $\mathcal{D}$ is a normal derivation of a unit U from a set of units Γ in an* ***IO***$(\mathcal{S}_b^{=})\iota$*-system, then each unit in $\mathcal{D}$ is a subexpression of an expression in $\Gamma \cup \{U\}$.*

PROOF: We proceed like in the corresponding proof for **IO**$(\mathcal{S}_b^{=})$-systems in [24]. As a result, all expressions in $\mathcal{D}$ are subexpressions of either the root or the leaves of $\mathcal{D}$. □

COROLLARY 3.20 (Subformula property). If $\mathcal{D}$ is a normal **IO**$(\mathcal{S}_b^{=})\iota$-derivation of formula A from a set of formulae Γ, then each formula in $\mathcal{D}$ is a subformula of a formula in $\Gamma \cup \{A\}$.

Remark 3.21. Since the identity predicates used in the proof systems [3, 4], are primitive, such a subformula result is not available for these systems. This remark also applies to other available intuitionistic natural deduction systems for definiteness (e.g., [11, 19]).

Corollary 3.22 (Internal completeness). Internal completeness of $\mathbf{I0}(\mathcal{S}_b^{=})\iota$-systems in the sense of [6] (adapted to natural deduction) is given by Corollary 3.20.[2] To establish internal completeness for them in the sense of [24, p. 127], we proceed like described therein.

4. A proof-theoretic semantics

On the basis of the results obtained, we may formulate a subatomic proof-theoretic semantics for qualified definiteness. For this purpose, we adjust the corresponding definitions form [24] to the present systems.

Definition 4.1.

1. A derivation $\mathcal{D}$ of a formula A in an $\mathbf{I0}(\mathcal{S}_b^{=})\iota$-system is a *canonical derivation* iff it derives A by means of an application of an I-rule (in the last step of $\mathcal{D}$).
2. A canonical derivation $\mathcal{D}$ of A in an $\mathbf{I0}(\mathcal{S}_b^{=})\iota$-system is a *canonical proof* of A in that system iff there are no applications of as-rules or $-as$-rules in $\mathcal{D}$ and all assumptions of $\mathcal{D}$ have been discharged.
3. The conclusions of canonical $\mathbf{I0}(\mathcal{S}_b^{=})\iota$-derivations are $\mathbf{I0}(\mathcal{S}_b^{=})\iota$-*theses* and the conclusions of $\mathbf{I0}(\mathcal{S}_b^{=})\iota$-derivations which are also proofs are $\mathbf{I0}(\mathcal{S}_b^{=})\iota$-*theorems*.

---

[2] Cf. Girard ([6, pp. 139–140]): "If we consider cut-free proofs, then all possible proofs are already there, there is no way to produce new ones. In other terms, the calculus is complete—nothing is missing. Observe that this completeness does not refer to any sort of model, it is an internal property of syntax. Such a property cannot be an accident, it should be given its real place, the first: *The subformula property is the actual completeness*."

Definition 4.2 (Meaning). Let I be an $\mathbf{I0}(\mathcal{S}_b^{=})\iota$-system.

1. The meaning of a *non-logical constant* τ is given by the term assumptions $\tau\Gamma$ for τ which are determined by the subatomic base of the $\mathcal{S}_b^{=}$-system of I.
2. The meaning of a *formula* A of $\mathcal{L}\iota$ is given by the set of canonical derivations of A in I.

Remark 4.3. The rules for qualified identity (Definition 3.5) allow not only for reductions in terms of detour conversions, but also for expansions (cf. [22, p. 256]). This is a further point, in which they differ from the standard natural deduction rules for identity (cf. [24, p. 104]). For an overview of the structural proof theory of identity see [9].

Remark 4.4. Note that this formal account of meaning does not make use of a semantic ontology (e.g., individuals, possible worlds), something essential to model-theoretic semantics. Specifically, the meaning of $\exists$-formulae does not presuppose a domain of individuals. Strictly speaking, $\exists xA$ reads: 'For at least one x, A', where x is a nominal variable ranging over $\mathcal{C}$. This feature of the present semantics makes it particularly natural for the analysis of constructions which involve non-denoting (or empty) terms (e.g., 'Pegasus', 'the captive unicorn').

5. On incomplete descriptions

Qualified uniqueness allows for fine-tuning.

Remark 5.1. Let $\{\varphi_i\} \subset \mathcal{Q}' \subset \mathcal{P}$ and $\varphi_i \in \mathcal{P}$, where $i \in \{1, ..., k\}$. We consider the following cases: (i) $\mathcal{Q} = \mathcal{P}$, (ii) $\mathcal{Q} = \mathcal{Q}'$, and (iii) $\mathcal{Q} = \{\varphi_i\}$.

Case (i): Like (3.2), but with $\mathcal{Q}$ replaced by $\mathcal{P}$. This case gives us the maximal degree of qualified uniqueness. For every x and y, if they are φ_i, then they are identical with respect to every predicate (i.e., they are indiscernible in every respect).

Case (ii): Like case (i), but with $\mathcal{P}$ replaced by $\mathcal{Q}'$ and with $\{\mathcal{D}\}$ replaced by $\{\mathcal{D}\}'$, where $\{\mathcal{D}\}' \subset \{\mathcal{D}\}$. This case gives us an intermediate degree of qualified uniqueness. For every x and y, if they are φ_i, then they are identical with respect to every predicate in $\mathcal{Q}'$ (i.e., they are indiscernible with respect to $\mathcal{Q}'$, but discernible with respect to $\mathcal{P} \setminus \mathcal{Q}'$).

Case (iii):

$$\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{[\varphi_i(\alpha)]^{(1_1)}}{\varphi_i\Gamma} \quad \ldots \quad \cfrac{\cfrac{[\varphi_i(\alpha)\&\varphi_i(\beta)]^{(1)}}{\varphi_i(\beta)}}{\beta\Gamma}}{\varphi_i(\beta)} \qquad \cfrac{\cfrac{[\varphi_i(\beta)]^{(1_2)}}{\varphi_i\Gamma} \quad \ldots \quad \cfrac{\cfrac{[\varphi_i(\alpha)\&\varphi_i(\beta)]^{(1)}}{\varphi_i(\alpha)}}{\alpha\Gamma}}{\varphi_i(\alpha)}}{\alpha \overset{\pm}{=}_{\{\varphi_i\}} \beta}\,1_1, 1_2}{(\varphi_i(\alpha)\&\varphi_i(\beta)) \supset \alpha \overset{\pm}{=}_{\{\varphi_i\}} \beta}\,1}{\forall y((\varphi_i(\alpha)\&\varphi_i(y)) \supset \alpha \overset{\pm}{=}_{\{\varphi_i\}} y)}\,\mathrm{iii}}{\forall x\forall y((\varphi_i(x)\&\varphi_i(y)) \supset x \overset{\pm}{=}_{\{\varphi_i\}} y)}\,\mathrm{iii} \tag{5.1}$$

This case gives us the minimal degree of qualified uniqueness. For every x and y, if they are φ_i, then they are identical with respect to every predicate in the singleton $\{\varphi_i\}$ (i.e., they are indiscernible with respect to the predicate φ_i, but discernible with respect to at least one predicate in $\mathcal{P} \setminus \{\varphi_i\}$). (Likewise for negative qualified uniqueness.)

Qualified definiteness allows for fine-tuning, since it involves qualified uniqueness.

Remark 5.2. Let $\{\varphi_i\} \subset \mathcal{Q}' \subset \mathcal{P}$, let $P = \varphi_i$, and $B = \varphi_j$ for $\varphi_i, \varphi_j \in \mathcal{Q}'$, where $i, j \in \{1, ..., k\}$ and $i \neq j$. P: '... is a pope'; B: '... is bald'. And let $\mathcal{D}_2(i)$ [$\mathcal{D}_2(ii)$, $\mathcal{D}_2(iii)$] refer to the derivation for case (i) [(ii), (iii)] mentioned in the previous remark. We may, then, distinguish three general cases of qualified definiteness.

Case (i). Maximal qualified definiteness:

$$\cfrac{\overset{\mathcal{D}_1}{\exists x\varphi_i(x)} \qquad \overset{\mathcal{D}_{2(i)}}{\forall u\forall v((\varphi_i(u)\&\varphi_i(v)) \supset u \overset{\pm}{=}_{\mathcal{P}} v)} \qquad \overset{\mathcal{D}_3}{\forall w(\varphi_i(w) \supset \varphi_j(w))}}{\varphi_j(\iota_{\mathcal{P}}x\varphi_i(x))}\,(\iota_{\mathcal{P}}\mathrm{I}) \tag{5.2}$$

The premisses of the $\iota_{\mathcal{P}}$I-application say that there is at least one thing which is φ_i, that any two things which are φ_i are the same in any respect, and that everything that is φ_i is φ_j. The conclusion $\varphi_j(\iota_{\mathcal{P}}x\varphi_i(x))$ can be read: 'the $\mathcal{P}$-unique x which is φ_i is φ_j', or, simplifying the reading of Definition 2.9(1) further, 'the only φ_i is φ_j'. We may use these readings only in case there is a single x that is φ_i. This is definiteness proper. We use it for the analysis of (1), in case there is no schism.

Case (ii). Intermediate qualified definiteness:

$$\frac{\overset{\mathcal{D}_1}{\exists x\varphi_i(x)} \qquad \overset{\mathcal{D}_{2(ii)}}{\forall u\forall v((\varphi_i(u)\&\varphi_i(v))\supset u \overset{\pm}{=}_{\mathcal{Q}'} v)} \qquad \overset{\mathcal{D}_3}{\forall w(\varphi_i(w)\supset\varphi_j(w))}}{\varphi_j(\iota_{\mathcal{Q}'}x\varphi_i(x))}\;(\iota_{\mathcal{Q}'}\mathrm{I}) \tag{5.3}$$

The premisses of the $\iota_{\mathcal{Q}'}$I-application say that there is at least one thing which is φ_i, that any two things which are φ_i are the same (only) in any $\mathcal{Q}'$-respect, and that everything that is φ_i is φ_j. The conclusion $\varphi_j(\iota_{\mathcal{Q}'}x\varphi_i(x))$ can be read: 'the $\mathcal{Q}'$-unique x which is φ_i is φ_j', or simply 'the φ_i is φ_j'. We may use these readings only in case there are at least two things that are φ_i which are discernible with respect to $\mathcal{P}\setminus\mathcal{Q}'$. It will be natural to use this restricted kind of definiteness for the analysis of (1) in times of schism.

Case (iii). Minimal qualified definiteness:

$$\frac{\overset{\mathcal{D}_1}{\exists x\varphi_i(x)} \qquad \overset{\mathcal{D}_{2(iii)}}{\forall u\forall v((\varphi_i(u)\&\varphi_i(v))\supset u \overset{\pm}{=}_{\{\varphi_i\}} v)} \qquad \overset{\mathcal{D}_3}{\forall w(\varphi_i(w)\supset\varphi_j(w))}}{\varphi_j(\iota_{\{\varphi_i\}}x\varphi_i(x))}\;(\iota_{\{\varphi_i\}}\mathrm{I}) \tag{5.4}$$

The premisses of the $\iota_{\{\varphi_i\}}$I-application say that there is at least one thing which is φ_i, that any two things which are φ_i are the same only with respect to $\{\varphi_i\}$, and that everything that is φ_i is φ_j. The conclusion $\varphi_j(\iota_{\{\varphi_i\}}x\varphi_i(x))$ can be read: 'the $\{\varphi_i\}$-unique x which is φ_i is φ_j'. We may use this reading only in case there are at least two things that are φ_i which are discernible with respect to at least one predicate in $\mathcal{P}\setminus\{\varphi_i\}$. In a sense, this minimal degree of definiteness (a special case of restricted definiteness; cf. Definition 2.9(2)) comes close to generic definiteness: 'the generic φ_i is φ_j'. Similarly for negative qualified definiteness.

Example 5.3. It is straightforward to construct canonical derivations for simple constructions such as those given below. For reasons of convenience, the predicates in the symbolizations are written out.

(1) The pope is bald. $Bald(\iota_{\mathcal{P}}xPope(x))$

(3) The king of France is not real. $-Real(\iota_{\mathcal{P}}x(King\text{-}of^2(x, France)))$

(4) The bishop is bald. $Bald(\iota_{\mathcal{Q}'}xBishop(x))$

(5) The Englishman is brave. $Brave(\iota_{\{Englishman\}}x(Englishman(x))$

(6) The non-smoker is healthy. $Healthy(\iota_{\{Smoker\}}x(-Smoker(x)))$

Concerning (3): cf. Remark 4.4. The symbolization of (6) in terms of negative predication is not entirely direct. The use of subatomic negation (cf. [23]) should be more adequate here.

6. Generalizations

Building on [3, 4], we now generalize the $\mathbf{I0}(\mathcal{S}_b^=)\iota$-systems for qualified definiteness described above ($\iota_{\mathcal{Q}}$*-systems*, for short) in order to obtain proof systems which are suitable for the analysis of constructions such as, e.g., (2) and:

(7) The dog descends from the wolf. (Cf. [12, (33)].)

(8) The pope puts the zucchetto on the zucchetto. (Cf. [12, (38)].)

(9) The king of the jungle loves the queen of the desert.

(10) Leo XIV is the bishop of Rome. (Cf. [12, (61)].)

(11) The rabbit in the box looks at the rabbit in the hat. (Cf. [8, p. 661].)

(12) The man wearing the beret with the button is French. ([10, p. 450].)

(13) The man wearing the beret and carrying the newspaper is French. ([10, p. 451].)

(14) The man wearing the beret and carrying the newspaper walks his dog.

We proceed in three generalization steps.

6.1. Generalization A: Parallel qualified definiteness

First, we turn the rules for parallel definiteness from [3, 4] into rules for parallel qualified definiteness. These rules will allow us to analyse constructions such as (2), (7), and (8).

Definition 6.1. *Contextual definitions ($\mathcal{L}\iota^A$).* We write $\varphi(x_1, ..., x_m)$, suppressing the arity of φ, for atomic formulae $\varphi^n o_1...o_n$ containing (possibly multiple occurrences of) x_i, where $i \in \{1, ..., m\}$. We generalize the contextual definitions for qualified definiteness (Definition 2.6) by replacing them with contextual definitions for parallel qualified definiteness. Let $\mathcal{Q}_1, ..., \mathcal{Q}_n \subseteq \mathcal{P}$.

1. *Parallel positive qualified definiteness*:
   $\psi(\iota_{\mathcal{Q}_1} x_1 \varphi_1(x_1), ..., \iota_{\mathcal{Q}_n} x_n \varphi_n(x_n)) =_{def}$

$$\begin{array}{l}(\exists x_1 \varphi_1(x_1) \ \& \ ... \ \& \ \exists x_n \varphi_n(x_n)) \ \& \\ (\forall u_1 \forall v_1((\varphi_1(u_1) \& \varphi_1(v_1)) \supset u_1 \overset{\pm}{=}_{\mathcal{Q}_1} v_1) \ \& \ ... \ \& \\ \forall u_n \forall v_n((\varphi_n(u_n) \& \varphi_n(v_n)) \supset u_n \overset{\pm}{=}_{\mathcal{Q}_n} v_n)) \ \& \\ (\forall w_1 ... \forall w_n((\varphi_1(w_1) \ \& \ ... \ \& \ \varphi_n(w_n)) \supset \psi(w_1, ..., w_n)))\end{array}$$

2. *Parallel negative qualified definiteness*:
   $\psi(\iota_{\mathcal{Q}_1} x_1 - \varphi_1(x_1), ..., \iota_{\mathcal{Q}_n} x_n - \varphi_n(x_n)) =_{def}$

$$\begin{array}{l}(\exists x_1 - \varphi_1(x_1) \ \& \ ... \ \& \ \exists x_n - \varphi_n(x_n)) \ \& \\ (\forall u_1 \forall v_1((-\varphi_1(u_1) \& - \varphi_1(v_1)) \supset u_1 \overset{-}{=}_{\mathcal{Q}_1} v_1) \ \& \ ... \ \& \\ \forall u_n \forall v_n((-\varphi_n(u_n) \& - \varphi_n(v_n)) \supset u_n \overset{-}{=}_{\mathcal{Q}_n} v_n)) \ \& \\ (\forall w_1 ... \forall w_n((-\varphi_1(w_1) \ \& \ ... \ \& \ -\varphi_n(w_n)) \supset \psi(w_1, ..., w_n)))\end{array}$$

(Likewise with $-\psi$.)

Example 6.2. Symbolizations:

(2) The pope blesses the bishop.
$Blesses^2(\iota_{\mathcal{P}}x(Pope(x)), \iota_{\mathcal{Q}}y(Bishop(y)))$

(7) The dog descends from the wolf. (Cf. [12, (33)].)
$Descends\text{-}from^2(\iota_{\{Dog\}}x(Dog(x)), \iota_{\{Wolf\}}y(Wolf(y)))$

(8) The pope puts the zucchetto on the zucchetto. (Cf. [12, (38)].)
$Puts\text{-}on^3(\iota_{\mathcal{P}}x(Pope(x)), \iota_{\mathcal{Q}'}y(Zucchetto(y)), \iota_{\mathcal{Q}''}z(Zucchetto(z)))$

Next, we generalize $\iota_{\mathcal{Q}}$-systems by replacing the rules for qualified definiteness (Definition 3.14) with rules for parallel qualified definiteness. We call the resulting systems $\iota_{\mathcal{Q}}$*A-systems*. To present the generalized rules in a more compact form we make use of abbreviations.

Definition 6.3. *Abbreviations ($\iota_{\mathcal{Q}}$A-systems).* Let $\mathcal{Q}_k \subseteq \mathcal{P}$ with $k \in \{1, ..., n\}$.

1. *Abbreviations for parallel positive qualified definiteness*:
   (a) E_k: $\exists x_k \varphi_k(x_k)$
   (b) QU_k: $\forall u_k \forall v_k((\varphi_k(u_k) \& \varphi_k(v_k)) \supset u_k \overset{+}{=}_{\mathcal{Q}_k} v_k)$
   (c) P: $\forall w_1 ... \forall w_n((\varphi_1(w_1) \& ... \& \varphi_n(w_n)) \supset \psi(w_1, ..., w_n))$
2. *Abbreviations for parallel negative qualified definiteness*:
   (a) $-E_k$: $\exists x_k - \varphi_k(x_k)$
   (b) $-QU_k$: $\forall u_k \forall v_k((-\varphi_k(u_k) \& -\varphi_k(v_k)) \supset u_k \overset{-}{=}_{\mathcal{Q}_k} v_k)$
   (c) $-P$: $\forall w_1 ... \forall w_n((-\varphi_1(w_1) \& ... \& -\varphi_n(w_n)) \supset \psi(w_1, ..., w_n))$

   (Likewise with $-\psi$.)

Definition 6.4. *Derivations ($\iota_{\mathcal{Q}}$A-systems).* The following rules replace the rules for qualified definiteness in Definition 3.14.

1. *Rules for parallel positive qualified definiteness*:

$$\dfrac{\begin{array}{ccccc}\mathcal{D}_{1_1} & \mathcal{D}_{1_n} & \mathcal{D}_{2_1} & \mathcal{D}_{2_n} & \mathcal{D}_3 \\ E_1 & ... \; E_n & QU_1 & ... \; QU_n & P\end{array}}{\psi(\iota_{\mathcal{Q}_1} x_1 \varphi_1(x_1), ..., \iota_{\mathcal{Q}_n} x_n \varphi_n(x_n))}\,(\iota_{\mathcal{Q}}\mathrm{I}_i^{\mathrm{A}})$$

$$\dfrac{\begin{array}{c}\mathcal{D}_1 \\ \psi(\iota_{\mathcal{Q}_1} x_1 \varphi_1(x_1), ..., \iota_{\mathcal{Q}_n} x_n \varphi_n(x_n))\end{array}}{E_k}\,(\iota_{\mathcal{Q}}\mathrm{E}_k^{\mathrm{A}}1)$$

$$\dfrac{\begin{array}{c}\mathcal{D}_1 \\ \psi(\iota_{\mathcal{Q}_1} x_1 \varphi_1(x_1), ..., \iota_{\mathcal{Q}_n} x_n \varphi_n(x_n))\end{array}}{QU_k}\,(\iota_{\mathcal{Q}}\mathrm{E}_k^{\mathrm{A}}2)$$

$$\dfrac{\begin{array}{c}\mathcal{D}_1 \\ \psi(\iota_{\mathcal{Q}_1} x_1 \varphi_1(x_1), ..., \iota_{\mathcal{Q}_n} x_n \varphi_n(x_n))\end{array}}{P}\,(\iota_{\mathcal{Q}}\mathrm{E}_i^{\mathrm{A}}3)$$

where $i \in \{1, ..., n\}$ (arity of ψ), $k \in \{1, ..., n\}$

2. *Rules for parallel negative qualified definiteness*:

$$\dfrac{\begin{array}{ccccc}\mathcal{D}_{1_1} & \mathcal{D}_{1_n} & \mathcal{D}_{2_1} & \mathcal{D}_{2_n} & \mathcal{D}_3 \\ -E_1 & ... \; -E_n & -QU_1 & ... \; -QU_n & -P\end{array}}{\psi(\iota_{\mathcal{Q}_1} x_1 - \varphi_1(x_1), ..., \iota_{\mathcal{Q}_n} x_n - \varphi_n(x_n))}\,(\iota_{\mathcal{Q}}-\mathrm{I}_i^{\mathrm{A}})$$

$$\dfrac{\begin{array}{c}\mathcal{D}_1 \\ \psi(\iota_{\mathcal{Q}_1} x_1 - \varphi_1(x_1), ..., \iota_{\mathcal{Q}_n} x_n - \varphi_n(x_n))\end{array}}{-E_k}\,(\iota_{\mathcal{Q}}-\mathrm{E}_k^{\mathrm{A}}1)$$

$$\dfrac{\begin{array}{c}\mathcal{D}_1 \\ \psi(\iota_{\mathcal{Q}_1} x_1 - \varphi_1(x_1), ..., \iota_{\mathcal{Q}_n} x_n - \varphi_n(x_n))\end{array}}{-QU_k}\,(\iota_{\mathcal{Q}}-\mathrm{E}_k^{\mathrm{A}}2)$$

$$\dfrac{\begin{array}{c}\mathcal{D}_1 \\ \psi(\iota_{\mathcal{Q}_1} x_1 - \varphi_1(x_1), ..., \iota_{\mathcal{Q}_n} x_n - \varphi_n(x_n))\end{array}}{-P}\,(\iota_{\mathcal{Q}}-\mathrm{E}_i^{\mathrm{A}}3)$$

where $i \in \{1, ..., n\}$ (arity of ψ), $k \in \{1, ..., n\}$

(Likewise with $-\psi$.)

Definition 6.5 (Detour conversions ($\iota_{\mathcal{Q}}A$-systems)). Like Definition 3.16, except that the conversions for qualified definiteness are replaced by the following ones.

1. *Detour conversions for parallel positive qualified definiteness*:

$$\dfrac{\dfrac{\begin{array}{ccccccc}\mathcal{D}_{1_1} & & \mathcal{D}_{1_n} & \mathcal{D}_{2_1} & & \mathcal{D}_{2_n} & \mathcal{D}_3 \\ E_1 & \ldots & E_n & QU_1 & \ldots & QU_n & P\end{array}}{\psi(\iota_{\mathcal{Q}_1}x_1\varphi_1(x_1),\ldots,\iota_{\mathcal{Q}_n}x_n\varphi_n(x_n))}\,(\iota_{\mathcal{Q}}\mathrm{I}^{\mathrm{A}}_i)}{E_k}\,(\iota_{\mathcal{Q}}\mathrm{E}^{\mathrm{A}}_k1) \quad \text{conv} \quad \begin{array}{c}\mathcal{D}_{1_k}\\ E_k\end{array}$$

$$\dfrac{\dfrac{\begin{array}{ccccccc}\mathcal{D}_{1_1} & & \mathcal{D}_{1_n} & \mathcal{D}_{2_1} & & \mathcal{D}_{2_n} & \mathcal{D}_3 \\ E_1 & \ldots & E_n & QU_1 & \ldots & QU_n & P\end{array}}{\psi(\iota_{\mathcal{Q}_1}x_1\varphi_1(x_1),\ldots,\iota_{\mathcal{Q}_n}x_n\varphi_n(x_n))}\,(\iota_{\mathcal{Q}}\mathrm{I}^{\mathrm{A}}_i)}{QU_k}\,(\iota_{\mathcal{Q}}\mathrm{E}^{\mathrm{A}}_k2) \quad \text{conv} \quad \begin{array}{c}\mathcal{D}_{2_k}\\ QU_k\end{array}$$

$$\dfrac{\dfrac{\begin{array}{ccccccc}\mathcal{D}_{1_1} & & \mathcal{D}_{1_n} & \mathcal{D}_{2_1} & & \mathcal{D}_{2_n} & \mathcal{D}_3 \\ E_1 & \ldots & E_n & QU_1 & \ldots & QU_n & P\end{array}}{\psi(\iota_{\mathcal{Q}_1}x_1\varphi_1(x_1),\ldots,\iota_{\mathcal{Q}_n}x_n\varphi_n(x_n))}\,(\iota_{\mathcal{Q}}\mathrm{I}^{\mathrm{A}}_i)}{P}\,(\iota_{\mathcal{Q}}\mathrm{E}^{\mathrm{A}}_i3) \quad \text{conv} \quad \begin{array}{c}\mathcal{D}_3\\ P\end{array}$$

2. *Detour conversions for parallel negative qualified definiteness*: *mutatis mutandis.*

Remark 6.6. Normalization, the subexpression property, the subformula property, and internal completeness for $\iota_{\mathcal{Q}}A$-systems are obtained in a manner analogous to Theorem 3.18, Theorem 3.19, and the corollaries of the latter. Specifically, for normalization also the detour conversions for parallel qualified definiteness have to be used. These involve no complications. This makes it rather straightforward to establish the results. Meaning is then explained like in Definition 4.2.

Example 6.7. Consider (8). We use the following symbolization:

$$P^3(\iota_{\mathcal{P}}x(P^1(x)),\iota_{\mathcal{Q}'}y(Z^1(y)),\iota_{\mathcal{Q}''}z(Z^1(z)))$$

Let $\mathcal{Q}',\mathcal{Q}'' \subset \mathcal{P}$ such that $\mathcal{Q}' \neq \mathcal{Q}''$. For reasons of illustration, let the derivations for the E-clauses have the form of (3.1).

$$\begin{array}{ccc}\mathcal{D}_{1(E)} & \mathcal{D}_{2(E)} & \mathcal{D}_{3(E)} \\ \exists xP^1(x) & \exists yZ^1(y) & \exists zZ^1(z)\end{array} \tag{6.1}$$

Let the derivations for the QU-clauses have the form of (3.2).

$$\begin{array}{c} \mathcal{D}_{1(QU)} \\ \forall u_1 \forall v_1 ((P^1(u_1) \& P^1(v_1)) \supset u_1 \overset{\pm}{=}_{\mathcal{P}} v_1) \\ \mathcal{D}_{2(QU)} \\ \forall u_2 \forall v_2 ((Z^1(u_2) \& Z^1(v_2)) \supset u_2 \overset{\pm}{=}_{\mathcal{Q}'} v_2) \\ \mathcal{D}_{3(QU)} \\ \forall u_3 \forall v_3 ((Z^1(u_3) \& Z^1(v_3)) \supset u_3 \overset{\pm}{=}_{\mathcal{Q}''} v_3) \end{array} \tag{6.2}$$

The conclusion of $\mathcal{D}_{1(QU)}$ says that there is at most one $\mathcal{P}$-qualified pope, that of $\mathcal{D}_{2(QU)}$ says that there is at most one $\mathcal{Q}'$-qualified zucchetto, and that of $\mathcal{D}_{3(QU)}$ says that there is at most one $\mathcal{Q}''$-qualified zucchetto. Accordingly, given the derivations of the E-clauses, there is a unique pope, but there is more than one zucchetto. Finally, let the derivation for the P-clause be an adjustment of (3.3).

$$\begin{array}{c} \mathcal{D}_{(P)} \\ \forall w_1 \forall w_2 \forall w_3 ((P^1(w_1) \& Z^1(w_2) \& Z^1(w_3)) \supset P^3(w_1, w_2, w_3)) \end{array} \tag{6.3}$$

Let $\{\mathcal{D}_{(E)}\} = \{\mathcal{D}_{1(E)}, \mathcal{D}_{2(E)}, \mathcal{D}_{3(E)}\}$ and $\{\mathcal{D}_{(QU)}\} = \{\mathcal{D}_{1(QU)}, \mathcal{D}_{2(QU)}, \mathcal{D}_{3(QU)}\}$. We combine these derivations by means of the I-rule for parallel positive qualified definiteness into a canonical derivation:

$$\frac{\{\mathcal{D}_{(E)}\} \qquad \{\mathcal{D}_{(QU)}\} \qquad \mathcal{D}_{(P)}}{P^3(\iota_{\mathcal{P}} x(P^1(x)), \iota_{\mathcal{Q}'} y(Z^1(y)), \iota_{\mathcal{Q}''} z(Z^1(z)))} \ (\iota_{\mathcal{Q}} \mathrm{I}_3^{\mathrm{A}}) \tag{6.4}$$

6.2. Generalization B: Parallel nested qualified definiteness

In a second generalization step, we adapt the rules for nested definiteness defined in [3, 4] to parallel nested qualified definiteness. This will allow us to analyse constructions like (9) and (10).

Definition 6.8 (Contextual definitions ($\mathcal{L}\iota^{\mathrm{B}}$)). We generalize the definitions for parallel qualified definiteness in Definition 6.1 by replacing them with definitions for parallel nested qualified definiteness. Let $\mathcal{Q}_{1_1}, \ldots, \mathcal{Q}_{n_m} \subseteq \mathcal{P}$.

1. *Parallel nested positive qualified definiteness*:
   $\psi(\iota_{\mathcal{Q}_{n_1}} x_{n_1} \varphi_{n_1}(x_{n_1}, ..., \iota_{\mathcal{Q}_{2_1}} x_{2_1} \varphi_{2_1}(x_{2_1}, ..., \iota_{\mathcal{Q}_{1_1}} x_{1_1} \varphi_{1_1}(x_{1_1}))...), ...,$
   $\iota_{\mathcal{Q}_{n_m}} x_{n_m} \varphi_{n_m}(x_{n_m}, ..., \iota_{\mathcal{Q}_{2_m}} x_{2_m} \varphi_{2_m}(x_{2_m}, ..., \iota_{\mathcal{Q}_{1_m}} x_{1_m} \varphi_{1_m}(x_{1_m}))...)) =_{def}$

   $(\exists x_{n_1} \varphi_{n_1}(x_{n_1}, ..., \iota_{\mathcal{Q}_{2_1}} x_{2_1} \varphi_{2_1}(x_{2_1}, ..., \iota_{\mathcal{Q}_{1_1}} x_{1_1} \varphi_{1_1}(x_{1_1})))\&...\&$
   $\exists x_{n_m} \varphi_{n_m}(x_{n_m}, ..., \iota_{\mathcal{Q}_{2_m}} x_{2_m} \varphi_{2_m}(x_{2_m}, ..., \iota_{\mathcal{Q}_{1_m}} x_{1_m} \varphi_{1_m}(x_{1_m}))))\&$

   $(\forall u_{n_1} \forall v_{n_1}((\varphi_{n_1}(u_{n_1}, ..., \iota_{\mathcal{Q}_{2_1}} x_{2_1} \varphi_{2_1}(x_{2_1}, ..., \iota_{\mathcal{Q}_{1_1}} x_{1_1} \varphi_{1_1}(x_{1_1})))\&$
   $\varphi_{n_1}(v_{n_1}, ..., \iota_{\mathcal{Q}_{2_1}} x_{2_1} \varphi_{2_1}(x_{2_1}, ..., \iota_{\mathcal{Q}_{1_1}} x_{1_1} \varphi_{n}(x_{1_1})))) \supset u_{n_1} \overset{+}{=}_{\mathcal{Q}_{n_1}} v_{n_1})\&...\&$
   $\forall u_{n_m} \forall v_{n_m}((\varphi_{n_m}(u_{n_m}, ..., \iota_{\mathcal{Q}_{2_m}} x_{2_m} \varphi_{2_m}(x_{2_m}, ..., \iota_{\mathcal{Q}_{1_m}} x_{1_m} \varphi_{1_m}(x_{1_m})))\&$
   $\varphi_{n_m}(v_{n_m}, ..., \iota_{\mathcal{Q}_{2_m}} x_{2_m} \varphi_{2_m}(x_{2_m}, ..., \iota_{\mathcal{Q}_{1_m}} x_{1_m} \varphi_{1_m}(x_{1_m}))))$
   $\supset u_{n_m} \overset{+}{=}_{\mathcal{Q}_{n_m}} v_{n_m}))\&$

   $(\forall w_{n_1} ... \forall w_{n_m}(\varphi_{n_1}(w_{n_1}, ..., \iota_{\mathcal{Q}_{2_1}} x_{2_1} \varphi_{2_1}(x_{2_1}, ..., \iota_{\mathcal{Q}_{1_1}} x_{1_1} \varphi_{1_1}(x_{1_1})))\&...\&$
   $\varphi_{n_m}(w_{n_m}, ..., \iota_{\mathcal{Q}_{2_m}} x_{2_m} \varphi_{2_m}(x_{2_m}, ..., \iota_{\mathcal{Q}_{1_m}} x_{1_m} \varphi_{1_m}(x_{1_m}))))$
   $\supset \psi(w_{n_1}, ..., w_{n_m})))$

2. *Parallel nested negative qualified definiteness*:
   $\psi(\iota_{\mathcal{Q}_{n_1}} x_{n_1} - \varphi_{n_1}(x_{n_1}, ..., \iota_{\mathcal{Q}_{2_1}} x_{2_1} - \varphi_{2_1}(x_{2_1}, ..., \iota_{\mathcal{Q}_{1_1}} x_{1_1} - \varphi_{1_1}(x_{1_1}))...), ...,$
   $\iota_{\mathcal{Q}_{n_m}} x_{n_m} - \varphi_{n_m}(x_{n_m}, ..., \iota_{\mathcal{Q}_{2_m}} x_{2_m} - \varphi_{2_m}(x_{2_m}, ..., \iota_{\mathcal{Q}_{1_m}} x_{1_m} - \varphi_{1_m}(x_{1_m}))...))$
   $=_{def}$

   $(\exists x_{n_1} - \varphi_{n_1}(x_{n_1}, ..., \iota_{\mathcal{Q}_{2_1}} x_{2_1} - \varphi_{2_1}(x_{2_1}, ..., \iota_{\mathcal{Q}_{1_1}} x_{1_1} - \varphi_{1_1}(x_{1_1})))\&...\&$
   $\exists x_{n_m} - \varphi_{n_m}(x_{n_m}, ..., \iota_{\mathcal{Q}_{2_m}} x_{2_m} - \varphi_{2_m}(x_{2_m}, ..., \iota_{\mathcal{Q}_{1_m}} x_{1_m} - \varphi_{1_m}(x_{1_m}))))\&$

   $(\forall u_{n_1} \forall v_{n_1}((-\varphi_{n_1}(u_{n_1}, ..., \iota_{\mathcal{Q}_{2_1}} x_{2_1} - \varphi_{2_1}(x_{2_1}, ..., \iota_{\mathcal{Q}_{1_1}} x_{1_1} - \varphi_{1_1}(x_{1_1})))\&$
   $-\varphi_{n_1}(v_{n_1}, ..., \iota_{\mathcal{Q}_{2_1}} x_{2_1} - \varphi_{2_1}(x_{2_1}, ..., \iota_{\mathcal{Q}_{1_1}} x_{1_1} - \varphi_{n}(x_{1_1})))) \supset u_{n_1} \overset{-}{=}_{\mathcal{Q}_{n_1}} v_{n_1})\&...\&$
   $\forall u_{n_m} \forall v_{n_m}((-\varphi_{n_m}(u_{n_m}, ..., \iota_{\mathcal{Q}_{2_m}} x_{2_m} - \varphi_{2_m}(x_{2_m}, ..., \iota_{\mathcal{Q}_{1_m}} x_{1_m} - \varphi_{1_m}(x_{1_m})))\&$
   $- \varphi_{n_m}(v_{n_m}, ..., \iota_{\mathcal{Q}_{2_m}} x_{2_m} - \varphi_{2_m}(x_{2_m}, ..., \iota_{\mathcal{Q}_{1_m}} x_{1_m} - \varphi_{1_m}(x_{1_m}))))$
   $\supset u_{n_m} \overset{-}{=}_{\mathcal{Q}_{n_m}} v_{n_m}))\&$

   $(\forall w_{n_1} ... \forall w_{n_m}(-\varphi_{n_1}(w_{n_1}, ..., \iota_{\mathcal{Q}_{2_1}} x_{2_1} - \varphi_{2_1}(x_{2_1}, ..., \iota_{\mathcal{Q}_{1_1}} x_{1_1} - \varphi_{1_1}(x_{1_1})))\&...\&$
   $- \varphi_{n_m}(w_{n_m}, ..., \iota_{\mathcal{Q}_{2_m}} x_{2_m} - \varphi_{2_m}(x_{2_m}, ..., \iota_{\mathcal{Q}_{1_m}} x_{1_m} - \varphi_{1_m}(x_{1_m}))))$
   $\supset \psi(w_{n_1}, ..., w_{n_m})))$

(Likewise with $-\psi$.)

Example 6.9. Symbolizations:

(9) The king of the jungle loves the queen of the desert.
$Loves^2(\iota_{\mathcal{Q}_{2_1}} x(King\text{-}of^2(x, \iota_{\mathcal{Q}_{1_1}} y(Jungle(y)))),$
$\iota_{\mathcal{Q}_{2_2}} z(Queen\text{-}of^2(z, \iota_{\mathcal{Q}_{1_2}} u(Desert(u)))))$

(10) Leo XIV is the bishop of Rome.
$Holds^2(LeoXIV, \iota_{\mathcal{P}} x(Office\text{-}of^2(x, \iota_{\mathcal{P}} y(Bishop\text{-}of^2(y, Rome)))))$

The symbolization of (10) does apparently not rest on an interpretation of that sentence which takes ‘is’ to express identity (cf. [16, p. 483]). It rests on an interpretation which takes the description in (10) to be predicative. However, it does not construe that description as a predicate (cf. [7]; for overview see [12, sect. 7.1]). Rather it takes the ‘is’ to indicate a relational predicate that needs to be specified.

In order to obtain $\iota_{\mathcal{Q}}B$*-systems*, we generalize $\iota_{\mathcal{Q}}A$-systems by replacing the rules for parallel qualified definiteness (Definition 6.4) with rules for parallel nested qualified definiteness.

Definition 6.10 (Abbreviations ($\iota_{\mathcal{Q}}B$-systems)).

1. *Abbreviations for parallel nested positive qualified definiteness*:
   (a) *E-abbreviations*:
   $\{E1_k\}$:
   $$\underbrace{\exists x1_1 \varphi1_1(x1_1)}_{E1_1}, ..., \underbrace{\exists x1_m \varphi1_m(x1_m)}_{E1_m}$$
   $\{E2_k\}$:
   $$\underbrace{\exists x2_1 \varphi2_1(x2_1, ..., \iota_{\mathcal{Q}1_1} x1_1 \varphi1_1(x1_1))}_{E2_1}, ...,$$
   $$\underbrace{\exists x2_m \varphi2_m(x2_m, ..., \iota_{\mathcal{Q}1_m} x1_m \varphi1(x1_m))}_{E2_m}$$
   $$\vdots$$
   $\{E{n_k}\}$:

$$\underbrace{\exists x_{n_1} \varphi_{n_1}(x_{n_1}, ..., \iota_{\mathcal{Q}_{2_1}} x_{2_1} \varphi_{2_1}(x_{2_1}, ..., \iota_{\mathcal{Q}_{1_1}} x_{1_1} \varphi_{1_1}(x_{1_1})))}_{E_{n_1}}, ...,$$

$$\underbrace{\exists x_{n_m} \varphi_{n_m}(x_{n_m}, ..., \iota_{\mathcal{Q}_{2_m}} x_{2_m} \varphi_{2_m}(x_{2_m}, ..., \iota_{\mathcal{Q}_{1_m}} x_{1_m} \varphi_{1_m}(x_{1_m})))}_{E_{n_m}}$$

(b) *QU-abbreviations*: see Figure 1.

(c) *P-abbreviations*:

P_1:

$$\forall w_{1_1} ... \forall w_{1_m}((\varphi_{1_1}(w_{1_1}) \& ... \& \varphi_{1_m}(w_{1_m})) \supset \psi_1(w_{1_1}, ..., w_{1_m}))$$

P_2:

$$\forall w_{2_1} ... \forall w_{2_m}((\varphi_{2_1}(w_{2_1}, ..., \iota_{\mathcal{Q}_{1_1}} x_{1_1} \varphi_{1_1}(x_{1_1})) \& ... \&$$
$$\varphi_{2_m}(w_{2_m}, ..., \iota_{\mathcal{Q}_{1_m}} x_{1_m} \varphi_{1_m}(x_{1_m}))) \supset \psi_2(w_{2_1}, ..., w_{2_m}))$$

$$\vdots$$

P_n:

$$\forall w_{n_1} ... \forall w_{n_m}((\varphi_{n_1}(w_{n_1}, ..., \iota_{\mathcal{Q}_{2_1}} x_{2_1} \varphi_{2_1}(w_{2_1}, ..., \iota_{\mathcal{Q}_{1_1}} x_{1_1} \varphi_{1_1}(x_{1_1}))) \& ... \&$$
$$\varphi_{n_m}(w_{n_m}, ..., \iota_{\mathcal{Q}_{2_m}} x_{2_m} \varphi_{2_m}(w_{2_m}, ..., \iota_{\mathcal{Q}_{1_m}} x_{1_m} \varphi_{1_m}(x_{1_m}))))$$
$$\supset \psi_n(w_{n_1}, ... w_{n_m}))$$

(Likewise with $-\psi_j$.)

(d) *QD-abbreviations*:

QD_1:

$$\psi_1(\iota_{\mathcal{Q}_{1_1}} x_{1_1} \varphi_{1_1}(x_{1_1}), ..., \iota_{\mathcal{Q}_{1_m}} x_{1_m} \varphi_{1_m}(x_{1_m}))$$

QD_2:

$$\psi_2(\iota_{\mathcal{Q}_{2_1}} x_{2_1} \varphi_{2_1}(x_{2_1}, ..., \iota_{\mathcal{Q}_{1_1}} x_{1_1} \varphi_{1_1}(x_{1_1})), ...,$$
$$\iota_{\mathcal{Q}_{2_m}} x_{2_m} \varphi_{2_m}(x_{2_m}, ..., \iota_{\mathcal{Q}_{1_m}} x_{1_m} \varphi_1(x_{1_m})))$$

$$\vdots$$

QD_n:

$$\psi_n(\iota_{\mathcal{Q}_{n_1}} x_{n_1} \varphi_{n_1}(x_{n_1}, ..., \iota_{\mathcal{Q}_{2_1}} x_{2_1} \varphi_{2_1}(x_{2_1}, ..., \iota_{\mathcal{Q}_{1_1}} x_{1_1} \varphi_{1_1}(x_{1_1}))...), ...,$$
$$\iota_{\mathcal{Q}_{n_m}} x_{n_m} \varphi_n(x_{n_m}, ..., \iota_{\mathcal{Q}_{2_m}} x_{2_m} \varphi_{2_m}(x_{2_m}, ..., \iota_{\mathcal{Q}_{1_m}} x_{1_m} \varphi_1(x_{1_m}))...))$$

(Likewise with $-\psi_j$.)

2. *Abbreviations for parallel nested negative qualified definiteness*: *mutatis mutandis.*

$\{QU_{1_k}\}$:

$$\underbrace{\forall u_{1_1}\forall v_{1_1}((\varphi_{1_1}(u_{1_1})\&\varphi_{1_1}(v_{1_1}))\supset u_{1_1}\overset{\pm}{=}_{\mathcal{Q}_{1_1}}\ v_{1_1})}_{QU_{1_1}},...,\underbrace{\forall u_{1_m}\forall v_{1_m}((\varphi_{1_m}(u_{1_m})\&\varphi_{1_m}(v_{1_m}))\supset u_{1_m}\overset{\pm}{=}_{\mathcal{Q}_{1_m}}\ v_{1_m})}_{QU_{1_m}}$$

$\{QU_{2_k}\}$:

$$\underbrace{\forall u_{2_1}\forall v_{2_1}((\varphi_{2_1}(u_{2_1},...,\iota_{\mathcal{Q}_{1_1}}x_{1_1}\varphi_{1_1}(x_{1_1}))\&\varphi_{2_1}(v_{2_1},...,\iota_{\mathcal{Q}_{1_1}}x_{1_1}\varphi_{1_1}(x_{1_1}))) \supset u_{2_1}\ \overset{\pm}{=}_{\mathcal{Q}_{2_1}}\ v_{2_1})}_{QU_{2_1}}\ ,...,$$

$$\underbrace{\forall u_{2_m}\forall v_{2_m}((\varphi_{2_m}(u_{2_m},...,\iota_{\mathcal{Q}_{1_m}}x_{1_m}\varphi_{1_m}(x_{1_m}))\&\varphi_{2_m}(v_{2_m},...,\iota_{\mathcal{Q}_{1_m}}x_{1_m}\varphi_{1_m}(x_{1_m}))) \supset u_{2_m}\ \overset{\pm}{=}_{\mathcal{Q}_{2_m}}\ v_{2_m})}_{QU_{2_m}}$$

$\vdots$

$\{QU_{n_k}\}$:

$$\underbrace{\forall u_{n_1}\forall v_{n_1}((\varphi_{n_1}(u_{n_1},...,\iota_{\mathcal{Q}_{2_1}}x_{2_1}\varphi_{2_1}(x_{2_1},...,\iota_{\mathcal{Q}_{1_1}}x_{1_1}\varphi_{1_1}(x_{1_1})))\&\varphi_{n_1}(v_{n_1},...,\iota_{\mathcal{Q}_{2_1}}x_{2_1}\varphi_{2_1}(x_{2_1},...,}_{QU_{n_1}}\cdots$$

$$\underbrace{\cdots\ \iota_{\mathcal{Q}_{1_1}}x_{1_1}\varphi_{1_1}(x_{1_1})))) \supset u_{n_1}\ \overset{\pm}{=}_{\mathcal{Q}_{n_1}}\ v_{n_1})},...,\underbrace{\forall u_{n_m}\forall v_{n_m}((\varphi_{n_m}(u_{n_m},...,\iota_{\mathcal{Q}}x_{2_m}\varphi_{2_m}(x_{2_m},...,}_{QU_{n_m}}\cdots$$

$$\underbrace{\cdots\ \iota_{\mathcal{Q}_{1_m}}x_{1_m}\varphi_{1_m}(x_{1_m})))\&\varphi_{n_m}(v_{n_m},...,\iota_{\mathcal{Q}_{2_m}}x_{2_m}\varphi_{2_m}(x_{2_m},...,\iota_{\mathcal{Q}_{1_m}}x_{1_m}\varphi_{1_m}(x_{1_m})))) \supset u_{n_m}\ \overset{\pm}{=}_{\mathcal{Q}_{n_m}}\ v_{n_m})}$$

Figure 1 : QU-Abbreviations (Generalization B)

Definition 6.11 (Derivations ($\iota_{\mathcal{Q}}B$-systems)). The following rules replace the rules for parallel qualified definiteness in Definition 6.4.

1. *Rules for parallel nested positive qualified definiteness*:

$$\dfrac{\begin{matrix}\{\mathcal{D}1_1\} & \{\mathcal{D}2_1\} & \mathcal{D}3_1\\ \{E1_k\} & \{QU1_k\} & P_1\end{matrix}}{QD_1}\,(\iota_{\mathcal{Q}}\mathrm{I}^{\mathrm{B}}_{i,1})$$

$$\vdots$$

$$\dfrac{\begin{matrix}\{\mathcal{D}1_n\} & \qquad\qquad\qquad\qquad & \{\mathcal{D}2_n\} & \mathcal{D}3_n\\ \{E_{n_k}\} & & \{QU_{n_k}\} & P_n\end{matrix}}{QD_n}\,(\iota_{\mathcal{Q}}\mathrm{I}^{\mathrm{B}}_{i,n})$$

$$\dfrac{\begin{matrix}\mathcal{D}1\\ QD_j\end{matrix}}{E_{j_k}}\,(\iota_{\mathcal{Q}}\mathrm{E}^{\mathrm{B}}_{k,j}1) \qquad \dfrac{\begin{matrix}\mathcal{D}1\\ QD_j\end{matrix}}{QU_{j_k}}\,(\iota_{\mathcal{Q}}\mathrm{E}^{\mathrm{B}}_{k,j}2) \qquad \dfrac{\begin{matrix}\mathcal{D}1\\ QD_j\end{matrix}}{P_j}\,(\iota_{\mathcal{Q}}\mathrm{E}^{\mathrm{B}}_{i,j}3)$$

where $i \in \{1, ..., m\}$ (arity of predicate in QD_j), $j \in \{1, ..., n\}$ (level of nesting), $k \in \{1, ..., m\}$

2. *Rules for parallel nested negative qualified definiteness*:

$$\dfrac{\begin{matrix}\{\mathcal{D}1_1\} & \{\mathcal{D}2_1\} & \mathcal{D}3_1\\ \{-E1_k\} & \{-QU1_k\} & -P_1\end{matrix}}{-QD_1}\,(\iota_{\mathcal{Q}-}\mathrm{I}^{\mathrm{B}}_{i,1})$$

$$\vdots$$

$$\dfrac{\begin{matrix}\{\mathcal{D}1_n\} & \qquad\qquad\qquad\qquad & \{\mathcal{D}2_n\} & \mathcal{D}3_n\\ \{-E_{n_k}\} & & \{-QU_{n_k}\} & -P_n\end{matrix}}{-QD_n}\,(\iota_{\mathcal{Q}-}\mathrm{I}^{\mathrm{B}}_{i,n})$$

$$\dfrac{\begin{matrix}\mathcal{D}1\\ -QD_j\end{matrix}}{-E_{j_k}}\,(\iota_{\mathcal{Q}-}\mathrm{E}^{\mathrm{B}}_{k,j}1) \qquad \dfrac{\begin{matrix}\mathcal{D}1\\ -QD_j\end{matrix}}{-QU_{j_k}}\,(\iota_{\mathcal{Q}-}\mathrm{E}^{\mathrm{B}}_{k,j}2) \qquad \dfrac{\begin{matrix}\mathcal{D}1\\ -QD_j\end{matrix}}{-P_j}\,(\iota_{\mathcal{Q}-}\mathrm{E}^{\mathrm{B}}_{i,j}3)$$

where $i \in \{1, ..., m\}$ (arity of predicate in QD_j), $j \in \{1, ..., n\}$ (level of nesting), $k \in \{1, ..., m\}$

(Likewise with $-\psi_j$.)

Definition 6.12 (Detour conversions ($\iota_Q B$-systems)).

1. *Detour conversions for parallel nested positive qualified definiteness*:

$$\begin{array}{c}
\dfrac{\begin{matrix}\{\mathcal{D}1_1\} & \{\mathcal{D}2_1\} & \mathcal{D}3_1\\ \{E1_k\} & \{QU1_k\} & P_1\end{matrix}}{QD_1}\,(\iota_Q\mathrm{I}^{\mathrm{B}}_{i,1})\\
\vdots\\
\dfrac{\dfrac{\begin{matrix}\{\mathcal{D}1_n\} & \qquad\qquad\qquad & \{\mathcal{D}2_n\} & \mathcal{D}3_n\\ \{E{n_k}\} & & \{QU{n_k}\} & P_n\end{matrix}}{QD_n}\,(\iota_Q\mathrm{I}^{\mathrm{B}}_{i,n})}{E_{n_k}}\,(\iota_Q\mathrm{E}^{\mathrm{B}}_{k,n}1)
\end{array}$$

conv

$$\begin{array}{c}
\dfrac{\begin{matrix}\{\mathcal{D}1_1\} & \{\mathcal{D}2_1\} & \mathcal{D}3_1\\ \{E1_k\} & \{QU1_k\} & P_1\end{matrix}}{QD_1}\,(\iota_Q\mathrm{I}^{\mathrm{B}}_{i,1})\\
\vdots\\
\mathcal{D}1_n\\
E_{n_k}
\end{array}$$

$$\begin{array}{c}
\dfrac{\begin{matrix}\{\mathcal{D}1_1\} & \{\mathcal{D}2_1\} & \mathcal{D}3_1\\ \{E1_k\} & \{QU1_k\} & P_1\end{matrix}}{QD_1}\,(\iota_Q\mathrm{I}^{\mathrm{B}}_{i,1})\\
\vdots\\
\dfrac{\dfrac{\begin{matrix}\{\mathcal{D}1_n\} & \qquad\qquad\qquad & \{\mathcal{D}2_n\} & \mathcal{D}3_n\\ \{E{n_k}\} & & \{QU{n_k}\} & P_n\end{matrix}}{QD_n}\,(\iota_Q\mathrm{I}^{\mathrm{B}}_{i,n})}{QU_{n_k}}\,(\iota_Q\mathrm{E}^{\mathrm{B}}_{k,n}2)
\end{array}
\quad \text{conv} \quad
\begin{array}{c}\mathcal{D}2_n\\ QU_{n_k}\end{array}$$

$$\cfrac{\cfrac{\begin{array}{c}\cfrac{\begin{array}{ccc}\{\mathcal{D}_{1_1}\} & \{\mathcal{D}_{2_1}\} & \mathcal{D}_{3_1}\\ \{E_{1_k}\} & \{QU_{1_k}\} & P_1\end{array}}{QD_1}\,(\iota_{\mathcal{Q}}\mathrm{I}^{\mathrm{B}}_{i,1})\\ \vdots\\ \begin{array}{ccc}\{\mathcal{D}_{1_n}\} & \{\mathcal{D}_{2_n}\} & \mathcal{D}_{3_n}\\ \{E_{n_k}\} & \{QU_{n_k}\} & P_n\end{array}\end{array}}{QD_n}\,(\iota_{\mathcal{Q}}\mathrm{I}^{\mathrm{B}}_{i,n})}{P_n}\,(\iota_{\mathcal{Q}}\mathrm{E}^{\mathrm{B}}_{i,n}3) \qquad \text{conv} \qquad \begin{array}{c}\mathcal{D}_{3_n}\\ P_n\end{array}$$

2. *Detour conversions for parallel nested negative qualified definiteness: mutatis mutandis.*

Remark 6.13. A remark analogous to Remark 6.6 applies to $\iota_{\mathcal{Q}}B$-systems.

Example 6.14. We construct a canonical derivation for (9). We use the following symbolization:

$$L^2(\iota_{\mathcal{Q}_{2_1}}x(K^2(x,\iota_{\mathcal{Q}_{1_1}}y(J^1(y)))),\iota_{\mathcal{Q}_{2_2}}z(Q^2(z,\iota_{\mathcal{Q}_{1_2}}u(D^1(u)))))$$

Let $\mathcal{Q}_{1_1}, \mathcal{Q}_{1_2}, \mathcal{Q}_{2_1}, \mathcal{Q}_{2_2} \subseteq \mathcal{P}$.

$$\begin{array}{ccc}\mathcal{D}_{1(E)} & \mathcal{D}_{2(QU)} & \mathcal{D}_{3(P)}\\ \exists y J^1(y) & \forall u_1\forall v_1((J^1(u_1)\&J^1(v_1))\supset u_1 \overset{\pm}{=}_{\mathcal{Q}_{1_1}} v_1) & \forall w_1(J^1(w_1)\supset K^2(\alpha_1,w_1))\end{array}$$

$$\begin{array}{ccc}\mathcal{D}'_{1(E)} & \mathcal{D}'_{2(QU)} & \mathcal{D}'_{3(P)}\\ \exists u D^1(u) & \forall u'_1\forall v'_1((D^1(u'_1)\&D^1(v'_1))\supset u'_1 \overset{\pm}{=}_{\mathcal{Q}_{1_2}} v'_1) & \forall w'_1(D^1(w'_1)\supset Q^2(\alpha_2,w'_1))\end{array}$$

The first level of parallel nesting:

$$\mathcal{D}_{4(E)} = \cfrac{\cfrac{\mathcal{D}_{1(E)} \quad \mathcal{D}_{2(QU)} \quad \mathcal{D}_{3(P)}}{K^2(\alpha_1,\iota_{\mathcal{Q}_{1_1}}y(J^1(y)))}\,(\iota_{\mathcal{Q}}\mathrm{I}^{\mathrm{B}}_{2,1})}{\exists x(K^2(x,\iota_{\mathcal{Q}_{1_1}}y(J^1(y))))}$$

$$\begin{array}{c}\mathcal{D}_{5(QU)}\\ \forall u_2\forall v_2((K^2(u_2,\iota_{\mathcal{Q}_{1_1}}y(J^1(y)))\&K^2(v_2,\iota_{\mathcal{Q}_{1_1}}y(J^1(y))))\supset u_2 \overset{\pm}{=}_{\mathcal{Q}_{2_1}} v_2)\end{array}$$

$$\mathcal{D}'_{4(E)} = \cfrac{\cfrac{\mathcal{D}'_{1(E)} \qquad \mathcal{D}'_{2(QU)} \qquad \mathcal{D}'_{3(P)}}{Q^2(\alpha_2, \iota_{\mathcal{Q}_{1_2}} u(D^1(u)))}\,(\iota_{\mathcal{Q}}\mathrm{I}^{\mathrm{B}}_{2,1})}{\exists z(Q^2(z, \iota_{\mathcal{Q}_{1_2}} u(D^1(u))))}$$

$$\begin{array}{c} \mathcal{D}'_{5(QU)} \\ \forall u'_2 \forall v'_2((Q^2(u'_2, \iota_{\mathcal{Q}_{1_2}} u(D^1(u)))\&Q^2(v'_2, \iota_{\mathcal{Q}_{1_2}} u(D^1(u)))) \supset u'_2 \overset{\pm}{=}_{\mathcal{Q}_{2_2}} v'_2) \end{array}$$

The second level of parallel nesting:

$$\{\mathcal{D}_{(E)}\} = \{\mathcal{D}_{4(E)}, \mathcal{D}'_{4(E)}\}, \{\mathcal{D}_{(QU)}\} = \{\mathcal{D}_{5(QU)}, \mathcal{D}'_{5(QU)}\}$$

$$\begin{array}{c} \mathcal{D}_{6(P)} \\ \forall w_2 \forall w_3((K^2(w_2, \iota_{\mathcal{Q}_{1_1}} y(J^1(y)))\&(Q^2(w_3, \iota_{\mathcal{Q}_{1_2}} u(D^1(u)))) \supset L^2(w_2, w_3)) \end{array}$$

$$\frac{\{\mathcal{D}_{(E)}\} \qquad \{\mathcal{D}_{(QU)}\} \qquad \mathcal{D}_{6(P)}}{L^2(\iota_{\mathcal{Q}_{2_1}} x(K^2(x, \iota_{\mathcal{Q}_{1_1}} y(J^1(y)))), \iota_{\mathcal{Q}_{2_2}} z(Q^2(z, \iota_{\mathcal{Q}_{1_2}} u(D^1(u)))))}\,(\iota_{\mathcal{Q}}\mathrm{I}^{\mathrm{B}}_{2,2}) \qquad (6.5)$$

A canonical derivation for (10) can be constructed in a similar manner.

6.3. Generalization C: Parallel conjunctively nested qualified definiteness

Finally, we generalize the systems for parallel nested qualified definiteness so as to allow for conjunctions also in the scope of qualified definiteness operators. This will allow us to deal with constructions such as (11)–(14).

Definition 6.15. *CN-formulae*:

1. A *positive* CN-formula is either
   (a) an atomic formula,
   (b) a parallel QD-formula, or
   (c) a parallel QD-formula containing a conjunction of formulae of the form of (1a), (1b), (1c).

2. A *negative* CN-formula is either
    (a) an atomic negative predication,
    (b) a parallel negative QD-formula, or
    (c) a parallel negative QD-formula containing a conjunction of formulae of the form of (2a), (2b), (2c).

DEFINITION 6.16 (Contextual definitions ($\mathcal{L}\iota^{\mathrm{C}}$)). We generalize the definitions for parallel nested qualified definiteness in Definition 6.8 by replacing them with definitions for parallel conjunctively nested qualified definiteness. Let $\mathcal{Q}_{1_1}, \ldots, \mathcal{Q}_{n_m} \subseteq \mathcal{P}$ and let $C(x_{j_k})$ (resp. $-C(x_{j_k})$) be a CN-formula (negative CN-formula) for $j \in \{1, \ldots, n\}$ and $k \in \{1, \ldots, m\}$.

1. *Parallel conjunctively nested positive qualified definiteness*: like Definition 6.8(1), but with the occurrences of 'φ' replaced by 'C'.

2. *Parallel conjunctively nested negative qualified definiteness*: like Definition 6.8(2), but with the occurrences of '$-\varphi$' replaced by '$-C$'.

(Likewise with $-\psi$.)

Example 6.17. The symbolizations of the following examples make use of conjunction in the scope of the $\iota_{\mathcal{Q}}$-operators.

(11) The rabbit in the box looks at the rabbit in the hat.
$Looks\text{-}at^2(\iota_{\mathcal{Q}'}x(Rabbit(x)\ \&\ In^2(x, \iota_{\mathcal{Q}_1}y(Box(y)))),$
$\iota_{\mathcal{Q}''}z(Rabbit(z)\ \&\ In^2(z, \iota_{\mathcal{Q}_2}u(Hat(u)))))$

(12) The man wearing the beret with the button is French.
$French(\iota_{\mathcal{Q}_3}x(Man(x)\ \&\ Wears^2(x, \iota_{\mathcal{Q}_2}y(Beret(y)\ \&\ Has^2(y,$
$\iota_{\mathcal{Q}_1}z(Button(z)))))))$

(13) The man wearing the beret and carrying the newspaper is French.
$French(\iota_{\mathcal{Q}_2}x(Man(x) \& Wears^2(x, \iota_{\mathcal{Q}_1}y(Beret(y)))\ \&\ Carries^2(x,$
$\iota_{\mathcal{Q}'_1}z(Newspaper(z)))))$

(14) The man wearing the beret and carrying the newspaper walks his dog.

$$Walks^2(\iota_{\mathcal{Q}_2}x(Man(x)\&Wears^2(x,\iota_{\mathcal{Q}_1}y(Beret(y)))\ \&\ Carries^2(x, \iota_{\mathcal{Q}'_1}z(Newspaper(z)))),\iota_{\mathcal{Q}_3}u(Dog(u)\&Owns^2(\iota_{\mathcal{Q}_2}x(Man(x)\& (Wears^2(x,\iota_{\mathcal{Q}_1}y(Beret(y)))\&Carries^2(x,\iota_{\mathcal{Q}'_1}z(Newspaper(z)))),u)))$$

No conjunction surfaces in (11) and (12). Moreover, (11) and (14) are parallel also on the outermost level. The latter contains the possessive construction ‘his dog’, where the possessive pronoun ‘his’ stands in for a definite description. Examples (1)–(10) are special cases of parallel conjunctively nested qualified definiteness.

Remark 6.18. Note that (11), like (8), involves multiple uses of an incomplete description (i.e., ‘the rabbit’). If our symbolization of (11) were to reflect the relevant part of the scenario depicted by Haddock (i.e., there are three rabbits, two hats, and one box, where one rabbit is in a hat and one is in the box; cf. [8, p. 661]), we might consider using the symbolization $Looks\text{-}at^2(\iota_{\mathcal{Q}'_2}x(Rabbit(x)\ \&\ In^2(x,\iota_{\mathcal{Q}_1}y(Box(y)))),\iota_{\mathcal{Q}''_2}z(Rabbit(z)\ \&\ In^2(z,\iota_{\mathcal{Q}'_3}u(Hat(u)))))$ instead, thereby leaving room for the symbolization of a further use of ‘the rabbit’ and a further use of ‘the hat’.[3]

Definition 6.19. *Abbreviations ($\iota_{\mathcal{Q}}C$-systems).* The abbreviations are like those of $\iota_{\mathcal{Q}}B$-systems in Definition 6.10, except that the occurrences of φ (resp. $-\varphi$) are replaced by C ($-C$).

Definition 6.20. *Derivations ($\iota_{\mathcal{Q}}C$-systems).* The following rules replace the rules for parallel nested qualified definiteness in Definition 6.11.

1. *Rules for parallel conjunctively nested positive qualified definiteness*:

$$\dfrac{\begin{array}{ccc}\{\mathcal{D}_{1_1}\} & \{\mathcal{D}_{2_1}\} & \mathcal{D}_{3_1}\\ \{E_{1_k}\} & \{QU_{1_k}\} & P_1\end{array}}{QD_1}\,(\iota_{\mathcal{Q}}\mathrm{I}^{\mathrm{C}}_{i,1})$$

$$\vdots$$

$$\dfrac{\begin{array}{ccc}\{\mathcal{D}_{1_n}\} & \{\mathcal{D}_{2_n}\} & \mathcal{D}_{3_n}\\ \{E_{n_k}\} & \{QU_{n_k}\} & P_n\end{array}}{QD_n}\,(\iota_{\mathcal{Q}}\mathrm{I}^{\mathrm{C}}_{i,n})$$

---

[3] I would like to thank Jan Köpping for making me aware of Haddock’s Puzzle [8] by which (11) is inspired.

$$\dfrac{\begin{array}{c}\mathcal{D}_1\\ QD_j\end{array}}{E_{j_k}}\,(\iota_{\mathcal{Q}}\mathrm{E}^{\mathrm{C}}_{k,j}1)\qquad \dfrac{\begin{array}{c}\mathcal{D}_1\\ QD_j\end{array}}{QU_{j_k}}\,(\iota_{\mathcal{Q}}\mathrm{E}^{\mathrm{C}}_{k,j}2)\qquad \dfrac{\begin{array}{c}\mathcal{D}_1\\ QD_j\end{array}}{P_j}\,(\iota_{\mathcal{Q}}\mathrm{E}^{\mathrm{C}}_{i,j}3)$$

where $i \in \{1, ..., m\}$ (arity of predicate in QD_j), $j \in \{1, ..., n\}$ (level of nesting), $k \in \{1, ..., m\}$

2. *Rules for parallel conjunctively nested negative qualified definiteness*:

$$\dfrac{\begin{array}{ccc}\{\mathcal{D}_{1_1}\} & \{\mathcal{D}_{2_1}\} & \mathcal{D}_{3_1}\\ \{-E_{1_k}\} & \{-QU_{1_k}\} & -P_1\end{array}}{-QD_1}\,(\iota_{\mathcal{Q}}{-}\mathrm{I}^{\mathrm{C}}_{i,1})$$

$$\vdots$$

$$\dfrac{\begin{array}{ccc}\{\mathcal{D}_{1_n}\} & \{\mathcal{D}_{2_n}\} & \mathcal{D}_{3_n}\\ \{-E_{n_k}\} & \{-QU_{n_k}\} & -P_n\end{array}}{-QD_n}\,(\iota_{\mathcal{Q}}{-}\mathrm{I}^{\mathrm{C}}_{i,n})$$

$$\dfrac{\begin{array}{c}\mathcal{D}_1\\ -QD_j\end{array}}{-E_{j_k}}\,(\iota_{\mathcal{Q}}{-}\mathrm{E}^{\mathrm{C}}_{k,j}1)\qquad \dfrac{\begin{array}{c}\mathcal{D}_1\\ -QD_j\end{array}}{-QU_{j_k}}\,(\iota_{\mathcal{Q}}{-}\mathrm{E}^{\mathrm{C}}_{k,j}2)\qquad \dfrac{\begin{array}{c}\mathcal{D}_1\\ -QD_j\end{array}}{-P_j}\,(\iota_{\mathcal{Q}}{-}\mathrm{E}^{\mathrm{C}}_{i,j}3)$$

where $i \in \{1, ..., m\}$ (arity of predicate in QD_j), $j \in \{1, ..., n\}$ (level of nesting), $k \in \{1, ..., m\}$

(Likewise with $-\psi_j$.)

DEFINITION 6.21 (Detour conversions ($\iota_{\mathcal{Q}}C$-systems)). The detour conversions for parallel conjunctively nested qualified definiteness are, *mutatis mutandis*, like those of Definition 6.12.

Remark 6.22. A remark analogous to Remark 6.6 applies also to $\iota_{\mathcal{Q}}C$-systems.

Example 6.23. We construct a canonical derivation for (12). Symbolization:

$$F^1(\iota_{\mathcal{Q}_3}x(M^1(x)\ \&\ W^2(x, \iota_{\mathcal{Q}_2}y(B^1_1(y)\ \&\ H^2(y, \iota_{\mathcal{Q}_1}z(B^1_2(z)))))))$$

Let $\mathcal{Q}_j, \mathcal{Q}'_j \subseteq \mathcal{P}$.

$$\begin{array}{ccc}\mathcal{D}_{1(E)} & \mathcal{D}_{2(QU)} & \mathcal{D}_{3(P)} \\ \exists z B^1_2(z) & \forall u_1 \forall v_1((B^1_2(u_1)\&B^1_2(v_1)) \supset u_1 \stackrel{\pm}{=}_{\mathcal{Q}_1} v_1) & \forall w_1(B^1_2(w_1) \supset H^2(\alpha_1, w_1))\end{array}$$

$$\mathcal{D}_{5(E)} = \dfrac{\dfrac{\begin{array}{c}\mathcal{D}_4\\ B^1_1(\alpha_1)\end{array} \qquad \dfrac{\mathcal{D}_{1(E)} \qquad \mathcal{D}_{2(QU)} \qquad \mathcal{D}_{3(P)}}{H^2(\alpha_1, \iota_{\mathcal{Q}_1} z(B^1_2(z)))}\,(\iota_{\mathcal{Q}}\mathrm{I}^{\mathrm{C}}_{2,1})}{B^1_1(\alpha_1)\&H^2(\alpha_1, \iota_{\mathcal{Q}_1} z(B^1_2(z)))}}{\exists y(B^1_1(y)\&H^2(y, \iota_{\mathcal{Q}_1} z(B^1_2(z))))}$$

$$\begin{array}{c}\mathcal{D}_{6(QU)}\\ \forall u_2 \forall v_2(((B^1_1(u_2)\&H^2(u_2, \iota_{\mathcal{Q}} z(B^1_2(z))))\&(B^1_1(v_2)\&H^2(v_2, \iota_{\mathcal{Q}} z(B^1_2(z))))) \supset u_2 \stackrel{\pm}{=}_{\mathcal{Q}_2} v_2)\end{array}$$

$$\begin{array}{c}\mathcal{D}_{7(P)}\\ \forall w_2((B^1_1(w_2)\&H^2(w_2, \iota_{\mathcal{Q}} z(B^1_2(z)))) \supset W^2(\alpha_2, w_2))\end{array}$$

$$\mathcal{D}_{9(E)} = \dfrac{\dfrac{\begin{array}{c}\mathcal{D}_8\\ M^1(\alpha_2)\end{array} \qquad \dfrac{\mathcal{D}_{5(E)} \qquad \mathcal{D}_{6(QU)} \qquad \mathcal{D}_{7(P)}}{W^2(\alpha_2, \iota_{\mathcal{Q}_2} y(B^1_1(y)\ \&\ H^2(y, \iota_{\mathcal{Q}_1} z(B^1_2(z)))))}\,(\iota_{\mathcal{Q}}\mathrm{I}^{\mathrm{C}}_{2,2})}{M^1(\alpha_2)\&W^2(\alpha_2, \iota_{\mathcal{Q}_2} y(B^1_1(y)\&H^2(y, \iota_{\mathcal{Q}_1} z(B^1_2(z)))))}}{\exists x(M^1(x)\&W^2(x, \iota_{\mathcal{Q}_2} y(B^1_1(y)\&H^2(y, \iota_{\mathcal{Q}_1} z(B^1_2(z))))))}$$

$$\begin{array}{c}\mathcal{D}_{10(QU)}\\ \forall u_3 \forall v_3(((M^1(u_3)\&W^2(u_3, \iota_{\mathcal{Q}_2} y(B^1_1(y)\&H^2(y, \iota_{\mathcal{Q}_1} z(B^1_2(z))))))\\ \&(M^1(v_3)\&W^2(v_3, \iota_{\mathcal{Q}_2} y(B^1_1(y)\&H^2(y, \iota_{\mathcal{Q}_1} z(B^1_2(z))))))) \supset u_3 \stackrel{\pm}{=}_{\mathcal{Q}_3} v_3)\end{array}$$

$$\begin{array}{c}\mathcal{D}_{11(P)}\\ \forall w_3((M^1(w_3)\&W^2(w_3, \iota_{\mathcal{Q}_2} y(B^1_1(y)\&H^2(y, \iota_{\mathcal{Q}_1} z(B^1_2(z)))))) \supset F^1(w_3))\end{array}$$

$$\dfrac{\mathcal{D}_{9(E)} \qquad \mathcal{D}_{10(QU)} \qquad \mathcal{D}_{11(P)}}{F^1(\iota_{\mathcal{Q}_3} x(M^1(x)\ \&\ W^2(x, \iota_{\mathcal{Q}_2} y(B^1_1(y)\ \&\ H^2(y, \iota_{\mathcal{Q}_1} z(B^1_2(z)))))))}\,(\iota_{\mathcal{Q}}\mathrm{I}^{\mathrm{C}}_{1,3}) \tag{6.6}$$

Example 6.24. Consider (14). Symbolization:

$$\begin{array}{l}W^2_1(\iota_{\mathcal{Q}_2} x(M^1(x)\&W^2_2(x, \iota_{\mathcal{Q}_1} y(B^1(y)))\ \&\ C^2(x, \iota_{\mathcal{Q}'_1} z(N^1(z)))), \iota_{\mathcal{Q}_3} u(D^1(u)\&\\ O^2(\iota_{\mathcal{Q}_2} x(M^1(x)\&(W^2_2(x, \iota_{\mathcal{Q}_1} y(B^1(y)))\ \&\ C^2(x, \iota_{\mathcal{Q}'_1} z(N^1(z)))), u)))\end{array}$$

Note that the second occurrence of & in the first description is the one which corresponds to the conjunction figuring at the surface of (14). Let $\mathcal{Q}_j, \mathcal{Q}'_j \subseteq \mathcal{P}$.

$$\begin{array}{ccc}
\mathcal{D}_{1(E)} & \mathcal{D}_{2(QU)} & \mathcal{D}_{3(P)} \\
\exists y B^1(y) & \forall u_1 \forall v_1((B^1(u_1)\&B^1(v_1)) \supset u_1 \overset{\pm}{=}_{\mathcal{Q}_1} v_1) & \forall w_1(B^1(w_1) \supset W^2_2(\alpha_1, w_1)) \\
\mathcal{D}'_{1(E)} & \mathcal{D}'_{2(QU)} & \mathcal{D}'_{3(P)} \\
\exists z N^1(z) & \forall u'_1 \forall v'_1((N^1(u'_1)\&N^1(v'_1)) \supset u'_1 \overset{\pm}{=}_{\mathcal{Q}'_1} v'_1) & \forall w'_1(N^1(w'_1) \supset C^2(\alpha_1, w'_1))
\end{array}$$

$$\mathcal{D}_{5(E)} = \cfrac{\cfrac{\begin{matrix}\mathcal{D}_4 \\ M^1(\alpha_1)\end{matrix} \qquad \cfrac{\cfrac{\mathcal{D}_{1(E)} \quad \mathcal{D}_{2(QU)} \quad \mathcal{D}_{3(P)}}{W^2_2(\alpha_1, \iota_{\mathcal{Q}_1} y(B^1(y)))}\,(\iota_{\mathcal{Q}}\mathrm{I}^{\mathrm{C}}_{2,1}) \quad \cfrac{\mathcal{D}'_{1(E)} \quad \mathcal{D}'_{2(QU)} \quad \mathcal{D}'_{3(P)}}{C^2(\alpha_1, \iota_{\mathcal{Q}'_1} z(N^1(z)))}\,(\iota_{\mathcal{Q}}\mathrm{I}^{\mathrm{C}}_{2,1})}{W^2_2(\alpha_1, \iota_{\mathcal{Q}_1} y(B^1(y)))\&C^2(\alpha_1, \iota_{\mathcal{Q}'_1} z(N^1(z)))}\,(\&\mathrm{I})}{M^1(\alpha_1)\&W^2_2(\alpha_1, \iota_{\mathcal{Q}_1} y(B^1(y)))\&C^2(\alpha_1, \iota_{\mathcal{Q}'_1} z(N^1(z)))}\,(\&\mathrm{I})}{\exists x(\underbrace{M^1(x)\&W^2_2(x, \iota_{\mathcal{Q}_1} y(B^1(y)))\&C^2(x, \iota_{\mathcal{Q}'_1} z(N^1(z))))}_{=A_1(x)})}$$

$$\begin{array}{c}
\mathcal{D}_{6(QU)} \\
\forall u_2 \forall v_2((A_1(u_2)\ \&\ A_1(v_2)) \supset u_2 \overset{\pm}{=}_{\mathcal{Q}_2} v_2) \\
\mathcal{D}_{7(P)} \\
\forall w_2((M^1(w_2)\&W^2_2(w_2, \iota_{\mathcal{Q}_1} y(B^1(y)))\&C^2(w_2, \iota_{\mathcal{Q}'_1} z(N^1(z)))) \supset O^2(w_2, \alpha_2))
\end{array}$$

$$\mathcal{D}_{9(E)} = \cfrac{\cfrac{\begin{matrix}\mathcal{D}_8 \\ D^1(\alpha_2)\end{matrix} \quad \cfrac{\mathcal{D}_{5(E)} \quad \mathcal{D}_{6(QU)} \quad \mathcal{D}_{7(P)}}{O^2(\iota_{\mathcal{Q}_2} x(M^1(x)\&W^2_2(x, \iota_{\mathcal{Q}_1} y(B^1(y)))\&C^2(x, \iota_{\mathcal{Q}'_1} z(N^1(z)))), \alpha_2)}\,(\iota_{\mathcal{Q}}\mathrm{I}^{\mathrm{C}}_{2,2})}{D^1(\alpha_2)\&O^2(\iota_{\mathcal{Q}_2} x(M^1(x)\&W^2_2(x, \iota_{\mathcal{Q}_1} y(B^1(y)))\&C^2(x, \iota_{\mathcal{Q}'_1} z(N^1(z)))), \alpha_2)}\,(\&\mathrm{I})}{\exists u(\underbrace{D^1(u)\&O^2(\iota_{\mathcal{Q}_2} x(M^1(x)\&W^2_2(x, \iota_{\mathcal{Q}_1} y(B^1(y)))\&C^2(x, \iota_{\mathcal{Q}'_1} z(N^1(z)))), u)}_{A_2(u)})}$$

$$\begin{array}{c}
\mathcal{D}_{10(QU)} \\
\forall u_3 \forall v_3((A_2(u_3)\&A_2(v_3)) \supset u_3 \overset{\pm}{=}_{\mathcal{Q}_3} v_3) \\
\mathcal{D}_{11(P)} \\
\forall w_3 \forall w_4((M^1(w_3)\&W^2_2(w_3, \iota_{\mathcal{Q}_1} y(B^1(y)))\&C^2(w_3, \iota_{\mathcal{Q}'_1} z(N^1(z)))) \ \& \\
(D^1(w_4)\&O^2(\iota_{\mathcal{Q}_2} x(M^1(x)\&W^2_2(x, \iota_{\mathcal{Q}_1} y(B^1(y))\&C^2(x, \iota_{\mathcal{Q}'_1} z(N^1(z)))), w_4))) \\
\supset W^2_1(w_3, w_4))
\end{array}$$

$$\cfrac{\mathcal{D}_{5(E)} \quad \mathcal{D}_{9(E)} \qquad \mathcal{D}_{6(QU)} \quad \mathcal{D}_{10(QU)} \qquad \mathcal{D}_{11(P)}}{\begin{array}{c} W^2_1(\iota_{\mathcal{Q}_2} x(M^1(x)\&W^2_2(x, \iota_{\mathcal{Q}_1} y(B^1(y)))\ \&\ C^2(x, \iota_{\mathcal{Q}'_1} z(N^1(z)))), \\ \iota_{\mathcal{Q}_3} u(D^1(u)\&O^2(\iota_{\mathcal{Q}_2} x(M^1(x)\&(W^2_2(x, \iota_{\mathcal{Q}_1} y(B^1(y)))\ \&\ C^2(x, \iota_{\mathcal{Q}'_1} z(N^1(z)))), u))) \end{array}}\,(\iota_{\mathcal{Q}}\mathrm{I}^{\mathrm{C}}_{2,3})$$

7. Concluding remarks

We have shown that dropping the standard primitive notion of identity from Russell's analysis of definite descriptions in favour of the defined notion of qualified identity [24] can lead us to proof systems for definiteness which not only enjoy good proof-theoretic properties (normalization, subexpression property, subformula property, internal completeness), but which also admit a formulation of an intuitionistically acceptable proof-theoretic semantics for natural language constructions which involve various kinds of definite descriptions (complete, incomplete, generic, parallel, nested, Haddock, predicative). It is hoped that the framework developed above is versatile enough to be adapted so as to handle a wider range of definiteness constructions.

Acknowledgements. Versions of this material have been presented at *Non-Classical Logics. Theory and Applications (NCL'24)* (University of Łódź, September of 2024), at the *ExtenDD Seminar* (University of Łódź, online, November of 2024), and at the *Philosophy of Language, Logic, and Information Colloquium* (Ruhr University Bochum, November of 2024). I would like to thank the audience members at each of these events for their valuable feedback, in particular, Andrzej Indrzejczak, Jan Köpping, Nils Kürbis, Kristina Liefke, and Michał Zawidzki. I am also indebted to two anonymous referees for this journal for comments which led to improvements.

References

[1] P. Elbourne, **Definite Descriptions**, Oxford Studies in Semantics and Pragmatics, Vol. 1, Oxford University Press, Oxford (2013), DOI: https://doi.org/10.1093/acprof:oso/9780199660193.001.0001.

[2] P. Elbourne, *Incomplete Descriptions and Indistinguishable Participants*, **Natural Language Semantics**, vol. 24(1) (2016), pp. 1–43, DOI: https://doi.org/10.1007/s11050-015-9118-8.

[3] N. Francez, B. Więckowski, *A Proof-Theoretic Semantics for Contextual Definiteness*, [in:] E. Moriconi, L. Tesconi (eds.), **Second Pisa Colloquium in Logic, Language and Epistemology**, Pisa: Edizioni ETS (2014), pp. 181–212.

[4] N. Francez, B. Więckowski, *A Proof-Theory for First-Order Logic With Definiteness*, **The IfCoLog Journal of Logics and their Applications**, vol. 4(2) (2017), pp. 313–331.

[5] P. T. Geach, *Identity*, **The Review of Metaphysics**, vol. 21(1) (1967), pp. 3–12, URL: https://www.jstor.org/stable/20124493.

[6] J.-Y. Girard, *From Foundations to Ludics*, **The Bulletin of Symbolic Logic**, vol. 9(2) (2003), pp. 131–168, DOI: https://doi.org/10.2178/bsl/1052669286.

[7] D. Graff, *Descriptions as Predicates*, **Philosophical Studies**, vol. 102(1) (2001), pp. 1–42, DOI: https://doi.org/10.1023/A:1010379409594.

[8] N. J. Haddock, *Incremental Interpretation and Combinatory Categorial Grammar*, [in:] J. P. McDermott (ed.), **Proceedings of the 10th International Joint Conference on Artificial Intelligence, Vol. 2**, San Francisco: Morgan Kaufmann Publishers (1987), pp. 661–663.

[9] A. Indrzejczak, *The Logicality of Equality*, [in:] T. Piecha, K. F. Wehmeier (eds.), **Peter Schroeder-Heister on Proof-Theoretic Semantics**, vol. 29 of Outstanding Contributions to Logic, Cham, Switzerland: Springer (2024), pp. 211–238, DOI: https://doi.org/10.1007/978-3-031-50981-0_7.

[10] S. T. Kuhn, *Embedded Definite Descriptions: Russellian Analysis and Semantic Puzzles*, **Mind**, vol. 109(435) (2000), pp. 443–454, DOI: https://doi.org/10.1093/mind/109.435.443.

[11] N. Kürbis, *A Binary Quantifier for Definite Descriptions in Intuitionist Negative Free Logic: Natural Deduction and Normalisation*, **Bulletin of the Section of Logic**, vol. 48(2) (2019), pp. 81–97, DOI: https://doi.org/10.18778/0138-0680.48.2.01.

[12] P. Ludlow, *Descriptions*, [in:] E. N. Zalta, U. Nodelman (eds.), **The Stanford Encyclopedia of Philosophy (Winter 2023 Edition)** (2023),

URL: https://plato.stanford.edu/archives/win2023/entries/descriptions.

[13] D. Prawitz, **Natural Deduction: A Proof-Theoretical Study**, Almqvist and Wiksell, Stockholm (1965 (Reprint 2006)).

[14] S. Read, *Identity and Harmony*, **Analysis**, vol. 64(2) (2004), pp. 113–119, DOI: https://doi.org/10.1093/analys/64.2.113.

[15] S. Read, *Harmonic Inferentialism and the Logic of Identity*, **The Review of Symbolic Logic**, vol. 9(2) (2016), pp. 408–420, DOI: https://doi.org/10.1017/S1755020316000010.

[16] B. Russell, *On Denoting*, **Mind**, vol. 14(56) (1905), pp. 479–493, DOI: https://doi.org/10.1093/mind/XIV.4.479.

[17] P. Schroeder-Heister, *Proof-Theoretic Semantics*, [in:] E. N. Zalta, U. Nodelman (eds.), **The Stanford Encyclopedia of Philosophy (Fall 2023 Edition)** (2023), URL: https://plato.stanford.edu/entries/proof-theoretic-semantics/.

[18] J. Stanley, Z. G. Szabó, *On Quantifier Domain Restriction*, **Mind and Language**, vol. 15(2–3) (2000), pp. 219–261, DOI: https://doi.org/10.1111/1468-0017.00130.

[19] S. Stenlund, *Descriptions in Intuitionistic Logic*, [in:] S. Kanger (ed.), **Proceedings of the Third Scandinavian Logic Symposium**, vol. 82 of Studies in Logic and the Foundations of Mathematics, Amsterdam: North-Holland (1975), pp. 197–212, DOI: https://doi.org/10.1016/S0049-237X(08)70732-8.

[20] Z. G. Szabó, *Definite Descriptions Without Uniqueness: A Reply to Abbott*, **Philosophical Studies**, vol. 114(3) (2003), pp. 279–291, DOI: https://doi.org/10.1023/A:1024904828212.

[21] A. S. Troelstra, H. Schwichtenberg, **Basic Proof Theory**, Cambridge Tracts in Theoretical Computer Science, Vol. 43, 2nd edition, Cambridge University Press, Cambridge (2000), DOI: https://doi.org/10.1017/CBO9781139168717.

[22] B. Więckowski, *Subatomic Natural Deduction for a Naturalistic First-Order Language With Non-Primitive Identity*, **Journal of Logic,**

Language and Information, vol. 25(2) (2016), pp. 215–268, DOI: https://doi.org/10.1007/s10849-016-9238-7.

[23] B. Więckowski, *Subatomic Negation*, **Journal of Logic, Language and Information**, vol. 30(1) (2021), pp. 207–262, DOI: https://doi.org/10.1007/s10849-020-09325-4.

[24] B. Więckowski, *Negative Predication and Distinctness*, **Logica Universalis**, vol. 17(1) (2023), pp. 103–138, DOI: https://doi.org/10.1007/s11787-022-00321-9.

[25] B. Więckowski, *Incomplete Descriptions and Qualified Definiteness*, [in:] A. Indrzejczak, M. Zawidzki (eds.), **Non-Classical Logics. Theory and Applications (NCL'24)**, vol. 415 of Electronic Proceedings in Theoretical Computer Science (2024), pp. 109–120, DOI: https://doi.org/https://doi.org/10.4204/EPTCS.415.12.

Bartosz Więckowski
Goethe-Universität Frankfurt am Main
Institut für Philosophie
Norbert-Wollheim-Platz 1
60629 Frankfurt am Main, Germany

e-mail: wieckowski@em.uni-frankfurt.de

Funding information: This work was supported by the Deutsche Forschungsgemeinschaft (grant number WI 3456/5-1).

Conflict of interests: None.

Ethical considerations: The Author assures of no violations of publication ethics and takes full responsibility for the content of the publication.

Declaration regarding the use of GAI tools: Not used.

Bulletin of the **S**ection of **L**ogic
Volume 54/4 (2025), pp. 519–576
https://doi.org/10.18778/0138-0680.2025.17

COPE
Member since 2018
JM13707

Cheng-Syuan Wan

SEMI-SUBSTRUCTURAL LOGICS À LA LAMBEK WITH SYMMETRY

Abstract

This work studies the proof theory and ternary relational semantics of left (right) skew monoidal closed categories and skew monoidal bi-closed categories, both symmetric and non-symmetric, from the perspective of non-associative Lambek calculus. Uustalu et al. used sequents with stoup (the leftmost position of an antecedent that can be either empty or a single formula) to deductively model left skew monoidal closed categories, yielding results regarding proof identities and categorical coherence. However, their syntax does not work well when modeling right skew monoidal closed and skew monoidal bi-closed categories, whether symmetric or non-symmetric.

We solve the problem via more flexible and equivalent frameworks to characterize the categories above: tree sequent calculus (where antecedents are binary trees) and axiomatic calculus (where antecedents are a single formula), inspired by works on non-associative Lambek calculus. Moreover, we prove that the axiomatic calculi are sound and complete with respect to their ternary relational

---

Presented by: Michał Zawidzki
Received: December 20, 2024, **Received in revised form**: October 22, 2025,
Accepted: October 28, 2025, **Published online**: March 13, 2026

models. We also prove a correspondence between frame conditions and structural laws, providing an algebraic way to understand the relationship between the left and right skew monoidal closed categories, encompassing both symmetric and non-symmetric variants.

Keywords: substructural logic, Lambek calculus, non-associative Lambek calculus, category theory, skew monoidal category, ternary relational semantics.

1. Introduction

Substructural logics are logic systems that lack at least one of the structural rules, weakening, contraction, and exchange. Joachim Lambek's syntactic calculus [18] is a well-known example that disallows weakening, contraction, and exchange. Another example, linear logic, proposed by Jean-Yves Girard [14], is a substructural logic in which weakening and contraction are in general disallowed but can be recovered for some formulae via modalities. Substructural logics have been found in numerous applications from computational analysis of natural languages to the development of resource-sensitive programming languages.

Left skew monoidal categories [26] are a weaker variant of Saunders Mac Lane's monoidal categories where the structural morphisms of associativity and unitality are not required to be bidirectional, they are natural transformations with a particular orientation. Therefore, they can be seen as *semi-associative* and *semi-unital* variants of monoidal categories. Left skew monoidal categories arise naturally in the semantics of programming languages [2], while the concept of semi-associativity is connected with combinatorial structures like the Tamari lattice and Stasheff associahedra [37, 22].

In recent years, Tarmo Uustalu, Niccolò Veltri, and Noam Zeilberger started a research project on *semi-substructural* logics, which is inspired by a series of developments on left skew monoidal categories and their variants by Szlachányi, Street, Bourke, Lack and others [26, 16, 25, 17, 8, 5, 6, 7].

We call the logics of left skew monoidal categories and their variants *semi-substructural* logics, because they are intermediate logics between (certain fragments of) non-associative and associative intuitionistic linear logic (or Lambek calculus). Semi-associativity and semi-unitality are encoded as follows. Sequents are in the form $S \mid \Gamma \vdash A$, where the antecedent consists of an optional formula S, called stoup, adapted from Girard [15], and an ordered list of formulae Γ. The succedent is a single formula A. We restrict the application of introduction rules in an appropriate way to allow only one of the directions of associativity and unitality.

This approach has successfully captured languages for several varieties of skew structured categories, including (i) left skew semigroup [37], (ii) left skew monoidal [31], (iii) left skew (prounital) closed [29], (iv) left skew monoidal closed categories [27, 33], and (v) distributive left skew monoidal categories with finite products and coproducts [35] through skew variants of fragments of non-commutative intuitionistic linear logic with different combinations of connectives $(\mathsf{I}, \otimes, \multimap, \wedge, \vee)$. Additionally, discussions have covered partial normality conditions, in which one or more structural morphisms are required to have an inverse [30], as well as extensions with skew exchange à la Bourke and Lack [32, 35, 34].

In all of the aforementioned works, internal languages of left skew monoidal categories and their variants are characterized in a similar way which we call sequent calculus à la Girard. These calculi with sequents of the form $S \mid \Gamma \vdash A$ are cut-free and by their rule design, they are decidable. Moreover, they all admit sound and complete subcalculi inspired by Andreoli's focusing [3] in which rules are restricted to be applied in a specific order. A focused calculus provides an algorithm to solve both the proof identity problems for its non-focused calculus and coherence problems for its corresponding variant of left skew monoidal category.

By reversing all structural morphisms and modifying the coherence conditions in left skew monoidal closed categories, right skew monoidal closed categories emerge [28]. Moreover, skew monoidal bi-closed categories are defined by appropriately integrating left and right skew monoidal closed structures. It is natural for us to consider sound sequent calculi for these categories. However, the implication rules are not well-behaved when just modeling right skew monoidal closed categories with sequent calculus à la Girard.

The problem stems from the skew structure concealed within the flat antecedent of $S \mid \Gamma \vdash A$. While the antecedent $S \mid \Gamma$ is defined similarly to an ordered list, it is actually a tree associating to the left. We start in Section 2 by introducing the sequent calculus à la Girard (`LSkG`) for left skew monoidal closed categories from [27] and its equivalent sequent calculus à la Lambek (`LSkT`)[1], which is inspired by sequent calculus for non-associative Lambek calculus [9, 23] with trees as antecedents.

Associative (non-associative) Lambek calculus can be extended with permutation by adding a rule of exchange [23]. In the commutative version of the Lambek calculus, two implications $\diagdown$ and $\diagup$ collapse into one, i.e. for any formulae A and B, $A \diagdown B$ is logically equivalent to $B \diagup A$. This leads to a natural question to consider semi-substructural logics with permutation. An immediate idea is adding an axiom of permutation directly into the calculus. Veltri addressed the addition of permutation to sequent calculi for symmetric skew monoidal and skew closed categories [32, 34]. Here, we extend this work by generalizing these results to symmetric right skew categories and symmetric skew monoidal bi-closed categories.

In Section 3, we introduce definitions of left (right) skew monoidal closed categories and skew monoidal bi-closed categories, and normality conditions for skew categories. In Section 4, we describe two calculi that characterize skew monoidal bi-closed categories: one is an axiomatic calculus (`SkMBiCA`), while the other is a sequent calculus (`SkMBiCT`) similar to the multimodal non-associative Lambek calculus [21]. In Section 5, we introduce the relational semantics for `SkMBiCA` via preordered sets of possible worlds with ternary relations. Furthermore, we show a correspondence theorem (Theorem 5.8) between conditions on ternary relations and structural laws on any frame. The theorem allows us to prove a thin version of main theorems in [28]. Finally, in Section 6, we incorporate commutativity into semi-substructural logics from both syntactic and semantic perspective following the method in [32, 34] and extend the result to symmetric right skew categories and symmetric skew monoidal bi-closed categories.

---

[1] We attribute this name to Lambek since he proposed the non-associative calculus in [19], though he did not discuss the tree sequent style presentation.

Publication History This paper is an extended version of [36]. Compared to the conference version, we have added Lemmata 2.10 and 4.4, which are essential to the proof of equivalence of calculi (`LSkG` and `LSkT` for the former and `SkMBiCA` and `SkMBiCT` for the latter) and detailed the proof of Theorem 4.6. The whole Section 6, studying the syntax and semantics of semi-substructural logics with permutation, is new.

2. Sequent Calculus

We recall the sequent calculus à la Girard for left skew monoidal closed categories from [27], which is a skew variant of non-commutative multiplicative intuitionistic linear logic.

Formulae (Fma) in `LSkG` are inductively generated by the grammar $A, B ::= X \mid \mathsf{I} \mid A \otimes B \mid A \multimap B$, where X comes from a set At of atoms, I is a multiplicative unit, $\otimes$ is multiplicative conjunction and $\multimap$ is a linear implication.

A sequent is a triple of the form $S \mid \Gamma \vdash_{\mathsf{G}} A$, where the antecedent splits into: an optional formula S, called *stoup* [15], and an ordered list of formulae Γ and succedent A is a single formula. The symbol S consistently denotes a stoup, meaning S can either be a single formula or empty, indicated as $S = -$; furthermore, X, Y, and Z always represent atomic formulae.

Definition 2.1. Derivations in `LSkG` are generated recursively by the following rules:

$$\frac{}{A \mid \quad \vdash_{\mathsf{G}} A}\ \mathsf{ax} \qquad \frac{- \mid \Gamma \vdash_{\mathsf{G}} A \quad B \mid \Delta \vdash_{\mathsf{G}} C}{A \multimap B \mid \Gamma, \Delta \vdash_{\mathsf{G}} C}\ {\multimap}\mathsf{L} \qquad \frac{- \mid \Gamma \vdash_{\mathsf{G}} C}{\mathsf{I} \mid \Gamma \vdash_{\mathsf{G}} C}\ \mathsf{IL}$$

$$\frac{A \mid B, \Gamma \vdash_{\mathsf{G}} C}{A \otimes B \mid \Gamma \vdash_{\mathsf{G}} C}\ {\otimes}\mathsf{L} \qquad \frac{A \mid \Gamma \vdash_{\mathsf{G}} C}{- \mid A, \Gamma \vdash_{\mathsf{G}} C}\ \mathsf{pass} \qquad \frac{S \mid \Gamma, A \vdash_{\mathsf{G}} B}{S \mid \Gamma \vdash_{\mathsf{G}} A \multimap B}\ {\multimap}\mathsf{R}$$

$$\frac{}{- \mid \quad \vdash_{\mathsf{G}} \mathsf{I}}\ \mathsf{IR} \qquad \frac{S \mid \Gamma \vdash_{\mathsf{G}} A \quad - \mid \Delta \vdash_{\mathsf{G}} B}{S \mid \Gamma, \Delta \vdash_{\mathsf{G}} A \otimes B}\ {\otimes}\mathsf{R}$$

The inference rules of `LSkG` are similar to the ones in the sequent calculus for non-commutative multiplicative intuitionistic linear logic (`NMILL`) [1], but with some crucial differences:

1. The left logical rules IL, $\otimes\mathsf{L}$ and $\multimap\mathsf{L}$, read bottom-up, are only allowed to be applied on the formula in the stoup position.
2. The right tensor rule $\otimes\mathsf{R}$, read bottom-up, splits the antecedent of a sequent $S \mid \Gamma, \Delta \vdash_{\mathsf{G}} A \otimes B$ and in the case where S is a formula, S is always moved to the stoup of the left premise, even if Γ is empty.
3. The presence of the stoup distinguishes two types of antecedents, $A \mid \Gamma$ and $- \mid A, \Gamma$. The structural rule pass (for 'passivation'), read bottom-up, allows the moving of the leftmost formula in the context to the stoup position whenever the stoup is empty.
4. The logical connectives of `NMILL` (and associative Lambek calculus) typically include two ordered implications $\diagdown$ and $\diagup$, which are two variants of linear implication arising from the removal of the exchange rule from intuitionistic linear logic. In `LSkG`, only the right residuation ($B \diagup A = A \multimap B$) of Lambek calculus is present.

For a more detailed explanation and a linear logical interpretation of `LSkG`, see [27, Section 2].

Theorem 2.2. *The rules*

$$\frac{S \mid \Gamma \vdash_{\mathsf{G}} A \quad A \mid \Delta \vdash_{\mathsf{G}} C}{S \mid \Gamma, \Delta \vdash_{\mathsf{G}} C}\;\mathsf{scut} \qquad \frac{- \mid \Gamma \vdash_{\mathsf{G}} A \quad S \mid \Delta_0, A, \Delta_1 \vdash_{\mathsf{G}} C}{S \mid \Delta_0, \Gamma, \Delta_1 \vdash_{\mathsf{G}} C}\;\mathsf{ccut}$$

are admissible in `LSkG`*.*

Proof: The proof proceeds by induction on the height of derivations and the complexity of cut formulae. Specifically, for scut, we first perform induction on the left premise f, and if necessary, we perform subinduction on g or the complexity of the cut formula A. For ccut, we start by performing induction on the right premise g instead. The cases other than $\multimap\mathsf{L}$ and

$\multimap\mathsf{R}$ have been discussed in [31, Lemma 5], so we will only elaborate on the cases of $\multimap$.
We first deal with scut. If $f = \multimap\mathsf{L}(f', f'')$, then we permute scut up, i.e.

$$\dfrac{\dfrac{\overset{f'}{- \mid \Gamma \vdash_{\mathsf{G}} A'} \quad \overset{f''}{B' \mid \Delta \vdash_{\mathsf{G}} A}}{A' \multimap B' \mid \Gamma, \Delta \vdash_{\mathsf{G}} A}\;\multimap\mathsf{L} \qquad \overset{g}{A \mid \Lambda \vdash_{\mathsf{G}} C}}{A' \multimap B' \mid \Gamma, \Delta, \Lambda \vdash_{\mathsf{G}} C}\;\mathsf{scut}$$

$$\mapsto \qquad \dfrac{\overset{f'}{- \mid \Gamma \vdash_{\mathsf{G}} A'} \qquad \dfrac{\overset{f''}{B' \mid \Delta \vdash_{\mathsf{G}} A} \quad \overset{g}{A \mid \Lambda \vdash_{\mathsf{G}} C}}{B' \mid \Delta, \Lambda \vdash_{\mathsf{G}} C}\;\mathsf{scut}}{A' \multimap B' \mid \Gamma, \Delta, \Lambda \vdash_{\mathsf{G}} C}\;\multimap\mathsf{L}$$

If $f = \multimap\mathsf{R}\ f'$, then we perform a subinduction on g:

- If $g = \multimap\mathsf{L}(g', g'')$, then

$$\dfrac{\dfrac{\overset{f'}{S \mid \Gamma, A \vdash_{\mathsf{G}} B}}{S \mid \Gamma \vdash_{\mathsf{G}} A \multimap B}\;\multimap\mathsf{R} \qquad \dfrac{\overset{g'}{- \mid \Delta \vdash_{\mathsf{G}} A} \quad \overset{g''}{B \mid \Lambda \vdash_{\mathsf{G}} C}}{A \multimap B \mid \Delta, \Lambda \vdash_{\mathsf{G}} C}\;\multimap\mathsf{L}}{S \mid \Gamma, \Delta, \Lambda \vdash_{\mathsf{G}} C}\;\mathsf{scut}$$

$$\mapsto \qquad \dfrac{\overset{g'}{- \mid \Delta \vdash_{\mathsf{G}} A} \qquad \dfrac{\overset{f'}{S \mid \Gamma, A \vdash_{\mathsf{G}} B} \quad \overset{g''}{B \mid \Lambda \vdash_{\mathsf{G}} C}}{S \mid \Gamma, A, \Lambda \vdash_{\mathsf{G}} C}\;\mathsf{scut}}{S \mid \Gamma, \Delta, \Lambda \vdash_{\mathsf{G}} C}\;\mathsf{ccut}$$

  where the complexity of the cut formulae is reduced.

- For other rules, we permute scut up. For example, if $g = \multimap\mathsf{R}\ g'$, then

$$\dfrac{\dfrac{\overset{f'}{S \mid \Gamma, A \vdash_{\mathsf{G}} B}}{S \mid \Gamma \vdash_{\mathsf{G}} A \multimap B}\ {\multimap}\mathsf{R} \qquad \dfrac{\overset{g'}{A \multimap B \mid \Delta, A' \vdash_{\mathsf{G}} B'}}{A \multimap B \mid \Delta \vdash_{\mathsf{G}} A' \multimap B'}\ {\multimap}\mathsf{R}}{S \mid \Gamma, \Delta \vdash_{\mathsf{G}} A' \multimap B'}\ \mathsf{scut}$$

$$\mapsto \qquad \dfrac{\dfrac{\dfrac{\overset{f'}{S \mid \Gamma, A \vdash_{\mathsf{G}} B}}{S \mid \Gamma \vdash_{\mathsf{G}} A \multimap B}\ {\multimap}\mathsf{R} \qquad \overset{g'}{A \multimap B \mid \Delta, A' \vdash_{\mathsf{G}} B'}}{S \mid \Gamma, \Delta, A' \vdash_{\mathsf{G}} B'}\ \mathsf{scut}}{S \mid \Gamma, \Delta \vdash_{\mathsf{G}} A' \multimap B'}\ {\multimap}\mathsf{R}$$

For ccut, if $g = {\multimap}\mathsf{R}\ g'$, then we permute ccut up. If $g = {\multimap}\mathsf{L}(g', g'')$, we permute ccut up as well, but depending on where the cut formula is placed, we either apply ccut on f and g' or f and g''. ☐

Moreover, `LSkG` is sound and complete wrt. left skew monoidal closed categories [27, Theorem 3.2].

By soundness and completeness, similar to the result in [31] for skew monoidal categories, we mean that `LSkG` is deductively equivalent to the axiomatic characterization of the free left skew monoidal closed category (`LSkA`).

Definition 2.3. Derivations in `LSkA` are generated by the following rules.

$$\dfrac{}{A \vdash_{\mathsf{L}} A}\ \mathsf{id} \qquad \dfrac{A \vdash_{\mathsf{L}} B \quad B \vdash_{\mathsf{L}} C}{A \vdash_{\mathsf{L}} C}\ \mathsf{comp} \qquad \dfrac{A \vdash_{\mathsf{L}} C \quad B \vdash_{\mathsf{L}} D}{A \otimes B \vdash_{\mathsf{L}} C \otimes D}\ \otimes$$

$$\dfrac{C \vdash_{\mathsf{L}} A \quad B \vdash_{\mathsf{L}} D}{A \multimap B \vdash_{\mathsf{L}} C \multimap D}\ \multimap \qquad \dfrac{}{\mathsf{I} \otimes A \vdash_{\mathsf{L}} A}\ \lambda \qquad \dfrac{}{A \vdash_{\mathsf{L}} A \otimes \mathsf{I}}\ \rho$$

$$\dfrac{}{(A \otimes B) \otimes C \vdash_{\mathsf{L}} A \otimes (B \otimes C)}\ \alpha \qquad \begin{array}{c} A \otimes B \vdash_{\mathsf{L}} C \\ \hline\hline A \vdash_{\mathsf{L}} B \multimap C \end{array}\ \pi$$

Throughout this paper, we will often treat derivations as formal objects. We use notation such as $f : A \vdash_{\mathsf{L}} B$ or $g : S \mid \Gamma \vdash_{\mathsf{G}} C$ to denote a specific derivation f or g of the sequent that follows the colon. This convention is adopted from type theory and the "proofs-as-morphisms" paradigm, where a derivation is treated as a concrete term or morphism, and the sequent is its type or specification.

This axiomatic calculus is a semi-unital and semi-associative variation of Moortgat and Oehrle's calculus [23, Chapter 4] of non-associative Lambek calculus (`NL`), where only right residuation is present.

However, different from `NL`, the rule comp cannot be eliminated from `LSkA`. For example, consider the following derivation:

$$\dfrac{\dfrac{}{\mathsf{I} \otimes (\mathsf{I} \otimes X) \vdash \mathsf{I} \otimes X}\,\lambda \qquad \dfrac{}{\mathsf{I} \otimes X \vdash X}\,\lambda}{\mathsf{I} \otimes (\mathsf{I} \otimes X) \vdash X}\,\mathsf{comp}$$

One can observe that no other axiom or inference rule can produce the endsequent in the comp-free calculus. Therefore, comp is an essential rule in this calculus.

We only care about sequent derivability in this section, therefore we omit the congruence relations on sets of derivations $A \vdash_{\mathsf{L}} B$ and $S \mid \Gamma \vdash_{\mathsf{G}} A$ that identify certain pairs of derivations. However, the congruence relations are essential for these calculi being correct characterizations of the free left skew monoidal closed category.

The calculus `LSkG`, being an equivalent presentation of a skew version of `NL`, provides an effective procedure to determine formulae derivability in `LSkA`. In other words, for any formula A, $\vdash_{\mathsf{L}} A$ if and only if $- \mid \ \vdash_{\mathsf{G}} A$. Exhaustive proof search in `LSkG` always terminates, so for any A, either it finds a proof or it fails and there is no proof

Adapted from [23], we define trees inductively by the grammar $T ::= \mathsf{Fma} \mid - \mid (T, T)$, where $-$ is an empty tree. A context is a tree with a hole defined recursively as $\mathcal{C} ::= [\cdot] \mid (\mathcal{C}, T) \mid (T, \mathcal{C})$. The substitution of a tree into a hole is defined recursively:

$$\begin{array}{rcl} \mathit{subst}([\cdot], U) & = & U \\ \mathit{subst}((T', \mathcal{C}), U) & = & (T', \mathit{subst}(\mathcal{C}, U)) \\ \mathit{subst}((\mathcal{C}, T'), U) & = & (\mathit{subst}(\mathcal{C}, U), T') \end{array}$$

We use $T[\cdot]$ to denote a context and $T[U]$ to abbreviate $\mathit{subst}(T[\cdot], U)$. Sometimes we omit parentheses for trees when it does not cause ambiguity.

Sequents in LSkT are in the form $T \vdash_{\mathsf{T}} A$ where T is a tree and A is a single formula.

Definition 2.4. Derivations in LSkT are generated recursively by following rules:

$$\frac{}{A \vdash_{\mathsf{T}} A}\,\mathsf{ax}$$

$$\frac{T[-] \vdash_{\mathsf{T}} C}{T[\mathsf{I}] \vdash_{\mathsf{T}} C}\,\mathsf{IL} \qquad \frac{}{- \vdash_{\mathsf{T}} \mathsf{I}}\,\mathsf{IR} \qquad \frac{T[A, B] \vdash_{\mathsf{T}} C}{T[A \otimes B] \vdash_{\mathsf{T}} C}\,\otimes\mathsf{L} \qquad \frac{T \vdash_{\mathsf{T}} A \quad U \vdash_{\mathsf{T}} B}{T, U \vdash_{\mathsf{T}} A \otimes B}\,\otimes\mathsf{R}$$

$$\frac{U \vdash_{\mathsf{T}} A \quad T[B] \vdash_{\mathsf{T}} C}{T[A \multimap B, U] \vdash_{\mathsf{T}} C}\,\multimap\mathsf{L} \qquad \frac{T, A \vdash_{\mathsf{T}} B}{T \vdash_{\mathsf{T}} A \multimap B}\,\multimap\mathsf{R}$$

$$\frac{T[U_0, (U_1, U_2)] \vdash_{\mathsf{T}} C}{T[(U_0, U_1), U_2] \vdash_{\mathsf{T}} C}\,\mathsf{assoc} \qquad \frac{T[U] \vdash_{\mathsf{T}} C}{T[-, U] \vdash_{\mathsf{T}} C}\,\mathsf{unitL} \qquad \frac{T[U, -] \vdash_{\mathsf{T}} C}{T[U] \vdash_{\mathsf{T}} C}\,\mathsf{unitR}$$

This calculus is similar to the ones for NL [23] and NL with unit [9] but with semi-associative (assoc) and semi-unital (unitL and unitR) rules. The structural rule unitL, read bottom-up, removes an empty tree from the left. It helps us to correctly characterize the axiom λ in LSkT, i.e. $\mathsf{I} \otimes A \vdash_{\mathsf{T}} A$ is derivable while $A \vdash_{\mathsf{T}} \mathsf{I} \otimes A$ is not. Analogously for the rule unitR, from a bottom-up perspective, adds an empty tree from the right, and we cannot capture ρ in LSkT without unitR (a double question mark ?? means that no rule can be applied to close the derivation):

$$\dfrac{\dfrac{\dfrac{\dfrac{}{A \vdash_{\mathsf{T}} A}\,\mathsf{ax}}{-, A \vdash_{\mathsf{T}} A}\,\mathsf{unitL}}{\mathsf{I}, A \vdash_{\mathsf{T}} A}\,\mathsf{IL}}{\mathsf{I} \otimes A \vdash_{\mathsf{T}} A}\,\otimes\mathsf{L} \qquad \dfrac{\dfrac{\overset{??}{X \vdash_{\mathsf{T}} \mathsf{I}} \quad \overset{??}{- \vdash_{\mathsf{T}} X}}{X, - \vdash_{\mathsf{T}} \mathsf{I} \otimes X}\,\otimes\mathsf{R}}{X \vdash_{\mathsf{T}} \mathsf{I} \otimes X}\,\mathsf{unitR}$$

$$\dfrac{\dfrac{\dfrac{}{A \vdash_{\mathsf{T}} A}\,\mathsf{ax} \quad \dfrac{}{- \vdash_{\mathsf{T}} \mathsf{I}}\,\mathsf{IR}}{A, - \vdash_{\mathsf{T}} A \otimes \mathsf{I}}\,\otimes\mathsf{R}}{A \vdash_{\mathsf{T}} A \otimes \mathsf{I}}\,\mathsf{unitR} \qquad \dfrac{\dfrac{\overset{??}{X, - \vdash_{\mathsf{T}} X}}{X, \mathsf{I} \vdash_{\mathsf{T}} X}\,\mathsf{IL}}{X \otimes \mathsf{I} \vdash_{\mathsf{T}} X}\,\otimes\mathsf{L}$$

Theorem 2.5. *The rule*

$$\frac{U \vdash_{\mathsf{T}} A \qquad T[A] \vdash_{\mathsf{T}} C}{T[U] \vdash_{\mathsf{T}} C}\;\mathsf{cut}$$

is admissible in `LSkT`*.*

Proof: We perform induction on the structure of derivation f of the left premise, and if necessary, we perform subinduction on the derivation g or the complexity of the cut formula A. Cases of logical rules ax, $\otimes\mathsf{L}$, $\otimes\mathsf{R}$, $\multimap\mathsf{L}$, and $\multimap\mathsf{R}$ have been discussed in [23], so we only elaborate on the new cases arising in `LSkT`.

- The first new case is that $f = \mathsf{IR}$, then we inspect the structure of g.
  - If $g = \mathsf{ax} : \mathsf{I} \vdash_{\mathsf{T}} \mathsf{I}$, then we define $\mathsf{cut}(\mathsf{IR}, \mathsf{ax}) = \mathsf{IR}$.
  - If $g = \mathsf{IL}\ g'$, then there are two subcases:
    - ∗ if the I introduced by IL is the cut formula, then we define

$$\dfrac{\dfrac{}{- \vdash_{\mathsf{T}} \mathsf{I}}\;\mathsf{IR} \qquad \dfrac{\overset{g'}{T[-] \vdash_{\mathsf{T}} C}}{T[\mathsf{I}] \vdash_{\mathsf{T}} C}\;\mathsf{IL}}{T[-] \vdash_{\mathsf{T}} C}\;\mathsf{cut} \qquad\mapsto\qquad \overset{g'}{T[-] \vdash_{\mathsf{T}} C}$$

    - ∗ if the I introduced by IL is not the cut formula, then we define

$$\dfrac{\dfrac{}{- \vdash_{\mathsf{T}} \mathsf{I}}\;\mathsf{IR} \qquad \dfrac{\overset{g'}{T[-] \vdash_{\mathsf{T}} C}}{T[\mathsf{I}] \vdash_{\mathsf{T}} C}\;\mathsf{IL}}{T^{\{\mathsf{I}:=-\}}[\mathsf{I}] \vdash_{\mathsf{T}} C}\;\mathsf{cut}$$

$$\mapsto \qquad \dfrac{\dfrac{\dfrac{}{- \vdash_{\mathsf{T}} \mathsf{I}}\;\mathsf{ax} \qquad \overset{g'}{T[-] \vdash_{\mathsf{T}} C}}{T^{\{\mathsf{I}:=-\}}[-] \vdash_{\mathsf{T}} C}\;\mathsf{cut}}{T^{\{\mathsf{I}:=-\}}[\mathsf{I}] \vdash_{\mathsf{T}} C}\;\mathsf{IL}$$

where $T^{\{\mathsf{I}:=-\}}[\cdot]$ means that a formula occurrence I at some fixed position in the context has been replaced by $-$.

  - If $g = \mathcal{R}\ g'$, where $\mathcal{R}$ is a one-premise rule other than IL, then $\mathsf{cut}(\mathsf{IR}, \mathcal{R}\ g') = \mathcal{R}(\mathsf{cut}(\mathsf{IR}, g'))$.
  - The cases of an arbitrary two-premises rule are similar.

- The only other new cases are IL and the structural rules, which are all one-premise left rules, where we can permute cut upwards. For example, if $f = \mathsf{unitL}\ f'$, then we define

$$\dfrac{\dfrac{\overset{f'}{T'[U] \vdash_{\mathsf{T}} A}}{T'[-,U] \vdash_{\mathsf{T}} A}\ \mathsf{unitL} \qquad \overset{g}{T[A] \vdash_{\mathsf{T}} C}}{T[T'[-,U]] \vdash_{\mathsf{T}} C}\ \mathsf{cut}$$

$$\mapsto \qquad \dfrac{\dfrac{\overset{f'}{T'[U] \vdash_{\mathsf{T}} A} \quad \overset{g}{T[A] \vdash_{\mathsf{T}}}}{T[T'[U]] \vdash_{\mathsf{T}} C}\ \mathsf{cut}}{T[T'[-,U]] \vdash_{\mathsf{T}} C}\ \mathsf{unitL}$$

The other cases are similar. □

The proof of equivalence relies on the following lemmata and definitions.

Definition 2.6. For any tree T, T^* is the formula obtained from T by replacing commas with $\otimes$ and $-$ with I, respectively.

Lemma 2.7. *For any context $T[\cdot]$ and tree U, $T[U]^* = T[U^*]^*$.*

Proof: The proof proceeds by induction on the structure of $T[\cdot]$.
If $T[\cdot] = [\cdot]$, then $[U]^* = U^*$ by the definition of substitution.
If $T[\cdot] = (T'[\cdot], T'')$, then by inductive hypothesis, we have $T'[U]^* = T'[U^*]^*$ and by definition, we have $(T'[U], T'')^* = T'[U]^* \otimes^{\mathsf{L}} T''^* = T'[U^*]^* \otimes^{\mathsf{L}} T''^* = (T'[U^*], T'')^*$.
The case $T[\cdot] = (T', T''[\cdot])$ is symmetric. □

In the remainder of the section, we will refer to uses of Lemma 2.7 by double lines.

Lemma 2.8. *Given a context $T[\cdot]$ and a derivation $f : A \vdash_{\mathsf{L}} B$, the following rule is admissible:*

$$\frac{\overset{f}{A \vdash_{\mathsf{L}} B}}{T[A]^* \vdash_{\mathsf{L}} T[B]^*}\; T[f]^*$$

Proof: The proof proceeds by induction on the structure of $T[\cdot]$.
If $T[\cdot] = [\cdot]$, then we have $T[A]^* = A$ and $T[B]^* = B$, and f is the desired derivation.
If $T[\cdot] = (T'[\cdot]; T'')$, then we construct the desired derivation as follows:

$$\frac{\dfrac{\dfrac{\overset{f}{A \vdash_{\mathsf{L}} B}}{T'[A]^* \vdash_{\mathsf{L}} T'[B]^*}\; T'[f]^* \qquad \dfrac{}{T''^* \vdash_{\mathsf{L}} T''^*}\; \mathsf{id}}{T'[A]^* \otimes T''^* \vdash_{\mathsf{L}} T'[B]^* \otimes T''^*}\; \otimes}{(T'[A], T'')^* \vdash_{\mathsf{L}} (T'[B], T'')^*}$$

The case $T[\cdot] = (T', T''[\cdot])$ is symmetric. □

Definition 2.9. We define an encoding function $[\![- \mid -]\!]$ that transforms a tree and an ordered list of formulae into a tree associating to the left:

$$[\![T \mid [\ ]]\!] = T$$
$$[\![T \mid B, \Gamma]\!] = [\![(T, B) \mid \Gamma]\!]$$

Lemma 2.10. *For any stoup S and contexts Γ and Δ, $[\![[\![S \mid \Gamma]\!] \mid \Delta]\!] = [\![S \mid \Gamma, \Delta]\!]$.*

Proof: The proof proceeds by induction on Δ.
If $\Delta = [\ ]$, then $[\![[\![S \mid \Gamma]\!] \mid [\ ]]\!] = [\![S \mid \Gamma]\!] = [\![S \mid \Gamma, [\ ]]\!]$ by definition.
If $\Delta = (A, \Delta')$, then by Definition 2.9, inductive hypothesis, and associativity of lists, we have $[\![[\![S \mid \Gamma]\!] \mid A, \Delta']\!] = [\![[\![S \mid \Gamma, A]\!] \mid \Delta']\!] \overset{\text{I.H.}}{=} [\![S \mid (\Gamma, A), \Delta']\!] = [\![S \mid \Gamma, (A, \Delta')]\!]$. □

With the above lemmata, definition, and the functions $s(S)$ that maps a stoup to a tree (i.e. $s(S) = -$ if $S = -$ or $s(S) = B$ if $S = B$), we can state and prove the equivalence between `LSkG` and `LSkT`.

Theorem 2.11. *The calculi* `LSkG` *and* `LSkT` *are equivalent, meaning that the two statements below are true:*

- *For any derivation* $f : S \mid \Gamma \vdash_{\mathsf{G}} C$*, there exists a derivation* $\mathsf{G2T}f : [\![s(S) \mid \Gamma]\!] \vdash_{\mathsf{T}} C$*.*
- *For any derivation* $f : T \vdash_{\mathsf{T}} C$*, there exists a derivation* $\mathsf{T2G}f : T^* \mid \ \vdash_{\mathsf{G}} C$*.*

Proof: Both $\mathsf{G2T}$ and $\mathsf{T2G}$ are constructed by induction on height of f.

For $\mathsf{G2T}$, the interesting cases are $\otimes\mathsf{R}$ and $\multimap\mathsf{L}$. For example, if $f = \otimes\mathsf{R}(f', f'')$, then by inductive hypothesis, we have two derivations $\mathsf{G2T}\ f' : [\![s(S) \mid \Gamma]\!] \vdash_{\mathsf{T}} A$ and $\mathsf{G2T}\ f'' : [\![\mathsf{I} \mid \Delta]\!] \vdash_{\mathsf{T}} B$. Our goal sequent is $[\![[\![s(S) \mid \Gamma]\!] \mid \Delta]\!] \vdash_{\mathsf{T}} A \otimes B$, which is constructed as follows:

$$\dfrac{\dfrac{\dfrac{\dfrac{\overset{\mathsf{G2T}\ f'}{[\![s(S) \mid \Gamma]\!] \vdash_{\mathsf{T}} A} \qquad \overset{\mathsf{G2T}\ f''}{[\![- \mid \Delta]\!] \vdash_{\mathsf{T}} B}}{[\![s(S) \mid \Gamma]\!], [\![- \mid \Delta]\!] \vdash_{\mathsf{T}} A \otimes B}\ \otimes\mathsf{R}}{[\![[\![s(S) \mid \Gamma]\!], - \mid \Delta]\!] \vdash_{\mathsf{T}} A \otimes B}\ \mathsf{assoc}^*}{[\![[\![s(S) \mid \Gamma]\!] \mid \Delta]\!] \vdash_{\mathsf{T}} A \otimes B}\ \mathsf{unitR}}{[\![s(S) \mid \Gamma, \Delta]\!] \vdash_{\mathsf{T}} A \otimes B}\ \text{Lemma 2.10}$$

where assoc^* means multiple applications of assoc. The case of $\multimap\mathsf{L}$ is similar.

For $\mathsf{T2G}$, the construction relies on Lemma 2.8 heavily. For example, when $f = \mathsf{unitR}\ g$, where we have $g : T[U, -] \vdash_{\mathsf{T}} C$. By inductive hypothesis, we have $\mathsf{T2G}\ g : T[U^* \otimes \mathsf{I}]^* \mid \ \vdash_{\mathsf{G}} C$. With Lemma 2.8, we construct the desired derivation as follows:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{U^* \mid \ \vdash_{\mathsf{G}} U^*}\ \mathsf{ax} \qquad \dfrac{}{- \mid \ \vdash_{\mathsf{G}} \mathsf{I}}\ \mathsf{IR}}{U^* \mid \ \vdash_{\mathsf{G}} U^* \otimes \mathsf{I}}\ \otimes\mathsf{R}}{T[U^*]^* \mid \ \vdash_{\mathsf{G}} T[U^* \otimes \mathsf{I}]^*}\ \text{Lemma 2.8} \qquad \overset{\mathsf{T2G}\ g}{T[U^* \otimes \mathsf{I}]^* \mid \ \vdash_{\mathsf{G}} C}}{T[U]^* \mid \ \vdash_{\mathsf{G}} C}\ \mathsf{scut}$$

The other cases are similar. □

3. Skew Categories

In this section, we present the definitions of left (right) skew monoidal closed categories, skew monoidal bi-closed categories, and various terms that will be used in the following section for discussion.

Definition 3.1. A *left skew monoidal closed category* $\mathbb{C}$ is a category with a unit object I and two functors $\otimes : \mathbb{C} \times \mathbb{C} \to \mathbb{C}$ and $\multimap : \mathbb{C}^{\mathsf{op}} \times \mathbb{C} \to \mathbb{C}$ forming an adjunction $- \otimes B \dashv B \multimap -$ for all B, and three natural transformations λ, ρ, α typed $\lambda_A : \mathsf{I} \otimes A \to A$, $\rho_A : A \to A \otimes \mathsf{I}$ and $\alpha_{A,B,C} : (A \otimes B) \otimes C \to A \otimes (B \otimes C)$, satisfying coherence conditions on morphisms due to Mac Lane [20]:

$$\begin{tikzcd}
 & \mathsf{I} \otimes \mathsf{I} \arrow[dr, "\lambda_{\mathsf{I}}"] & \\
\mathsf{I} \arrow[ur, "\rho_{\mathsf{I}}"] \arrow[rr, equal] & & \mathsf{I}
\end{tikzcd}
\qquad
\begin{tikzcd}
(A \otimes \mathsf{I}) \otimes B \arrow[r, "\alpha_{A,\mathsf{I},B}"] & A \otimes (\mathsf{I} \otimes B) \arrow[d, "A \otimes \lambda_B"] \\
A \otimes B \arrow[u, "\rho_A \otimes B"] \arrow[r, equal] & A \otimes B
\end{tikzcd}$$

$$\begin{tikzcd}
(\mathsf{I} \otimes A) \otimes B \arrow[rr, "\alpha_{\mathsf{I},A,B}"] \arrow[dr, "\lambda_A \otimes B"'] & & \mathsf{I} \otimes (A \otimes B) \arrow[dl, "\lambda_{A \otimes B}"] \\
 & A \otimes B &
\end{tikzcd}$$

$$\begin{tikzcd}
(A \otimes B) \otimes \mathsf{I} \arrow[rr, "\alpha_{A,B,\mathsf{I}}"] & & A \otimes (B \otimes \mathsf{I}) \\
 & A \otimes B \arrow[ul, "\rho_{A \otimes B}"] \arrow[ur, "A \otimes \rho_B"'] &
\end{tikzcd}$$

$$\begin{tikzcd}
(A \otimes (B \otimes C)) \otimes D \arrow[rr, "\alpha_{A,B \otimes C,D}"] & & A \otimes ((B \otimes C) \otimes D) \arrow[d, "A \otimes \alpha_{B,C,D}"] \\
((A \otimes B) \otimes C) \otimes D \arrow[u, "\alpha_{A,B,C} \otimes D"] \arrow[r, "\alpha_{A \otimes B,C,D}"'] & (A \otimes B) \otimes (C \otimes D) \arrow[r, "\alpha_{A,B,C \otimes D}"'] & A \otimes (B \otimes (C \otimes D))
\end{tikzcd}$$

Left skew monoidal closed category has other equivalent characterizations [25, 28], because natural transformations (λ, ρ, α) are in bijective correspondence with tuples of (extra)natural transformations (j, i, L) typed $j_A : \mathsf{I} \to A \multimap A$, $i_A : \mathsf{I} \multimap A \to A$, and $L_{A,B,C} : B \multimap C \to (A \multimap B) \multimap (A \multimap C)$. In particular, in a left skew *non-monoidal* closed category,

(λ, ρ, α) are not available and one has to work with (j, i, L) and the corresponding equations.

Definition 3.2. A *right skew monoidal closed category* $(\mathbb{C}, \mathsf{I}, \otimes, \multimap)$ is defined with the same objects and adjoint functors as in left skew monoidal closed category but three natural transformations λ^{R}, ρ^{R}, α^{R} are typed $\lambda^{\mathsf{R}}_A : A \to \mathsf{I} \otimes A$, $\rho^{\mathsf{R}}_A : A \otimes \mathsf{I} \to A$ and $\alpha^{\mathsf{R}}_{A,B,C} : A \otimes (B \otimes C) \to (A \otimes B) \otimes C$. The equations on morphisms are analogous but modified to fit the definition.

Similar to left skew monoidal closed categories, natural transformations $(\lambda^{\mathsf{R}}, \rho^{\mathsf{R}}, \alpha^{\mathsf{R}})$ are in bijective correspondence with tuples $(j^{\mathsf{R}}, i^{\mathsf{R}}, L^{\mathsf{R}})$ typed $j^{\mathsf{R}}_{A,B} : \mathbb{C}(\mathsf{I}, A \multimap B) \to \mathbb{C}(A, B)$, $i^{\mathsf{R}}_A : A \to \mathsf{I} \multimap A$, and $L^{\mathsf{R}}_{A,B,C,D} : \mathbb{C}(A, B \multimap (C \multimap D)) \to \int^X \mathbb{C}(A, X \multimap D) \times \mathbb{C}(B, C \multimap X)$, where $\int^X$ is a coend, cf. [28, Section 4], and $\mathbb{C}(A, B)$ means the set of morphisms from A to B. In parts of the next sections, where we only work with thin categories (for any two objects A and B, $\mathbb{C}(A, B)$ is either empty or a singleton set), it is safe to replace $\int^X$ with an existential quantifier.
In the rest of the paper, we usually omit subscripts of natural transformations.

Definition 3.3. A left skew monoidal closed category is called

- *associative normal* if α is a natural isomorphism;
- *left unital normal* if λ is a natural isomorphism;
- *right unital normal* if ρ is a natural isomorphism.
- Fully normal if α, λ, and ρ are all natural isomorphisms.

Each normality condition can be expressed equivalently using j, i, and L. The normality conditions for right skew monoidal closed categories follow the same pattern, but with α^R, λ^R, and ρ^R instead of α, λ, and ρ.

Definition 3.4. A category $(\mathbb{C}, \mathsf{I}, \otimes^{\mathsf{L}}, \multimap^{\mathsf{L}}, \otimes^{\mathsf{R}}, \multimap^{\mathsf{R}})$ is skew monoidal bi-closed (SkMBiC) if there exists a natural isomorphism $\gamma : A \otimes^{\mathsf{L}} B \to B \otimes^{\mathsf{R}} A$, $(\mathbb{C}, \mathsf{I}, \otimes^{\mathsf{L}}, \multimap^{\mathsf{L}})$ is left skew monoidal closed such that right skew structural rules are dictated by the left skew ones via γ, i.e. $\lambda^{\mathsf{R}} = \gamma \circ \rho$, $\rho^{\mathsf{R}} = \gamma^{-1} \circ \lambda$, and $\alpha^{\mathsf{R}} = (\gamma \otimes^{\mathsf{R}} C) \circ \gamma \circ \alpha \circ \gamma^{-1} \circ (A \otimes^{\mathsf{R}} \gamma^{-1})$ diagrammatically:

$$\begin{tikzcd}
A \arrow[r, "\lambda^{\mathsf{R}}"] \arrow[d, equal] & \mathsf{I} \otimes^{\mathsf{R}} A & A \otimes^{\mathsf{R}} \mathsf{I} \arrow[r, "\rho^{\mathsf{R}}"] \arrow[d, "\gamma^{-1}"'] & A \arrow[d, equal] \\
A \arrow[r, "\rho"'] & A \otimes^{\mathsf{L}} \mathsf{I} \arrow[u, "\gamma"'] & \mathsf{I} \otimes^{\mathsf{L}} A \arrow[r, "\lambda"'] & A
\end{tikzcd}$$

$$\begin{tikzcd}
A \otimes^{\mathsf{R}} (B \otimes^{\mathsf{R}} C) \arrow[r, "\alpha^{\mathsf{R}}"] \arrow[d, "A \otimes^{\mathsf{R}} \gamma^{-1}"'] & (A \otimes^{\mathsf{R}} B) \otimes^{\mathsf{R}} C \\
A \otimes^{\mathsf{R}} (C \otimes^{\mathsf{L}} B) \arrow[d, "\gamma^{-1}"'] & (B \otimes^{\mathsf{L}} A) \otimes^{\mathsf{R}} C \arrow[u, "\gamma \otimes^{\mathsf{R}} C"'] \\
(C \otimes^{\mathsf{L}} B) \otimes^{\mathsf{L}} A \arrow[r, "\alpha"'] & C \otimes^{\mathsf{L}} (B \otimes^{\mathsf{L}} A) \arrow[u, "\gamma"']
\end{tikzcd}$$

This definition combines concepts from skew bi-monoidal and bi-closed categories as introduced in [28].

In contrast to the categorical model of associative Lambek calculus, the monoidal bi-closed category, we do not have both left ($\backslash$) and right residuation ($/$), but instead have two right residuations corresponding to different tensor products. However, with the natural isomorphism γ, and selecting a specific tensor, we can simulate both left and right residuations.

In the remainder of the paper, we will develop axiomatic and sequent calculi for SkMBiC and explore its relational semantics.

4. Calculi for SkMBiC

By defining new formulae and adding rules in `LSkA`, we can have an axiomatic calculus `SkMBiCA`, where formulae (Fma) are inductively generated by the grammar $A, B ::= X \mid \mathsf{I} \mid A \otimes^{\mathsf{L}} B \mid A \multimap^{\mathsf{L}} B \mid A \otimes^{\mathsf{R}} B \mid A \multimap^{\mathsf{R}} B$.

X and I adhere to the definitions provided in Section 2, and $\otimes^{\mathsf{L}}$ and $\multimap^{\mathsf{L}}$ ($\otimes^{\mathsf{R}}$ and $\multimap^{\mathsf{R}}$) represent left (right) skew multiplicative conjunction and implication, respectively.
Derivations in `SkMBiCA` are inductively generated by the following rules:

$$\dfrac{}{A \vdash_{\mathsf{L}} A}\ \mathsf{id} \qquad \dfrac{A \vdash_{\mathsf{L}} B \quad B \vdash_{\mathsf{L}} C}{A \vdash_{\mathsf{L}} C}\ \mathsf{comp}$$

$$\dfrac{A \vdash_{\mathsf{L}} C \quad B \vdash_{\mathsf{L}} D}{A \otimes^{\mathsf{L}} B \vdash_{\mathsf{L}} C \otimes^{\mathsf{L}} D}\ \otimes^{\mathsf{L}}$$

$$\dfrac{C \vdash_{\mathsf{L}} A \quad B \vdash_{\mathsf{L}} D}{A \multimap^{\mathsf{L}} B \vdash_{\mathsf{L}} C \multimap^{\mathsf{L}} D}\ \multimap^{\mathsf{L}} \qquad \dfrac{C \vdash_{\mathsf{L}} A \quad B \vdash_{\mathsf{L}} D}{A \multimap^{\mathsf{R}} B \vdash_{\mathsf{L}} C \multimap^{\mathsf{R}} D}\ \multimap^{\mathsf{R}}$$

$$\dfrac{}{\mathsf{I} \otimes^{\mathsf{L}} A \vdash_{\mathsf{L}} A}\ \lambda \qquad \dfrac{}{A \vdash_{\mathsf{L}} A \otimes^{\mathsf{L}} \mathsf{I}}\ \rho \qquad \dfrac{}{(A \otimes^{\mathsf{L}} B) \otimes^{\mathsf{L}} C \vdash_{\mathsf{L}} A \otimes^{\mathsf{L}} (B \otimes^{\mathsf{L}} C)}\ \alpha$$

$$\dfrac{}{A \otimes^{\mathsf{L}} B \vdash_{\mathsf{L}} B \otimes^{\mathsf{R}} A}\ \gamma \qquad \dfrac{}{A \otimes^{\mathsf{R}} B \vdash_{\mathsf{L}} B \otimes^{\mathsf{L}} A}\ \gamma^{-1}$$

$$\dfrac{A \otimes^{\mathsf{L}} B \vdash_{\mathsf{L}} C}{\overline{A \vdash_{\mathsf{L}} B \multimap^{\mathsf{L}} C}}\ \pi \qquad \dfrac{A \otimes^{\mathsf{R}} B \vdash_{\mathsf{L}} C}{\overline{A \vdash_{\mathsf{L}} B \multimap^{\mathsf{R}} C}}\ \pi^{\mathsf{R}}$$

For any $f : A \vdash_{\mathsf{L}} B$ and $g : C \vdash_{\mathsf{L}} D$, we define $f \otimes^{\mathsf{R}} g$ as $\gamma \circ (g \otimes^{\mathsf{L}} f) \circ \gamma^{-1}$. λ^{R}, ρ^{R}, and α^{R} are also derivable.

One might note that the connective $\otimes^{\mathsf{R}}$ is not strictly necessary from a logical perspective, as it will be inter-definable with $\otimes^{\mathsf{L}}$ via the γ and γ^{-1} axioms above. However, we include both sets of connectives explicitly in the syntax.

Definition 4.1. The congruence relation on derivations in `SkMBiCA`is defined by the following:

$$\begin{array}{llll}
\text{(category laws)} & \mathsf{id} \circ f \doteq f & f \doteq f \circ \mathsf{id} & (f \circ g) \circ h \doteq f \circ (g \circ h) \\
(\otimes^{\mathsf{L}} \text{ functorial}) & \mathsf{id} \otimes^{\mathsf{L}} \mathsf{id} \doteq \mathsf{id} & & (h \circ f) \otimes^{\mathsf{L}} (k \circ g) \doteq h \otimes^{\mathsf{L}} k \circ f \otimes^{\mathsf{L}} g \\
(\multimap^{\mathsf{L}} \text{ functorial}) & \mathsf{id} \multimap^{\mathsf{L}} \mathsf{id} \doteq \mathsf{id} & & (f \circ h) \multimap^{\mathsf{L}} (k \circ g) \doteq h \multimap^{\mathsf{L}} k \circ f \multimap^{\mathsf{L}} g \\
(\multimap^{\mathsf{R}} \text{ functorial}) & \mathsf{id} \multimap^{\mathsf{R}} \mathsf{id} \doteq \mathsf{id} & & (f \circ h) \multimap^{\mathsf{R}} (k \circ g) \doteq h \multimap^{\mathsf{R}} k \circ f \multimap^{\mathsf{R}} g
\end{array}$$

(λ, ρ, α nat. trans.)

$$\begin{gathered}\lambda \circ \mathsf{id} \otimes^{\mathsf{L}} f \doteq f \circ \lambda \\ \rho \circ f \doteq f \otimes^{\mathsf{L}} \mathsf{id} \circ \rho \\ \alpha \circ (f \otimes^{\mathsf{L}} g) \otimes^{\mathsf{L}} h \doteq f \otimes^{\mathsf{L}} (g \otimes^{\mathsf{L}} h) \circ \alpha\end{gathered}$$

(Mac Lane axioms)

$$\begin{gathered}\lambda \circ \rho \doteq \mathsf{id} \qquad \mathsf{id} \doteq \mathsf{id} \otimes^{\mathsf{L}} \lambda \circ \alpha \circ \rho \otimes^{\mathsf{L}} \mathsf{id} \\ \lambda \circ \alpha \doteq \lambda \otimes^{\mathsf{L}} \mathsf{id} \qquad \alpha \circ \rho \doteq \mathsf{id} \otimes^{\mathsf{L}} \rho \\ \alpha \circ \alpha \doteq \mathsf{id} \otimes^{\mathsf{L}} \alpha \circ \alpha \circ \alpha \otimes^{\mathsf{L}} \mathsf{id}\end{gathered}$$

(γ isomorphism)

$$\gamma \circ \gamma^{-1} \doteq \mathsf{id} \qquad \gamma^{-1} \circ \gamma \doteq \mathsf{id}$$

($\pi^{(\mathsf{R})}$ nat. trans.)

$$\begin{gathered}\pi f \circ g \doteq \pi(f \circ (g \otimes^{\mathsf{L}} \mathsf{id})) \qquad \pi(f \circ g) \doteq (\mathsf{id} \multimap^{\mathsf{L}} f) \circ \pi g \\ \pi(\mathsf{id} \otimes^{\mathsf{L}} f) \doteq (f \multimap^{\mathsf{L}} \mathsf{id}) \circ \pi \mathsf{id} \quad \pi^{\mathsf{R}}(\mathsf{id} \otimes^{\mathsf{R}} f) \doteq (f \multimap^{\mathsf{R}} \mathsf{id}) \circ \pi^{\mathsf{R}} \mathsf{id} \\ \pi^{\mathsf{R}} f \circ g \doteq \pi^{\mathsf{R}}(f \circ (g \otimes^{\mathsf{R}} \mathsf{id})) \qquad \pi^{\mathsf{R}}(f \circ g) \doteq (\mathsf{id} \multimap^{\mathsf{R}} f) \circ \pi^{\mathsf{R}} g\end{gathered}$$

($\pi^{(\mathsf{R})}$ isomorphism)

$$\begin{gathered}\pi(\pi^{-1} f) \doteq f \qquad \pi^{-1}(\pi f) \doteq f \\ \pi^{\mathsf{R}}(\pi^{\mathsf{R}-1} f) \doteq f \qquad \pi^{\mathsf{R}-1}(\pi^{\mathsf{R}} f) \doteq f\end{gathered}$$

Similar to the constructions in [31, 30, 29, 32, 27], `SkMBiCA` generates the free SkMBiC (FSkMBiC(At)) over a set At in the following way:

- Objects of FSkMBiC(At) are formulae (Fma).
- Morphisms between formulae A and B are derivations of sequents $A \vdash_{\mathsf{L}} B$ and identified up to the congruence relation $\doteq$ in Definition 4.1. Notice that by the definition of $f \otimes^{\mathsf{R}} g$ and γ being an isomorphism, γ and γ^{-1} are natural transformations. For example, $\gamma \circ f \otimes^{\mathsf{L}} g \doteq \gamma \circ f \otimes^{\mathsf{L}} g \circ \mathsf{id} \doteq \gamma \circ f \otimes^{\mathsf{L}} g \circ \gamma^{-1} \circ \gamma = g \otimes^{\mathsf{R}} f \circ \gamma$. Similarly, naturality of $(\lambda^{\mathsf{R}}, \rho^{\mathsf{R}}, \alpha^{R})$ and the Mac Lane axioms corresponding to them hold as well.

Given a skew monoidal bi-closed category $\mathbb{D}$ with function $G : \mathsf{At} \to \mathbb{D}$, we can define functions $\overline{G}_0 : \mathsf{Fma} \to \mathbb{D}_0$ ($\mathbb{D}_0$ is the collection of objects in $\mathbb{D}$) and $\overline{G}_1 : \mathsf{FSkMBiC(At)}(A, B) \to \mathbb{D}(\overline{G}_0(A), \overline{G}_0(B))$ by induction on complexity of formulae and height of derivations respectively. This construction uniquely specifies a strict skew monoidal bi-closed functor $\overline{G} : \mathsf{FSkMBiC(At)} \to \mathbb{D}$ satisfying $\overline{G}(X) = G(X)$.

However, it remains unclear how to construct a sequent calculus à la Girard for SkMBiC.[2] A simpler scenario to consider is the sequent calculus for

---

[2] An anonymous reviewer has suggested that an Andreoli-style focusing calculus

right skew monoidal closed categories. In this context, recalling Definition 3.2, where natural transformations are in an opposite direction compared to left skew monoidal closed categories. One approach is to propose a dual sequent calculus to `LSkG`. Here, sequents would be of the form $\Gamma \mid S \vdash_{\mathsf{G}} A$, indicating a reversal of stoup and context, with all left rules applicable solely to the stoup. We should think of the antecedents as trees associating to the right, structured as $(A_n, (\ldots, (A_1, A_0)) \ldots)$. Nevertheless, $\multimap^{\mathsf{R}}$, by definition, is again a right residuation, implying that $\multimap^{\mathsf{R}}\mathsf{L}$ and $\multimap^{\mathsf{R}}\mathsf{R}$ should resemble those in `LSkG`. This requirement then necessitates contexts to appear on the right-hand side of the stoup.

Fortunately, we can develop a sequent calculus, denoted as `SkMBiCT`, which is inspired by `LSkT` to characterize SkMBiC categories. Specifically, `SkMBiCT` is an instantiation of Moortgat's multimodal Lambek calculus [21] with unit, semi-unital, and semi-associative structural rules.

Trees in `SkMBiCT` are inductively defined by the grammar $T ::= \mathsf{Fma} \mid - \mid (T,T) \mid (T;T)$. What we have defined are trees with two different ways of linking nodes: through the use of commas and semicolons, corresponding to $\otimes^{\mathsf{L}}$ and $\otimes^{\mathsf{R}}$, respectively. Contexts and substitution are defined analogously to those of `LSkT`. Sequents are in the form $T \vdash_{\mathsf{T}} A$ analogous to those in Section 2.

Derivations in `SkMBiCT` are generated recursively by the following rules:

$$\frac{}{A \vdash_{\mathsf{T}} A}\ \mathsf{ax} \qquad \frac{}{- \vdash_{\mathsf{T}} \mathsf{I}}\ \mathsf{IR} \qquad \frac{T[-] \vdash_{\mathsf{T}} C}{T[\mathsf{I}] \vdash_{\mathsf{T}} C}\ \mathsf{IL}$$

$$\frac{T[A,B] \vdash_{\mathsf{T}} C}{T[A \otimes^{\mathsf{L}} B] \vdash_{\mathsf{T}} C}\ \otimes^{\mathsf{L}}\mathsf{L} \qquad \frac{T \vdash_{\mathsf{T}} A \quad U \vdash_{\mathsf{T}} B}{T, U \vdash_{\mathsf{T}} A \otimes^{\mathsf{L}} B}\ \otimes^{\mathsf{L}}\mathsf{R}$$

$$\frac{U \vdash_{\mathsf{T}} A \quad T[B] \vdash_{\mathsf{T}} C}{T[A \multimap^{\mathsf{L}} B, U] \vdash_{\mathsf{T}} C}\ \multimap^{\mathsf{L}}\mathsf{L} \qquad \frac{T, A \vdash_{\mathsf{T}} B}{T \vdash_{\mathsf{T}} A \multimap^{\mathsf{L}} B}\ \multimap^{\mathsf{L}}\mathsf{R}$$

---

might also be possible, where the focused formula is not separated into a stoup but instead keeps its place within the sequence of antecedent formulae. This presents an interesting direction for future investigation, especially for studying the equational theory of derivations. The present work, however, concentrates on the explicit use of structural rules on tree structures to achieve the required flexibility.

$$\frac{T[U_0,(U_1,U_2)] \vdash_{\mathsf{T}} C}{T[(U_0,U_1),U_2] \vdash_{\mathsf{T}} C}\ \mathsf{assoc}^{\mathsf{L}} \quad \frac{T[U] \vdash_{\mathsf{T}} C}{T[-,U] \vdash_{\mathsf{T}} C}\ \mathsf{unitL}^{\mathsf{L}} \quad \frac{T[U,-] \vdash_{\mathsf{T}} C}{T[U] \vdash_{\mathsf{T}} C}\ \mathsf{unitR}^{\mathsf{L}}$$

$$\begin{array}{c} T[U_0,U_1] \vdash_{\mathsf{T}} C \\ \hline\hline T[U_1;U_0] \vdash_{\mathsf{T}} C \end{array}\ \otimes\mathsf{comm}$$

$$\frac{T[A;B] \vdash_{\mathsf{T}} C}{T[A \otimes^{\mathsf{R}} B] \vdash_{\mathsf{T}} C}\ \otimes^{\mathsf{R}}\mathsf{L} \quad \frac{T \vdash_{\mathsf{T}} A \quad U \vdash_{\mathsf{T}} B}{T;U \vdash_{\mathsf{T}} A \otimes^{\mathsf{R}} B}\ \otimes^{\mathsf{R}}\mathsf{R}$$

$$\frac{U \vdash_{\mathsf{T}} A \quad T[B] \vdash_{\mathsf{T}} C}{T[A \multimap^{\mathsf{R}} B;U] \vdash_{\mathsf{T}} C}\ \multimap^{\mathsf{R}}\mathsf{L} \quad \frac{T;A \vdash_{\mathsf{T}} B}{T \vdash_{\mathsf{T}} A \multimap^{\mathsf{R}} B}\ \multimap^{\mathsf{R}}\mathsf{R}$$

$$\frac{T[(U_0;U_1);U_2] \vdash_{\mathsf{T}} C}{T[U_0;(U_1;U_2)] \vdash_{\mathsf{T}} C}\ \mathsf{assoc}^{\mathsf{R}} \quad \frac{T[U] \vdash_{\mathsf{T}} C}{T[U;-] \vdash_{\mathsf{T}} C}\ \mathsf{unitL}^{\mathsf{R}} \quad \frac{T[-;U] \vdash_{\mathsf{T}} C}{T[U] \vdash_{\mathsf{T}} C}\ \mathsf{unitR}^{\mathsf{R}}$$

We can think of these rules as originating from two separate calculi: `LSkT` (the red part with ax, IR, and IL) and another for right skew monoidal closed categories (`RSkT`, the blue part with ax, IR, and IL), linked by ⊗comm, in other words, we can mimic all the blue rules in the style of `LSkT` (only commas appear in antecedents) and conversely, the red rules can be expressed using the blue rules. For example, we can express $\otimes^{\mathsf{R}}\mathsf{L}$, $\otimes^{\mathsf{R}}\mathsf{R}$ and $\multimap^{\mathsf{R}}\mathsf{L}$ in the style of `LSkT`:

$$\frac{T[A,B] \vdash_{\mathsf{T}} C}{T[B \otimes^{\mathsf{R}} A] \vdash_{\mathsf{T}} C}\ \otimes^{\mathsf{R}}\mathsf{L}' \quad = \quad \frac{\dfrac{T[A,B] \vdash_{\mathsf{T}} C}{T[B;A] \vdash_{\mathsf{T}} C}\ \otimes\mathsf{comm}}{T[B \otimes^{\mathsf{R}} A] \vdash_{\mathsf{T}} C}\ \otimes^{\mathsf{R}}\mathsf{L}$$

$$\frac{T \vdash_{\mathsf{T}} A \quad U \vdash_{\mathsf{T}} B}{U,T \vdash_{\mathsf{T}} A \otimes^{\mathsf{R}} B}\ \otimes^{\mathsf{R}}\mathsf{R}' \quad = \quad \frac{\dfrac{T \vdash_{\mathsf{T}} A \quad U \vdash_{\mathsf{T}} B}{T;U \vdash_{\mathsf{T}} A \otimes^{\mathsf{R}} B}\ \otimes^{\mathsf{R}}\mathsf{L}}{U,T \vdash_{\mathsf{T}} A \otimes^{\mathsf{R}} B}\ \otimes\mathsf{comm}$$

$$\frac{U \vdash_{\mathsf{T}} A \quad T[B] \vdash_{\mathsf{T}} C}{T[U,A \multimap^{\mathsf{R}} B] \vdash_{\mathsf{T}} C}\ \multimap^{\mathsf{R}}\mathsf{L}' \quad = \quad \frac{\dfrac{U \vdash_{\mathsf{T}} A \quad T[B] \vdash_{\mathsf{T}} C}{T[A \multimap^{\mathsf{R}} B;U] \vdash_{\mathsf{T}} C}\ \multimap^{\mathsf{R}}\mathsf{L}}{T[U,A \multimap^{\mathsf{R}} B] \vdash_{\mathsf{T}} C}\ \otimes\mathsf{comm}$$

$$\dfrac{A, T \vdash_{\mathsf{T}} B}{T \vdash_{\mathsf{T}} A \multimap^{\mathsf{R}} B}\ {\multimap^{\mathsf{R}}}\mathsf{R}' \quad = \quad \dfrac{\dfrac{A, T \vdash_{\mathsf{T}} B}{T ; A \vdash_{\mathsf{T}} B}\ \otimes\mathsf{comm}}{T \vdash_{\mathsf{T}} A \multimap^{\mathsf{R}} B}\ {\multimap^{\mathsf{R}}}\mathsf{R}$$

Theorem 4.2. *Similar to* `LSkT`*,* `cut` *is admissible in* `SkMBiCT`*.*

$$\dfrac{U \vdash_{\mathsf{T}} A \quad T[A] \vdash_{\mathsf{T}} C}{T[U] \vdash_{\mathsf{T}} C}\ \mathsf{cut}$$

Proof: The proof proceeds similarly to that of Theorem 2.5. For the new logical rules in blue, the proofs follow the same pattern as their red counterparts. Since $\otimes$comm and all the logical and structural rules in blue are one-premise left rules, we can permute `cut` upwards. ☐

The equivalence between `SkMBiCA` and `SkMBiCT` can be proved by induction on height of derivations with the following admissible rules, definition, and lemmata:

Definition 4.3. For any tree T, $T^{\#}$ is the formula obtained from T by replacing commas with $\otimes^{\mathsf{L}}$ and semicolons with $\otimes^{\mathsf{R}}$, and $-$ with I, respectively.

Lemma 4.4. *For any context* $T[\cdot]$ *and tree* U*,* $T[U]^{\#} = T[U^{\#}]^{\#}$*.*

Proof: The proof proceeds by induction on the structure of $T[\cdot]$.
If $T[\cdot] = [\cdot]$, then $[U]^{\#} = U^{\#}$ by the definition of substitution.
If $T[\cdot] = (T'[\cdot], T'')$, then by inductive hypothesis, we have $T'[U]^{\#} = T'[U^{\#}]^{\#}$ and by the definition of $()^{\#}$, we have $(T'[U], T'')^{\#} = T'[U]^{\#} \otimes^{\mathsf{L}} T''^{\#} = T'[U^{\#}]^{\#} \otimes^{\mathsf{L}} T''^{\#} = (T'[U^{\#}], T'')^{\#}$.
Other cases are similar. ☐

In the remainder of the paper, we will refer to uses of Lemma 4.4 by double lines.

LEMMA 4.5. *Given a context $T[\cdot]$ and a derivation $f : A \vdash_{\mathsf{L}} B$, the following rule is admissible:*

$$\dfrac{\overset{f}{A \vdash_{\mathsf{L}} B}}{T[A]^{\#} \vdash_{\mathsf{L}} T[B]^{\#}}\; T[f]^{\#}$$

PROOF: The proof proceeds by induction on the structure of $T[\cdot]$.
If $T[\cdot] = [\cdot]$, then we have $T[A]^{\#} = A$ and $T[B]^{\#} = B$, and f is the desired derivation.
If $T[\cdot] = (T'[\cdot]; T'')$, then we construct the desired derivation as follows:

$$\dfrac{\dfrac{\overset{f}{T'[A]^{\#} \vdash_{\mathsf{L}} T'[B]^{\#}} \quad \dfrac{}{T''^{\#} \vdash_{\mathsf{L}} T''^{\#}}\,\mathsf{id}}{T'[A]^{\#} \otimes^{\mathsf{R}} T''^{\#} \vdash_{\mathsf{L}} T'[B]^{\#} \otimes^{\mathsf{R}} T''^{\#}}\,\otimes^{\mathsf{R}}}{(T'[A]; T'')^{\#} \vdash_{\mathsf{L}} (T'[B]; T'')^{\#}}$$

The case $T[\cdot] = (T'; T''[\cdot])$ is symmetric, while other cases are covered in the proof of Lemma 2.8. □

THEOREM 4.6. `SkMBiCT` *is equivalent to* `SkMBiCA`*, meaning that the following two statements are true:*

1. *For any derivation $f : A \vdash_{\mathsf{L}} C$, there exists a derivation $\mathsf{A2T} f : A \vdash_{\mathsf{T}} C$.*
2. *For any derivation $f : T \vdash_{\mathsf{T}} C$, there exists a derivation $\mathsf{T2A} f : T^{\#} \vdash_{\mathsf{L}} C$.*

PROOF: We first construct A2T by structural induction on the derivation f.

Case $f = \mathsf{id}$.

$$\dfrac{}{A \vdash_{\mathsf{L}} A}\,\mathsf{id} \quad \mapsto \quad \dfrac{}{A \vdash_{\mathsf{T}} A}\,\mathsf{ax}$$

Case $f = \mathsf{comp}(f', f'')$.

$$\dfrac{\overset{f'}{A \vdash_{\mathsf{L}} B} \quad \overset{f''}{B \vdash_{\mathsf{L}} C}}{A \vdash_{\mathsf{L}} C}\,\mathsf{comp} \quad \mapsto \quad \dfrac{\overset{\mathsf{A2T}f'}{A \vdash_{\mathsf{T}} B} \quad \overset{\mathsf{A2T}f''}{B \vdash_{\mathsf{T}} C}}{A \vdash_{\mathsf{T}} C}\,\mathsf{cut}$$

Case $f = \otimes^{\mathsf{L}}(f', f'')$.

$$\dfrac{\overset{f'}{A \vdash_{\mathsf{L}} C} \quad \overset{f''}{B \vdash_{\mathsf{L}} D}}{A \otimes^{\mathsf{L}} B \vdash_{\mathsf{L}} C \otimes^{\mathsf{L}} D}\,\otimes^{\mathsf{L}} \quad \mapsto \quad \dfrac{\dfrac{\overset{\mathsf{A2T}f'}{A \vdash_{\mathsf{T}} C} \quad \overset{\mathsf{A2T}f''}{B \vdash_{\mathsf{T}} D}}{A, B \vdash_{\mathsf{T}} C \otimes^{\mathsf{L}} D}\,\otimes^{\mathsf{L}}\mathsf{R}}{A \otimes^{\mathsf{L}} B \vdash_{\mathsf{T}} C \otimes^{\mathsf{L}} D}\,\otimes^{\mathsf{L}}\mathsf{L}$$

Case $f = \multimap^{\mathsf{L}} (f', f'')$.

$$\dfrac{\overset{f'}{C \vdash_{\mathsf{L}} A} \quad \overset{f''}{B \vdash_{\mathsf{L}} D}}{A \multimap^{\mathsf{L}} B \vdash_{\mathsf{L}} C \multimap^{\mathsf{L}} D}\,\multimap^{\mathsf{L}} \quad \mapsto \quad \dfrac{\dfrac{\overset{\mathsf{A2T}f'}{C \vdash_{\mathsf{T}} A} \quad \overset{\mathsf{A2T}f''}{B \vdash_{\mathsf{T}} D}}{A \multimap^{\mathsf{L}} B, C \vdash_{\mathsf{T}} D}\,\multimap^{\mathsf{L}}\mathsf{L}}{A \multimap^{\mathsf{L}} B \vdash_{\mathsf{T}} C \multimap^{\mathsf{L}} D}\,\multimap^{\mathsf{L}}\mathsf{R}$$

Case $f = \lambda$.

$$\dfrac{}{\mathsf{I} \otimes^{\mathsf{L}} A \vdash_{\mathsf{L}} A}\,\lambda \quad \mapsto \quad \dfrac{\dfrac{\dfrac{\dfrac{}{A \vdash_{\mathsf{T}} A}\,\mathsf{ax}}{-, A \vdash_{\mathsf{T}} A}\,\mathsf{unitL}^{\mathsf{L}}}{\mathsf{I}, A \vdash_{\mathsf{T}} A}\,\mathsf{IL}}{\mathsf{I} \otimes^{\mathsf{L}} A \vdash_{\mathsf{T}} A}\,\otimes^{\mathsf{L}}\mathsf{L}$$

Case $f = \rho$.

$$\dfrac{}{A \vdash_{\mathsf{L}} A \otimes^{\mathsf{L}} \mathsf{I}}\,\rho \quad \mapsto \quad \dfrac{\dfrac{\dfrac{}{A \vdash_{\mathsf{T}} A}\,\mathsf{ax} \quad \dfrac{}{- \vdash_{\mathsf{T}} \mathsf{I}}\,\mathsf{IR}}{A, - \vdash_{\mathsf{T}} A \otimes^{\mathsf{L}} \mathsf{I}}\,\otimes^{\mathsf{R}}\mathsf{R}}{A \vdash_{\mathsf{T}} A \otimes^{\mathsf{L}} \mathsf{I}}\,\mathsf{unitR}^{\mathsf{L}}$$

Case $f = \alpha$.

$$\dfrac{}{(A \otimes^{\mathsf{L}} B) \otimes^{\mathsf{L}} C \vdash_{\mathsf{L}} A \otimes^{\mathsf{L}} (B \otimes^{\mathsf{L}} C)}\ \alpha$$

$$\mapsto\quad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{A \vdash_{\mathsf{T}} A}\ \mathsf{ax} \quad \dfrac{\dfrac{}{B \vdash_{\mathsf{T}} B}\ \mathsf{ax} \quad \dfrac{}{C \vdash_{\mathsf{T}} C}\ \mathsf{ax}}{B, C \vdash_{\mathsf{T}} B \otimes^{\mathsf{L}} C}\ \otimes^{\mathsf{L}}\mathsf{R}}{A, (B, C) \vdash_{\mathsf{T}} A \otimes^{\mathsf{L}} (B \otimes^{\mathsf{L}} C)}\ \otimes^{\mathsf{L}}\mathsf{R}}{(A, B), C \vdash_{\mathsf{T}} A \otimes^{\mathsf{L}} (B \otimes^{\mathsf{L}} C)}\ \mathsf{assoc}^{\mathsf{L}}}{(A \otimes^{\mathsf{L}} B), C \vdash_{\mathsf{T}} A \otimes^{\mathsf{L}} (B \otimes^{\mathsf{L}} C)}\ \otimes^{\mathsf{L}}\mathsf{L}}{(A \otimes^{\mathsf{L}} B) \otimes^{\mathsf{L}} C \vdash_{\mathsf{T}} A \otimes^{\mathsf{L}} (B \otimes^{\mathsf{L}} C)}\ \otimes^{\mathsf{L}}\mathsf{L}$$

Case $f = \gamma$.

$$\dfrac{}{A \otimes^{\mathsf{L}} B \vdash_{\mathsf{L}} B \otimes^{\mathsf{R}} A}\ \gamma \quad\mapsto\quad \dfrac{\dfrac{\dfrac{\dfrac{}{B \vdash_{\mathsf{T}} B}\ \mathsf{ax} \quad \dfrac{}{A \vdash_{\mathsf{T}} A}\ \mathsf{ax}}{B; A \vdash_{\mathsf{T}} B \otimes^{\mathsf{R}} A}\ \otimes^{\mathsf{R}}\mathsf{R}}{A, B \vdash_{\mathsf{T}} B \otimes^{\mathsf{R}} A}\ \otimes\mathsf{comm}}{A \otimes^{\mathsf{L}} B \vdash_{\mathsf{T}} B \otimes^{\mathsf{R}} A}\ \otimes^{\mathsf{L}}\mathsf{L}$$

Case $f = \gamma^{-1}$.

$$\dfrac{}{A \otimes^{\mathsf{R}} B \vdash_{\mathsf{L}} B \otimes^{\mathsf{L}} A}\ \gamma^{-1} \quad\mapsto\quad \dfrac{\dfrac{\dfrac{\dfrac{}{B \vdash_{\mathsf{T}} B}\ \mathsf{ax} \quad \dfrac{}{A \vdash_{\mathsf{T}} A}\ \mathsf{ax}}{B, A \vdash_{\mathsf{T}} B \otimes^{\mathsf{L}} A}\ \otimes^{\mathsf{L}}\mathsf{R}}{A; B \vdash_{\mathsf{T}} B \otimes^{\mathsf{L}} A}\ \otimes\mathsf{comm}^{-1}}{A \otimes^{\mathsf{R}} B \vdash_{\mathsf{T}} B \otimes^{\mathsf{L}} A}\ \otimes^{\mathsf{R}}\mathsf{L}$$

Case $f = \pi\ f'$.

$$\dfrac{\overset{f'}{A \otimes^{\mathsf{L}} B \vdash_{\mathsf{L}} C}}{A \vdash_{\mathsf{L}} B \multimap^{\mathsf{L}} C}\ \pi \quad\mapsto\quad \dfrac{\dfrac{\overset{\mathsf{A2T}\,f'}{A \otimes^{\mathsf{L}} B \vdash_{\mathsf{T}} C}}{A, B \vdash_{\mathsf{T}} C}\ \otimes^{\mathsf{L}}\mathsf{L}^{-1}}{A \vdash_{\mathsf{T}} B \multimap^{\mathsf{L}} C}\ \multimap^{\mathsf{L}}\mathsf{R}$$

Case $f = \pi^{-1}\ f'$.

$$\dfrac{\begin{array}{c} f' \\ A \vdash_{\mathsf{L}} B \multimap^{\mathsf{L}} C \end{array}}{A \otimes^{\mathsf{L}} B \vdash_{\mathsf{L}} C}\ \pi^{-1} \quad \mapsto \quad \dfrac{\dfrac{\begin{array}{c} \mathsf{A2T} f' \\ A \vdash_{\mathsf{T}} B \multimap^{\mathsf{L}} C \end{array}}{A, B \vdash_{\mathsf{T}} C}\ \multimap^{\mathsf{L}}\mathsf{R}^{-1}}{A \otimes^{\mathsf{L}} B \vdash_{\mathsf{T}} C}\ \otimes^{\mathsf{L}}\mathsf{L}$$

Other cases for $\multimap^{\mathsf{R}}$ and π^{R} are similar.
We construct T2A by structural induction on f as well.

Case $f = \mathsf{ax}$.

$$\dfrac{}{A \vdash_{\mathsf{T}} A}\ \mathsf{ax} \quad \mapsto \quad \dfrac{}{A \vdash_{\mathsf{L}} A}\ \mathsf{id}$$

Case $f = \mathsf{IR}$.

$$\dfrac{}{- \vdash_{\mathsf{T}} \mathsf{I}}\ \mathsf{IR} \quad \mapsto \quad \dfrac{}{\mathsf{I} \vdash_{\mathsf{L}} \mathsf{I}}\ \mathsf{id}$$

Case $f = \mathsf{IL}\ f'$.

$$\dfrac{\begin{array}{c} f' \\ T[-] \vdash_{\mathsf{T}} C \end{array}}{T[\mathsf{I}] \vdash_{\mathsf{T}} C}\ \mathsf{IL} \quad \mapsto \quad \dfrac{\begin{array}{c} \mathsf{T2A} f' \\ T[-]^{\#} \vdash_{\mathsf{L}} C \end{array}}{T[\mathsf{I}]^{\#} \vdash_{\mathsf{L}} C}$$

Case $f = \otimes\mathsf{comm}\ f'$

$$\dfrac{\begin{array}{c} f' \\ T[U_0, U_1] \vdash_{\mathsf{T}} C \end{array}}{T[U_1; U_0] \vdash_{\mathsf{T}} C}\ \otimes\mathsf{comm}$$

$$\mapsto \dfrac{\dfrac{\dfrac{\dfrac{}{U_1^{\#} \otimes^{\mathsf{R}} U_0^{\#} \vdash_{\mathsf{L}} U_0^{\#} \otimes^{\mathsf{L}} U_1^{\#}}\ \gamma^{-1}}{T[U_1^{\#} \otimes^{\mathsf{R}} U_0^{\#}]^{\#} \vdash_{\mathsf{L}} T[U_0^{\#} \otimes^{\mathsf{L}} U_1^{\#}]^{\#}}\ \text{Lemma 4.5}}{T[U_1; U_0]^{\#} \vdash_{\mathsf{L}} T[U_0, U_1]^{\#}} \qquad \begin{array}{c} \mathsf{T2Af'} \\ T[U_0, U_1]^{\#} \vdash_{\mathsf{L}} C \end{array}}{T[U_1; U_0]^{\#} \vdash_{\mathsf{L}} C}\ \mathsf{comp}$$

Case $f = \otimes\mathsf{L}\ f'$

$$\dfrac{\overset{f'}{T[A,B] \vdash_{\mathsf{T}} C}}{T[A \otimes^{\mathsf{L}} B] \vdash_{\mathsf{T}} C}\ \otimes^{\mathsf{L}}\mathsf{L} \quad\mapsto\quad \dfrac{\overset{\mathsf{T2A}f'}{T[A,B]^{\#} \vdash_{\mathsf{L}} C}}{T[A \otimes^{\mathsf{L}} B]^{\#} \vdash_{\mathsf{L}} C}$$

Case $f = \otimes^{\mathsf{L}}\mathsf{R}(f', f'')$.

$$\dfrac{\overset{f'}{T \vdash_{\mathsf{T}} A} \quad \overset{f''}{U \vdash_{\mathsf{T}} B}}{T, U \vdash_{\mathsf{T}} A \otimes^{\mathsf{L}} B}\ \otimes^{\mathsf{L}}\mathsf{R} \quad\mapsto\quad \dfrac{\dfrac{\overset{\mathsf{T2A}f'}{T^{\#} \vdash_{\mathsf{L}} A} \quad \overset{\mathsf{T2A}f''}{U^{\#} \vdash_{\mathsf{L}} B}}{T^{\#} \otimes^{\mathsf{L}} U^{\#} \vdash_{\mathsf{L}} A \otimes^{\mathsf{L}} B}\ \otimes^{\mathsf{L}}}{(T, U)^{\#} \vdash_{\mathsf{L}} A \otimes^{\mathsf{L}} B}$$

Case $f = \multimap^{\mathsf{L}}\mathsf{L}$.

$$\dfrac{\overset{f'}{U \vdash_{\mathsf{T}} A} \quad \overset{f''}{T[B] \vdash_{\mathsf{T}} C}}{T[A \multimap^{\mathsf{L}} B, U] \vdash_{\mathsf{T}} C}\ \multimap^{\mathsf{L}}\mathsf{L}$$

$$\mapsto \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{A \multimap^{\mathsf{L}} B \vdash_{\mathsf{L}} A \multimap^{\mathsf{L}} B}\ \mathsf{id} \quad \overset{\mathsf{T2A}f'}{U^{\#} \vdash_{\mathsf{L}} A}}{(A \multimap^{\mathsf{L}} B) \otimes^{\mathsf{L}} U^{\#} \vdash_{\mathsf{L}} (A \multimap^{\mathsf{L}} B) \otimes^{\mathsf{L}} A}\ \otimes^{\mathsf{L}}}{T[(A \multimap^{\mathsf{L}} B) \otimes^{\mathsf{L}} U^{\#}]^{\#} \vdash_{\mathsf{L}} T[(A \multimap^{\mathsf{L}} B) \otimes^{\mathsf{L}} A]^{\#}}\ \text{Lem. 4.5} \quad \dfrac{\dfrac{\dfrac{}{A \multimap^{\mathsf{L}} B \vdash_{\mathsf{L}} A \multimap^{\mathsf{L}} B}\ \mathsf{id}}{(A \multimap^{\mathsf{L}} B) \otimes^{\mathsf{L}} A \vdash_{\mathsf{L}} B}\ \pi^{-1}}{T[(A \multimap^{\mathsf{L}} B) \otimes^{\mathsf{L}} A]^{\#} \vdash_{\mathsf{L}} T[B]^{\#}}\ \text{Lem. 4.5}}{T[(A \multimap^{\mathsf{L}} B) \otimes^{\mathsf{L}} U^{\#}]^{\#} \vdash_{\mathsf{L}} T[B]^{\#}}\ \mathsf{comp}}{T[(A \multimap^{\mathsf{L}} B), U]^{\#} \vdash_{\mathsf{L}} T[B]^{\#}} \quad \overset{\mathsf{T2A}f''}{T[B]^{\#} \vdash_{\mathsf{L}} C}}{T[(A \multimap^{\mathsf{L}} B), U^{\#}]^{\#} \vdash_{\mathsf{L}} C}\ \mathsf{comp}$$

Case $f = \multimap^{\mathsf{L}}\mathsf{R}\ f'$

$$\dfrac{\overset{f'}{T, A \vdash_{\mathsf{T}} B}}{T \vdash_{\mathsf{T}} A \multimap^{\mathsf{L}} B}\ \multimap^{\mathsf{L}}\mathsf{R} \quad\mapsto\quad \dfrac{\overset{\mathsf{T2A}f'}{T^{\#} \otimes^{\mathsf{L}} A \vdash_{\mathsf{L}} B}}{T^{\#} \vdash_{\mathsf{L}} A \multimap^{\mathsf{L}} B}\ \pi$$

Case $f = \mathsf{assoc}^{\mathsf{L}}\ f'$

$$\dfrac{\overset{f'}{T[U_0,(U_1,U_2)] \vdash_{\mathsf{T}} C}}{T[(U_0,U_1),U_2] \vdash_{\mathsf{T}} C}\ \mathsf{assoc}^{\mathsf{L}}$$

$$\mapsto \dfrac{\dfrac{\dfrac{\dfrac{}{(U_0^{\#} \otimes^{\mathsf{L}} U_1^{\#}) \otimes^{\mathsf{L}} U_2^{\#} \vdash_{\mathsf{L}} U_0^{\#} \otimes^{\mathsf{L}} (U_1^{\#} \otimes^{\mathsf{L}} U_2^{\#})}\ \alpha}{T[(U_0^{\#} \otimes^{\mathsf{L}} U_1^{\#}) \otimes^{\mathsf{L}} U_2^{\#}]^{\#} \vdash_{\mathsf{L}} T[U_0^{\#} \otimes^{\mathsf{L}} (U_1^{\#} \otimes^{\mathsf{L}} U_2^{\#})]^{\#}}\ \text{Lemma 4.5}}{T[(U_0,U_1),U_2]^{\#} \vdash_{\mathsf{L}} T[U_0,(U_1,U_2)]^{\#}} \qquad \overset{\mathsf{T2A}f'}{T[U_0,(U_1,U_2)]^{\#} \vdash_{\mathsf{L}} C}}{T[(U_0,U_1),U_2]^{\#} \vdash_{\mathsf{T}} C}\ \mathsf{comp}$$

Case $f = \mathsf{unitL}^{\mathsf{L}}\ f'$

$$\dfrac{\overset{f'}{T[U] \vdash_{\mathsf{T}} C}}{T[-,U] \vdash_{\mathsf{T}} C}\ \mathsf{unitL}^{\mathsf{L}}$$

$$\mapsto \dfrac{\dfrac{\dfrac{\dfrac{}{\mathsf{I} \otimes^{\mathsf{L}} U^{\#} \vdash_{\mathsf{L}} U^{\#}}\ \lambda}{T[\mathsf{I} \otimes^{\mathsf{L}} U^{\#}]^{\#} \vdash_{\mathsf{L}} T[U^{\#}]^{\#}}\ \text{Lemma 4.5}}{T[-,U]^{\#} \vdash_{\mathsf{L}} T[U]^{\#}} \qquad \overset{\mathsf{T2A}f'}{T[U]^{\#} \vdash_{\mathsf{T}} C}}{T[-,U]^{\#} \vdash_{\mathsf{L}} C}\ \mathsf{comp}$$

Case $f = \mathsf{unitR}^{\mathsf{L}}\ f'$

$$\dfrac{\overset{f'}{T[U,-] \vdash_{\mathsf{T}} C}}{T[U] \vdash_{\mathsf{T}} C}\ \mathsf{unitR}^{\mathsf{L}}$$

$$\mapsto \dfrac{\dfrac{\dfrac{\dfrac{}{U^{\#} \vdash_{\mathsf{L}} U^{\#} \otimes^{\mathsf{L}} \mathsf{I}}\ \rho}{T[U^{\#}]^{\#} \vdash_{\mathsf{L}} T[U^{\#} \otimes^{\mathsf{L}} \mathsf{I}]^{\#}}\ \text{Lemma 4.5}}{T[U]^{\#} \vdash_{\mathsf{L}} T[U,-]^{\#}} \qquad \overset{\mathsf{T2A}f'}{T[U,-]^{\#} \vdash_{\mathsf{T}} C}}{T[U]^{\#} \vdash_{\mathsf{L}} C}\ \mathsf{comp}$$

Other cases for right skew rules are similar. □

5. Relational Semantics of `SkMBiCA` and Application

In this section, we present the relational semantics of `SkMBiCA`. Furthermore, the relational semantics for `SkMBiCA` is characterized modularly, allowing us to construct models for semi-substructural logics step by step by incorporating additional structural conditions into the frame. The modularity allows us to provide an algebraic proof for the main theorems concerning the interdefinability of a series of skew structured categories as discussed in [28].

Definition 5.1. A preordered ternary frame with a special subset is $\langle W, \leq, \mathbb{I}, \mathbb{L} \rangle$, where:

- W is a set of worlds, and $\leq$ is a preorder relation on W.
- $\mathbb{I}$ is a downwards closed subset of W.
- $\mathbb{L}$ is a ternary relation on W that is upwards closed in its first two arguments and downwards closed in its last argument with respect to $\leq$.

The intended meaning of a relation $\mathbb{L}abc$ is that worlds a (left daughter) and b (right daughter) combine to form world c (root). Notice that the unit is interpreted as a set of worlds $\mathbb{I}$, rather than a single world, to ensure that the interpretation can include all formulae provably equivalent to the unit, e.g. $\mathsf{I} \otimes \mathsf{I}$.

Definition 5.2. We list properties of ternary relations which we will focus on.

Left Skew Associativity (LSA)	$\forall a, b, c, d, x \in W, \mathbb{L}abx \;\&\; \mathbb{L}xcd$ $\longrightarrow \exists y \in W$ such that $\mathbb{L}bcy \;\&\; \mathbb{L}ayd$.
Left Skew Left Unitality (LSLU)	$\forall a, b \in W, e \in \mathbb{I}, \mathbb{L}eab \longrightarrow b \leq a$.
Left Skew Right Unitality (LSRU)	$\forall a \in W, \exists e \in \mathbb{I}$ such that $\mathbb{L}aea$.
Right Skew Associativity (RSA)	$\forall a, b, c, d, x \in W, \mathbb{L}bcx \;\&\; \mathbb{L}axd$ $\longrightarrow \exists y \in W$ such that $\mathbb{L}aby \;\&\; \mathbb{L}ycd$.
Right Skew Left Unitality (RSLU)	$\forall a \in W, \exists e \in \mathbb{I}$ such that $\mathbb{L}eaa$.
Right Skew Right Unitality (RSRU)	$\forall a, b \in W, e \in \mathbb{I}, \mathbb{L}aeb \longrightarrow b \leq a$.

Given another ternary relation $\mathbb{R}$, we define

$$\mathbb{LR}\text{-reverse} \quad \forall a, b, c \in W, \mathbb{L}abc \longleftrightarrow \mathbb{R}bac.$$

The associativity and unitality conditions are adapted from the theory of relational monoids [24] and relational semantics for Lambek calculus [12].

A `SkMBiCA` frame is a quintuple $\langle W, \leq, \mathbb{I}, \mathbb{L}, \mathbb{R}\rangle$, where $\mathbb{LR}$-reverse is satisfied, $\mathbb{L}$ satisfies LSA, LSLU, LSRU, and $\mathbb{R}$ automatically satisfies RSA, RSLU, RSRU because of $\mathbb{LR}$-reverse.

Unlike studies in `NL` e.g. [12, 21, 23], where two associativity conditions simultaneously hold for a relation or not, we explore two relations where one satisfies LSA and the other satisfies RSA. Another distinction from the existing studies on semantics for `NL` with unit [9] (or non-commutative linear logic [1]) is that while W is commonly assumed to be an unital magma (or monoid in the case of linear logic), here, we should consider that the unit behaves differently for different relations.

We denote the set of downwards closed subsets of W as $\mathcal{P}_{\downarrow}(W)$.

Definition 5.3. A function $v : \mathsf{Fma} \to \mathcal{P}_{\downarrow}(W)$ on a `SkMBiCA` frame is a valuation if it satisfies:

$$\begin{array}{ll}
v(\mathsf{I}) & = \mathbb{I} \\
v(A \otimes^{\mathsf{L}} B) & = \{c : \exists a \in v(A), \exists b \in v(B),\ \mathbb{L}abc\} \\
v(A \multimap^{\mathsf{L}} B) & = \{c : \forall a \in v(A), \forall b \in W,\ \mathbb{L}cab \Rightarrow b \in v(B)\} \\
v(A \otimes^{\mathsf{R}} B) & = \{c : \exists a \in v(A), \exists b \in v(B),\ \mathbb{R}abc\} \\
v(A \multimap^{\mathsf{R}} B) & = \{c : \forall a \in v(A), \forall b \in W,\ \mathbb{R}cab \Rightarrow b \in v(B)\}
\end{array}$$

Notice that $v(A \otimes^{\mathsf{L}} B)$ and $v(A \otimes^{\mathsf{R}} B)$ are downwards closed since $\mathbb{L}$ is downwards closed at its third argument. On the other hand, $v(A \multimap^{\mathsf{L}} B)$ and $v(A \multimap^{\mathsf{R}} B)$ are downwards closed since the first argument of $\mathbb{L}$ is upwards closed. For example, consider any $c \in v(A \multimap^{\mathsf{L}} B)$ and any $c' \in W$ with $c' \leq c$. If $\forall a \in v(A), \forall b \in W, \mathbb{L}c'ab$, then by upwards closedness of $\mathbb{L}$, we have $\mathbb{L}cab$ and then $b \in v(B)$, which further implies $c' \in v(A \multimap^{\mathsf{L}} B)$.

We define a `SkMBiCA` model to be a `SkMBiCA` frame with a valuation function, i.e. $\langle W, \leq, \mathbb{I}, \mathbb{L}, \mathbb{R}, v\rangle$. A sequent $A \vdash_{\mathsf{L}} B$ is valid in a model $\langle W, \leq, \mathbb{I}, \mathbb{L}, \mathbb{R}, v\rangle$ if $v(A) \subseteq v(B)$ and is valid in a frame if for any v for that frame, $v(A) \subseteq v(B)$.

THEOREM 5.4 (Soundness). *If a sequent* $A \vdash_{\mathsf{L}} B$ *is provable in* `SkMBiCA` *then it is valid in any* `SkMBiCA` *model.*

PROOF: The proof is adapted from [12, 23], where the cases of α and α^{R} have been discussed. Therefore, we only elaborate on new cases arising in `SkMBiCA`.

- If the derivation is the axiom $\lambda : \mathsf{I} \otimes^{\mathsf{L}} A \vdash_{\mathsf{L}} A$, then for any `SkMBiCA` model $\langle W, \mathbb{I}, \mathbb{L}, \mathbb{R}, v \rangle$ and any $a \in v(\mathsf{I} \otimes^{\mathsf{L}} A)$, there exist $e \in \mathbb{I}$, $a' \in v(A)$, and $\mathbb{L}ea'a$. By LSLU, we know that $a \leq a'$, and then $a \in v(A)$.
- If the derivation is the axiom $\rho : A \vdash_{\mathsf{L}} A \otimes^{\mathsf{L}} \mathsf{I}$, then for any `SkMBiCA` model $\langle W, \mathbb{I}, \mathbb{L}, \mathbb{R}, v \rangle$ and any $a \in v(A)$, by LSRU, there exists $e \in \mathbb{I}$ such that $\mathbb{L}aea$, which means that $a \in v(A \otimes^{\mathsf{L}} \mathsf{I})$.
- If the derivation is the axiom $\gamma : A \otimes^{\mathsf{L}} B \vdash_{\mathsf{L}} B \otimes^{\mathsf{R}} A$, then for any `SkMBiCA` model $\langle W, \mathbb{I}, \mathbb{L}, \mathbb{R}, v \rangle$ and any $c \in v(A \otimes^{\mathsf{L}} B)$, there exist $a \in v(A)$ and $b \in v(B)$ such that $\mathbb{L}abc$. By $\mathbb{L}\mathbb{R}$-reverse, we have $\mathbb{R}bac$, therefore $c \in v(B \otimes^{\mathsf{R}} A)$.
- The case of γ^{-1} is similar.

□

DEFINITION 5.5. The canonical model of `SkMBiCA`$_{\mathsf{e}}$ is $\langle W, \leq, \mathbb{I}, \mathbb{L}, \mathbb{R}, v \rangle$ where:

- $W = \mathsf{Fma}$ and $A \leq B$ if and only if $A \vdash_{\mathsf{L}} B$,
- $\mathbb{I} = v(\mathsf{I})$,
- $\mathbb{L}ABC$ if and only if $C \vdash_{\mathsf{L}} A \otimes^{\mathsf{L}} B$,
- $\mathbb{R}ABC$ if and only if $C \vdash_{\mathsf{L}} A \otimes^{\mathsf{R}} B$, and
- $v(A) = \{B : B \vdash_{\mathsf{L}} A$ is provable in `SkMBiCA`$\}$.

LEMMA 5.6. *The canonical model is a* `SkMBiCA` *model.*

Proof:

- The set $(\mathsf{Fma}, \vdash_{\mathsf{L}})$ is a preorder because of the rules id and comp, and the set $\mathbb{I}$ is downwards closed because of comp. The relations $\mathbb{L}$ and $\mathbb{R}$ are downwards closed in their last argument because of the rule comp. They are upwards closed in their first two arguments due to the rules $\otimes^{\mathsf{L}}$ and $\otimes^{\mathsf{R}}$, respectively. These facts ensure that $\langle \mathsf{Fma}, \vdash_{\mathsf{L}}, \mathbb{I}, \mathbb{L}, \mathbb{R} \rangle$ is a ternary frame.
- We show two cases (LSRU and LSRU) of the proof that $\mathbb{L}, \mathbb{R}$ satisfy their corresponding conditions, while other cases are similar.

  (LSLU) Given any two formulae A and B, and $J \in \mathbb{I}$ with $\mathbb{L}JAB$, we have $J \vdash_{\mathsf{L}} \mathsf{I}$, and $B \vdash_{\mathsf{L}} J \otimes^{\mathsf{L}} A$, then we can construct $B \vdash_{\mathsf{L}} A$ as follows:

$$\dfrac{\dfrac{B \vdash_{\mathsf{L}} J \otimes^{\mathsf{L}} A \quad \dfrac{J \vdash_{\mathsf{L}} \mathsf{I} \quad \dfrac{}{A \vdash_{\mathsf{L}} A}\,\mathsf{id}}{J \otimes^{\mathsf{L}} A \vdash_{\mathsf{L}} \mathsf{I} \otimes^{\mathsf{L}} A}\,\otimes^{\mathsf{L}}}{B \vdash_{\mathsf{L}} \mathsf{I} \otimes^{\mathsf{L}} A}\,\mathsf{comp} \quad \dfrac{}{\mathsf{I} \otimes^{\mathsf{L}} A \vdash_{\mathsf{L}} A}\,\lambda}{B \vdash_{\mathsf{L}} A}\,\mathsf{comp}$$

  (LSRU) By the axiom ρ, for any formula A, we have $A \vdash_{\mathsf{L}} A \otimes^{\mathsf{L}} \mathsf{I}$, i.e. $\mathbb{L}AIA$.
- The valuation v is downwards closed because of the rule comp. The other conditions on connectives are satisfied by definition.

Therefore, $\langle \mathsf{Fma}, \vdash_{\mathsf{L}}, \mathbb{I}, \mathbb{L}, \mathbb{R}, v \rangle$ is a `SkMBiCA` model. ☐

Theorem 5.7 (Completeness). *If $A \vdash_{\mathsf{L}} B$ is valid in any* `SkMBiCA` *model, then it is provable in* `SkMBiCA`*.*

Proof: If $A \vdash_{\mathsf{L}} B$ is valid in any `SkMBiCA` model, then it is valid in the canonical model, i.e. $v(A) \subseteq v(B)$ in the canonical model. From $A \vdash_{\mathsf{L}} A$, by definition of v, we have $A \in v(A)$, and because $v(A) \subseteq v(B)$, we know that $A \in v(B)$, therefore $A \vdash_{\mathsf{L}} B$. ☐

We show a correspondence between frame conditions and the validity of structural laws in frames.

Theorem 5.8. *For any ternary frame* $\langle W, \leq, \mathbb{I}, \mathbb{L}, \mathbb{R} \rangle$,

$$\begin{array}{ccccc}
 & \mathbb{LR}\textit{-reverse holds} & \longleftrightarrow & \gamma \textit{ and } \gamma^{-1} \textit{are valid} & \\
\alpha^{(\mathsf{R})} \textit{ valid} & \longleftrightarrow & \textit{LSA (RSA) holds} & \longleftrightarrow & L^{(\mathsf{R})} \textit{ valid} \\
\lambda^{(\mathsf{R})} \textit{ valid} & \longleftrightarrow & \textit{LSLU (RSLU) holds} & \longleftrightarrow & j^{(\mathsf{R})} \textit{ valid} \\
\rho^{(\mathsf{R})} \textit{ valid} & \longleftrightarrow & \textit{LSRU (RSRU) holds} & \longleftrightarrow & i^{(\mathsf{R})} \textit{ valid}
\end{array}$$

Proof: The first case is that $\mathbb{LR}$-reverse holds if and only if γ and γ^{-1} are valid, i.e. $v(A \otimes^{\mathsf{L}} B) = v(B \otimes^{\mathsf{R}} A)$.

($\longrightarrow$) For any $x \in v(A \otimes^{\mathsf{L}} B) \subseteq W$, there exists $a \in v(A), b \in v(B)$ and $\mathbb{L}abx$. By $\mathbb{LR}$-reverse, we have $\mathbb{R}bax$ meaning that $x \in v(B \otimes^{\mathsf{R}} A)$. The other way around is similar.

($\longleftarrow$) Suppose that for any v, A, B, we have $v(A \otimes^{\mathsf{L}} B) = v(B \otimes^{\mathsf{R}} A)$. Consider any $a, b, x \in W$ such that $\mathbb{L}abx$. We take $v(A) = a{\downarrow}$ and $v(B) = b{\downarrow}$ for some $A, B \in \mathsf{At}$. By the definition of v and assumption, x belongs to $v(A \otimes^{\mathsf{L}} B)$ which is equal to $v(B \otimes^{\mathsf{R}} A)$, therefore $\mathbb{R}bax$. The other direction is similar.

λ : LSLU holds if and only if λ is valid.

($\longrightarrow$) This is similar to case of λ in the proof of Theorem 5.4.

($\longleftarrow$) Suppose that λ is valid, i.e. for any A and v, we have $v(\mathsf{I} \otimes^{\mathsf{L}} A) \subseteq v(A)$. Consider any $a, b \in W$, $e \in \mathbb{I}$ such that $\mathbb{L}eab$. We take $v(A) = a{\downarrow}$ for some $A \in \mathsf{At}$. By $\mathbb{L}eab$ and the assumption, we know that $b \in v(A)$, which means that $b \leq a$.

ρ : LSRU holds if and only if ρ is valid.

($\longrightarrow$) This is similar to case of ρ in the proof of Theorem 5.4.

($\longleftarrow$) Suppose ρ is valid, i.e. for any A and v, $v(A) \subseteq v(A \otimes^{\mathsf{L}} \mathsf{I})$. Consider any $a \in W$. We take $v(A) = a{\downarrow}$ for some $A \in \mathsf{At}$. By the assumption, there exist $a' \in v(A)$ and $e \in \mathbb{I}$ such that $\mathbb{L}a'ea$.

Because $\mathbb{L}$ is upwards closed in its first argument, we know that $\mathbb{L}aea$.

α : LSA holds if and only if α is valid.

($\longrightarrow$) For any $s \in v((A \otimes^{\mathsf{L}} B) \otimes^{\mathsf{L}} C)$, there exists $a \in v(A), b \in v(B), x \in v(A \otimes^{\mathsf{L}} B), c \in v(C), \mathbb{L}abx$, and $\mathbb{L}xcs$. By LSA, there exists $y \in W$ such that $\mathbb{L}bcy$ and $\mathbb{L}ays$, then by definition of v, $y \in v(B \otimes^{\mathsf{L}} C)$ and $s \in v(A \otimes^{\mathsf{L}} (B \otimes^{\mathsf{L}} C))$.

($\longleftarrow$) Suppose that α is valid, i.e. for any A, B, C, v, we have $v((A \otimes^{\mathsf{L}} B) \otimes^{\mathsf{L}} C) \subseteq v(A \otimes^{\mathsf{L}} (B \otimes^{\mathsf{L}} C))$. Consider any $a, b, x, c, d \in W$ such that $\mathbb{L}abx$ and $\mathbb{L}xcd$. We take $v(A) = a{\downarrow}, v(B) = b{\downarrow}$, $v(C) = c{\downarrow}$ for some $A, B, C \in \mathsf{At}$, then we know that $x \in v(A \otimes^{\mathsf{L}} B)$ and $d \in v((A \otimes^{\mathsf{L}} B) \otimes^{\mathsf{L}} C)$. By the assumption, d belongs to $v(A \otimes^{\mathsf{L}} (B \otimes^{\mathsf{L}} C))$ as well, which means that there exist $a', b', y, c' \in W$ such that $\mathbb{L}b'c'y$ and $\mathbb{L}a'yd$. Because $\mathbb{L}$ is upwards closed in its first and second arguments, we have $\mathbb{L}bcy$ and $\mathbb{L}ayd$ as desired.

L : LSA holds if and only if for any A, B, C and v, $v(B \multimap^{\mathsf{L}} C) \subseteq v((A \multimap^{\mathsf{L}} B) \multimap^{\mathsf{L}} (A \multimap^{\mathsf{L}} C))$.

($\longrightarrow$) For any $s \in v(B \multimap^{\mathsf{L}} C)$, we show $s \in v((A \multimap^{\mathsf{L}} B) \multimap^{\mathsf{L}} (A \multimap^{\mathsf{L}} C))$. By definition, from assumptions $x \in v(A \multimap^{\mathsf{L}} B)$, $\mathbb{L}sxy$, $y \in v(A \multimap^{\mathsf{L}} C)$, $a \in A$, $c \in W$, and $\mathbb{L}yac$, we have to prove that $c \in C$. By LSA, there exists $x' \in W$ such that $\mathbb{L}xax'$ and $\mathbb{L}sx'c$. We get $x' \in B$ due to $x \in v(A \multimap^{\mathsf{L}} B)$. Thus, we have $c \in C$ because $s \in v(B \multimap^{\mathsf{L}} C)$.

($\longleftarrow$) Suppose that for any A, B, C and v, we have $v(B \multimap^{\mathsf{L}} C) \subseteq v((A \multimap^{\mathsf{L}} B) \multimap^{\mathsf{L}} (A \multimap^{\mathsf{L}} C))$. Consider $a, b, x, c, d \in W$ such that $\mathbb{L}abx$ and $\mathbb{L}xcd$. Take $v(A) = c{\downarrow}$, $v(B) = \{y : \mathbb{L}bcy\}$, and $v(C) = \{d' : \exists y \in v(B), \mathbb{L}ayd'\}$ for some $A, B, C \in \mathsf{At}$. Given any $y \in v(B)$ and any $d' \in W$, if $\mathbb{L}ayd'$, then by definition of $v(C)$, $d' \in v(C)$, therefore $a \in v(B \multimap^{\mathsf{L}} C)$. By assumption, $a \in v((A \multimap^{\mathsf{L}} B) \multimap^{\mathsf{L}} (A \multimap^{\mathsf{L}} C))$ as well, which means that, for any $b' \in v(A \multimap^{\mathsf{L}} B)$, $x' \in W$, $c' \in v(A)$ and $d' \in W$, if

$\mathbb{L}ab'x'$, then $x' \in v(A \multimap^{\mathsf{L}} C)$, and if $\mathbb{L}x'c'd'$, then $d' \in C$. By the definition of $v(B)$ and assumptions $\mathbb{L}abx$ and $\mathbb{L}xcd$, we have $b \in v(A \multimap^{\mathsf{L}} B)$, $x \in v(A \multimap^{\mathsf{L}} C)$, therefore $d \in v(C)$, which means that there exists $y \in W$ such that $\mathbb{L}bcy$ and $\mathbb{L}ayd$.

j^{R} : RSLU holds if and only if for any A, B and v, if $\mathbb{I} \subseteq v(A \multimap^{\mathsf{R}} B)$, then $v(A) \subseteq v(B)$.

($\longrightarrow$) By RSLU, for all $a \in v(A)$, there exists $e \in \mathbb{I}$ such that $\mathbb{R}eaa$, then we have $a \in v(B)$ because $e \in v(A \multimap^{\mathsf{R}} B)$.

($\longleftarrow$) Suppose that for any A, B and v, if $\mathbb{I} \subseteq v(A \multimap^{\mathsf{R}} B)$, then $v(A) \subseteq v(B)$. Consider any $a \in W$. We take $v(A) = a{\downarrow}$ and $v(B) = \{b : \exists e \in \mathbb{I}, \mathbb{R}eab\}$ for some $A, B \in \mathsf{At}$. For any $e' \in \mathbb{I}$, $a' \in v(A)$, and $b' \in W$, if $\mathbb{R}e'a'b'$, then because $\mathbb{R}$ is upwards closed in its second argument, we have $b' \in v(B)$, which means $e' \in v(A \multimap^{\mathsf{R}} B)$. Therefore $\mathbb{I} \subseteq v(A \multimap^{\mathsf{R}} B)$. From the assumption, we can now conclude that $v(A) \subseteq v(B)$. In particular, $a \in v(B)$, which means that there exists $e \in \mathbb{I}$ such that $\mathbb{R}eaa$.

L^{R} : RSA holds if and only if for any A, B, C, D and v, if $v(A) \subseteq v(B \multimap^{\mathsf{R}} (C \multimap^{\mathsf{R}} D))$ then there exists X such that $v(A) \subseteq v(X \multimap^{\mathsf{R}} D)$ and $v(B) \subseteq v(C \multimap^{\mathsf{R}} X)$.

($\longrightarrow$) We expand the assumption.

For any A, B, C, D, $a \in v(A)$, and $b, z \in W$, if $b \in v(B)$ and $\mathbb{R}abz$ then $z \in v(C \multimap^{\mathsf{R}} D)$ and for all $z \in v(C \multimap^{\mathsf{R}} D)$, for all $c, d \in W$ if $c \in v(C)$ and $\mathbb{R}zcd$, then $d \in v(D)$. In other words, for any $z, d \in W$, if there are $a \in v(A)$, $b \in v(B)$, $c \in v(C)$, $\mathbb{R}abz$, and $\mathbb{R}zcd$, then $d \in v(D)$.

We take $X = B \otimes^{\mathsf{R}} C$ and show it satisfies the two following statements:

- For any $a \in v(A)$, we show that $a \in v((B \otimes^{\mathsf{R}} C) \multimap^{\mathsf{R}} D)$. For any $x \in v(B \otimes^{\mathsf{R}} C)$ and $d \in W$, if $\mathbb{R}axd$, then by definition of $\otimes^{\mathsf{R}}$, we have $\mathbb{R}bcx$, where $b \in v(B)$ and $c \in v(C)$. By RSA, there exists $z \in W$ such that $\mathbb{R}abz$, and $\mathbb{R}zcd$. By the expanded assumption, $d \in v(D)$. Therefore $a \in v((B \otimes^{\mathsf{R}} C) \multimap^{\mathsf{R}} D)$.

- For any $b \in v(B)$, $c \in v(C)$, and $x \in W$, suppose $\mathbb{R}bcx$, then $x \in v(B \otimes^{\mathsf{R}} C)$ by definition of $\otimes^{\mathsf{R}}$. Therefore $b \in v(C \multimap^{\mathsf{R}} (B \otimes^{\mathsf{R}} C))$.

($\longleftarrow$) Assume that for any A, B, C, D and v, if $v(A) \subseteq v(B \multimap^{\mathsf{R}} (C \multimap^{\mathsf{R}} D))$, then there exists X such that $v(A) \subseteq v(X \multimap^{\mathsf{R}} D)$ and $v(B) \subseteq v(C \multimap^{\mathsf{R}} X)$. Suppose that we have $a, b, c, d, x \in W$ such that $\mathbb{R}axd$ and $\mathbb{R}bcx$, then we take $v(A) = a{\downarrow}$, $v(B) = b{\downarrow}$, $v(C) = c{\downarrow}$, and $v(D) = \{d' : \exists y, \mathbb{R}aby \& \mathbb{R}ycd'\}$ for some $A, B, C, D \in \mathsf{At}$. For any $a' \in v(A)$, given any $b' \in v(B)$, $x' \in W$, $c' \in v(C)$, $d' \in W$ such that $\mathbb{R}a'b'x'$ and $\mathbb{R}x'c'd'$. Because $\mathbb{R}$ is upwards closed in its first and second arguments, by the definition of $v(D)$, we have $d' \in v(D)$, which means $v(A) \subseteq v(B \multimap^{\mathsf{R}} (C \multimap^{\mathsf{R}} D))$. By the assumption, there exists X such that

1. $v(A) \subseteq v(X \multimap^{\mathsf{R}} D)$, which means that for any $a' \in v(A)$, given any $x' \in X$, $d' \in W$, if $\mathbb{R}a'x'd'$, then $d' \in v(D)$, and
2. $v(B) \subseteq v(C \multimap^{\mathsf{R}} X)$, which means that for any $b' \in v(B)$, given any $c' \in v(C)$ and $x' \in W$, if $\mathbb{R}b'c'x'$, then $x' \in v(X)$.

By $\mathbb{R}bcx$, and (2), we know that $x \in v(X)$. By $\mathbb{R}axd$, and (1), we know that $d \in v(D)$, which means that there exists $y \in W$ such that $\mathbb{R}aby$ and $\mathbb{R}ycd$.

The other cases are similar to the arguments above. □

A frame $\langle W, \leq, \mathbb{I}, \mathbb{L}\rangle$ is left (right) skew associative if $\mathbb{L}$ satisfies LSA (RSA). For other conditions, the naming is similar. If $\langle W, \leq, \mathbb{I}, \mathbb{L}\rangle$ satisfies LSA, LSLU, and LSRU (respectively RSA, RSLU, RSRU), then it is a left (respectively right) skew frame.

We can think of a `SkMBiCA` frame $\langle W, \leq, \mathbb{I}, \mathbb{L}, \mathbb{R}\rangle$ as a combination of two ternary frames $\langle W, \leq, \mathbb{I}, \mathbb{L}\rangle$ (left skew frame) and $\langle W, \leq, \mathbb{I}, \mathbb{R}\rangle$ (right skew frame) sharing the same set of possible worlds, where the ternary relations are interdefinable by $\mathbb{LR}$-reverse. Whenever $\mathbb{LR}$-reverse holds, then $\langle W, \leq, \mathbb{I}, \mathbb{L}\rangle$ is left skew if and only if $\langle W, \leq, \mathbb{I}, \mathbb{R}\rangle$ is right skew. In fact, we have:

$$\begin{array}{lcl} \langle W, \leq, \mathbb{I}, \mathbb{L}\rangle \text{ left skew associative} & \longleftrightarrow & \langle W, \leq, \mathbb{I}, \mathbb{R}\rangle \text{ right skew associative} \\ \langle W, \leq, \mathbb{I}, \mathbb{L}\rangle \text{ left skew left unital} & \longleftrightarrow & \langle W, \leq, \mathbb{I}, \mathbb{R}\rangle \text{ right skew right unital} \\ \langle W, \leq, \mathbb{I}, \mathbb{L}\rangle \text{ left skew right unital} & \longleftrightarrow & \langle W, \leq, \mathbb{I}, \mathbb{R}\rangle \text{ right skew left unital} \end{array}$$

If we state the structural laws semantically rather than syntactically, as in the sequent calculus `SkMBiCA`, we can reformulate Theorem 5.8 without referring to sequents and valuations. For example, we can define $\otimes^{\mathsf{L}}$ on downwards closed sets of worlds as $A \otimes^{\mathsf{L}} B = \{c : \exists a \in A \;\&\; \exists b \in B \;\&\; \mathbb{L}abc\}$ and express α as $(A \otimes^{\mathsf{L}} B) \otimes^{\mathsf{L}} C \subseteq A \otimes^{\mathsf{L}} (B \otimes^{\mathsf{L}} C)$. It is the case that α holds in a frame if and only if it satisfies LSA.

We construct a thin SkMBiC from the frame $\langle W, \leq, \mathsf{I}, \mathbb{L}, \mathbb{R}\rangle$ and provide algebraic proofs for the main theorems in [28]. The objects in the category are downwards closed subsets of W and for A, B, we have a map $A \to B$ if and only if $A \subseteq B$.

COROLLARY 5.9. The category $(\mathcal{P}_{\downarrow}(W), \subseteq)$ generated from any `SkMBiCA` frame is a thin SkMBiC.

A frame $\langle W, \leq, \mathbb{I}, \mathbb{L}\rangle$ is associative normal if it satisfies LSA and RSA simultaneously, and left (right) unital normal if LSLU and RSLU (LSRU and RSRU) are satisfied. Therefore, by Theorem 5.8, we have a thin version of the main results in [28].

COROLLARY 5.10. Given any frame, for the category $(\mathcal{P}_{\downarrow}(W), \subseteq)$ generated from the frame we have:

$$\begin{array}{lcl} (\mathbb{I}, \otimes^{\mathsf{L}}) \text{ left skew monoidal} & \longleftrightarrow & (\mathbb{I}, \multimap^{\mathsf{L}}) \text{ left skew closed} \\ (\mathbb{I}, \otimes^{\mathsf{R}}) \text{ right skew monoidal} & \longleftrightarrow & (\mathbb{I}, \multimap^{\mathsf{R}}) \text{ right skew closed} \end{array}$$

Moreover, if the frame satisfies $\mathbb{L}\mathbb{R}$-reverse then:

$$\begin{array}{lcl} (\mathbb{I}, \otimes^{\mathsf{L}}) \text{ left skew monoidal} & \longleftrightarrow & (\mathbb{I}, \otimes^{\mathsf{R}}) \text{ right skew monoidal} \\ (\mathbb{I}, \multimap^{\mathsf{L}}) \text{ left skew closed} & \longleftrightarrow & (\mathbb{I}, \multimap^{\mathsf{R}}) \text{ right skew closed} \\ (\mathbb{I}, \otimes^{\mathsf{L}}) \text{ associative normal} & \longleftrightarrow & (\mathbb{I}, \otimes^{\mathsf{R}}) \text{ associative normal} \\ (\mathbb{I}, \otimes^{\mathsf{L}}) \text{ left unital normal} & \longleftrightarrow & (\mathbb{I}, \otimes^{\mathsf{R}}) \text{ right unital normal} \end{array}$$

$$\begin{array}{lcl}
(\mathbb{I}, \otimes^{\mathsf{L}}) \text{ right unital normal} & \longleftrightarrow & (\mathbb{I}, \otimes^{\mathsf{R}}) \text{ left unital normal} \\
(\mathbb{I}, \multimap^{\mathsf{L}}) \text{ associative normal} & \longleftrightarrow & (\mathbb{I}, \multimap^{\mathsf{R}}) \text{ associative normal} \\
(\mathbb{I}, \multimap^{\mathsf{L}}) \text{ left unital normal} & \longleftrightarrow & (\mathbb{I}, \multimap^{\mathsf{R}}) \text{ right unital normal} \\
(\mathbb{I}, \multimap^{\mathsf{L}}) \text{ right unital normal} & \longleftrightarrow & (\mathbb{I}, \multimap^{\mathsf{R}}) \text{ left unital normal}
\end{array}$$

6. `SkMBiCA` with Symmetry

Two implications $\backslash$ and $/$ collapse into one in commutative Lambek calculus, i.e. for any formulae A and B, $A \backslash B$ is logically equivalent to B / A. In particular, consider an axiomatic presentation of non-associative Lambek calculus with exchange $\mathsf{ex} : A \otimes B \vdash_{\mathsf{L}} B \otimes A$, both $A \backslash B \vdash_{\mathsf{L}} B / A$ and $B / A \vdash_{\mathsf{L}} A \backslash B$ are provable. We adapt the notations in [23, Section 4] to fit in our discussion.

$$\dfrac{\dfrac{\dfrac{}{(A \backslash B) \otimes A \vdash_{\mathsf{L}} A \otimes (A \backslash B)}\,\mathsf{ex} \qquad \dfrac{\dfrac{}{A \backslash B \vdash_{\mathsf{L}} A \backslash B}\,\mathsf{id}}{A \otimes (A \backslash B) \vdash_{\mathsf{L}} B}\,\pi_{\backslash}^{-1}}{(A \backslash B) \otimes A \vdash_{\mathsf{L}} B}\,\mathsf{comp}}{A \backslash B \vdash_{\mathsf{L}} B / A}\,\pi_{/}$$

$$\dfrac{\dfrac{\dfrac{}{A \otimes (B / A) \vdash_{\mathsf{L}} (B / A) \otimes A}\,\mathsf{ex} \qquad \dfrac{\dfrac{}{(B / A) \vdash_{\mathsf{L}} B / A}\,\mathsf{id}}{(B / A) \otimes A \vdash_{\mathsf{L}} B}\,\pi_{/}^{-1}}{A \otimes (B / A) \vdash_{\mathsf{L}} B}\,\mathsf{comp}}{B / A \vdash_{\mathsf{L}} A \backslash B}\,\pi_{\backslash}$$

It leads to a natural question to consider semi-substructural logics with permutation. An immediate idea is to adding the following axiom to `LSkA`:

$$\dfrac{}{A \otimes B \vdash_{\mathsf{L}} B \otimes A}\,\mathsf{ex}$$

Following this axiom, we can define a derivable rule ex' that swaps any two adjacent formulae in the antecedent. This rule is defined through combinations of the axioms ex and id and the rules comp and $\otimes$. For example, given a derivation $f : (A \otimes B) \otimes C \vdash_{\mathsf{L}} D$ and the goal sequent $(B \otimes A) \otimes C \vdash_{\mathsf{L}} D$, we can use the derivable rule:

$$\dfrac{\overset{f}{(A\otimes B)\otimes C \vdash_{\mathsf{L}} D}}{(B\otimes A)\otimes C \vdash_{\mathsf{L}} D}\,\mathsf{ex}'$$

$$= \dfrac{\dfrac{\dfrac{}{B\otimes A\vdash_{\mathsf{L}} A\otimes B}\,\mathsf{ex} \qquad \dfrac{}{C\vdash_{\mathsf{L}} C}\,\mathsf{id}}{(B\otimes A)\otimes C \vdash_{\mathsf{L}} (A\otimes B)\otimes C}\,\otimes \qquad \overset{f}{(A\otimes B)\otimes C\vdash_{\mathsf{L}} D}}{(B\otimes A)\otimes C\vdash_{\mathsf{L}} D}\,\mathsf{comp}$$

However, as observed by Bourke and Lack [7], the axiom ex makes the calculus fully normal, i.e. λ^{-1}, ρ^{-1}, and α^{-1} are provable.

$$\lambda^{-1} \quad = \quad \dfrac{\dfrac{}{A\otimes \mathsf{I}\vdash_{\mathsf{L}} \mathsf{I}\otimes A}\,\mathsf{ex}\qquad \dfrac{}{\mathsf{I}\otimes A\vdash_{\mathsf{L}} A}\,\lambda}{A\otimes \mathsf{I}\vdash_{\mathsf{L}} A}\,\mathsf{comp}$$

$$\rho^{-1} \quad = \quad \dfrac{\dfrac{}{A\vdash_{\mathsf{L}} A\otimes \mathsf{I}}\,\rho\qquad \dfrac{}{A\otimes \mathsf{I}\vdash_{\mathsf{L}} \mathsf{I}\otimes A}\,\mathsf{ex}}{A\vdash_{\mathsf{L}} \mathsf{I}\otimes A}\,\mathsf{comp}$$

$$\alpha^{-1} \quad = \quad \dfrac{\dfrac{\dfrac{\dfrac{}{(C\otimes B)\otimes A\vdash_{\mathsf{L}} C\otimes(B\otimes A)}\,\alpha \qquad \dfrac{\dfrac{\dfrac{}{(A\otimes B)\otimes C\vdash_{\mathsf{L}} (A\otimes B)\otimes C}\,\mathsf{id}}{(B\otimes A)\otimes C\vdash_{\mathsf{L}} (A\otimes B)\otimes C}\,\mathsf{ex}'}{C\otimes(B\otimes A)\vdash_{\mathsf{L}} (A\otimes B)\otimes C}\,\mathsf{ex}'}{(C\otimes B)\otimes A\vdash_{\mathsf{L}} (A\otimes B)\otimes C}\,\mathsf{comp}}{(B\otimes C)\otimes A\vdash_{\mathsf{L}} (A\otimes B)\otimes C}\,\mathsf{ex}'}{A\otimes(B\otimes C)\vdash_{\mathsf{L}} (A\otimes B)\otimes C}\,\mathsf{ex}'$$

Therefore semi-substructural logics need a different treatment of commutativity.

Veltri has recently investigated the proof theory of *symmetric* left skew monoidal categories and *symmetric* left skew closed categories [32, 34]. These are variants of Mac Lane's symmetric monoidal categories and de Shippers' symmetric closed categories [11] which are originally introduced by Bourke and Lack [7] where the natural isomorphism representing symmetry involves *three* objects rather than two. Following the design of axiomatic calculus (called Hilbert-style calculus in the original papers) in Veltri's studies, where symmetry is represented by the following axioms (notations are modified to fit our discussion):

$$\frac{}{(A \otimes B) \otimes C \vdash_{\mathsf{L}} (A \otimes C) \otimes B}\ s \qquad \frac{}{B \multimap (A \multimap C) \vdash_{\mathsf{L}} A \multimap (B \multimap C)}\ s'$$

The axiom s is introduced for the axiomatic calculus of symmetric left skew monoidal categories where $\multimap$ is not present, while s' is the dual case for symmetric left skew closed categories.

These axioms only take care of symmetric left skew categories. In the remainder of the section, we first extend the proof-theoretical analysis to symmetric right skew and symmetric skew monoidal bi-closed categories. We will first introduce the definition of symmetric left (and right) skew monoidal closed categories, then prove the equivalence of the axioms of symmetry proof-theoretically. After that we introduce the commutative extension of `SkMBiCA` (`SkMBiCT`), called $\mathtt{SkMBiCA}_{\mathsf{e}}$ ($\mathtt{SkMBiCT}_{\mathsf{e}}$) and prove the equivalence of the axiomatic and tree calculi. Finally, we prove that $\mathtt{SkMBiCA}_{\mathsf{e}}$ is sound and complete with respect to the preordered ternary relation model and extend the correspondence theorem (Theorem 5.8) with axioms of symmetry.

Definition of symmetric left skew monoidal closed category:

Definition 6.1. A *symmetric left skew monoidal closed category* $\mathbb{C}$ is a left skew monoidal closed category equipped with a natural isomorphism $s_{A,B,C} : (A \otimes B) \otimes C \to (A \otimes C) \otimes B$ satisfying the equations in Figure 1.

Similar to left skew monoidal closed categories, left skew symmetric monoidal closed categories admit an equivalent characterization, i.e. the natural isomorphism s is bijective with the natural isomorphism $s' : B \multimap (A \multimap C) \to A \multimap (B \multimap C)$ [7]. In other words, s' correctly characterizes symmetry in a symmetric left skew *non-monoidal* closed category.

Definition 6.2. A *symmetric right skew monoidal closed category* $\mathbb{C}$ is a right skew monoidal closed category equipped with a natural isomorphism $s^{\mathsf{R}}_{A,B,C} : A \otimes (B \otimes C) \to B \otimes (A \otimes C)$ satisfying the equations in Figure 2, which are similar to the ones in Figure 1 with modified bracketing.

$$\begin{tikzcd}
((A \otimes B) \otimes C) \otimes D \arrow[r, "s_{A \otimes B,C,D}"] \arrow[d, "s_{A,B,C} \otimes D"'] & ((A \otimes B) \otimes D) \otimes C \arrow[r, "s_{A,B,D} \otimes C"] & ((A \otimes D) \otimes B) \otimes C \arrow[d, "s_{A \otimes D,B,C}"] \\
((A \otimes C) \otimes B) \otimes D \arrow[r, "s_{A \otimes C,B,D}"] & ((A \otimes C) \otimes D) \otimes B \arrow[r, "s_{A,C,D} \otimes B"] & ((A \otimes D) \otimes C) \otimes B
\end{tikzcd}$$

$$\begin{tikzcd}
((A \otimes B) \otimes C) \otimes D \arrow[r, "s_{A,B,C} \otimes D"] \arrow[d, "\alpha_{A \otimes B,C,D}"'] & ((A \otimes C) \otimes B) \otimes D \arrow[r, "s_{A \otimes C,B,D}"] & ((A \otimes C) \otimes D) \otimes B \arrow[d, "\alpha_{A,C,D} \otimes B"] \\
(A \otimes B) \otimes (C \otimes D) \arrow[rr, "s_{A,B,C \otimes D}"'] & & (A \otimes (C \otimes D)) \otimes B
\end{tikzcd}$$

$$\begin{tikzcd}
((A \otimes B) \otimes C) \otimes D \arrow[r, "s_{A \otimes B,C,D}"] \arrow[d, "\alpha_{A,B,C} \otimes D"'] & ((A \otimes B) \otimes D) \otimes C \arrow[r, "s_{A,B,D} \otimes C"] & ((A \otimes D) \otimes B) \otimes C \arrow[d, "\alpha_{A \otimes D,B,C}"] \\
(A \otimes (B \otimes C)) \otimes D \arrow[rr, "s_{A,B \otimes C,D}"'] & & (A \otimes D) \otimes (B \otimes C)
\end{tikzcd}$$

$$\begin{tikzcd}
((A \otimes B) \otimes C) \otimes D \arrow[r, "\alpha_{A,B,C} \otimes D"] \arrow[d, "s_{A \otimes B,C,D}"'] & (A \otimes (B \otimes C)) \otimes D \arrow[r, "\alpha_{A,B \otimes C,D}"] & A \otimes ((B \otimes C) \otimes D) \arrow[d, "A \otimes s_{B,C,D}"] \\
((A \otimes B) \otimes D) \otimes C \arrow[r, "\alpha_{A,B,D} \otimes C"] & (A \otimes (B \otimes D)) \otimes C \arrow[r, "\alpha_{A,B \otimes D,C}"] & A \otimes ((B \otimes D) \otimes C)
\end{tikzcd}$$

$$\begin{tikzcd}
& (A \otimes C) \otimes B \arrow[dr, "s_{A,C,B}"] & \\
(A \otimes B) \otimes C \arrow[ur, "s_{A,B,C}"] \arrow[rr, equal] & & (A \otimes B) \otimes C
\end{tikzcd}$$

Figure 1: Equations of morphisms in symmetric left skew monoidal closed category.

$$\begin{tikzcd}
A \otimes (B \otimes (C \otimes D)) \arrow[r, "s^{\mathsf{R}}_{A,B,C\otimes D}"] \arrow[d, "A \otimes s^{\mathsf{R}}_{B,C,D}"'] & B \otimes (A \otimes (C \otimes D)) \arrow[r, "B \otimes s^{\mathsf{R}}_{A,C,D}"] & B \otimes (C \otimes (A \otimes D)) \arrow[d, "s^{\mathsf{R}}_{B,C,A\otimes D}"] \\
A \otimes (C \otimes (B \otimes D)) \arrow[r, "s^{\mathsf{R}}_{A,C,B\otimes D}"] & C \otimes (A \otimes (B \otimes D)) \arrow[r, "C \otimes s^{\mathsf{R}}_{A,B,D}"] & C \otimes (B \otimes (A \otimes D))
\end{tikzcd}$$

$$\begin{tikzcd}
A \otimes (B \otimes (C \otimes D)) \arrow[r, "A \otimes s^{\mathsf{R}}_{A,C,B\otimes D}"] \arrow[d, "\alpha^{\mathsf{R}}_{A,B,C\otimes D}"'] & A \otimes (C \otimes (B \otimes D)) \arrow[r, "s^{\mathsf{R}}_{A,C,B\otimes D}"] & C \otimes (A \otimes (B \otimes D)) \arrow[d, "C \otimes \alpha^{\mathsf{R}}_{A,B,D}"] \\
(A \otimes B) \otimes (C \otimes D) \arrow[rr, "s^{\mathsf{R}}_{A\otimes B,C,D}"'] & & C \otimes ((A \otimes B) \otimes D)
\end{tikzcd}$$

$$\begin{tikzcd}
A \otimes (B \otimes (C \otimes D)) \arrow[r, "s^{\mathsf{R}}_{A,B,C\otimes D}"] \arrow[d, "A \otimes \alpha^{\mathsf{R}}_{B,C,D}"'] & B \otimes (A \otimes (C \otimes D)) \arrow[r, "B \otimes s^{\mathsf{R}}_{A,C,D}"] & B \otimes (C \otimes (A \otimes D)) \arrow[d, "\alpha^{\mathsf{R}}_{B,C,A\otimes D}"] \\
A \otimes ((B \otimes C) \otimes D) \arrow[rr, "s^{\mathsf{R}}_{A,B\otimes C,D}"'] & & (B \otimes C) \otimes (A \otimes D)
\end{tikzcd}$$

$$\begin{tikzcd}
A \otimes (B \otimes (C \otimes D)) \arrow[r, "A \otimes \alpha^{\mathsf{R}}_{B,C,D}"] \arrow[d, "s_{A\otimes B,C,D}"'] & A \otimes ((B \otimes C) \otimes D) \arrow[r, "\alpha^{\mathsf{R}}_{A,B\otimes C,D}"] & (A \otimes (B \otimes C)) \otimes D \arrow[d, "s^{\mathsf{R}}_{A,B,C\otimes D}"] \\
B \otimes (A \otimes (C \otimes D)) \arrow[r, "B \otimes \alpha^{\mathsf{R}}_{A,C,D}"] & B \otimes ((A \otimes C) \otimes D) \arrow[r, "\alpha^{\mathsf{R}}_{B,A\otimes C,D}"] & (B \otimes (A \otimes C)) \otimes D
\end{tikzcd}$$

$$\begin{tikzcd}
& B \otimes (A \otimes C) \arrow[dr, "s^{\mathsf{R}}_{B,A,C}"] & \\
A \otimes (B \otimes C) \arrow[ur, "s^{\mathsf{R}}_{A,B,C}"] \arrow[rr, equal] & & A \otimes (B \otimes C)
\end{tikzcd}$$

Figure 2: Equations of morphisms in symmetric right skew monoidal closed category.

There exists a bijective correspondence with natural isomorphisms $\mathsf{s'^R} : \int^Y \mathbb{C}(B, Y \multimap D) \times \mathbb{C}(A, C \multimap Y) \to \int^X \mathbb{C}(A, X \multimap D) \times \mathbb{C}(B, C \multimap X)$ in a symmetric right skew *non-monoidal* closed category. We prove the bijective correspondence between s and s^{R} and s' and s'^{R} proof-theoretically.

Theorem 6.3. *In an extension of* `LSkA`, *if*

$$\frac{}{(A \otimes B) \otimes C \vdash_{\mathsf{L}} (A \otimes C) \otimes B}\; s$$

is derivable in the calculus, then s' is derivable and vice versa.

Proof: <u>From s to s'.</u>

$$\dfrac{\dfrac{\dfrac{\dfrac{}{((B \multimap (A \multimap C)) \otimes A) \otimes B \vdash_{\mathsf{L}} ((B \multimap (A \multimap C)) \otimes B) \otimes A}\;\mathsf{s} \qquad \dfrac{\dfrac{\dfrac{}{B \multimap (A \multimap C) \vdash_{\mathsf{L}} B \multimap (A \multimap C)}\;\mathsf{ax}}{B \multimap (A \multimap C) \otimes B \vdash_{\mathsf{L}} A \multimap C}\;\pi^{-1}}{((B \multimap (A \multimap C)) \otimes B) \otimes A \vdash_{\mathsf{L}} C}\;\pi^{-1}}{((B \multimap (A \multimap C)) \otimes A) \otimes B \vdash_{\mathsf{L}} C}\;\mathsf{comp}}{(B \multimap (A \multimap C)) \otimes A \vdash_{\mathsf{L}} B \multimap C}\;\pi}{B \multimap (A \multimap C) \vdash_{\mathsf{L}} A \multimap (B \multimap C)}\;\pi$$

<u>From s' to s.</u>

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{(A \otimes C) \otimes B \vdash_{\mathsf{L}} (A \otimes C) \otimes B}\;\mathsf{ax}}{A \otimes C \vdash_{\mathsf{L}} B \multimap ((A \otimes C) \otimes B)}\;\pi}{A \vdash_{\mathsf{L}} C \multimap (B \multimap ((A \otimes C) \otimes B))}\;\pi \qquad \dfrac{}{C \multimap (B \multimap ((A \otimes C) \otimes B)) \vdash_{\mathsf{L}} B \multimap (C \multimap ((A \otimes C) \otimes B))}\;\mathsf{s'}}{A \vdash_{\mathsf{L}} B \multimap (C \multimap ((A \otimes C) \otimes B))}\;\mathsf{comp}}{A \otimes B \vdash_{\mathsf{L}} C \multimap ((A \otimes C) \otimes B)}\;\pi^{-1}}{(A \otimes B) \otimes C \vdash_{\mathsf{L}} (A \otimes C) \otimes B}\;\pi^{-1}$$

□

In this context, we overload the notations X and Y to represent unknown formulae rather than atomic ones.

Theorem 6.4. *In an extension of* `LSkA`, *if*

$$\frac{}{A \otimes (B \otimes C) \vdash_{\mathsf{L}} B \otimes (A \otimes C)}\; s^{\mathsf{R}}$$

is derivable, then the following statement holds:

s'^{R} : *If there exists a formula Y such that two sequents $B \vdash_{\mathsf{L}} Y \multimap D$ and $A \vdash_{\mathsf{L}} C \multimap Y$ hold, then there exists a formula X such that two sequents $A \vdash_{\mathsf{L}} X \multimap D$ and $B \vdash_{\mathsf{L}} C \multimap X$ hold.*

Conversely, if s'^{R} is true in the calculus, then s^{R} is derivable.

PROOF: From s^{R} to s'^{R}. Suppose that there exists a formula Y such that two sequents $B \vdash_{\mathsf{L}} Y \multimap D$ and $A \vdash_{\mathsf{L}} C \multimap Y$ hold, then we take $X = B \otimes C$ and construct the desired sequents $A \vdash_{\mathsf{L}} (B \otimes C) \multimap D$ and $B \vdash_{\mathsf{L}} C \multimap (B \otimes C)$ as follows:

$$\dfrac{\dfrac{\dfrac{}{A\otimes(B\otimes C)\vdash_{\mathsf{L}} B\otimes(A\otimes C)}\,s^{\mathsf{R}} \qquad \dfrac{\dfrac{\dfrac{}{B\vdash_{\mathsf{L}} B}\,\mathsf{id} \quad \dfrac{\overset{\text{Assumption}}{A\vdash_{\mathsf{L}} C\multimap Y}}{A\otimes C\vdash_{\mathsf{L}} Y}\,\pi^{-1}}{B\otimes(A\otimes C)\vdash_{\mathsf{L}} B\otimes Y}\,\otimes \quad \dfrac{\overset{\text{Assumption}}{B\vdash_{\mathsf{L}} Y\multimap D}}{B\otimes Y\vdash_{\mathsf{L}} D}\,\pi^{-1}}{B\otimes(A\otimes C)\vdash_{\mathsf{L}} D}\,\mathsf{comp}}{A\otimes(B\otimes C)\vdash_{\mathsf{L}} D}\,\mathsf{comp}}{A\vdash_{\mathsf{L}}(B\otimes C)\multimap D}\,\pi$$

$$\dfrac{\dfrac{}{B\otimes C\vdash_{\mathsf{L}} B\otimes C}\,\mathsf{id}}{B\vdash_{\mathsf{L}} C\multimap(B\otimes C)}\,\pi$$

Then the formula X is $B \otimes C$, where $B \vdash_{\mathsf{L}} C \multimap (B \otimes C)$ is derivable.
From s'^{R} to s'^{R}. To prove the sequent $A \otimes (B \otimes C) \vdash_{\mathsf{L}} B \otimes (A \otimes C)$, we start from the following two axiom sequents $\mathsf{id} : B \otimes (A \otimes C) \vdash_{\mathsf{L}} B \otimes (A \otimes C)$ and $\mathsf{id} : A \otimes C \vdash_{\mathsf{L}} A \otimes C$. By applying π on both sequents, we obtain $\pi\ \mathsf{id} : B \vdash_{\mathsf{L}} (A \otimes C) \multimap (B \otimes (A \otimes C))$ and $\pi\ \mathsf{id} : A \vdash_{\mathsf{L}} C \multimap (A \otimes C)$. We take $A \otimes C = Y$ to apply s'^{R}, then there exists a formula X such that two sequents $A \vdash_{\mathsf{L}} X \multimap (B \otimes (A \otimes C))$ and $B \vdash_{\mathsf{L}} C \multimap X$ hold. The desired derivation is constructed as follows:

$$\dfrac{\dfrac{\dfrac{}{A\vdash_{\mathsf{L}} A}\,\mathsf{id} \quad \dfrac{\overset{\text{By } s'^{\mathsf{R}}}{B\vdash_{\mathsf{L}} C\multimap X}}{B\otimes C\vdash_{\mathsf{L}} X}\,\pi^{-1}}{A\otimes(B\otimes C)\vdash_{\mathsf{L}} A\otimes X}\,\otimes \qquad \dfrac{\overset{\text{By } s'^{\mathsf{R}}}{A\vdash_{\mathsf{L}} X\multimap(B\otimes(A\otimes C))}}{A\otimes X\vdash_{\mathsf{L}} B\otimes(A\otimes C)}\,\pi^{-1}}{A\otimes(B\otimes C)\vdash_{\mathsf{L}} B\otimes(A\otimes C)}\,\mathsf{comp}$$

□

Definition 6.5. A *symmetric skew monoidal bi-closed category* SymSkMBiC is a skew monoidal bi-closed category with the left skew symmetry s. s^{R} is defined as $B \otimes^{\mathsf{L}} \gamma \circ \gamma \circ s \circ \gamma^{-1} \circ A \otimes^{\mathsf{R}} \gamma^{-1}$, diagrammatically:

$$\begin{tikzcd}
A \otimes^{\mathsf{R}} (B \otimes^{\mathsf{R}} C) \arrow[r, "A \otimes^{\mathsf{R}} \gamma^{-1}"] \arrow[d, "s^{\mathsf{R}}"'] & A \otimes^{\mathsf{R}} (C \otimes^{\mathsf{L}} B) \arrow[r, "\gamma^{-1}"] & (C \otimes^{\mathsf{L}} B) \otimes^{\mathsf{L}} A \arrow[d, "s"] \\
B \otimes^{\mathsf{R}} (A \otimes^{\mathsf{R}} C) & B \otimes^{\mathsf{R}} (C \otimes^{\mathsf{L}} A) \arrow[l, "B \otimes^{\mathsf{R}} \gamma"] & (C \otimes^{\mathsf{L}} A) \otimes^{\mathsf{L}} B \arrow[l, "\gamma"]
\end{tikzcd}$$

The axiomatic calculus that is sound and complete with respect to SymSkMBiC is $\mathtt{SkMBiCA_e}$ which is extended from `SkMBiCA` by adding the axiom:

$$\frac{}{(A \otimes^{\mathsf{L}} B) \otimes^{\mathsf{L}} C \vdash_{\mathsf{L}} (A \otimes^{\mathsf{L}} C) \otimes^{\mathsf{L}} B}\ s$$

The axiom s^{R} is defined by transforming the diagram in Definition 6.5 into a proof in $\mathtt{SkMBiCA_e}$, and then by Theorems 6.3 and 6.4, s' and s'^{R} are derivable in $\mathtt{SkMBiCA_e}$.

Moreover, we can construct the free SymSkMBiC (FSymSkMBiC(At)) over a set At by a similar construction of FSkMBiC(At) in Section 4:

- Objects of FSymSkMBiC(At) are formulae (Fma).
- Morphisms between A and B are derivations of sequents $A \vdash_{\mathsf{L}} B$ and identified up to the congruence relation $\doteq$ defined in Definition 4.1 with following additional equations:

$$\text{(sym. axioms)} \qquad \begin{array}{c} s \otimes^{\mathsf{L}} \mathsf{id} \circ s \circ s \otimes^{\mathsf{L}} \mathsf{id} \doteq s \circ s \otimes^{\mathsf{L}} \mathsf{id} \circ s \\ s \circ \alpha \doteq \alpha \otimes^{\mathsf{L}} \mathsf{id} \circ s \circ s \otimes^{\mathsf{L}} \mathsf{id} \quad s \circ \alpha \otimes^{\mathsf{L}} \mathsf{id} \doteq \alpha \circ s \otimes^{\mathsf{L}} \mathsf{id} \circ s \\ \alpha \circ \alpha \otimes^{\mathsf{L}} \mathsf{id} \circ s \doteq \mathsf{id} \otimes^{\mathsf{L}} s \circ \alpha \circ \alpha \otimes^{\mathsf{L}} \mathsf{id} \end{array}$$

$$(s \text{ symmetry}) \qquad s \circ s \doteq \mathsf{id}$$

On the other hand, the commutative extension of `SkMBiCT` ($\mathtt{SkMBiCT_e}$) is defined by adding the following two rules:

$$\frac{T[(U_0, U_1), U_2] \vdash_{\mathsf{T}} C}{T[(U_0, U_2), U_1] \vdash_{\mathsf{T}} C}\ \mathsf{ex}^{\mathsf{L}} \qquad \frac{T[U_0; (U_1; U_2)] \vdash_{\mathsf{T}} C}{T[U_1; (U_0; U_2)] \vdash_{\mathsf{T}} C}\ \mathsf{ex}^{\mathsf{R}}$$

A result similar to Theorems 6.3 and 6.4 can also be proved in $\mathtt{SkMBiCT_e}$. We adopt a symmetric presentation to emphasize that $\mathtt{SkMBiCT_e}$ should be viewed as a combination of two distinct calculi, connected through the rule $\otimes\mathsf{comm}$.

Moreover, $\mathtt{SkMBiCA_e}$ and $\mathtt{SkMBiCT_e}$ are equivalent.

THEOREM 6.6. *$\mathtt{SkMBiCA_e}$ is equivalent to $\mathtt{SkMBiCT_e}$, meaning that the following two statements hold:*

- *For any derivation $f : A \vdash_{\mathsf{L}} C$, there exists a derivation $\mathsf{A2T}f : A \vdash_{\mathsf{T}} C$.*
- *For any derivation $f : T \vdash_{\mathsf{T}} C$, there exists a derivation $\mathsf{T2A}f : T^{\#} \vdash_{\mathsf{L}} C$, where $T^{\#}$ transforms a tree into a formula by replacing commas with $\otimes^{\mathsf{L}}$ and semicolons with $\otimes^{\mathsf{R}}$, and $-$ with I, respectively.*

PROOF: We extend the proof of Theorem 4.6 by examining the additional cases of s (for $\mathsf{A2T}$) and ex^{L} and ex^{R} (for $\mathsf{T2A}$).

Case $f = s$

$$\frac{}{(A \otimes^{\mathsf{L}} B) \otimes^{\mathsf{L}} C \vdash_{\mathsf{L}} (A \otimes^{\mathsf{L}} C) \otimes^{\mathsf{L}} B}\ s$$

$$\mapsto \quad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{A \vdash_{\mathsf{T}} A}\ \mathsf{ax} \quad \dfrac{}{C \vdash_{\mathsf{T}} C}\ \mathsf{ax}}{A, C \vdash_{\mathsf{T}} A \otimes^{\mathsf{L}} C}\ \otimes^{\mathsf{L}}\mathsf{R} \quad \dfrac{}{B \vdash_{\mathsf{T}} B}\ \mathsf{ax}}{(A, C), B \vdash_{\mathsf{T}} (A \otimes^{\mathsf{L}} C) \otimes^{\mathsf{L}} B}\ \otimes^{\mathsf{L}}\mathsf{R}}{(A, B), C \vdash_{\mathsf{T}} (A \otimes^{\mathsf{L}} C) \otimes^{\mathsf{L}} B}\ \mathsf{ex}^{\mathsf{L}}}{(A \otimes^{\mathsf{L}} B), C \vdash_{\mathsf{T}} (A \otimes^{\mathsf{L}} C) \otimes^{\mathsf{L}} B}\ \otimes^{\mathsf{L}}\mathsf{L}}{(A \otimes^{\mathsf{L}} B) \otimes^{\mathsf{L}} C \vdash_{\mathsf{T}} (A \otimes^{\mathsf{L}} C) \otimes^{\mathsf{L}} B}\ \otimes^{\mathsf{L}}\mathsf{L}$$

<u>Case $f = \mathsf{ex}^{\mathsf{L}}\ f'$</u>

$$\dfrac{\overset{f'}{T[(U_0, U_1), U_2] \vdash_{\mathsf{T}} C}}{T[(U_0, U_2), U_1] \vdash_{\mathsf{T}} C}\ \mathsf{ex}^{\mathsf{L}}$$

$$\mapsto \dfrac{\begin{array}{c} \dfrac{\dfrac{}{(U_0^{\#} \otimes^{\mathsf{L}} U_2^{\#}) \otimes^{\mathsf{L}} U_1^{\#} \vdash_{\mathsf{T}} (U_0^{\#} \otimes^{\mathsf{L}} U_1^{\#}) \otimes^{\mathsf{L}} U_2^{\#}}\ s}{T[(U_0^{\#} \otimes^{\mathsf{L}} U_2^{\#}) \otimes^{\mathsf{L}} U_1^{\#}]^{\#} \vdash_{\mathsf{T}} T[(U_0^{\#} \otimes^{\mathsf{L}} U_1^{\#}) \otimes^{\mathsf{L}} U_2^{\#}]^{\#}}\ \text{Lemma 4.5} \\ \hline\hline T[(U_0, U_2), U_1]^{\#} \vdash_{\mathsf{L}} T[(U_0, U_1), U_2]^{\#} \end{array} \qquad \overset{\mathsf{T2A}f'}{T[(U_0, U_1), U_2]^{\#} \vdash_{\mathsf{T}} C}}{T[(U_0, U_2), U_1]^{\#} \vdash_{\mathsf{T}} C}\ \mathsf{comp}$$

<u>Case $f = \mathsf{ex}^{\mathsf{R}}\ f'$</u>

$$\dfrac{\overset{f'}{T[U_0; (U_1; U_2)] \vdash_{\mathsf{T}} C}}{T[U_1; (U_0; U_2)] \vdash_{\mathsf{T}} C}\ \mathsf{ex}^{\mathsf{R}}$$

$$\mapsto \dfrac{\begin{array}{c} \dfrac{\dfrac{}{U_1^{\#} (\otimes^{\mathsf{R}} U_0^{\#} \otimes^{\mathsf{R}} U_2^{\#}) \vdash_{\mathsf{T}} U_0^{\#} \otimes^{\mathsf{R}} (U_1^{\#} \otimes^{\mathsf{R}} U_2^{\#})}\ s^{\mathsf{R}}}{T[U_1^{\#} \otimes^{\mathsf{R}} (U_0^{\#} \otimes^{\mathsf{R}} U_2^{\#})]^{\#} \vdash_{\mathsf{T}} T[U_0^{\#} \otimes^{\mathsf{R}} (U_1^{\#} \otimes^{\mathsf{R}} U_2^{\#})]^{\#}}\ \text{Lemma 4.5} \\ \hline\hline T[U_1; (U_0; U_2)]^{\#} \vdash_{\mathsf{L}} T[U_0; (U_1; U_2)]^{\#} \end{array} \qquad \overset{\mathsf{T2A}f'}{T[U_0; (U_1; U_2)]^{\#} \vdash_{\mathsf{T}} C}}{T[U_1; (U_0; U_2)]^{\#} \vdash_{\mathsf{T}} C}\ \mathsf{comp}$$

$\square$

Recall that in commutative Lambek calculus (both associative and non-associative), the two implications collapse into one. However, this is not the case in either `SkMBiCA`$_{\mathsf{e}}$ or `SkMBiCT`$_{\mathsf{e}}$. Specifically, for any formulae A and B, neither of the sequents $A \multimap^{\mathsf{L}} B \vdash_i A \multimap^{\mathsf{R}} B$ nor $A \multimap^{\mathsf{R}} B \vdash_i A \multimap^{\mathsf{L}} B$ ($i \in \{\mathsf{L}, \mathsf{T}\}$) is provable. We demonstrate this non-provability first in the cut-free sequent calculus `SkMBiCT`$_{\mathsf{e}}$[3], by taking A and B as atomic formulae (a double question mark ?? means that no rule can be applied to close the derivation):

---

[3] The proof of cut admissibility for `SkMBiCT`$_{\mathsf{e}}$ is a straightforward extension of the proof of Theorem 4.2.

$$\dfrac{\dfrac{\overset{??}{(X \multimap^{\mathsf{L}} Y); X \vdash_{\mathsf{T}} Y}}{(X \multimap^{\mathsf{L}} Y) \otimes^{\mathsf{R}} X \vdash_{\mathsf{T}} Y}\ \otimes^{\mathsf{R}}\mathsf{L}}{X \multimap^{\mathsf{L}} Y \vdash_{\mathsf{T}} X \multimap^{\mathsf{R}} Y}\ \multimap^{\mathsf{R}}\mathsf{R} \qquad \dfrac{\dfrac{\overset{??}{(X \multimap^{\mathsf{R}} Y), X \vdash_{\mathsf{T}} Y}}{(X \multimap^{\mathsf{R}} Y) \otimes^{\mathsf{L}} X \vdash_{\mathsf{T}} Y}\ \otimes^{\mathsf{L}}\mathsf{L}}{X \multimap^{\mathsf{R}} Y \vdash_{\mathsf{T}} X \multimap^{\mathsf{L}} Y}\ \multimap^{\mathsf{R}}\mathsf{R}$$

By Theorem 6.6, we know both sequents are not provable in $\mathtt{SkMBiCA}_{\mathsf{e}}$ as well.

Lastly, we can analyze skew symmetry through the lens of ternary relational semantics and obtain a sound and complete model of $\mathtt{SkMBiCA}_{\mathsf{e}}$. Furthermore, we obtain the correspondence theorem of ternary frame conditions and validity of structural laws.

DEFINITION 6.7. We list the frame conditions properties of skew commutativity:

Left Skew Commutativity (LSC)	$\forall a, b, c, d, x \in W, \mathbb{L}abx \ \& \ \mathbb{L}xcd$ $\longrightarrow \exists y \in W$ s.t. $\mathbb{L}acy \ \& \ \mathbb{L}ybd.$
Right Skew Commutativity (RSC)	$\forall a, b, c, d, x \in W, \mathbb{L}bcx \ \& \ \mathbb{L}axd$ $\longrightarrow \exists y \in W$ s.t. $\mathbb{L}acy \ \& \ \mathbb{L}byd.$

A $\mathtt{SkMBiCA}_{\mathsf{e}}$ frame is a $\mathtt{SkMBiCA}$ frame where $\mathbb{L}$ additionally satisfies LSC, which implies $\mathbb{R}$ satisfies RSC. A $\mathtt{SkMBiCA}_{\mathsf{e}}$ model is a $\mathtt{SkMBiCA}_{\mathsf{e}}$ frame with a valuation function.

THEOREM 6.8 (Soundness). *If a sequent* $A \vdash_{\mathsf{L}} B$ *is provable in* $\mathtt{SkMBiCA}_{\mathsf{e}}$ *then it is valid in any* $\mathtt{SkMBiCA}_{\mathsf{e}}$ *model.*

PROOF: The proof is extended from the proof of Theorem 5.4 by examining one additional case, $f = s : (A \otimes^{\mathsf{L}} B) \otimes^{\mathsf{L}} C \vdash_{\mathsf{L}} (A \otimes^{\mathsf{L}} C) \otimes^{\mathsf{L}} B$. For any $\mathtt{SkMBiCA}_{\mathsf{e}}$ model $\langle W, \mathbb{I}, \mathbb{L}, \mathbb{R}, v\rangle$ and any $d \in v((A \otimes^{\mathsf{L}} B) \otimes^{\mathsf{L}} C)$, there exist $x \in v(A \otimes^{\mathsf{L}} B)$ and $c \in v(C)$ such that $\mathbb{L}xcd$. Moreover, there exist $a \in v(A)$ and $b \in v(B)$ such that $\mathbb{L}abx$. By LSC, we know that there exist $y \in W$ such that $\mathbb{L}acy$ and $\mathbb{L}ybd$, which means that $d \in v((A \otimes^{\mathsf{L}} C) \otimes^{\mathsf{L}} B)$.
□

Definition 6.9. The canonical model of `SkMBiCA` is $\langle W, \leq, \mathbb{I}, \mathbb{L}, \mathbb{R}, v \rangle$ where:

- $W = \mathsf{Fma}$ and $A \leq B$ if and only if $A \vdash_{\mathsf{L}} B$,
- $\mathbb{I} = v(\mathsf{I})$,
- $\mathbb{L}ABC$ if and only if $C \vdash_{\mathsf{L}} A \otimes^{\mathsf{L}} B$,
- $\mathbb{R}ABC$ if and only if $C \vdash_{\mathsf{L}} A \otimes^{\mathsf{R}} B$, and
- $v(A) = \{B : B \vdash_{\mathsf{L}} A$ is provable in $\mathtt{SkMBiCA}_{\mathsf{e}}\}$.

Lemma 6.10. *The canonical model is a* $\mathtt{SkMBiCA}_{\mathsf{e}}$ *model.*

Proof: The proof proceeds similarly to the proof of Lemma 5.6 but with one additional case showing that LSC is satisfied.
Given five formulae A, B, C, C', D and two derivations $f : C' \vdash_{\mathsf{L}} A \otimes^{\mathsf{L}} B$ and $g : D \vdash_{\mathsf{L}} C' \otimes^{\mathsf{L}} C$, then we take $A \otimes^{\mathsf{L}} C$ as the desired formula. The first desired sequent $A \otimes^{\mathsf{L}} C \vdash_{\mathsf{L}} A \otimes^{\mathsf{L}} C$ is derivable and the other desired sequent $D \vdash_{\mathsf{L}} (A \otimes^{\mathsf{L}} C) \otimes^{\mathsf{L}} B$ is constructed as follows:

$$\dfrac{\overset{g}{D \vdash_{\mathsf{L}} C' \otimes^{\mathsf{L}} C} \qquad \dfrac{\dfrac{\overset{f}{C' \vdash_{\mathsf{L}} A \otimes^{\mathsf{L}} B} \quad \dfrac{}{C \vdash_{\mathsf{L}} C}\,\mathsf{ax}}{C' \otimes^{\mathsf{L}} C \vdash_{\mathsf{L}} (A \otimes^{\mathsf{L}} B) \otimes^{\mathsf{L}} C}\,\otimes^{\mathsf{L}} \quad \dfrac{}{(A \otimes^{\mathsf{L}} C) \otimes^{\mathsf{L}} B \vdash_{\mathsf{L}} (A \otimes^{\mathsf{L}} B) \otimes^{\mathsf{L}} C}\,s}{C' \otimes^{\mathsf{L}} C \vdash_{\mathsf{L}} (A \otimes^{\mathsf{L}} C) \otimes^{\mathsf{L}} B}\,\mathsf{comp}}{D \vdash_{\mathsf{L}} (A \otimes^{\mathsf{L}} C) \otimes^{\mathsf{L}} B}\,\mathsf{comp}$$

□

Following the same argument in the proof of Theorem 5.7, we have:

Theorem 6.11 (Completeness). *If* $A \vdash_{\mathsf{L}} B$ *is valid in any* $\mathtt{SkMBiCA}_{\mathsf{e}}$ *model, then it is provable in* $\mathtt{SkMBiCA}_{\mathsf{e}}$.

Finally, we extend the correspondence between frame conditions and validity of structural laws to the symmetric case.

THEOREM 6.12. *For any ternary frame* $\langle W, \leq, \mathbb{I}, \mathbb{L}\rangle$,

$$\begin{array}{ccccc} s \textit{ valid} & \longleftrightarrow & \textit{LSC holds} & \longleftrightarrow & s' \textit{ valid} \\ s^{\mathsf{R}} \textit{ valid} & \longleftrightarrow & \textit{RSC holds} & \longleftrightarrow & s'^{\mathsf{R}} \textit{ valid} \end{array}$$

PROOF:

$s :$ LSC holds if and only if s is valid.

($\longrightarrow$) This is similar to the case of s in the proof of Theorem 6.8.

($\longleftarrow$) Suppose that s is valid, i.e. for any A, B, C, $v((A \otimes^{\mathsf{L}} B) \otimes^{\mathsf{L}} C) \subseteq v((A \otimes^{\mathsf{L}} C) \otimes^{\mathsf{L}} B)$. Consider any $a, b, c, d, x \in W$ such that $\mathbb{L}abx$ and $\mathbb{L}xcd$. We take $v(A) = a{\downarrow}, v(B) = b{\downarrow}$, $v(C) = c{\downarrow}$ for some $A, B, C \in \mathsf{At}$, then we know that $x \in v(A \otimes^{\mathsf{L}} B)$ and $d \in v((A \otimes^{\mathsf{L}} B) \otimes^{\mathsf{L}} C)$. By the assumption, $d \in v((A \otimes^{\mathsf{L}} C) \otimes^{\mathsf{L}} B)$ as well, which means that there exist $a' \in v(A)$, $c' \in v(C)$, $y \in v(A \otimes^{\mathsf{L}} C)$, and $b' \in v(B)$ such that $\mathbb{L}a'c'y$ and $\mathbb{L}yb'd$. Because $\mathbb{L}$ is upward closed in its first and second argument, we have $\mathbb{L}acy$ and $\mathbb{L}ybd$ as desired.

$s' :$ LSC holds if and only if s' is valid.

($\longrightarrow$) Suppose that LSC holds, we show that for any A, B, C, $v(B \multimap^{\mathsf{L}} (A \multimap^{\mathsf{L}} C)) \subseteq v(A \multimap^{\mathsf{L}} (B \multimap^{\mathsf{L}} C))$. Consider any $d \in v(B \multimap^{\mathsf{L}} (A \multimap^{\mathsf{L}} C))$. Assume that there exists $a \in v(A)$, $b \in v(B)$, and $x, c \in W$ such that $\mathbb{L}dax$ and $\mathbb{L}xbc$. Our goal is to prove that $c \in v(C)$. By LSC, there exists $y \in W$ such that $\mathbb{L}dby$ and $\mathbb{L}yac$, then by the assumption $d \in v(B \multimap^{\mathsf{L}} (A \multimap^{\mathsf{L}} C))$, we know that $c \in v(C)$.

($\longleftarrow$) Suppose that s' is valid, i.e. for any A, B, C, $v(B \multimap^{\mathsf{L}} (A \multimap^{\mathsf{L}} C)) \subseteq v(A \multimap^{\mathsf{L}} (B \multimap^{\mathsf{L}} C))$. Consider any $a, b, c, d, x \in W$ such that $\mathbb{L}abx$ and $\mathbb{L}xcd$. Take $v(A) = b{\downarrow}$, $v(B) = c{\downarrow}$, and $v(C) = \{d' : \exists y.\mathbb{L}acy \& \mathbb{L}ybd\}$ for some $A, B, C \in \mathsf{At}$. Consider any $c' \in v(B)$, $b' \in v(A)$, $y', d' \in W$, $\mathbb{L}ac'y'$ and $\mathbb{L}y'b'd'$. Because $\mathbb{L}$ is upwards closed in its second argument, we have $\mathbb{L}acy'$ and

$\mathbb{L}y'bd'$, which means that $d' \in v(C)$ and $y' \in v(A \multimap^{\mathsf{L}} C)$, therefore $a \in v(B \multimap^{\mathsf{L}} (A \multimap^{\mathsf{L}} C))$. By validity of s', $\mathbb{L}abx$, and $\mathbb{L}xcd$, we know that $x \in v(B \multimap^{\mathsf{L}} C)$ and $d \in v(C)$, i.e. there exists $y \in W$ such that $\mathbb{L}acy$ and $\mathbb{L}ybd$.

s^{R} : RSC holds if and only if s^{R} is valid.

- $(\longrightarrow)$ Suppose that RSC holds, we show that for any A, B, C, $v(A \otimes^{\mathsf{R}} (B \otimes^{\mathsf{R}} C)) \subseteq v(B \otimes^{\mathsf{R}} (A \otimes^{\mathsf{R}} C))$. Consider any $d \in v(A \otimes^{\mathsf{R}} (B \otimes^{\mathsf{R}} C))$. By definition, there exists $a \in v(A)$, $b \in v(B)$, $c \in v(C)$, $x \in v(B \otimes^{\mathsf{R}} C)$ such that $\mathbb{L}bcx$ and $\mathbb{L}axd$. By RSC, there exists $y \in W$ such that $\mathbb{L}acy$ and $\mathbb{L}byd$, then by definition, we know that $y \in v(A \otimes^{\mathsf{R}} C)$ and therefore $d \in v(B \otimes^{\mathsf{R}} (A \otimes^{\mathsf{R}} C))$.
- $(\longleftarrow)$ Suppose that s^{R} is valid. Consider any $a, b, c, d, x \in W$ such that $\mathbb{L}bcx$ and $\mathbb{L}axd$. We take $v(A) = a{\downarrow}, v(B) = b{\downarrow}$, $v(C) = c{\downarrow}$ for some $A, B, C \in \mathsf{At}$, then we know that $x \in v(B \otimes^{\mathsf{R}} C)$ and $d \in v(A \otimes^{\mathsf{R}} (B \otimes^{\mathsf{R}} C))$. By the assumption, $d \in v(B \otimes^{\mathsf{R}} (A \otimes^{\mathsf{R}} C))$ as well, which means that that there exist $a', b', y, c' \in W$ such that $\mathbb{L}a'c'y$ and $\mathbb{L}b'yd$. Because $\mathbb{L}$ is upwards closed in its first and second argument, we have $\mathbb{L}acy$ and $\mathbb{L}byd$ as desired.

s'^{R} : RSC holds if and only if s'^{R} is valid.

- $(\longrightarrow)$ Suppose that RSC holds, we show that for any formulae A, B, C, D, if there exists a formula Y such that $v(B) \subseteq v(Y \multimap^{\mathsf{R}} D)$ and $v(A) \subseteq v(C \multimap^{\mathsf{R}} Y)$ then there exists a formula X such that $v(A) \subseteq v(X \multimap^{\mathsf{R}} D)$ and $v(B) \subseteq v(C \multimap^{\mathsf{R}} X)$. Take $X = B \otimes^{\mathsf{R}} C$, then clearly $v(B) \subseteq v(C \multimap^{\mathsf{R}} (B \otimes^{\mathsf{R}} C))$. For any $a \in v(A)$, if there exist $x \in v(B \multimap^{\mathsf{R}} C)$ and $d \in W$ such that $\mathbb{L}axd$, then by definition, there exist $b \in v(B)$ and $c \in v(C)$ such that $\mathbb{L}bcx$. By RSC, there exists $y \in W$ such that $\mathbb{L}acy$ and $\mathbb{L}byd$, then by $v(A) \subseteq v(C \multimap^{\mathsf{R}} Y)$, we have $y \in v(Y)$, and further by $v(B) \subseteq v(Y \multimap^{\mathsf{R}} D)$, we have $d \in v(D)$.
- $(\longleftarrow)$ Suppose that s'^{R} is valid. Consider any $a, b, c, d, x \in W$ such that $\mathbb{L}bcx$ and $\mathbb{L}axd$. Take $v(A) = a{\downarrow}$, $v(B) = b{\downarrow}$, $v(C) = c{\downarrow}$,

and $v(D) = \{d' : \exists y.\mathbb{L}acy \& \mathbb{L}byd\}$ for some $A, B, C, D \in \mathsf{At}$. Clearly, $v(A)$ is a subset of $v(C \multimap^{\mathsf{R}} (A \otimes^{\mathsf{R}} C))$. For any $b' \in v(B)$, if there exist $y' \in v(A \otimes^{\mathsf{R}} C)$ and $d' \in W$ and $\mathbb{L}b'y'd'$, then by definition, there exist $a' \in v(A)$ and $c' \in v(C)$ such that $\mathbb{L}a'c'y'$. Because $\mathbb{L}$ is upwards closed in its first and second argument, we have $\mathbb{L}acy'$ and $\mathbb{L}by'd'$, which means that $d' \in v(D)$ and therefore $v(B) \in v((A \otimes^{\mathsf{R}} C) \multimap^{\mathsf{R}} D)$. Take $Y = A \otimes^{\mathsf{R}} C$, then by s'^{R}, there exists a formula X such that $v(A) \subseteq v(X \multimap^{\mathsf{R}} D)$ and $v(B) \subseteq v(C \multimap^{\mathsf{R}} X)$. By $b \in v(C \multimap^{\mathsf{R}} X)$ and $\mathbb{L}bcx$, we have $x \in v(X)$. By $a \in v(X \multimap^{\mathsf{R}} D)$ and $\mathbb{L}axd$, we have $d \in v(D)$, which means that there exists $y \in W$ such that $\mathbb{L}acy$ and $\mathbb{L}byd$, as desired. □

7. Concluding remarks

This paper discusses sequent calculi for (symmetric) left (right) skew monoidal categories and (symmetric) skew monoidal bi-closed categories in the style of non-associative Lambek calculus. Compared to the sequent calculi with stoup, the calculi à la Lambek are more flexible in the sense that the sequent calculi for right skew monoidal closed categories (`RSkT`) and skew monoidal bi-closed categories (`SkMBiCT`) can be formulated. Moreover, we show that they are cut-free and equivalent to the calculus with stoup (Theorem 2.11) and the axiomatic calculus (Theorem 4.6).

Moreover, we discuss the relational semantics of `SkMBiCA` ($\mathtt{SkMBiCA}_{\mathsf{e}}$) via the ternary frame $\langle W, \leq, \mathbb{I}, \mathbb{L}, \mathbb{R}\rangle$ where $\mathbb{L}$ and $\mathbb{R}$ are connected by $\mathbb{L}\mathbb{R}$-reverse and therefore if $\mathbb{L}$ satisfies left skew structural conditions then $\mathbb{R}$ satisfies right skew structural conditions automatically. By Theorem 5.8, for any `SkMBiCA` model, we can construct a thin skew monoidal bi-closed category $(\mathcal{P}_{\downarrow}(W), \subseteq)$ and obtain algebraic proofs of the main theorems in [28].

A deeper exploration of symmetric right skew closed categories remains as future work, particularly in identifying appropriate coherence conditions without relying on monoidal structures. This investigation will be built upon the foundational work by Day and Laplaza [10], who explored a hierarchy of closed categories, from symmetric monoidal closed through symmetric closed and closed, to non-associative closed categories. Their research provided concrete examples where the Day convolution version of structural laws are not bijective. This approach will extend the framework by studying symmetric skew closed categories.

In Section 6, we established results for the special case of posetal (thin) symmetric skew monoidal bi-closed categories, where there is at most one morphism between any pair of objects. The natural progression is to extend these results to non-posetal categories, requiring the coherence conditions for symmetric right skew closed categories. This extension will extend the Eilenberg-Kelly theorem [13, 28] to the symmetric skew monoidal closed categories.

Another possible future direction is to incorporate modalities (exponentials in linear logical terminology) into semi-substructural logic as in [21] (modalities) and [4] (subexponentials) into non-associative Lambek calculus and non-commutative and non-associative linear logic.

Similar to the equational theories for `SkMBiCA` discussed in Section 4, we also plan to investigate the equational theories on the derivations of `LSkT` and `SkMBiCT` in the future as well as their commutative version.

Acknowledgements We thank Giulio Fellin, Tarmo Uustalu, and Niccolò Veltri for invaluable discussions and the anonymous reviewers for constructive feedback and comments. Special thanks to Tarmo Uustalu and Niccolò Veltri for thorough review, for highlighting some inaccuracies in the draft, and their assistance in resolving these issues. This work was supported by the Estonian Research Council grant PSG749.

References

[1] V. M. Abrusci, *Non-commutative intuitionistic linear logic*, **Mathematical Logic Quarterly**, vol. 36(4) (1990), pp. 297–318, DOI: https://doi.org/10.1002/malq.19900360405.

[2] T. Altenkirch, J. Chapman, T. Uustalu, *Monads need not be endofunctors*, **Logical Methods in Computer Science**, vol. 11(1) (2015), 3, DOI: https://doi.org/10.2168/lmcs-11(1:3)2015.

[3] J.-M. Andreoli, *Logic programming with focusing proofs in linear logic*, **Journal of Logic and Computation**, vol. 2(3) (1992), pp. 297–347, DOI: https://doi.org/10.1093/logcom/2.3.297.

[4] E. Blaisdell, M. Kanovich, S. L. Kuznetsov, E. Pimentel, A. Scedrov, *Non-associative, Non-commutative Multi-modal Linear Logic*, [in:] J. Blanchette, L. Kovács, D. Pattinson (eds.), **Proceedings of the 11th International Joint Conference on Automated Reasoning, IJCAR 2022**, vol. 13385 of Lecture Notes in Computer Science, Springer (2022), pp. 449–467, DOI: https://doi.org/10.1007/978-3-031-10769-6_27.

[5] J. Bourke, *Skew structures in 2-category theory and homotopy theory*, **Journal of Homotopy and Related Structures**, vol. 12(1) (2017), pp. 31–81, DOI: https://doi.org/10.1007/s40062-015-0121-z.

[6] J. Bourke, S. Lack, *Skew monoidal categories and skew multicategories*, **Journal of Algebra**, vol. 506 (2018), pp. 237–266, DOI: https://doi.org/10.1016/j.jalgebra.2018.02.039.

[7] J. Bourke, S. Lack, *Braided skew monoidal categories*, **Theory and Applications of Categories**, vol. 35(2) (2020), pp. 19–63, URL: http://www.tac.mta.ca/tac/volumes/35/2/35-02abs.html.

[8] M. Buckley, R. Garner, S. Lack, R. Street, *The Catalan simplicial set*, **Mathematical Proceedings of Cambridge Philosophical Society**, vol. 158(2) (2015), pp. 211–222, DOI: https://doi.org/10.1017/s0305004114000498.

[9] M. Bulińska, *On the complexity of nonassociative Lambek calculus with unit*, **Studia Logica**, vol. 93(1) (2009), pp. 1–14, DOI: https://doi.org/

10.1007/s11225-009-9205-2.

[10] B. Day, M. Laplaza, *On embedding closed categories*, **Bulletin of the Australian Mathematical Society**, vol. 18(3) (1978), pp. 357–371, DOI: https://doi.org/10.1017/s0004972700008236.

[11] W. J. de Schipper, **Symmetric closed categories**, vol. 64 of Mathematical Centre Tracts, CWI, Amsterdam (1975).

[12] K. Došen, *A brief survey of frames for the Lambek calculus*, **Mathematical Logic Quarterly**, vol. 38(1) (1992), pp. 179–187, DOI: https://doi.org/10.1002/malq.19920380113.

[13] S. Eilenberg, G. M. Kelly, *Closed categories*, [in:] S. Eilenberg, D. K. Harrison, S. Mac Lane, H. Röhrl (eds.), **Proceedings of Conference on Categorical Algebra (La Jolla, 1965)**, Springer (1966), pp. 421–562, DOI: https://doi.org/10.1007/978-3-642-99902-4_22.

[14] J.-Y. Girard, *Linear logic*, **Theoretical Computer Science**, vol. 50 (1987), pp. 1–102, DOI: https://doi.org/10.1016/0304-3975(87)90045-4.

[15] J.-Y. Girard, *A new constructive logic: classical logic*, **Mathematical Structures in Computer Science**, vol. 1(3) (1991), pp. 255–296, DOI: https://doi.org/10.1017/s0960129500001328.

[16] S. Lack, R. Street, *Skew monoidales, skew warpings and quantum categories*, **Theory and Applications of Categories**, vol. 26(15) (2012), pp. 385–402, URL: http://www.tac.mta.ca/tac/volumes/26/15/26-15abs.html.

[17] S. Lack, R. Street, *Triangulations, orientals, and skew monoidal categories*, **Advances in Mathematics**, vol. 258 (2014), pp. 351–396, DOI: https://doi.org/10.1016/j.aim.2014.03.003.

[18] J. Lambek, *The mathematics of sentence structure*, **American Mathematical Monthly**, vol. 65(3) (1958), pp. 154–170, DOI: https://doi.org/10.2307/2310058.

[19] J. Lambek, *On the Calculus of Syntactic Types*, [in:] R. Jakobson (ed.), **Structure of Language and Its Mathematical Aspects**, AMS, Providence (1961), pp. 166–178, DOI: https://doi.org/10.1090/psapm/012.

[20] S. Mac Lane, *Natural associativity and commutativity*, **Rice University Studies**, vol. 49(4) (1963), pp. 28–46, URL: http://hdl.handle.net/1911/62865.

[21] M. Moortgat, *Multimodal linguistic inference*, **Journal of Logic, Language and Information**, vol. 5(3–4) (1996), pp. 349–385, DOI: https://doi.org/10.1007/bf00159344.

[22] M. Moortgat, *The Tamari order for D^3 and derivability in semi-associative Lambek-Grishin Calculus*, Talk at 16th Workshop on Computational Logic and Applications, CLA 2020 (2020), slides available at: http://cla.tcs.uj.edu.pl/history/2020/pdfs/CLA_slides_Moortgat.pdf.

[23] R. Moot, C. Retoré, **The logic of categorial grammars: A deductive account of natural language syntax and semantics**, vol. 6850 of Lecture Notes in Computer Science, Springer (2012), DOI: https://doi.org/10.1007/978-3-642-31555-8.

[24] K. Rosenthal, *Relational monoids, multirelations, and quantalic recognizers*, **Cahiers de Topologie et Géométrie Différentielle Catégoriques**, vol. 38(2) (1997), pp. 161–171.

[25] R. Street, *Skew-closed categories*, **Journal of Pure and Applied Algebra**, vol. 217(6) (2013), pp. 973–988, DOI: https://doi.org/10.1016/j.jpaa.2012.09.020.

[26] K. Szlachányi, *Skew-monoidal categories and bialgebroids*, **Advances in Mathematics**, vol. 231(3–4) (2012), pp. 1694–1730, DOI: https://doi.org/10.1016/j.aim.2012.06.027.

[27] T. Uustalu, N. Veltri, C.-S. Wan, *Proof theory of skew non-commutative MILL*, [in:] A. Indrzejczak, M. Zawidzki (eds.), **Proceedings of 10th International Conference on Non-classical Logics: Theory and Applications, NCL 2022**, vol. 358 of Electronic Proceedings in Theoretical Computer Science, Open Publishing Association (2022), pp. 118–135, DOI: https://doi.org/10.4204/eptcs.358.9.

[28] T. Uustalu, N. Veltri, N. Zeilberger, *Eilenberg-Kelly reloaded*, **volume 352 of Electronic Notes in Theoretical Computer Science**, vol. 352 (2020), pp. 233–256, DOI: https://doi.org/10.1016/j.entcs.2020.09.012.

[29] T. Uustalu, N. Veltri, N. Zeilberger, *Deductive systems and coherence for skew prounital closed categories*, [in:] C. Sacerdoti Coen, A. Tiu (eds.), **Proceedings of 15th Workshop on Logical Frameworks and Meta-Languages: Theory and Practice, LFMTP 2020**, vol. 332 of Electronic Proceedings in Theoretical Computer Science, Open Publishing Association (2021), pp. 35–53, DOI: https://doi.org/10.4204/eptcs.332.3.

[30] T. Uustalu, N. Veltri, N. Zeilberger, *Proof theory of partially normal skew monoidal categories*, [in:] D. I. Spivak, J. Vicary (eds.), **Proceedings of 3rd Annual International Applied Category Theory Conference 2020, ACT 2020**, vol. 333 of Electronic Proceedings in Theoretical Computer Science, Open Publishing Association (2021), pp. 230–246, DOI: https://doi.org/10.4204/eptcs.333.16.

[31] T. Uustalu, N. Veltri, N. Zeilberger, *The sequent calculus of skew monoidal categories*, [in:] C. Casadio, P. J. Scott (eds.), **Joachim Lambek: The Interplay of Mathematics, Logic, and Linguistics**, vol. 20 of Outstanding Contributions to Logic, Springer (2021), pp. 377–406, DOI: https://doi.org/10.1007/978-3-030-66545-6_11.

[32] N. Veltri, *Coherence via focusing for symmetric skew monoidal categories*, [in:] A. Silva, R. Wassermann, R. de Queiroz (eds.), **Proceedings of 27th International Workshop on Logic, Language, Information, and Computation, WoLLIC 2021**, vol. 13028 of Lecture Notes in Computer Science, Springer (2021), pp. 184–200, DOI: https://doi.org/10.1007/978-3-030-88853-4_12.

[33] N. Veltri, *Maximally multi-focused proofs for skew non-commutative MILL*, [in:] H. H. Hansen, A. Scedrov, R. J. G. B. de Queiroz (eds.), **Proceedings of 29th International Workshop on Logic, Language, Information, and Computation, WoLLIC 2023**, vol. 13923 of Lecture Notes in Computer Science, Springer (2023), pp. 377–393, DOI: https://doi.org/10.1007/978-3-031-39784-4_24.

[34] N. Veltri, *Coherence via focusing for symmetric skew monoidal and symmetric skew closed categories*, **Journal of Logic and Computation**, (to appear), DOI: https://doi.org/10.1093/logcom/exae059.

[35] N. Veltri, C.-S. Wan, *Semi-substructural logics with additives*, [in:] T. Kutsia, D. Ventura, D. Monniaux, J. F. Morales (eds.), **Proceedings of 18th International Workshop on Logical and Semantic Frameworks, with Applications and 10th Workshop on Horn Clauses for Verification and Synthesis, LSFA/HCVS 2023**, vol. 402 of Electronic Proceedings in Theoretical Computer Science, Open Publishing Association (2024), pp. 63–80, DOI: https://doi.org/10.4204/eptcs.402.8.

[36] C.-S. Wan, *Semi-substructural logics à la Lambek*, [in:] A. Indrzejczak, M. Zawidzki (eds.), **Proceedings of 11th International Conference on Non-classical Logics: Theory and Applications, NCL 2024**, vol. 415 of Electronic Proceedings in Theoretical Computer Science, Open Publishing Association (2024), pp. 195–213, DOI: https://doi.org/10.4204/eptcs.415.18.

[37] N. Zeilberger, *A sequent calculus for a semi-associative law*, **Logical Methods in Computer Science**, vol. 15(1) (2019), 9, DOI: https://doi.org/10.23638/lmcs-15(1:9)2019.

Cheng-Syuan Wan
Tallinn University of Technology
Department of Software Science
Akadeemia tee 21b, 12913
Tallinn, Estonia
e-mail: cswan@cs.ioc.ee

Funding information: Estonian Research Council grant PSG749.

Conflict of interests: None.

Ethical considerations: The Author assures of no violations of publication ethics and takes full responsibility for the content of the publication.

Declaration regarding the use of GAI tools: Not used.

Bulletin of the **S**ection of **L**ogic
Volume 54/4 (2025), pp. 577–605
https://doi.org/10.18778/0138-0680.2025.18

Paweł Płaczek

COMPLEXITY OF NONASSOCIATIVE LAMBEK CALCULUS WITH CLASSICAL AND INTUITIONISTIC LOGIC

Abstract

The Nonassociative Lambek Calculus (NL) represents a logic devoid of the structural rules of exchange, weakening, and contraction, and it does not presume the associativity of its connectives. Its finitary consequence relation is decidable in polynomial time. However, the addition of classical connectives conjunction and disjunction (FNL) makes the consequence relation undecidable. Interestingly, if these connectives are distributive, the consequence relation is decidable in exponential time. This paper provides the proof, that we can merge classical logic with NL (i.e. BFNL) and intuitionistic logic with NL (i.e. HFNL), and still consequence relations are decidable in exponential time.

Keywords: Lambek calculus, nonassociative logics, non-commutative logics, substructural logics, consequence relation, nonlogical axioms.

Presented by: Michał Zawidzki
Received: January 1, 2025, **Received in revised form**: August 21, 2025,
Accepted: September 17, 2025, **Published online**: March 13, 2026

1. Introduction and preliminaries

Lambek Calculus L was introduced by Lambek [5] under the name *Syntactic Calculus*. L is a propositional logic with three connectives $\otimes$ (product), $\backslash$ and $/$ (residuations of product). Lambek [6] introduced the nonassociative version of this logic, nowadays called Nonassociative Lambek Calculus (NL). From a logical perspective, NL can be seen as the pure logic of residuation, and L as its stronger version for associative product. For both L and NL, Lambek provided a sequent system and proved cut elimination [5, 6].

The product for both L and NL derives from conjunction after dropping the structural rules of exchange, weakening, and contraction in terms of sequent systems. NL additionally does not require being an associative operator in terms of algebra. In effect, we obtain a pure operation joining two formulas. This operation may be seen as a binary modality.

Definition 1.1. Let $\mathbf{G} = (G, \otimes, \backslash, /, \leq)$ be a structure such that $(G, \otimes)$ is a groupoid, $(G, \leq)$ is a poset, and the following holds:

(RES) $a \otimes b \leq c$ iff $b \leq a \backslash c$ iff $a \leq c/b$

for all $a, b, c \in G$. Then $\mathbf{G}$ is called a *residuated groupoid*.

By *groupoid* we mean a set closed under a binary operation without any specific properties required. The residuated groupoids are models of NL. The residuated groupoids where the product is associative are called *residuated semigroups* and are models of L.

The most popular extensions of L and NL are: adding the (multiplicative) constant 1 or adding conjunction and disjunction. The constant 1 in algebras is a unit for the product. The conjunction and disjunction replace the partial order with the lattice structure and lattice order. We can also add the boundaries, i.e., $\top$ and $\bot$, as respectively, the greatest and lowest elements. In this paper we use the same symbol for both syntactic and semantic purposes and the exact meaning is clear from the context.

Definition 1.2. Let $(G, \otimes, \backslash, /, \leq)$ be a residuated groupoid and let $1 \in G$ be an element such that:

$$1 \otimes a = a = a \otimes 1$$

for all $a \in G$. Then $(G, \otimes, \backslash, /, 1, \leq)$ is a unital residuated groupoid. 1 is said to be a unit.

The unital residuated groupoids are models for NL with constant 1 and unital residuated semigroups are models for L with constant 1.

Lambek Calculus with additive connectives (conjunction and disjunction) is called Full Lambek Calculus and denoted FL. Some authors also require the presence of 1 (multiplicative constant) and $\top, \bot$ (additive constants). In this paper, we follow this convention, so FL admits all these constants. Analogously, FNL is an extension of NL with additive connectives and all constants.

Definition 1.3. Let $(G, \otimes, \backslash, /, 1, \leq)$ be a unital residuated groupoid and $(G, \vee, \wedge, \top, \bot, \leq)$ be a bounded lattice. Then, $(G, \otimes, \backslash, /, \vee, \wedge, 1, \top, \bot, \leq)$ is a *residuated lattice.*

The residuated lattices are models for FNL. Residuated lattices where $\otimes$ is associative are models for FL.

Pentus [7] proves that pure[1] L is NP-complete and Buszkowski [1] proves that its finitary consequence relation is undecidable. A similar situation applies if we add the constant 1. FL is a strongly conservative extension[2] of L, so its finitary consequence relation is also undecidable. The same applies to all strongly conservative extensions of L. In this paper, we focus on extensions of NL because of that.

Buszkowski [1] proves that the finitary consequence relation for NL is in *PTIME*. The same applies if we admit the constant 1. Unfortunately, FNL has an undecidable consequence relation [3].

The lattices in the algebras of FNL are not necessarily distributive. If we consider logic with such an axiom for additive connectives, we talk about Distributive Full Nonassociative Lambek Calculus and denote it DFNL. The models for this logic are residuated distributive lattices.

---

[1] By *pure* we mean the logic without nonlogical axioms (assumptions).

[2] A logic $\mathcal{L}_2$, extending $\mathcal{L}_1$, is a (resp. strongly) conservative extension of $\mathcal{L}_1$, if both logics have the same theorems (resp. the same consequence relation) in language of $\mathcal{L}_1$.

The finitary consequence relation of DFNL is *EXPTIME*-complete if we do not admit the constant 1 and is in *EXPTIME* if we admit the constant, which was proved in [9].[3] The lower bound of complexity of the consequence relation for DFNL with constant 1 remains an open problem.

The other interesting extensions of FNL are BFNL and HFNL, i.e., Boolean FNL and Heyting FNL. These logics may be seen as extensions of NL with Boolean and Heyting algebras or as extensions of classical logic and intuitionistic logic with NL. Such logics have been studied by Galatos and Jipsen [4], Buszkowski [2], and others.

DEFINITION 1.4. Let $(G, \otimes, \backslash, /, 1, \leq)$ be a unital residuated groupoid and $(G, \vee, \wedge, \neg, \bot, \top, \leq)$ be a Boolean algebra. Then, $(G, \otimes, \backslash, /, \vee, \wedge, \neg, 1, \top, \bot, \leq)$ is a *residuated Boolean algebra*.

DEFINITION 1.5. Let $(G, \otimes, \backslash, /, 1, \leq)$ be a unital residuated groupoid and $(G, \vee, \wedge, \rightarrow, \bot, \top, \leq)$ be a Heyting algebra. Then, $(G, \otimes, \backslash, /, \vee, \wedge, \rightarrow, 1, \top, \bot, \leq)$ is a *residuated Heyting algebra*.

In this paper, we provide the proof of the upper bound of the complexity of the consequence relations for BFNL and HFNL, extending the results of [9], using the same methods. We also use the results from [10], where distributive lattices, Heyting algebras, and Boolean algebras are considered. The differences between [9, 10] and this paper lay in the details. An experienced reader can easily deduce the results of this paper by reading cited papers, but some changes are subtle, e.g. in some places we do not use families of upsets, but the whole powerset, because we have negation here. Moreover, the results in [9, 10] are described in only algebraic terms and use first-order formulas. Here, we use syntactic notion more directly, still using algebraic methods in proofs.

We show the full proof only for the version with the constant 1 because the proofs for logics without that constant can be easily obtained by omitting some parts.

Since HFNL and BFNL without 1 are strongly conservative extensions

---

[3] Shkatov and Van Alten [9] show that the satisfiability problem of quantifier-free first-order formulas in the language of bounded distributive residuated lattices is *EXPTIME*-complete.

of DFNL,[4] we know their finitary consequence relations are *EXPTIME*-hard and, in effect, are *EXPTIME*-complete. The lower bound for HFNL and BFNL with 1 is still an open problem.

In the second section, we provide the sequent systems for BFNL and HFNL. These systems come from [4], where the authors prove the cut-elimination theorem. In the third section, we study partial structures connected with models of BFNL. We prove important theorems that allow us to check whether a given partial structure is a partial residuated algebra. In the fourth section, we use these theorems to prove *EXPTIME* complexity of the consequence relation for BFNL.

This paper is an extension of the conference paper [8]. The novelty is the last section (fifth), where we add the detailed instructions how to modify definitions, theorems and proofs to obtain the result for HFNL, since it is analogous.

2. Sequent systems

The language of BFNL is defined as follows. We admit a countable set of variables, which we denote by small Latin letters. The formulas are constructed from this set of variables by five binary connectives $(\otimes, \backslash, /, \vee, \wedge)$, one unary connective $(\neg)$ and three constants $(1, \top, \bot)$.

Usual notion of sequents using sequeces of formulas is not applicable in nonassociative framework. The comma in sequences is a concatenation operation which is associative. We need to change the structure to something more flexible. Moreover, we need to have two types of commas: one for $\otimes$ and one for $\wedge$ with different properites.

We define bunches. The bunches are elements of free biunital bi-groupoid, i.e. the algebra with two binary operations with a unit for both of them, generated from the set of all formulas. We denote first operator by comma and the second one by semicolon. The unit for comma is denoted ϵ and unit for semicolon is δ.

---

[4] See Remark 5 in [2].

One may think of bunches as of binary trees in which leaves are formulas or ϵ or δ and every node besides leaves is labeled by comma or semicolon.

The bunch ϵ is called an *empty bunch*. All the other bunches are nonempty. We reserve Latin capital letters for formulas and Greek capital letters for bunches. A *context* is a bunch with an anonymous variable. Contexts are denoted by $\Gamma[\_]$, and when we perform the substitution of Δ in place of $\_$, we represent it as $\Gamma[\Delta]$.

A *sequent* is a pair Γ, A, where Γ is a bunch and A is a formula. We write $\Gamma \Rightarrow A$ for the sequent.

The axioms and the rules for BFNL are as follows:

$$\text{(id)} \quad A \Rightarrow A \quad \text{(cut)} \quad \frac{\Gamma \Rightarrow A \quad \Delta[A] \Rightarrow C}{\Delta[\Gamma] \Rightarrow C}$$

$$(\otimes \Rightarrow) \quad \frac{\Gamma[(A, B)] \Rightarrow C}{\Gamma[A \otimes B] \Rightarrow C} \quad (\Rightarrow \otimes) \quad \frac{\Gamma \Rightarrow A \quad \Delta \Rightarrow B}{\Gamma, \Delta \Rightarrow A \otimes B}$$

$$(\backslash \Rightarrow) \quad \frac{\Gamma[B] \Rightarrow C \quad \Theta \Rightarrow A}{\Gamma[(\Theta, A\backslash B)] \Rightarrow C} \quad (\Rightarrow \backslash) \quad \frac{A, \Gamma \Rightarrow B}{\Gamma \Rightarrow A\backslash B}$$

$$(/ \Rightarrow) \quad \frac{\Gamma[A] \Rightarrow C \quad \Theta \Rightarrow B}{\Gamma[(A/B, \Theta)] \Rightarrow C} \quad (\Rightarrow /) \quad \frac{\Gamma, B \Rightarrow A}{\Gamma \Rightarrow A/B}$$

$$(\wedge \Rightarrow) \quad \frac{\Gamma[(A; B)] \Rightarrow C}{\Gamma[A \wedge B] \Rightarrow C} \quad (\Rightarrow \wedge) \quad \frac{\Gamma \Rightarrow A \quad \Gamma \Rightarrow B}{\Gamma \Rightarrow A \wedge B}$$

$$(\vee \Rightarrow) \quad \frac{\Gamma[A] \Rightarrow C \quad \Gamma[B] \Rightarrow C}{\Gamma[A \vee B] \Rightarrow C} \quad (\Rightarrow \vee) \quad \frac{\Gamma \Rightarrow A}{\Gamma \Rightarrow A \vee B} \quad \frac{\Gamma \Rightarrow B}{\Gamma \Rightarrow A \vee B}$$

$$(\top \Rightarrow) \quad \frac{\Gamma[\Delta] \Rightarrow C}{\Gamma[(\top; \Delta)] \Rightarrow C} \; \frac{\Gamma[\Delta] \Rightarrow C}{\Gamma[(\Delta; \top)] \Rightarrow C} \qquad (\Rightarrow \top) \quad \Gamma \Rightarrow \top$$

$$(\bot \Rightarrow) \quad \Gamma[\bot] \Rightarrow C$$

$$(\wedge\text{-ass}) \quad \begin{array}{c} \Gamma[\Delta_1; (\Delta_2; \Delta_3)] \Rightarrow C \\ \hline\hline \Gamma[(\Delta_1; \Delta_2); \Delta_3] \Rightarrow C \end{array} \quad (\wedge\text{-ex}) \quad \frac{\Gamma[\Delta; \Theta] \Rightarrow C}{\Gamma[\Theta; \Delta] \Rightarrow C}$$

$$(\wedge\text{-weak}) \quad \frac{\Gamma[\Delta] \Rightarrow C}{\Gamma[\Delta; \Theta] \Rightarrow C} \quad (\wedge\text{-cont}) \quad \frac{\Gamma[\Delta; \Delta] \Rightarrow C}{\Gamma[\Delta] \Rightarrow C}$$

$$(1 \Rightarrow)\quad \frac{\Gamma[\Delta] \Rightarrow C}{\Gamma[(1, \Delta)] \Rightarrow C} \quad \frac{\Gamma[\Delta] \Rightarrow C}{\Gamma[(\Delta, 1)] \Rightarrow C} \qquad (\Rightarrow 1)\quad \epsilon \Rightarrow 1$$

$$(\neg \Rightarrow)\quad A \wedge \neg A \Rightarrow \bot \quad (\Rightarrow \neg)\quad \top \Rightarrow A \vee \neg A$$

We shortly describe the semantics of BFNL. The models for BNFL are residuated Boolean algebras. The valuation is a homomorphism μ from the free algebra of formulas to a residuated Boolean algebra $\mathbf{B}$ extended to bunches inductively as follows:

$$\mu(\epsilon) = 1$$
$$\mu(\delta) = \top$$
$$\mu((\Gamma, \Delta)) = \mu(\Gamma) \otimes \mu(\Delta)$$
$$\mu((\Gamma; \Delta)) = \mu(\Gamma) \wedge \mu(\Delta)$$

The sequent $\Gamma \Rightarrow A$ is said to be true in $\mathbf{B}$ under the valuation μ if $\mu(\Gamma) \leq \mu(A)$.

The language of HFNL is defined as follows. We admit a countable set of variables, which we denote by small Latin letters. The formulas are constructed from this set of variables by six binary connectives $(\otimes, \backslash, /, \vee, \wedge, \rightarrow)$ and three constants $(1, \top, \bot)$.

We define bunches and sequents analogously. The axioms and the rules of the sequent system for HFNL are similar. We replace the negation axioms $(\neg \Rightarrow)$ and $(\Rightarrow \neg)$ with the following rules:

$$(\rightarrow\Rightarrow)\quad \frac{\Gamma[B] \Rightarrow C \quad \Theta \Rightarrow A}{\Gamma[(\Theta; A \rightarrow B)] \Rightarrow C} \quad (\Rightarrow\rightarrow)\quad \frac{\Gamma; A \Rightarrow B}{\Gamma \Rightarrow A \rightarrow B}$$

The models for HFNL are residuated Hayting algebras. We define valuation analogously.

3. Partial residuated Boolean algebras

In this section we provide the notion of partial structures and we prove some properties. The most important result here is Theorem 3.19 which helps in identifying partial residuated Boolean algebras in exponential time in the next section.

3.1. Partial structures

Definition 3.1. A function $f : U \mapsto Y$, where $U \subseteq X$, is called a *partial function* from X to Y (we write $f : X \to Y$). If $U = X$, then the function is said to be *total*.

We write $f(x) = \infty$, if the function f on the argument x is undefined.

Definition 3.2. Let I, J, K be finite indexing sets. We say

$$(U, \{f_i^{n_i}\}_{i\in I}, \{a_j\}_{j\in J}, \{R_k^{m_k}\}_{k\in K})$$

is a *partial structure*, if $\{a_j\}_{j\in J} \subseteq U$ and $f_i^{n_i} : U^{n_i} \to U$ is a partial function for all $i \in I$ and $R_k^{m_k} \subseteq U^{m_k}$ for all $k \in K$. If all operations are total, then we say the structure is *total*.

Definition 3.3. Let I, J, K be finite indexing sets. Let

$$(U, \{f_i^{n_i}\}_{i\in I}, \{a_j\}_{j\in J}, \{R_k^{m_k}\}_{k\in K})$$

be a partial structure and

$$(U', \{f_i'^{n_i}\}_{i\in I}, \{a_j'\}_{j\in J}, \{R_k'^{m_k}\}_{k\in K})$$

be a total structure. Let $\iota : U \to U'$ be an injection. We say ι is an *embedding*, if:

(i) for all $j \in J$ we have $\iota(a_j) = a_j'$,

(ii) for all $i \in I$ and all $x_1, x_2, \ldots, x_{n_i} \in U$, if $f_i^{n_i}(x_1, x_2, \ldots, x_{n_i}) \neq \infty$, then $\iota(f_i^{n_i}(x_1, x_2, \ldots, x_{n_i}) = f_i'^{n_i}(\iota(x_1), \iota(x_2), \ldots, \iota(x_{n_i}))$,

(iii) for all $k \in K$ we have $(\iota(x_1), \iota(x_2), \ldots, \iota(x_{m_k})) \in (R_k'^{m_k}) \iff (x_1, x_2, \ldots, x_{m_k}) \in R_k^{m_k}$ for all $x_1, x_2, \ldots, x_{m_k} \in U$.

If $\mathbf{A}$ is a partial structure, $\mathbf{B}$ is a total structure and there exists an embedding from $\mathbf{A}$ to $\mathbf{B}$, then we say $\mathbf{A}$ is *embeddable* into $\mathbf{B}$. If $\mathbf{A}$ is embeddable into $\mathbf{B}$ and $A \subseteq B$, then we say $\mathbf{A}$ is a *partial substructure* of $\mathbf{B}$. Let $\mathcal{K}$ be a class of structures. By $\mathcal{K}^P$ we denote the class of all partial substructures of structures of $\mathcal{K}$.

Definition 3.4. Let $\mathbf{L} = (L, \vee, \wedge, \top, \bot, \leq)$ be a partial structure. We say $\mathbf{L}$ is a *partial lattice*, if there exists a total lattice $\mathbf{L}'$ such that $\mathbf{L}$ is embeddable into it. If $\mathbf{L}'$ is distributive, then $\mathbf{L}$ is a *partial distributive lattice*.

One shows that a partial structure $(L, \vee, \wedge, \top, \bot, \leq)$ is a partial bounded lattice, if $(L, \leq)$ is a poset, $\top$ and $\bot$ are bounds of $\leq$ and $\vee, \wedge$ are compatible with $\leq$, i.e. if $a \vee b \neq \infty$, then $a \vee b$ is the supremum of $\{a, b\}$ with respect to $\leq$ and if $a \wedge b \neq \infty$, then $a \wedge b$ is the infimum of $\{a, b\}$ with respect to $\leq$. See [9].

Definition 3.5. Let $\mathbf{B} = (B, \otimes, \backslash, /, \vee, \wedge, \neg, 1, \top, \bot, \leq)$ be a partial structure. We say $\mathbf{B}$ is a *partial residuated Boolean algebra*, if there exists a total residuated Boolean algebra such that $\mathbf{B}$ is embeddable into it and for all $a \in B$ we have $\neg a \neq \infty$, $\neg a \in B$, $a \vee \neg a = \top$ and $a \wedge \neg a = \bot$. One notices that $(B, \otimes, \backslash, /, \vee, \wedge, \top, \bot, \leq)$ is a partial bounded distributive residuated lattice.

3.2. Filters

Let $(P, \leq)$ be a poset and let $A \subseteq P$. We say A is an *upset*, if for all $a \in A$ and all $b \in P$ such that $a \leq b$ we have $b \in A$. Analogously, A is a *downset*, if for all $a \in A$ and $b \in P$ such that $b \leq a$ we have $b \in A$.

For every poset $(P, \leq)$ and every element $a \in P$ we define:

$$[a) = \{b \in P : a \leq b\} \qquad (a] = \{b \in P : b \leq a\}$$

One notices $[a)$ is an upset and $(a]$ is a downset.

Definition 3.6. Let $(L, \vee, \wedge)$ be a lattice and let $F \subseteq L$. We say F is a *filter*, if the following conditions hold:

(F1) if $a \leq b$ and $a \in F$, then $b \in F$

(F2) if $a \in F$ and $b \in F$, then $a \wedge b \in F$

We say F is *proper*, if $F \neq L$. The filter F is *prime*, if it is proper and:

(F3) if $a \vee b \in F$, then $a \in F$ or $b \in F$

Let $(L, \vee, \wedge)$ be a lattice and F be a filter. We use the following notion:

$$F_a = \left\{ y \in L : \underset{x \in F}{\exists}\, x \wedge a \leq y \right\}$$

One proves F_a is a filter.

If we consider filters on residuated Boolean algebras, then (F3) is replaced with the following condition:

(FB) $\neg a \in F$ iff $a \notin F$

Considering filters on partial residuated Boolean algebras, we must change definition. We replace (F2) with the following condition:

(F2') if $a \in F$ and $b \in F$, then $a \wedge b \in F$ or $a \wedge b = \infty$

for all $a, b \in B$.

The following properties of filters are useful and may be easily proved.

Lemma 3.7. *Let $(B, \vee, \wedge, \neg, \top, \bot)$ be a Boolean algebra and let $F \subseteq B$ be a proper filter. The filter F is prime if, and only if, $a \in F$ or $\neg a \in F$ for all $a \in B$.*

This lemma remains true for residuated Boolean algebras.

Proof: Let F be a prime filter. Then $a \vee \neg a = \top \in F$ for all $a \in B$, so the condition of lemma holds. Now let $a \in F$ or $\neg a \in F$ for all $a \in B$. Let $a \vee b \in F$ and suppose $a \notin F$ and $b \notin F$. Then $\neg a \in F$ and $\neg b \in F$, by assumption. By (F2), $\neg a \wedge \neg b \in F$. So, $\neg(a \vee b) \in F$. Hence, $(a \vee b) \wedge \neg(a \vee b) = \bot \in F$, by (F2). This is impossible. □

Lemma 3.8. *Let $(L, \vee, \wedge)$ be a distributive lattice and let $F \subseteq L$ be a filter and $b \in L$ be such that $b \notin F$. There exists a prime filter $P \subseteq L$ such that $F \subseteq P$ and $b \notin P$.*

Proof: Let F be a filter, $b \in L$ and $b \notin F$. We construct a prime filter as an extension of F, but we need to avoid adding b.

Let $\mathcal{E}$ be a family of filters of L containing F and not containing b. The family is nonempty, since $F \in \mathcal{E}$. Let $C \subseteq \mathcal{E}$ be any nonempty chain in $\mathcal{E}$. Then $F \subseteq \bigcup C$ and $b \notin \bigcup C$. We show $\bigcup C$ is a filter. Let $c, d \in \bigcup C$, then $c \in G$ and $d \in G'$ for some $G, G' \in C$. Since C is a chain, then $G \subseteq G'$ or $G' \subseteq G$, so both c and d are elements of G or G'. Then, by (F2), $c \wedge d \in G$ or $c \wedge d \in G'$, so $c \wedge d \in \bigcup C$. So $\bigcup C$ satisfies (F2). (F1) is obvious. Hence, $\bigcup C$ is a filter.

By Kuratowski–Zorn's lemma, there exists $P \in \mathcal{E}$, which is a maximal element of $\mathcal{E}$. We need to show P is prime. Let $c, d \notin P$ and $c \vee d \in P$. Since $c \notin P$, then $P \subseteq P_c$, and, since P is a maximal element of $\mathcal{E}$, $P_c \notin \mathcal{E}$. Clearly, $F \subseteq P_c$, so $b \in P_c$. Analogously, since $d \notin P$, then $b \in P_d$.

By definition of P_c, P_d, for some $x, y \in P$ we have $x \wedge c \leq b$ and $y \wedge d \leq b$. Hence, $x \wedge y \wedge c \leq b$ and $x \wedge y \wedge d \leq b$ and so $(x \wedge y \wedge c) \vee (x \wedge y \wedge d) \leq b$. By distributivity, $x \wedge y \wedge (c \vee d) \leq b$. Since $x, y, c \vee d \in P$, then $b \in P$. Thus, if $c, d \notin P$, when $c \vee d \in P$, then $b \in P$, which is impossible by definition of P. □

Corollary 3.9. Let $(L, \vee, \wedge)$ be a distributive lattice and let $a, b \in L$ be such that $a \not\leq b$. There exists a prime filter $F \subseteq L$ such that $a \in F$ and $b \notin F$.

Proof: The set $[a)$ is a filter such that $b \notin [a)$. Then, by Theorem 3.8, there exists a prime filter P such that $a \in P$ and $b \notin P$. □

Lemma 3.10. *Let* $\mathbf{B}$ *be a total residuated Boolean algebra and let* F, G *be proper filters of* $\mathbf{B}$ *and* H *be a prime filter of* $\mathbf{B}$ *such that* $\{x \otimes y : x \in F \text{ and } y \in G\} \subseteq H$. *Then, there exist prime filters* F' *and* G' *such that* $F \subseteq F'$ *and* $G \subseteq G'$ *and* $\{x \otimes y : x \in F' \text{ and } y \in G\} \subseteq H$ *and* $\{x \otimes y : x \in F \text{ and } y \in G'\} \subseteq H$.

Proof: Let F, G be proper filters and H be a prime filter such that $\{x \otimes y : x \in F \text{ and } y \in G\} \subseteq H$. We show there exists a prime filter F' such that $F \subseteq F'$ and $\{x \otimes y : x \in F' \text{ and } y \in G\} \subseteq H$.

Let $\mathcal{E}$ be the family of filters Q of $\mathbf{B}$ such that $\{x \otimes y : x \in Q \text{ and } y \in G\} \subseteq H$. This family is nonempty, since $F \in \mathcal{E}$. Clearly, all filters in $\mathcal{E}$ are proper; otherwise $\perp = \perp \otimes 1 \in H$, which is impossible. We show that $\bigcup C \in \mathcal{E}$ for every nonempty chain $C \subseteq \mathcal{E}$. Now, let $a \in \bigcup C$. Then, for some $Q \in C$ we have $a \in Q$ and $\{x \otimes y : x \in Q \text{ and } y \in G\} \subseteq H$. Hence, for some $y \in G$, we have $a \otimes y \in H$. So, $\bigcup C \in \mathcal{E}$.

By Kuratowski–Zorn's lemma, there exists $P \in \mathcal{E}$, which is a maximal element of $\mathcal{E}$. We show P is a prime filter. Let $a \vee b \in P$ and suppose $a, b \notin P$. We consider P_a, P_b. Clearly, $P \subset P_a$ and $P \subset P_b$. So, since P is a maximal element, $P_a, P_b \notin \mathcal{E}$. So $\{x \otimes y : x \in P_a \text{ and } y \in G\} \nsubseteq H$ and $\{x \otimes y : x \in P_b \text{ and } y \in G\} \nsubseteq H$.

So, for some $x, y \in P$ and some $z_1, z_2 \in G$ we have $(x \wedge a) \otimes z_1 \notin H$ and $(y \wedge b) \otimes z_2 \notin H$. Since $x, y, a \vee b \in P$, then $x \wedge y \wedge (a \vee b) \in P$. So we have $(x \wedge y \wedge (a \vee b)) \otimes (z_1 \wedge z_2) \in H$. But:

$$\begin{aligned}(x \wedge y \wedge (a \vee b)) \otimes (z_1 \wedge z_2) &= ((x \wedge y \wedge a) \vee (x \wedge y \wedge b)) \otimes (z_1 \wedge z_2) = \\ &= (x \wedge y \wedge a) \otimes (z_1 \wedge z_2) \vee (x \wedge y \wedge b) \otimes (z_1 \wedge z_2)\end{aligned}$$

So, since H is a prime filter, $(x \wedge y \wedge a) \otimes (z_1 \wedge z_2) \in H$ or $(x \wedge y \wedge b) \otimes (z_1 \wedge z_2) \in H$. Because H is a filter, then $(x \wedge a) \otimes z_1 \in H$ or $(y \wedge b) \otimes z_2 \in H$. This contradicts the assumptions. Hence, $a \in P$ or $b \in P$.

We put $F' = P$. We show that there exists G' such that $G \subseteq G'$ and $\{x \otimes y : x \in F \text{ and } y \in G'\} \subseteq H$ analogously. ☐

3.3. Residuated frames

Definition 3.11. Let $\mathfrak{F} = (P, I, R)$. We say $\mathfrak{F}$ is a *residuated frame*, when $I \subset P$ and R is a ternary relation on P and the following conditions hold:

(U1) $\underset{x,x',y,z \in P}{\forall} \big(\text{if } R(x,y,z) \text{ and } x' = x, \text{ then } R(x',y,z)\big)$

(U2) $\underset{x,y,y',z \in P}{\forall} \big(\text{if } R(x,y,z) \text{ and } y' = y, \text{ then } R(x,y',z)\big)$

(U3) $\underset{x,y,z,z' \in P}{\forall} \big(\text{if } R(x,y,z) \text{ and } z = z', \text{ then } R(x,y,z')\big)$

(U4) $\underset{x \in P}{\forall}\ \underset{y,z \in I}{\exists} \big(R(x,y,x) \text{ and } R(z,x,x)\big)$

(U5) $\underset{x,z \in P}{\forall}\ \underset{y \in I}{\forall} \big(\text{if } R(x,y,z) \text{ or } R(y,x,z), \text{ then } x = z\big)$

Residuated frames are the relational structures similar to groupoids. Instead of a binary operation we use a ternary relation.

Definition 3.12. Let $\mathbf{B} = (B, \otimes, \backslash, /, \vee, \wedge, \neg, 1, \top, \bot, \leq)$ be a partial residuated Boolean algebra. We define the *associated residuated frame* $\mathfrak{F}_{\mathbf{B}} = (\mathscr{F}(B), \mathscr{I}_{\mathbf{B}}, \mathscr{R}_{\mathbf{B}})$, where $\mathscr{F}(B)$ is the set of prime filters of $\mathbf{B}$, $\mathscr{I}_{\mathbf{B}}$ is the set of all prime filters containing 1 and:

$$\begin{aligned}\mathscr{R}_{\mathbf{B}}(F, G, H) \iff & \left(\underset{a,b\in B}{\forall} \text{ if } a \in F \text{ and } b \in G, \text{ then } a \otimes b \in H \vee a \otimes b = \infty\right)\\ \text{and } & \left(\underset{a,b\in B}{\forall} \text{ if } a \in F \text{ and } a\backslash b \in G \text{ and } a\backslash b \neq \infty, \text{ then } b \in H\right)\\ \text{and } & \left(\underset{a,b\in B}{\forall} \text{ if } b/a \in F \text{ and } a \in G \text{ and } a/b \neq \infty, \text{ then } b \in H\right).\end{aligned}$$

Proposition 3.13. Let $\mathbf{B}$ be a residuated Boolean algebra and let $F \in \mathscr{F}(B)$. Then, there exist prime filters $P, Q \in \mathscr{F}(B)$ such that $\mathscr{R}_{\mathbf{B}}(F, P, F)$ and $\mathscr{R}_{\mathbf{B}}(Q, F, F)$ and $1 \in P, 1 \in Q$.

Proof: Let $F \in \mathscr{F}(L)$, we show there exists a prime filter P such that $1 \in P$ and $\mathscr{R}_{\mathbf{L}}(F, P, F)$. The proof for $\mathscr{R}_{\mathbf{L}}(Q, F, F)$ is similar.

Let $\mathcal{E}$ be the family of filters of $\mathbf{L}$ such that for every filter $G \in \mathcal{E}$ we have $1 \in G$ and $f \otimes g \in F$ for all $f \in F$ and $g \in G$. Clearly, all filters in $\mathcal{E}$ are proper. This family is nonempty, since $[1) \in \mathcal{E}$. One shows that $\bigcup C$ is a filter for every nonempty chain $C \subseteq \mathcal{E}$ analogously like in the proof of Theorem 3.8. We show $\bigcup C \in \mathcal{E}$. Clearly, $1 \in \bigcup C$. Let $f \in F$ and $g \in \bigcup C$. Then, $g \in G$ for some $G \in C$. So, $f \otimes g \in F$.

By Kuratowski–Zorn's lemma, there exists $P \in \mathcal{E}$, which is a maximal element of $\mathcal{E}$. We show that P is a prime filter. Assume $a \vee b \in P$. Suppose $a, b \notin P$.

We consider P_a and P_b. Clearly, $P \subset P_a$ and $P \subset P_b$. Since P is a maximal element of $\mathcal{E}$, then $P_a, P_b \notin \mathcal{E}$.

We have $1 \in P_a, P_b$. Then, for some $f_a \in F$ and some $x \in P$, we have $f_a \otimes (x \wedge a) \notin F$ and for some $f_b \in F$ and some $y \in P$ we have $f_b \otimes (y \wedge b) \notin F$. Since $f_a, f_b \in F$, then $f_a \wedge f_b \in F$, by (F2). Since

$a \vee b \in P$, then $(x \wedge y) \wedge (a \vee b) = (x \wedge y \wedge a) \vee (x \wedge y \wedge b) \in P$.

So, $(f_a \wedge f_b) \otimes [(x \wedge a) \vee (y \wedge b)] \in F$. As a consequence:

$$(f_a \wedge f_b) \otimes [(x \wedge a) \vee (y \wedge b)] = ((f_a \wedge f_b) \otimes (x \wedge a)) \vee ((f_a \wedge f_b) \otimes (y \wedge b))$$

Because F is a prime filter, then $(f_a \wedge f_b) \otimes (x \wedge a) \in F$ or $(f_a \wedge f_b) \otimes (y \wedge b) \in F$. Assume $(f_a \wedge f_b) \otimes (x \wedge a) \in F$. Then $f_a \otimes (x \wedge a) \in F$, by (F1) and monotonicity of $\otimes$. Assume $(f_a \wedge f_b) \otimes (y \wedge b) \in F$. Then $f_b \otimes (y \wedge b) \in F$. Both possibilites lead to the contradiction with assumptions. Hence, $a \in P$ or $b \in P$.

Therefore, $\mathscr{R}_{\mathbf{L}}(F, P, F)$. □

Lemma 3.14. *Let* $\mathbf{B}$ *be a total residuated Boolean algebra and* $\mathfrak{F}_{\mathbf{B}} = (\mathscr{F}(B), \subseteq, \mathscr{R}_{\mathbf{B}})$ *its associated residuated frame. Then, for* $F, G, H \in \mathscr{F}(B)$, *the following are equivalent:*

(i) if $a \in F$ *and* $b \in G$, *then* $a \otimes b \in H$ *for all* $a, b \in B$

(ii) if $a \in F$ *and* $a \backslash b \in G$, *then* $b \in H$ *for all* $a, b \in B$

(iii) if $b / a \in F$ *and* $a \in G$, *then* $b \in H$ *for all* $a, b \in B$

Proof: We assume (i). Let $a \in F$ and $a \backslash b \in G$. Since $\mathscr{R}_{\mathbf{B}}(F, G, H)$, $a \otimes (a \backslash b) \in H$ and then $b \in H$, because $a \otimes (a \backslash b) \leq b$. Hence (ii) holds. Now we assume (ii). Let $a \in F$ and $b \in G$. Since $b \leq a \backslash (a \otimes b)$, then $a \backslash (a \otimes b) \in G$, so, by (ii), $a \otimes b \in H$ and (i) holds. The proof of equivalence of (i) and (iii) is similar. □

Corollary 3.15. Let $\mathbf{B}$ be a total residuated Boolean algebra and let F, G be proper filters of $\mathbf{L}$ and $\mathbf{H}$ be a prime filter of $\mathbf{H}$ such that $\{x \otimes y : x \in F \text{ and } y \in G\} \subseteq H$. Then, there exist prime filters F' and G' such that $F \subseteq F'$ and $G \subseteq G'$ and $\mathscr{R}_{\mathbf{L}}(F', G', H)$.

Proof: First, we construct F' such that $\{x \otimes y : x \in F' \text{ and } y \in G\} \subseteq H$, by Theorem 3.10. Then, we construct G' such that $\{x \otimes y : x \in F' \text{ and } y \in G'\} \subseteq H$, by Theorem 3.10. Then, by Theorem 3.14, $\mathscr{R}_{\mathbf{L}}(F', G', H)$. □

We construct a residuated Boolean algebras from the arbitrary residuated frame $\mathfrak{F} = (P, I, R)$. Let $X, Y \subseteq P$, we define:

$$X \otimes' Y = \left\{ z \in P : \underset{x,y \in P}{\exists} x \in X \text{ and } y \in Y \text{ and } R(x,y,z) \right\}$$
$$X \backslash' Y = \left\{ y \in P : \underset{x,z \in P}{\forall} \text{if } R(x,y,z) \text{ and } x \in X, \text{ then } z \in Y \right\}$$
$$Y /' X = \left\{ x \in P : \underset{y,z \in P}{\forall} \text{if } R(x,y,z) \text{ and } y \in X, \text{ then } z \in Y \right\}$$

Then, $\mathbf{B}_{\mathfrak{F}} = (\mathscr{P}(P), \otimes', \backslash', /', \cup, \cap, {}^c, I, P, \emptyset, \subseteq)$ is a residuated Boolean algebra, where $X^c = \mathscr{P}(P) \setminus X$ for all $X \in \mathscr{P}(P)$. We call it the *complex Boolean algebra of the residuated frame* $\mathfrak{F}$.

Lemma 3.16. *Let* $\mathbf{B}$ *be a total residuated Boolean algebra and* $\mathfrak{F}_{\mathbf{B}} = (\mathscr{F}(B), \subseteq, \mathscr{R}_{\mathbf{B}})$ *its associated residuated frame. Let* $a, b \in B$.

(i) *If* $H \in \mathscr{F}(B)$ *and* $a \otimes b \in H$, *then there exist* $F, G \in \mathscr{F}(B)$ *such that* $a \in F$, $b \in G$ *and* $\mathscr{R}_{\mathbf{B}}(F, G, H)$.

(ii) *If* $G \in \mathscr{F}(B)$ *and* $a \backslash b \notin G$, *then there exist* $F, H \in \mathscr{F}(B)$ *such that* $a \in F$, $b \notin H$ *and* $\mathscr{R}_{\mathbf{B}}(F, G, H)$.

(iii) *If* $F \in \mathscr{F}(B)$ *and* $b/a \notin F$, *then there exist* $G, H \in \mathscr{F}(B)$ *such that* $a \in G$, $b \notin H$ *and* $\mathscr{R}_{\mathbf{B}}(F, G, H)$.

Proof: We show (i). Since $a \otimes b \in H$, then $x \otimes y \in H$ for all $a \leq x$ and $b \leq y$. So, $\{x \otimes y : x \in [a) \text{ and } y \in [b)\} \subseteq H$ and, by Corollary 3.15, there exist prime filters F, G such that $\mathscr{R}_{\mathbf{B}}(F, G, H)$.

We show (ii). Let G be a prime filter such that $a \backslash b \notin G$. We consider $aG = \{a \otimes x : x \in G\}$. We extend aG to be filter. Let $Q = \{x \in L : \underset{y \in aG}{\exists} y \leq x\}$. Clearly, (F1) holds. Let $x, y \in Q$. Then, for some $x', y' \in G$ we have $a \otimes x' \leq x$ and $a \otimes y' \leq y$. Since $x', y' \in G$, then $x' \wedge y' \in G$ and $a \otimes (x' \wedge y') \in aG$. So:

$$a \otimes (x' \wedge y') \leq (a \otimes x') \wedge (a \otimes y') \leq x \wedge y$$

Hence, $x \wedge y \in Q$. We show $b \notin Q$. Suppose $b \in Q$, then, for some $x \in G$, $a \otimes x \leq b$. By (RES), $x \leq a \backslash b$. Hence, $a \backslash b \in G$ – contradiction. So, Q is a filter and $b \notin Q$. By Theorem 3.8, there exists a prime filter H such that

$Q \subseteq H$ and $b \notin H$. So, we have $\{x \otimes y : x \in [a) \text{ and } y \in G\} \subseteq H$. By Theorem 3.10, there exists a prime filter F such that $\mathscr{R}_{\mathbf{L}}(F, G, H)$.

One shows (iii) analogously. □

Lemma 3.17. *Let* $\mathbf{B}$ *be a partial residuated Boolean algebra and let* $a, b \in L$ *be such that* $a \not\leq b$. *There exists a prime filter* $F \subseteq B$ *such that* $a \in F$ *and* $b \notin F$.

Proof: By definition of a partial residuated Boolean algebra, there exists a total residuated Boolean algebra $\mathbf{B}'$ such that ι is an embedding of $\mathbf{B}$ into $\mathbf{B}'$. Then, by Corollary 3.9, there exists a prime filter $F \subseteq B'$ such that $a \in F$ and $b \notin F$. Clearly, $\iota^{-1}(F)$ is a prime filter of $\mathbf{B}$ and $a \in \iota^{-1}(F)$ and $b \notin \iota^{-1}(F)$. □

Proposition 3.18. Let $\mathbf{B} = (B, \otimes, \backslash, /, \vee, \wedge, \neg, 1, \top, \bot, \leq)$ be a partial residuated Boolean algebra. Let $\mathbf{B}_{\mathfrak{F}_{\mathbf{B}}}$ be the complex Boolean algebra of the associated residuated frame. We define $\iota(a) = \{F \in \mathcal{F}_B : a \in F\}$ for all $a \in B$. Then, ι is an embedding.

Proof: Let $a \leq b$. Then, for all $H \in \iota(a)$, we have $b \in H$, so $H \in \iota(b)$. Hence, $\iota(a) \subseteq \iota(b)$. Let $a \not\leq b$. By Theorem 3.17, there exists a prime filter H such that $a \in H$ and $b \notin H$. Hence, $\iota(a) \not\subseteq \iota(b)$. Therefore, $a \leq b$ iff $\iota(a) \subseteq \iota(b)$. As a consequence, ι is injective.

Since prime filters are proper filters, $\iota(\bot) = \emptyset$. $\top$ is an element of every filter, so $\iota(\top) = \mathscr{F}(B)$.

Let $a, b \in B$ and $a \otimes b \neq \infty$. By definition:

$$\iota(a) \otimes' \iota(b) = \left\{ H \in \mathscr{F}(B) : \underset{F,G \in \mathscr{F}(B)}{\exists} F \in \iota(a) \text{ and } G \in \iota(b) \text{ and } \mathscr{R}_{\mathbf{B}}(F, G, H) \right\}.$$

We show $\iota(a \otimes b) \subseteq \iota(a) \otimes' \iota(b)$. Let $H \in \iota(a \otimes b)$. Then, $a \otimes b \in H$ and by Theorem 3.16(i), there exist $F, G \in \mathscr{F}(L)$ such that $a \in F$, i.e. $F \in \iota(a)$ and $b \in G$, i.e. $G \in \iota(b)$ and $\mathscr{R}_{\mathbf{B}}(F, G, H)$.

We show $\iota(a) \otimes' \iota(b) \subseteq \iota(a \otimes b)$. Let $H \in \iota(a) \otimes' \iota(b)$. Then, for some $F \in \iota(a)$ and $G \in \iota(b)$ we have $\mathscr{R}_{\mathbf{B}}(F, G, H)$. In particular, $a \in F$, $b \in G$, so $a \otimes b \in H$, by definition of $\mathscr{R}_{\mathbf{B}}$. Hence, $H \in \iota(a \otimes b)$.

For $a \backslash b$ and a / b we prove analogously, using (ii) and (iii) of Theorem 3.16 and Theorem 3.14.

Let $a \vee b \neq \infty$. We show $\iota(a \vee b) \subseteq \iota(a) \cup \iota(b)$. Let $H \in \iota(a \vee b)$, then $a \vee b \in H$. Since H is a prime filter, $a \in H$ or $b \in H$. Hence, $H \in \iota(a)$ or $H \in \iota(b)$. Conversely, let $a \in H$ or $b \in H$. Then, $a \vee b \in H$, by (F1). So, $\iota(a) \cup \iota(b) \subseteq \iota(a \vee b)$.

Let $a \wedge b \neq \infty$. Let $H \in \iota(a \wedge b)$. Then, $a \in H$ and $b \in H$, by (F1). Hence, $H \in \iota(a)$ and $H \in \iota(b)$, i.e. $H \in \iota(a)$. Conversely, let $H \in \iota(a)$. Then, by (F2'), $a \wedge b \in H$, so $H \in \iota(a \wedge b)$. ☐

The following theorem allows us to identify the partial residuated Boolean algebras. Its proof is a merge of the proofs from [9] and [10]. We skip identical parts and we focus on nontrivial differences.

THEOREM 3.19. *Let* $\mathbf{B} = (B, \otimes, \backslash, /, \vee, \wedge, \neg, 1, \top, \bot, \leq)$ *be a partial structure such that* $\neg a \neq \infty$, $\neg a \in B$, $a \vee \neg a = \top$, $a \wedge \neg a = \bot$ *and* $1 \otimes a = a = a \otimes 1$ *for all* $a \in B$. *Then,* $\mathbf{B}$ *is a partial residuated Boolean algebra if, and only if, it is a partial bounded lattice and there exists a set* $\mathcal{F}$ *of prime filters of* $\mathbf{B}$ *and a set* $\mathcal{I} \subseteq \mathcal{F}$ *such that* $1 \in F$ *for all* $F \in \mathcal{I}$ *such that the following conditions hold:*

$$\textit{(S)}\quad \underset{a,b\in L}{\forall}\Big(\textit{if } a \not\leq b,\ \textit{then } \underset{F\in\mathcal{F}}{\exists}\, a \in F \textit{ and } b \notin F\Big)$$

$$\textit{(M}\otimes\textit{)}\quad \underset{H\in\mathcal{F}}{\forall}\ \underset{a,b\in L}{\forall}\Big(\textit{if } a \otimes b \in H,\ \textit{then } \underset{F,G\in\mathcal{F}}{\exists}\, a \in F \textit{ and } b \in G \textit{ and } \mathscr{R}_{\mathbf{L}}(F, G, H)\Big)$$

$$\textit{(M}\backslash\textit{)}\quad \underset{G\in\mathcal{F}}{\forall}\ \underset{a,b\in L}{\forall}\Big(\textit{if } a\backslash b \neq \infty \textit{ and } a\backslash b \notin G,\ \textit{then } \underset{F,H\in\mathcal{F}}{\exists}\, a \in F \textit{ and } b \notin H \textit{ and } \mathscr{R}_{\mathbf{L}}(F, G, H)\Big)$$

$$\textit{(M/)}\quad \underset{F\in\mathcal{F}}{\forall}\ \underset{a,b\in L}{\forall}\Big(\textit{if } a/b \neq \infty \textit{ and } a/b \notin F,\ \textit{then } \underset{G,H\in\mathcal{F}}{\exists}\, a \in G \textit{ and } b \notin H \textit{ and } \mathscr{R}_{\mathbf{L}}(F, G, H)\Big)$$

$$\textit{(M1)}\quad \underset{F\in\mathcal{F}}{\forall}\ \underset{G_1,G_2\in\mathcal{I}}{\exists}\Big(\mathscr{R}_{\mathbf{L}}(F, G_1, F) \textit{ and } \mathscr{R}_{\mathbf{L}}(G_2, F, F)\Big)$$

PROOF: Let $\mathbf{B} = (B, \otimes, \backslash, /, \vee, \wedge, \neg, 1, \top, \bot, \leq)$ be a partial unital residuated Boolean algebra and let $\mathbf{A} = (A, \otimes', \backslash', /', \vee', \wedge', \neg', 1', \top', \bot', \leq')$ be a total unital residuated Boolean algebra and let ι be an embedding of $\mathbf{B}$ into $\mathbf{A}$. We show that there exists a set $\mathcal{F}$ of prime filters of $\mathbf{B}$ that satisfies (S), (M$\otimes$), (M$\backslash$), (M/) and (M1). We define:

$$\mathcal{F} = \{\iota^{-1}(F) : F \text{ is a prime filter of } \mathbf{A}\}$$

For better readability we use the following notion: let F be a prime filter of $\mathbf{A}$, then $F_\iota = \iota^{-1}(F)$. We prove (S), (M$\otimes$), (M$\backslash$) and (M/) like in [9].

We show there exists $\mathcal{I} \subseteq \mathcal{F}$ such that (M1) holds. We define:

$$\mathcal{I} = \{F \in \mathcal{F} : 1 \in F\}$$

Let $F_\iota \in \mathcal{F}$, then, by Proposition 3.13 there exists a prime filter G of $\mathbf{A}$ such that $1 \in G$ and $\mathscr{R}_{\mathbf{A}}(F, G, F)$. Then, $G_\iota \in \mathcal{I}$ and $\mathscr{R}_{\mathbf{B}}(F_\iota, G_\iota, F_\iota)$. Similarly, there exists H such that $H_\iota \in \mathcal{I}$ and $\mathscr{R}_{\mathbf{B}}(H_\iota, F_\iota, F_\iota)$.

Now we assume $\mathbf{B}$ is a partial structure satisfying the assumptions of the theorem. We construct the residuated Boolean algebra $\mathbf{A}$ and the embedding of $\mathbf{B}$ into $\mathbf{A}$. We see $\mathfrak{F} = (\mathcal{F}, \mathcal{I}, \mathscr{R}_{\mathbf{B}})$ satisfies (U1)–(U4). We show (U5). Let $F, H \in \mathcal{F}$ and $G \in \mathcal{I}$ be such that $\mathscr{R}_{\mathbf{B}}(F, G, H)$. Then, for all $a \in F$, since $1 \in G$, we have $a \otimes 1 \in H$, so $F \subseteq H$. Suppose there exists $a \in H$ such that $a \notin F$. Then, by (FB), $\neg a \in F$, which is impossible.

Let $\mathbf{A} = (\mathscr{P}(\mathcal{F}), \otimes, \backslash, /, \cup, \cap, \mathcal{I}, \mathcal{F}, \emptyset, \subseteq)$ be the complex algebra of $\mathfrak{F}$. We define the mapping ι for every $a \in L$ by $\iota(a) = \{F \in \mathcal{F} : a \in F\}$. We show ι is an embedding.

Let $a, b \in L$ and $a \leq b$. Then, $\iota(a) \subseteq \iota(b)$, by (F1). Let $a \not\leq b$, then by (S) there exists $F \in \mathcal{F}$ such that $a \in F$ and $b \notin F$, so $\iota(a) \not\subseteq \iota(b)$. Hence $a \leq b$ iff $\iota(a) \subseteq \iota(b)$ and ι is injective.

One shows ι preserves $\otimes, \backslash, /, \vee, \wedge, \top, \bot$, analogously like in [9].

We show $\iota(1) = \mathcal{I}$. The inclusion $\mathcal{I} \subseteq \iota(1)$ is trivial, since 1 belongs to every element of $\mathcal{I}$. Let $F \in \iota(1)$. By (M1), there exists $G \in \mathcal{I}$ such that $\mathscr{R}_{\mathbf{B}}(F, G, F)$. Since $1 \in F$, then $G \subseteq F$. Suppose $a \in F$ and $a \notin G$. Then, by (FB), $\neg a \in G$ and then $\neg a \in F$, which is impossible. So, $G = F$ and $F \in \mathcal{I}$.

Let $a \in B$, then $\iota(\neg a) = \{F \in \mathcal{F} : \neg a \in F\} = \{F \in \mathcal{F} : a \notin F\}$, by (FB). Thus, $\{F \in \mathcal{F} : a \notin F\} = \{F \in \mathcal{F} : a \in F\}^c$. □

4. The upper bound of complexity

In this section we show that the finitary consequence relation for BFNL is decidable in exponential time.

LEMMA 4.1. *Let* $\mathbf{B} = (B, \otimes, \backslash, /, \vee, \wedge, \neg, 1, \top, \bot, \leq)$ *be a partial structure. We can verify whether* $\mathbf{B}$ *is a partial residuated Boolean algebra in exponential time (depending on* $|B|$*).*

By definition, $\mathbf{B}$ is a partial residuated Boolean algebra if it is embeddable in a total residuated Boolean algebra. Such a total algebra may have the same set of elements, but may also have additional elements to satisfy all the properties. Hence, to check if $\mathbf{B}$ is a partial residuated Boolean algebra by definition, we need to embed $\mathbf{B}$ in every possible total structure until we find one where all the properties of residuated Boolean algebra hold. Even with the limit on the maximal size of such a structure, it would be 2EXPTIME problem.

Hence, we use Theorem 3.19 to identify partial residuated Boolean algebras.

PROOF: We provide an algorithm to verify whether $\mathbf{B}$ is a partial residuated Boolean algebra. We follow the analogous lemma and its proof from [9].

Step 1. We check whether $\leq$ is a partial order, $\top, \bot$ are bounds and the lattice operators are compatible with $\leq$. If it fails, the algorithm stops with negative answer. It can be done in the polynomial time.

Step 2. We check whether $1 \otimes a = a$ and $a \otimes 1 = a$ for all $a \in L$. If it fails, the algorithm stops with negative answer. It can be done in the polynomial time.

Step 3. We check whether $\neg a \neq \infty$, $\neg a \in B$, $a \vee \neg a = \top$ and $a \wedge \neg a = \bot$ for all $a \in B$. If it fails, the algorithm stops with negative answer. It can be done in the polynomial time.

Step 4. We construct a descreasing sequence of families of filters $\mathcal{F}_n$. We construct the set $\mathcal{F}_0$ of all prime filters of $\mathbf{B}$. For every subset $S \subseteq B$ we check the definition of prime filter. It can be done in $\mathcal{O}(2^{2|B|})$.

We set $i = 0$.

Step 4.1. We define $\mathcal{I}_i = \{F \in \mathcal{F}_i : 1 \in F\}$. For every prime filter $F \in \mathcal{F}_i$ we check (M$\otimes$), (M$\backslash$), (M/) and (M1). If every of these condition holds for F, then we add F to set $\mathcal{F}_{i+1}$.

Step 4.2. If $\mathcal{F}_{i+1} = \emptyset$, then the algorithm stops with negative answer. If $\mathcal{F}_i = \mathcal{F}_{i+1}$, then the algorithm proceeds to the next step. Else, the algorithm goes back to Step 0.1 with $i + 1$.

Checking conditions for arbitrary F can be done in $\mathcal{O}(2^{3|B|})$. Number of filters in $\mathcal{F}_i$ is $\mathcal{O}(2^{|B|})$. Maximal i does not exceed $2^{|B|}$. So this step can be done in $\mathcal{O}(2^{5|B|})$.

Step 5. We check (S). If (S) does not hold, then the algorithm stops with negative answer. If (S) does not hold for a family of filters, then it does not hold for any smaller family. It can be done in $\mathcal{O}(|B|^2 2^{|B|})$ time. □

We notice that every sequent $\Gamma \Rightarrow C$ can be represented as $G \Rightarrow C$, where G is a formula arising from Γ by replacing every comma by $\otimes$, every semicolon by $\wedge$, ϵ by 1 and δ by $\top$. So, we consider only sequents of this form.

Let $G \Rightarrow A$ be a sequent. We define the size of $G \Rightarrow A$ as follows:

$$s(p) = 1 \qquad s(1) = 1$$

$$s(\top) = 1 \qquad s(\bot) = 1$$

$$s(A \otimes B) = s(A) + s(B) + 1$$

$$s(A\backslash B) = s(A) + s(B) + 1 \qquad s(A/B) = s(A) + s(B) + 1$$

$$s(A \wedge B) = s(A) + s(B) + 1 \qquad s(A \vee B) = s(A) + s(B) + 1$$

$$s(\neg A) = s(A) + 1 \qquad s(A \to B) = s(A) + s(B) + 1$$

$$s(G \Rightarrow A) = s(G) + s(A)$$

Definition 4.2. Let $\mathbf{A}$ be a partial residuated Boolean algebra. Let μ be a partial function from the free algebra of $\mathcal{L}$–formulas into $\mathbf{A}$. We say μ is a *valuation*, if the following conditions hold:

- $\mu(\top) = \top$, $\mu(\bot) = \bot$;
- $\mu(1) = 1$;
- if $\mu(D \otimes E) \neq \infty$, then $\mu(D) \neq \infty, \mu(E) \neq \infty$ and $\mu(D \otimes E) = \mu(D) \otimes \mu(E)$;
- if $\mu(D \backslash E) \neq \infty$, then $\mu(D) \neq \infty, \mu(E) \neq \infty$ and $\mu(D \backslash E) = \mu(D) \backslash \mu(E)$;
- if $\mu(D/E) \neq \infty$, then $\mu(D) \neq \infty, \mu(E) \neq \infty$ and $\mu(D/E) = \mu(D)/\mu(E)$;
- if $\mu(D \wedge E) \neq \infty$, then $\mu(D) \neq \infty, \mu(E) \neq \infty$ and $\mu(D \wedge E) = \mu(D) \wedge \mu(E)$;
- if $\mu(D \vee E) \neq \infty$, then $\mu(D) \neq \infty, \mu(E) \neq \infty$ and $\mu(D \vee E) = \mu(D) \vee \mu(E)$;
- if $\mu(\neg D) \neq \infty$, then $\mu(D) \neq \infty$ and $\mu(\neg D) = \neg\mu(D)$;

Let $G \Rightarrow C$ be a sequent and μ be a valuation. We say $G \Rightarrow C$ is satisfied under the valuation μ, if $\mu(G) \neq \infty$, $\mu(C) \neq \infty$ and $\mu(G) \leq \mu(C)$.

Now we are ready to prove the *EXPTIME* complexity of the consequence relations. The following theorem was formulated in [9] in algebraic terms of satisfiability of quantifier–free first–order formulas of the language of residuated distributive lattices.

Theorem 4.3. *The finitary consequence relation of BFNL is EXPTIME.*

PROOF:

(i) Let $\mathcal{K}$ be the class of residuated Boolean algebras, $\Phi = \{G_1 \Rightarrow C_1, G_2 \Rightarrow C_2, \ldots, G_k \Rightarrow C_k\}$ be a set of sequents and $G \Rightarrow C$ a sequent. Let:

$$n := 2(s(G_1 \Rightarrow C_1)+s(G_2 \Rightarrow C_2)+\cdots+s(G_k \Rightarrow C_k)+s(G \Rightarrow C))+4.$$

We show that Φ entails $G \Rightarrow C$, if, and only if, for all $\mathbf{A} \in \mathcal{K}^P$ such that $|A| \leq n$ and all valuations μ, if all sequents from Φ are satisfied in $\mathbf{A}$ under the valuation μ and both $\mu(G)$ and $\mu(C)$ are defined, then $G \Rightarrow C$ is satisfied in $\mathbf{A}$ under the valuation μ.

(1.1) Let $\mathbf{A} \in \mathcal{K}^P$, $|A| \leq n$ and μ be a valuation. Assume all sequents from Φ are satisfied in $\mathbf{A}$ under the valuation μ and both $\mu(G)$ and $\mu(C)$ are defined, but $G \Rightarrow C$ is not satisfied, i.e. $\mu(G) \not\leq \mu(C)$. Then, for some $\mathbf{A}' \in \mathcal{K}$, we have an embedding ι of $\mathbf{A}$ into $\mathbf{A}'$. Then, $\iota(\mu(G_i)) \leq' \iota(\mu(C_i))$ for all $i = 1, \ldots, k$ and $\iota(\mu(G)) \not\leq' \iota(\mu(C))$ in $\mathbf{A}'$. Hence, for the valuation $\mu' = \iota \circ \mu$ all sequents from Φ are satisfied, but $G \Rightarrow C$ is not satisfied in $\mathbf{A}'$. Thus, Φ does not entail $G \Rightarrow C$.

(1.2) Now let $G \Rightarrow C$ not be satisfied in $\mathbf{A}' \in \mathcal{K}$ under the valuation μ', but all sequents from Φ be satisfied under μ'. We construct $\mathbf{A} \in \mathcal{K}^P$.

First, we define T as the set consisting of $1, \top, \bot$ and all subformulas of $G_1, C_1, \ldots, G_k, C_k, G, C$. We put $A = \{\mu'(D) : D \in T\} \cup \{\neg'\mu'(D) : D \in T\}$. In effect, negation is a total operation, but doing this does not change final complexity. We define partial operations as follows:

- if $D \in T$ and $D = E \otimes F$, then $\mu'(E) \otimes \mu'(F) := \mu'(E \otimes F)$;
- if $D \in T$ and $D = E \backslash F$, then $\mu'(E) \backslash \mu'(F) := \mu'(E \backslash F)$;
- if $D \in T$ and $D = E/F$, then $\mu'(E)/\mu'(F) := \mu'(E/F)$;
- if $D \in T$ and $D = E \vee F$, then $\mu'(E) \vee \mu'(F) := \mu'(E \vee F)$;
- if $D \in T$ and $D = E \wedge F$, then $\mu'(E) \wedge \mu'(F) := \mu'(E \wedge F)$;

We define $1 \otimes a := a$ and $a \otimes 1 := a$ and $\neg a := \neg' a$ and $a \vee \neg a := \top$ and $a \wedge \neg a := \bot$ for all $a \in A$.

We also define $\leq = \leq' \cap A^2$. By the construction, $|A| \leq n$ and $\mathbf{A} \in \mathcal{K}^P$. We define $\mu = \mu'_{|T}$. Clearly, μ satisfies the conditions of Definition 4.2 and $\mu(G_i) \leq \mu(C_i)$ for $i = 1, \ldots, k$ and $\mu(G) \not\leq \mu(C)$ and both $\mu(G)$ and $\mu(C)$ are defined.

(ii) Thus, to verify whether $\Phi \vdash G \Rightarrow C$ we check whether $G \Rightarrow C$ is satisfied in all $\mathbf{A} \in \mathcal{K}^P$ under every valuation μ such that $|A| \leq n$ and all sequents from Φ are satisfied in $\mathbf{A}$ under μ and both $\mu(G)$ and $\mu(C)$ are defined.

We construct all partial residuated Boolean algebras with cardinality not exceeding n. Each such a structure can be encoded by matrices. Every binary operation and order is encoded by a matrix of size $\mathcal{O}(n^2)$ and negation is encoded by matrix of size $\mathcal{O}(n)$. Each entry in the matrix can take $\mathcal{O}(n)$ values (including ∞). Hence, we have $\mathcal{O}(2^{Ln^3})$ possibilities, where L is a positive integer. We check whether such a structure is a partial residuated Boolean algebra, using Theorem 4.1. This step can be done in $\mathcal{O}(2^{Ln^3} 2^{5n})$.

For a given residuated Boolean algebra $\mathbf{A}$ the number of all possible valuations is $\mathcal{O}(|A|^n)$. Checking if all sequents from Φ and $G \Rightarrow C$ are satisfied under the arbitrary valuation is $\mathcal{O}(n)$. Hence, checking whether Φ entails $G \Rightarrow C$ in $\mathbf{A}$ is $\mathcal{O}(2^{n^3})$.

The time of the whole algorithm is $\mathcal{O}(2^{Ln^3} 2^{5n} 2^{n^3}) = \mathcal{O}(2^{(L+1)n^3+5n})$.

□

The analogous result for BFL (associative version of BFNL) does not hold. BFL is a strongly conservative extension of L and the consequence relation of L is undecidable [1].

If we exclude the constant 1 from BFNL, the result remains true. Moreover, for 1-free BFNL the lower bound of complexity of the consequence relation is also EXPTIME, since 1-free BFNL is a strongly conservative extension of 1-free DFNL which is EXPTIME-complete [9]. The lower bound of complexity for BFNL or DFNL with 1 remains an open problem.

5. Complexity of HFNL

In this section, we provide detailed instructions on how to prove the same result for HFNL, modifying the definitions, theorems and proofs from previous sections.

We start with the following definition.

Definition 5.1. Let $\mathbf{H} = (H, \otimes, \backslash, /, \vee, \wedge, \rightarrow, 1, \top, \bot, \leq)$ be a partial structure. We say $\mathbf{H}$ is a *partial residuated Heyting algebra*, if there exists a total residuated Heyting algebra such that $\mathbf{H}$ is embeddable into it. One notices that $(H, \otimes, \backslash, /, \vee, \wedge, 1, \top, \bot, \leq)$ is a partial bounded distributive residuated lattice.

One sees that the definition is analogous to the definition of partial residuated Boolean algebra. The most important is that it is still a partial distributive lattice. Therefore, we can define filters in the analogous way.

Let $\mathbf{H} = (H, \otimes, \backslash, /, \vee, \wedge, \rightarrow, 1, \top, \bot, \leq)$ be a partial residuated Heyting algebra and $F \subseteq H$. We say F is a filter, if (F1), (F2') and the following condition are satisfied.

(FH) if $a \rightarrow b \neq \infty$ and $a \in F$ and $a \rightarrow b \in F$, then $b \in F$

We notice that (FB) follows from (F1) and (F2') in total residuated Heyting algebras. A filter is prime if it is proper and satisfies (F3).

We define associated residuated frames to partial residuated Heyting algebras the same way we defined them for partial residuated Boolean algebras, since the definition again does not depend on negation.

Theorems 3.8, 3.10 and 3.14, corollaries 3.9 and 3.15, and proposition 3.13 do not depend on negation, so they remain true for (partial) residuated Heyting algebras.

The first nontrivial difference is the construction of total residuated Heyting algebra from residuated frames. Let $\mathfrak{F} = (P, I, R)$ be an arbitrary residuated frame and $X, Y \subseteq P$, we define:

$$X \otimes' Y = \left\{ z \in P : \underset{x,y \in P}{\exists} x \in X \text{ and } y \in Y \text{ and } R(x,y,z) \right\}$$

$$X \backslash' Y = \left\{ y \in P : \underset{x,z \in P}{\forall} \text{if } R(x,y,z) \text{ and } x \in X, \text{ then } z \in Y \right\}$$

$$Y /' X = \left\{ x \in P : \underset{y,z \in P}{\forall} \text{if } R(x,y,z) \text{ and } y \in X, \text{ then } z \in Y \right\}$$

$$X \to' Y = \bigcup \{ Z \in \mathscr{P}(P) : X \cap Z \subseteq Y \}$$

Then $\mathbf{H}_{\mathfrak{F}} = (\mathscr{P}(P), \otimes', \backslash', /', \cup, \cap, \to', I, P, \emptyset, \subseteq)$ is a residuated Heyting algebra.

The analogues of theorems 3.16 and 3.17 are true, since the proofs do not use negations.

The most important difference lies in the formulation of the following theorem.

Theorem 5.2. *Let* $\mathbf{H} = (H, \otimes, \backslash, /, \vee, \wedge, \to, 1, \top, \bot, \leq)$ *be a partial structure such that* $1 \otimes a = a = a \otimes 1$ *for all* $a \in H$*. Then,* $\mathbf{H}$ *is a partial unital residuated Heyting algebra if, and only if, it is a partial bounded lattice and there exists a set* $\mathcal{F}$ *of prime H–filters of* $\mathbf{H}$ *such that (S), (M⊗), (M\\), (M/) are satisfied and there exists a set* $\mathcal{I} \subseteq \mathcal{F}$ *such that* $1 \in F$ *for all* $F \in \mathcal{I}$ *and (M1) is satisfied, and the following conditions hold:*

(H1) $\underset{a,b \in H}{\forall}$ *if* $a \to b \neq \infty$*, then* $b \leq a \to b$

(H2) $\underset{F \in \mathcal{F}}{\forall} \underset{a,b \in H}{\forall} \Big($*if* $a \to b \neq \infty$ *and* $a \notin F$ *and* $a \to b \notin F$*,*

then $\underset{F' \in \mathcal{F}}{\exists} \big(F \subseteq F'$ *and* $a \in F'$ *and* $a \to b \notin F'\big)\Big)$

The proof is similar. We show only the important parts.

Proof: Let $\mathbf{H} = (H, \otimes, \backslash, /, \vee, \wedge, \to, \top, \bot, \leq)$ be a partial residuated Heyting algebra and let $\mathbf{A} = (A, \otimes', \backslash', /', \vee', \wedge', \to', \top', \bot', \leq')$ be a residuated Heyting algebra and let ι be an embedding of $\mathbf{H}$ into $\mathbf{A}$. Clearly, (H1) holds. We show that there exists a set $\mathcal{F}$ of prime filters of $\mathbf{H}$ satisfying (S), (H2), (M⊗), (M\) and (M/). We define:

$$\mathcal{F} = \{\iota^{-1}(F) : F \text{ is a prime filter of } \mathbf{A}\}$$

We prove (H2). Let $a, b \in H$ be such that $a \to b \neq \infty$. Assume $a, a \to b \notin F_\iota$ for some prime filter F of $\mathbf{A}$. Then, $\iota(a), \iota(a \to b) \notin F$. We have $\iota(a \to b) = \iota(a) \to' \iota(b)$. We take $F_{\iota(a)}$, so $\iota(a) \in F_{\iota(a)}$. Suppose, $\iota(a) \to' \iota(b) \in F_{\iota(a)}$. Then, for some $x \in F$, $\iota(a) \wedge x \leq' \iota(a) \to' \iota(b)$. By residuation law, $\iota(a) \wedge \iota(a) \wedge x \leq' \iota(b)$. Clearly, $\iota(a) \wedge \iota(a) \wedge x = \iota(a) \wedge x$, so $\iota(a) \wedge x \leq' \iota(b)$ and $x \leq' \iota(a) \to' \iota(b)$. Hence, $\iota(a) \to' \iota(b) \in F$, which contradicts the assumption. Thus, $\iota(a) \to' \iota(b) \notin F_{\iota(a)}$. By Theorem 3.8, there exists a prime filter F' of $\mathbf{A}$ such that $F_{\iota(a)} \subseteq F'$ and $\iota(a) \to' \iota(b) \notin F'$.

The rest of this part proceeds like in Theorem 3.19.

Now we assume $\mathbf{H}$ is a partial structure satisfying the assumptions of the theorem. We construct the residuated Heyting algebra $\mathbf{A}$ and the embedding ι of $\mathbf{H}$ into $\mathbf{A}$. We see $\mathfrak{F} = (\mathcal{F}, \subseteq, \mathscr{R}_{\mathbf{H}})$ is a residuated frame. Let $\mathbf{A} = (\mathscr{P}(\mathcal{F}), \otimes, \backslash, /, \cup, \cap, \to', \mathcal{F}, \emptyset, \subseteq)$ be the complex algebra of $\mathfrak{F}$. We define the mapping ι for every $a \in H$ by $\iota(a) = \{F \in \mathcal{F} : a \in F\}$. We show ι is an embedding.

Since $\mathcal{F}$ satisfies (S), (M$\otimes$), (M$\backslash$) and (M/), we need to prove only that $\iota(a \to b) = \iota(a) \to' \iota(b)$. The rest can be shown in a similar way like in Theorem 3.19. Let $a, b \in H$ and $a \to b \neq \infty$. We recall that:

$$\iota(a) \to' \iota(b) = \bigcup\{X \in \mathcal{F} : \iota(a) \cap X \subseteq \iota(b)\}$$

We show $\iota(a \to b) = \iota(a) \to \iota(b)$. One notices $F \in \iota(a) \cap \iota(a \to b)$ iff $a \in F$ and $a \to b \in F$. By (FH), $b \in F$, so $F \in \iota(b)$. Hence, $\iota(a \to b) \subseteq \iota(a) \to' \iota(b)$.

Let $\{X_i\}_{i \in I}$ be an arbitrary family such that $X_i = \bigcap\{\iota(c_{i,j}) : j \in J_i\}$ and $\iota(a) \cap X_i \subseteq \iota(b)$ for some family $\{c_{i,j}\}_{j \in J_i}$ for all $i \in I$. Then, for all $F \in \mathcal{F}$ such that $a \in F$ and $\{c_{i,j}\}_{j \in J_i} \subseteq F$ we have $b \in F$, since $F \in \iota(a)$ and $F \in X$.

Let $F \in X_i$. Assume $a \in F$, then $b \in F$ and $F \in \iota(b)$. By (H1), $b \leq a \to b$, so $a \to b \in F$ and $F \in \iota(a \to b)$. Assume $a \notin F$ and suppose $a \to b \notin F$. By (H2), there exists $F' \in \mathcal{F}$ such that $F \subseteq F'$ and $a \in F'$ and $a \to b \notin F'$. Then, $b \notin F'$. But $\{c_{i,j}\}_{j \in J_i} \subseteq F \subseteq F'$, which contradicts

$b \notin F'$. Hence, $a \to b \in F$.

Let $X = \bigcup\{X_i : i \in I\}$. Clearly, $\iota(a) \cap X = \bigcup\{\iota(a) \cap X_i : i \in I\} \subseteq \iota(b)$. For every $i \in I$ we have $X_i \subseteq \iota(a \to b)$, hence $X \subseteq \iota(a \to b)$. Thus, $\iota(a) \to \iota(b) \subseteq \iota(a \to b)$. □

Now we are ready to provide the complexity results.

LEMMA 5.3. *Let* $\mathbf{H} = (H, \otimes, \backslash, /, \vee, \wedge, \to, 1, \top, \bot, \leq)$ *be a partial structure. We can verify whether* $\mathbf{H}$ *is a partial unital residuated Heyting algebra in exponential time (depending on* $|H|$*).*

PROOF: We modify the algorithm provided in the proof of Theorem 4.1.

Step 1. We check whether $\leq$ is a partial order, $\top, \bot$ are bounds and the lattice operators are compatible with $\leq$. If it fails, the algorithm stops with negative answer. It can be done in the polynomial time.

Step 2. We check whether $1 \otimes a = a$ and $a \otimes 1 = a$ for all $a \in L$. If it fails, the algorithm stops with negative answer. It can be done in the polynomial time.

Step 3. We check (H1) in polynomial time.

Step 4. We construct a descreasing sequence of families of filters $\mathcal{F}_n$. We construct the set $\mathcal{F}_0$ of all prime filters of $\mathbf{B}$. For every subset $S \subseteq B$ we check the definition of prime filter. It can be done in $\mathcal{O}(2^{2|H|})$.

We set $i = 0$.

Step 4.1. We define $\mathcal{I}_i = \{F \in \mathcal{F}_i : 1 \in F\}$. For every prime filter $F \in \mathcal{F}_i$ we check (M$\otimes$), (M$\backslash$), (M/), (M1) and (H2). If every of these condition holds for F, then we add F to set $\mathcal{F}_{i+1}$.

Step 4.2. If $\mathcal{F}_{i+1} = \emptyset$, then the algorithm stops with negative answer. If $\mathcal{F}_i = \mathcal{F}_{i+1}$, then the algorithm proceeds to the next step. Else, the algorithm goes back to Step 0.1 with $i + 1$.

Checking conditions for arbitrary F can be done in $\mathcal{O}(2^{3|H|})$. Number of filters in $\mathcal{F}_i$ is $\mathcal{O}(2^{|H|})$. Maximal i does not exceed $2^{|H|}$. So this step can be done in $\mathcal{O}(2^{5|H|})$.

Step 5. We check (S). If (S) does not hold, then the algorithm stops with negative answer. If (S) does not hold for a family of filters, then it does not hold for any smaller family. It can be done in $\mathcal{O}(|H|^2 2^{|H|})$ time.

Hence, time of the whole algorithm remains $\mathcal{O}(2^{5|H|})$. □

Theorem 5.4. *The finitary consequence relation of HFNL is EXPTIME.*

The proof is analogous. We just skip the parts regarding negations and add a new connective → similarly as the rest of binary connectives.

The analogous result for HFL (associative version of HFNL) does not hold. HFL is a strongly conservative extension of L and the consequence relation of L is undecidable [1].

If we exclude the constant 1 from HFNL, the result remains true. Moreover, for 1-free HFNL the lower bound of complexity of the consequence relation is also EXPTIME, since 1-free HFNL is a strongly conservative extension of 1-free DFNL which is EXPTIME-complete [9]. The lower bound of complexity for HFNL remains an open problem.

References

[1] W. Buszkowski, *Lambek calculus with nonlogical axioms*, [in:] C. Casadio, P. J. Scott, R. A. G. Seely (eds.), **Language and Grammar. Studies in Mathematical Linguistics and Natural Language** (2005), pp. 77–93.

[2] W. Buszkowski, *Lambek Calculus with Classical Logic*, [in:] R. Loukanova (ed.), **Natural Language Processing in Artificial Intelligence—NLPinAI 2020**, Springer International Publishing (2021), pp. 1–36, DOI: https://doi.org/10.1007/978-3-030-63787-3_1.

[3] K. Chvalovskỳ, *Undecidability of consequence relation in full non-associative Lambek calculus*, **Journal of Symbolic Logic**, vol. 80(2) (2015), pp. 567—586, DOI: https://doi.org/10.1017/jsl.2014.39.

[4] N. Galatos, P. Jipsen, *Distributive residuated frames and generalized bunched implication algebras*, **Algebra universalis**, vol. 78(3) (2017), pp. 303–336, DOI: https://doi.org/10.1007/s00012-017-0456-x.

[5] J. Lambek, *The mathematics of sentence structure*, **The American Mathematical Monthly**, vol. 65(3) (1958), pp. 154–170.

[6] J. Lambek, *On the calculus of syntactic types*, [in:] R. Jakobson (ed.), **Structure of Language and Its Mathematical Aspects**, vol. 12, Providence, RI: American Mathematical Society (1961), pp. 166–178.

[7] M. Pentus, *Lambek calculus is NP-complete*, **Theoretical Computer Science**, vol. 357(1) (2006), pp. 186–201, DOI: https://doi.org/10.1016/j.tcs.2006.03.018.

[8] P. Płaczek, *Complexity of Nonassociative Lambek Calculus with classical logic*, [in:] A. Indrzejczak, M. Zawidzki (eds.), **Proceedings of the 11th International Conference on Non-Classical Logics. Theory and Applications**, vol. 415 of Electronic Proceedings in Theoretical Computer Science, Open Publishing Association (2024), pp. 150–164, DOI: https://doi.org/10.4204/eptcs.415.15.

[9] D. Shkatov, C. J. Van Alten, *Complexity of the universal theory of bounded residuated distributive lattice-ordered groupoids.*, **Algebra Universalis**, vol. 80(3) (2019), DOI: https://doi.org/10.1007/s00012-019-0609-1.

[10] C. J. van Alten, *Partial algebras and complexity of satisfiability and universal theory for distributive lattices, boolean algebras and Heyting algebras*, **Theoretical Computer Science**, vol. 501 (2013), pp. 82–92, DOI: https://doi.org/10.1016/j.tcs.2013.05.012.

Paweł Płaczek
WSB Merito University in Poznań
Faculty of Finance and Banking
ul. Powstańców Wielkopolskich 5
61-895 Poznań, Poland

e-mail: pawel.placzek@poznan.merito.pl

Funding information: Not applicable.

Conflict of interests: None.

Ethical considerations: The Author assures of no violations of publication ethics and takes full responsibility for the content of the publication.

Declaration regarding the use of GAI tools: Not used.

Bulletin of the **S**ection of **L**ogic
Volume 54/4 (2025), pp. 607–642
https://doi.org/10.18778/0138-0680.2025.19

Yuki Nishimura

AGENT-KNOWLEDGE LOGIC FOR ALTERNATIVE EPISTEMIC LOGIC

Abstract

Epistemic logic is known as a logic that captures the knowledge and beliefs of agents and has undergone various developments. In this paper, we propose a new logic called agent-knowledge logic by taking the product of individual knowledge structures and the set of relationships among agents. This logic is based on the Facebook logic and the Logic of Hide and Seek Game. We show two main results; one is that this logic can embed the standard epistemic logic, and the other is that there is a proof system of tableau calculus that works in finite time. We also discuss various sentences and inferences that this logic can express.

Keywords: agent-knowledge logic, modal logic, epistemic logic, hybrid logic, tableau calculus.

Presented by: Michał Zawidzki
Received: December 25, 2024, **Received in revised form**: December 13, 2025,
Accepted: December 29, 2025, **Published online**: March 13, 2026

1. Introduction

Investigations into knowledge and beliefs form the part of philosophy, which is now called epistemology. This area has been the subject of various studies from a logical standpoint. One of these was conducted by applying modal logic, which is nowadays called epistemic logic. The operator K_i, which is the key element of this logic, constitutes formulas of the form $K_i\varphi$, which express that “agent i knows that φ.” On this basis, it is possible to represent various concepts related to knowledge and belief in formal language. Hintikka did pioneering work on epistemic logic in 1962 [9], and there is a wide range of research today; see Fagin et al. [6] and van Benthem [18].

A more recent logic for human knowledge is Facebook logic, developed by Seligman et al. in 2011 [16]. This logic was invented to describe personal knowledge plus the friendships of agents in a two-dimensional hybrid logic. For instance, consider this sentence: “An agent is Andy’s friend, and Andy knows he has a pollen allergy. Then, one of the agent’s friends knows that they have a pollen allergy.” This inference can be written using the language of Facebook logic as follows:

$$\langle \texttt{Friend} \rangle i \wedge @_i [\texttt{Know}] p \to \langle \texttt{Friend} \rangle [\texttt{Know}] p,$$

where $p =$ “they have pollen allergy.” and $i =$ “This is Andy.” The at sign @ in the logical formula is the operator of hybrid logic, where $@_i p$ can be read as “p holds at point i.” Facebook logic uses nominals, a tool of hybrid logic, to make reference to individual agents. For a thorough introduction to hybrid logics, we refer the reader to Blackburn & ten Cate [2], Indrzejczak [10], and Braüner [4]. Sano [14] provides further details on two-dimensional hybrid logic.

In fact, Facebook logic treats propositional variables differently from epistemic logic. The truth of a propositional variable p depends not only on the epistemic alternative but also on the agent under consideration. Therefore, the propositions represented by the propositional variables here are personal properties, such as, “I have a pollen allergy.”

The new logic proposed in this paper — we will call it *agent-knowledge logic* — is a modification of the aforementioned Facebook logic. One feature of this logic is that the fragment of it is compatible with epistemic logic. This property allows us to use agent-knowledge logic as an alternative to epistemic logic. Indeed, this paper shows how to embed epistemic logic into our new logic. Furthermore, agent-knowledge logic is able to formalize a variety of sentences that cannot be represented by traditional epistemic logic, such as "one of an agent's friends knows p."

In this paper, we also introduce a proof system for the logic by constructing a suitable tableau calculus. The tableau calculus is not only a proof system but also a system for discovering a counterexample model in which the formula is not valid. In particular, by constructing a tableau calculus with the termination property — in short, that the proof ends in finite time — we can show that the logic is decidable.

This logic has two parents: one is Facebook logic, and the other is a logic which seems to have nothing to do with epistemic logic, namely the Logic of Hide and Seek Game (LHS, in short) created by Li et al. in 2021 [11, 12]. This logic was originally invented to illustrate the hide and seek game (also known as cops and robbers). In LHS, propositional variables are split into two sets, which are related to hider and seeker, respectively. We borrow this idea to express the *agent-free* propositions ("the sun rises in the east," for example).

We proceed as follows: Section 2 reviews the well-known epistemic logic and explains the parents of agent-knowledge logic, Facebook logic, and LHS, briefly. In Section 3, we introduce our new logic, that is, agent-knowledge logic. Section 4 shows how we embed epistemic logic into our new logic. In Section 5, we construct a tableau calculus with the termination property and completeness. Finally, in Section 6, we write about some future prospects.

2. Preliminary

2.1. Epistemic Logic

This section is mostly based on the work of Fagin et al. [6, Chapter 2].

In epistemic logic, we have another set **A** of agents besides a usual set **Prop** of propositional variables. The elements of **A** occur in a new operator K_i. The intuitive meaning of $K_i\varphi$ is that "agent i knows φ."

Definition 2.1. We have two disjoint sets, **Prop** and **A**. A formula φ of the *epistemic logic* $\mathcal{L}_{\mathrm{EL}}$ is defined as follows:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid K_i\varphi$$

where $p \in \mathbf{Prop}$ and $i \in \mathbf{A}$.

We only use $\neg$ and $\wedge$ as primitives since other Boolean operators, such as $\vee$ and $\rightarrow$, can be defined as compounds of the first two operators.

Definition 2.2. A *Kripke model for epistemic logic* (we call it *EL model*) $\mathcal{M}_{\mathrm{EL}}$ is a tuple $(W, (R_i)_{i\in\mathbf{A}}, V)$ where:

- W is a non-empty set.
- For each $i \in \mathbf{A}$, R_i is a binary relation on W.
- $V : \mathbf{Prop} \rightarrow \mathcal{P}(W)$.

Definition 2.3. Given an EL model $\mathcal{M}_{\mathrm{EL}}$, its point w, and a formula $\varphi \in \mathcal{L}_{\mathrm{EL}}$, the *satisfaction relation* $\mathcal{M}_{\mathrm{EL}}, w \models \varphi$ is defined inductively as follows:

$$\begin{aligned}
\mathcal{M}_{\mathrm{EL}}, w \models p &\iff w \in V(p), \text{ where } p \in \mathbf{Prop}\\
\mathcal{M}_{\mathrm{EL}}, w \models \neg\varphi &\iff \text{Not } \mathcal{M}_{\mathrm{EL}}, w \models \varphi\ (\mathcal{M}_{\mathrm{EL}}, w \not\models \varphi)\\
\mathcal{M}_{\mathrm{EL}}, w \models \varphi \wedge \psi &\iff \mathcal{M}_{\mathrm{EL}}, w \models \varphi \text{ and } \mathcal{M}_{\mathrm{EL}}, w \models \psi\\
\mathcal{M}_{\mathrm{EL}}, w \models K_i\varphi &\iff \text{For all } v \in W, wR_iv \text{ implies } \mathcal{M}_{\mathrm{EL}}, v \models \varphi.
\end{aligned}$$

As for epistemic logic, we define the validity of a formula. Later we discuss embedding epistemic logic into our new logic, so the formal definition is needed.

Definition 2.4. A formula φ is *valid* with respect to the class of EL models (denoted by $\models_{\mathrm{EL}} \varphi$) if $\mathcal{M}_{\mathrm{EL}}, w \models \varphi$ for every model $\mathcal{M}_{\mathrm{EL}}$ and its every world w.

2.2. Facebook Logic

Facebook logic, first invented by Seligman et al. [16], has two characteristics compared to classical modal logic.

First, we have two modal operators, K and F. These modal operators correspond to knowledge and friendship, respectively. Correspondingly, a possible world is decomposed into two components: one representing an agent and the other representing an epistemic alternative of an individual.

Another addition is the introduction of special propositional variables called *nominals*. A nominal n is a proposition corresponding to only one agent, that is, it is a proposition for the *name* of the agent. In addition, we introduce the satisfaction operator @ used in hybrid logic. The intuitive meaning of $@_n p$ is that "p holds for agent n."

Let us introduce a formal definition. We have two disjoint infinite sets, **Prop** of propositional variables and **Nom** of nominals. A formula φ of the *Facebook logic* is defined as follows:

$$\varphi ::= p \mid n \mid \neg\varphi \mid \varphi \wedge \varphi \mid K\varphi \mid F\varphi \mid @_n\varphi,$$

where $p \in$ **Prop** and $n \in$ **Nom**. If needed, we can define the dual $\langle K \rangle$ and $\langle F \rangle$ of each modal operator as $\langle K \rangle\varphi := \neg K \neg\varphi$ and $\langle F \rangle\varphi := \neg F \neg\varphi$.

The semantics of Facebook logic is based on *epistemic social network models*. An epistemic social network model is a tuple $(W, A, (\sim_a)_{a \in A}, (\asymp_w)_{w \in W}, V)$, where:

- W is a set of epistemic alternatives.
- A is a set of agents.
- For each $a \in A$, $\sim_a$ is an equivalence relation on W.

- For each $w \in W$, $\asymp_w$ is an irreflexive and symmetric relation of friendship on A.
- V is a valuation function, which assigns a propositional variable p to a subset of $W \times A$ and a nominal n to a set $W \times \{a\}$ for some $a \in A$.

The reason for a relation $\asymp_w$ over A being irreflexive and symmetric can be understood when we assume it as a friendship; no one is a friend to oneself, and if a person is your friend, then you are a friend of them.

Then, the truth of formulas in Facebook logic is defined inductively. The Boolean cases are omitted since they are the same as those in classical modal logic. Also, the element $a \in A$ such that $V(n) = W \times \{a\}$ holds is abbreviated as n^V.

$$\begin{aligned}
\mathcal{M}, w, a \models p &\iff (w, a) \in V(p) \text{ where } p \in \mathbf{Prop},\\
\mathcal{M}, w, a \models n &\iff n^V = a, \text{ where } n \in \mathbf{Nom}\\
\mathcal{M}, w, a \models K\varphi &\iff \mathcal{M}, v, a \models \varphi \text{ for every } v \sim_a w,\\
\mathcal{M}, w, a \models F\varphi &\iff \mathcal{M}, w, b \models \varphi \text{ for every } b \asymp_w a,\\
\mathcal{M}, w, a \models @_n\varphi &\iff \mathcal{M}, w, n^V \models \varphi.
\end{aligned}$$

As mentioned in the Introduction, the truth of a propositional variable depends on both an epistemic alternative and an agent.

Example 2.5. The following formulas of Facebook logic can be translated into natural language as follows:

- Kp: An agent knows that they are p.
- KFp: An agent knows that all of their friends are p.
- FKp: Each of an agent's friends knows that they are p.
- $\langle F \rangle n$: An agent has a friend n.
- $@_n Kp$: An agent n knows that they are p.

For readers who would like to study it deeper, Seligman et al. [16] and its sequel, Seligman et al. [17], should be of help.

2.3. Logic of Hide and Seek Game

The logic of hide and seek game (LHS), as the name implies, is a logic for describing a hide and seek game. There are two players, a hider and a seeker, and a set of propositional variables $\mathbf{Prop}_H$ and $\mathbf{Prop}_S$ for each player to describe their state. Moreover, there is a special propositional variable I. This is a proposition to describe that the hider and seeker are in the same place, i.e., expressing "I find you!"

The main difference from Facebook logic is that we use the same structure (W, R, V) as in usual modal logic, which is appropriate considering that the hide and seek game is played by two players on the same board.

Here is a definition of a formula of LHS φ, where $p_H \in \mathbf{Prop}_H$ and $p_S \in \mathbf{Prop}_S$:

$$\varphi ::= p_H \mid p_S \mid I \mid \neg\varphi \mid \varphi \wedge \varphi \mid \Diamond_H \varphi \mid \Diamond_S \varphi.$$

The truth value of LHS formulas is defined inductively as follows (note that both x and y are elements of W):

$$\begin{aligned}
\mathcal{M}, x, y \models p_H &\iff x \in V(p_H) \text{ where } p_H \in \mathbf{Prop}_H \\
\mathcal{M}, x, y \models p_S &\iff y \in V(p_S) \text{ where } p_S \in \mathbf{Prop}_S \\
\mathcal{M}, x, y \models I &\iff x = y \\
\mathcal{M}, x, y \models \Diamond_H \varphi &\iff \text{there is some } x' \text{ such that } xRx' \text{ and } \mathcal{M}, x', y \models \varphi \\
\mathcal{M}, x, y \models \Diamond_S \varphi &\iff \text{there is some } y' \text{ such that } yRy' \text{ and } \mathcal{M}, x, y' \models \varphi.
\end{aligned}$$

Using this language, we can describe the hide and seek game. For example, $\Box_H \Diamond_S I$ means that no matter how the hider moves, the seeker has a one-step move to catch the hider. This expression shows the existence of a winning strategy for the seeker.

In addition to the already mentioned Li et al. [11], Li et al. [12] may also help readers who want to know more about LHS.

3. Agent-Knowledge Logic

Here, we introduce a new logic, called *agent-knowledge logic*. As you read in Section 1, this logic is a mixture of Facebook logic and LHS. We have two dimensions, which correspond to agents and their knowledge, respectively. This structure and the intention behind it are very similar to that of Facebook logic. On the other hand, the idea that we use both $\mathbf{Prop}_A$ and $\mathbf{Prop}_K$ is unique for LHS.

3.1. Agent-Knowledge Model

To construct the vocabulary, we require four sets of variables in total: two for propositional variables and two for nominals, each associated with agents and knowledge, respectively. Among these, $p_K \in \mathbf{Prop}_K$ can be viewed as a proposition that does not depend on an agent, such as "the Earth goes around the Sun." Furthermore, the two types of nominals, $a \in \mathbf{Nom}_A$ and $k \in \mathbf{Nom}_K$, can be interpreted as an agent name and a label for an epistemic alternative, respectively. Interpreting the elements of $\mathbf{Prop}_A$ is somewhat more difficult by comparison, but $p_A \in \mathbf{Prop}_A$ could be understood as a proposition representing some property of an agent, such as "an agent has a pollen allergy."

Definition 3.1. We have four disjoint sets $\mathbf{Prop}_A$, $\mathbf{Prop}_K$, $\mathbf{Nom}_A$, and $\mathbf{Nom}_K$. A formula φ of the *agent-knowledge logic* $\mathcal{L}_{\mathrm{AK}}$ is defined as follows:

$$\varphi ::= p_A \mid p_K \mid a \mid k \mid \neg\varphi \mid \varphi \wedge \varphi \mid \Box_A\varphi \mid \Box_K\varphi \mid @_a\varphi \mid @_k\varphi,$$

where $p_A \in \mathbf{Prop}_A$, $p_K \in \mathbf{Prop}_K$, $a \in \mathbf{Nom}_A$, and $k \in \mathbf{Nom}_K$.

As we mentioned in Section 2.2, nominals in $\mathbf{Nom}_A$ and $\mathbf{Nom}_K$ point to a specific agent and a specific epistemic alternative, respectively. As well as $\vee$ and $\rightarrow$, if we need, we can define $\Diamond_A$ and $\Diamond_K$ in the usual way.

Definition 3.2. An *agent-knowledge model (AK model)* $\mathcal{M}_{\mathrm{AK}}$ is a tuple $(W_A \times W_K, (R_y)_{y\in W_K}, (S_x)_{x\in W_A}, V_A, V_K)$ where:

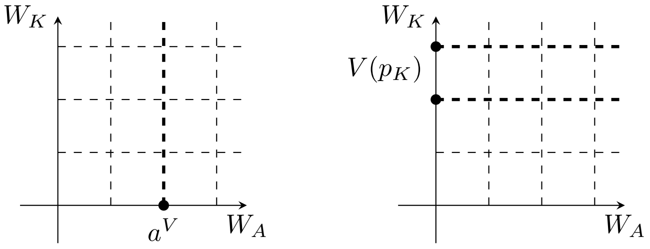

Figure 1: An agent-knowledge model.

- W_A, W_K are disjoint non-empty sets.
- For each $y \in W_K$, R_y is a binary relation on W_A.
- For each $x \in W_A$, S_x is a binary relation on W_K.
- $V_A : \mathbf{Prop}_A \cup \mathbf{Nom}_A \to \mathcal{P}(W_A)$, where if $a \in \mathbf{Nom}_A$, then $V_A(a) = \{x\}$ for some $x \in W_A$.
- $V_K : \mathbf{Prop}_K \cup \mathbf{Nom}_K \to \mathcal{P}(W_K)$, where if $k \in \mathbf{Nom}_K$, then $V_K(k) = \{y\}$ for some $y \in W_K$.

Note that the image of a nominal $\mathbf{Nom}_A$ by V_A is a singleton (the same fact holds for $\mathbf{Nom}_K$ and V_K). Owing to this definition, a nominal behaves as a *name* for each possible world.

We can illustrate an agent-knowledge model as if we write Cartesian coordinates in Figure 1. In this circumstance, a nominal is true in the worlds on the corresponding horizontal or vertical line. Likely, a propositional variable holds in the worlds on some set of parallel lines.

We write V to express $V_A \cup V_K$. For instance, $V(p_A) = V_A(p_A)$. Moreover, we abbreviate $x \in W_A$ such that $V_A(a) = \{x\}$ by a^V. We do the same for k^V.

Definition 3.3. Given a model $\mathcal{M}_{\mathrm{AK}}$, its points $(x, y) \in W_A \times W_K$, and a formula $\varphi \in \mathcal{L}_{\mathrm{AK}}$, the *satisfaction relation* $\mathcal{M}_{\mathrm{AK}}, (x, y) \models \varphi$ is defined inductively as follows:

$$\begin{array}{rcl}
\mathcal{M}_{\mathrm{AK}}, (x, y) \models p_A & \Longleftrightarrow & x \in V(p_A), \text{ where } p_A \in \mathbf{Prop}_A \\
\mathcal{M}_{\mathrm{AK}}, (x, y) \models p_K & \Longleftrightarrow & y \in V(p_K), \text{ where } p_K \in \mathbf{Prop}_K \\
\mathcal{M}_{\mathrm{AK}}, (x, y) \models a & \Longleftrightarrow & x = a^V, \text{ where } a \in \mathbf{Nom}_A \\
\mathcal{M}_{\mathrm{AK}}, (x, y) \models k & \Longleftrightarrow & y = k^V, \text{ where } k \in \mathbf{Nom}_K \\
\mathcal{M}_{\mathrm{AK}}, (x, y) \models \neg\varphi & \Longleftrightarrow & \text{Not } \mathcal{M}_{\mathrm{AK}}, (x, y) \models \varphi \ (\mathcal{M}_{\mathrm{AK}}, (x, y) \not\models \varphi) \\
\mathcal{M}_{\mathrm{AK}}, (x, y) \models \varphi \wedge \psi & \Longleftrightarrow & \mathcal{M}_{\mathrm{AK}}, (x, y) \models \varphi \text{ and } \mathcal{M}_{\mathrm{AK}}, (x, y) \models \psi \\
\mathcal{M}_{\mathrm{AK}}, (x, y) \models \Box_A\varphi & \Longleftrightarrow & \text{For all } x' \in W_A, xR_yx' \text{ implies} \\
 & & \mathcal{M}_{\mathrm{AK}}, (x', y) \models \varphi \\
\mathcal{M}_{\mathrm{AK}}, (x, y) \models \Box_K\varphi & \Longleftrightarrow & \text{For all } y' \in W_K, yS_xy' \text{ implies} \\
 & & \mathcal{M}_{\mathrm{AK}}, (x, y') \models \varphi \\
\mathcal{M}_{\mathrm{AK}}, (x, y) \models @_a\varphi & \Longleftrightarrow & \mathcal{M}_{\mathrm{AK}}, (a^V, y) \models \varphi \\
\mathcal{M}_{\mathrm{AK}}, (x, y) \models @_k\varphi & \Longleftrightarrow & \mathcal{M}_{\mathrm{AK}}, (x, k^V) \models \varphi.
\end{array}$$

The truth of each propositional variable is determined by either $x \in W_A$ or $y \in W_K$. Especially whether p_K is true or false is independent of the element of W_A, so p_K can be assumed as an agent-free proposition.

The usage of the satisfaction operator @ should also be mentioned. It refers to a specific agent or epistemic alternative while ignoring the current one. For example, the meaning of $@_a\varphi$ is "for an agent whose name is a, φ holds." The current element of W_A is no longer necessary information to determine the truth of that formula.

Definition 3.4. A formula φ is *valid* with respect to the class of $\mathcal{M}_{\mathrm{AK}}$ (denoted by $\models_{\mathrm{AK}} \varphi$) if $\mathcal{M}_{\mathrm{AK}}, (x, y) \models \varphi$ for every model $\mathcal{M}_{\mathrm{AK}}$ and its every pair (x, y).

Before finishing this section, it is necessary to mention the globality of the satisfaction operator for readers familiar with hybrid logic. In ordinary (one-dimensional, you might say) hybrid logic, a formula of the form $@_i\varphi$

has globality, that is, if $@_i\varphi$ is true in any possible world, then it is true in every possible world.

On the other hand, this is not the case with agent-knowledge logic. Suppose $\mathcal{M}_{\mathrm{AK}}, (x, y) \models @_a\varphi$ ($a \in \mathbf{Nom}_A$). Then, we have $\mathcal{M}_{\mathrm{AK}}, (a^V, y) \models \varphi$. However, it can be the case that there is some $z \in W_K$ such that $\mathcal{M}_{\mathrm{AK}}, (a^V, z) \not\models \varphi$. In this case, $@_a\varphi$ is not true in (x, z). Likewise, $@_k\varphi$ ($k \in \mathbf{Nom}_K$) is not a global expression.

However, this does not mean that the agent-knowledge logic has completely lost its global expression. If we use two satisfaction operators together and create a formula $@_a@_k\varphi$ ($a \in \mathbf{Nom}_A, k \in \mathbf{Nom}_K$), then it has globality. In fact, $\mathcal{M}_{\mathrm{AK}}, (x, y) \models @_a@_k\varphi$ is equivalent to $\mathcal{M}_{\mathrm{AK}}, (a^V, k^V) \models \varphi$, which shows that the truth of $@_a@_k\varphi$ does not depend at all on the current state.

3.2. Examples

As we do in Facebook logic, we can compound friendship and knowledge in agent-knowledge logic. We read $\Box_K\varphi$ as "An agent knows φ," and $\Box_A\varphi$ as "All of an agent's friends are φ." For example, we can write some sentences as follows:

- $\Box_A\Box_K p_K$: All of an agent's friends know p_K.
- $\Diamond_A\Box_K p_K$: Some of an agent's friends know p_K.
- $\Box_K\Diamond_A\Box_K p_K$: An agent knows that some of their friends know p_K.

Moreover, we can designate an individual by calling their name owing to nominals. Consider this sentence:

> An agent is Andy's friend, and if Andy knows that the Earth goes around the Sun, then one of the agent's friends knows the heliocentric theory.

This inference can be symbolized in the agent-knowledge logic as follows:

$$\Diamond_A a \wedge @_a\Box_K p_K \to \Diamond_A\Box_K p_K,$$

where p_K shows "the Earth goes around the Sun" and a shows "This is Andy."

The difference between agent-knowledge logic and Facebook logic becomes more pronounced when we assume that the binary relations over epistemic alternatives are equivalence relations. For example, in Facebook logic, the formula $@_n Kp \to p$ is not valid even if $\asymp_w$ is an equivalence relation. Define $\mathcal{M} = (W, A, (\sim_a)_{a \in A}, (\asymp_w)_{w \in W}, V)$ as follows:

$$\begin{aligned} W &= \{w, v\} \\ A &= \{a, b\} \\ \sim_a &= \sim_b = W \times W \\ \asymp_w &= \asymp_v = A \times A \\ V(p) &= \{(w, b), (v, b)\} \\ V(n) &= W \times \{b\}. \end{aligned}$$

Then, $\mathcal{M}, (w, a) \models @_n Kp$ holds but we have $\mathcal{M}, (w, a) \not\models p$. However, in agent-knowledge logic, the situation changes.

PROPOSITION 3.5. The formula $@_a \Box_K p_K \to p_K$ is valid with respect to the class of $\mathcal{M}_{\mathrm{AK}}$ where all of S_x are equivalence relations.

PROOF: Suppose that $\mathcal{M}_{\mathrm{AK}}, (x, y) \models @_a \Box_K p_K$. Then, we have $\mathcal{M}_{\mathrm{AK}}, (a^V, y) \models \Box_K p_K$. By the reflexivity of S_y, especially we have $\mathcal{M}_{\mathrm{AK}}, (a^V, y) \models p_K$. Since the truth value of p_K is determined only by an element of W_K, we have $\mathcal{M}_{\mathrm{AK}}, (x, y) \models p_K$. $\Box$

This fact may be better understood if we interpret those formulas in natural language. Even though Andy knows he has a pollen allergy, it does not mean so do all agents. However, if he knows that the Earth goes around the Sun, then it is true; the Earth really goes around the Sun.

In addition to the relationships between epistemic alternatives, we can also impose restrictions on the relationships between agents as needed. For example, in Facebook logic, the relationship between agents should be irreflexive and symmetric. Also, we have another way to capture relationships between agents, for example, to read xR_yx' as "in the situation y, the agent x can see the post of x'" on X[1]. Then, we can read $\Box_A \Box_K p_K$ as "all the people know p_K, *as far as I know*."

4. Embedding Epistemic Logic into Agent-Knowledge Logic

One of the aims of our new logic is to make it an alternative to Facebook logic. In fact, any sentence we can express in basic epistemic logic can be rewritten in this agent-knowledge logic. In this section, we show that we can embed epistemic logic into agent-knowledge logic.

To begin with, let us define how to translate a formula of epistemic logic.

Definition 4.1. We define a *translation* $T : \mathcal{L}_{\mathrm{EL}} \to \mathcal{L}_{\mathrm{AK}}$ as follows:

$$\begin{aligned} T : \mathbf{Prop} \ni p \mapsto p_K \in \mathbf{Prop}_K & \text{ is a bijection} \\ T : \mathbf{A} \ni i \mapsto a \in \mathbf{Nom}_A & \text{ is a bijection} \\ T(\neg\varphi) &= \neg T(\varphi) \\ T(\varphi \wedge \psi) &= T(\varphi) \wedge T(\psi) \\ T(K_i\varphi) &= @_{T(i)} \Box_K T(\varphi). \end{aligned}$$

Example 4.2. Here is one example of translation:

$$T(K_i(p \wedge K_j \neg q)) = @_{a_i} \Box_K (p_K \wedge @_{a_j} \Box_K \neg q_K).$$

We write a_i to abbreviate $T(i)$ ($i \in \mathbf{A}$).

In fact, the idea of rewriting $K_i\varphi$ as $@_{T(i)}\Box_K T(\varphi)$ was presented in Sano's review in 2011 [15]. This was written in Japanese to introduce the paper by Seligman et al. [16]. Unfortunately, this translation does not work for Facebook logic, but it does work when the target logic is agent-knowledge logic.

We aim to prove the following theorem.

---

[1] Most of the readers are familiar with the name once it had: Twitter.

Theorem 4.3. *For all* $\varphi \in \mathcal{L}_{EL}$*,*

$$\models_{EL} \varphi \iff \models_{AK} T(\varphi).$$

First, we construct an AK model from an EL model.

Definition 4.4. Given an EL model $\mathcal{M}_{\mathrm{EL}} = (W, (R_i)_{i\in\mathbf{A}}, V)$, the induced AK model $\mathcal{M}^{\alpha}_{\mathrm{AK}}$ is defined as follows:

$$\mathcal{M}^{\alpha}_{\mathrm{AK}} = (\mathbf{A} \times W, \emptyset, (R_i)_{i\in\mathbf{A}}, V^{\alpha}),$$

where:

- For any $p_A \in \mathbf{Prop}_A$, $V^{\alpha}(p_A) = \emptyset$.
- For any $p_K \in \mathbf{Prop}_K$, $V^{\alpha}(p_K) = V(T^{-1}(p_K))$.
- For any $a \in \mathbf{Nom}_A$, $V^{\alpha}(a) = \{T^{-1}(a)\}$.
- Take one $y_0 \in W$, and for any $k \in \mathbf{Nom}_K$, $V^{\alpha}(k) = \{y_0\}$.

Note that we do not care about the definitions of $(R_y)_{y\in W_K}, V^{\alpha}(p_A)$, and $V^{\alpha}(k)$. It is because the formula translated by T requires only $\mathbf{Prop}_K$, $\mathbf{Nom}_A$, Boolean operators, $\Box_K$, and $@_a$.

Lemma 4.5. *For any* $\varphi \in \mathcal{L}_{EL}$ *and for any* $i \in \mathbf{A}$*, we have:*

$$\mathcal{M}_{EL}, w \models \varphi \iff \mathcal{M}^{\alpha}_{AK}, (i, w) \models T(\varphi).$$

Proof: By induction on the complexity of φ.

$(\varphi = p)$ For all $i \in \mathbf{A}$,

$$\begin{aligned} \mathcal{M}_{\mathrm{EL}}, w \models p &\iff w \in V(p) \\ &\iff w \in V^{\alpha}(T(p)) \\ &\iff \mathcal{M}^{\alpha}_{\mathrm{AK}}, (i, w) \models T(p). \end{aligned}$$

$(\varphi = \neg\psi, \psi \wedge \chi)$ Straightforward.

$(\varphi = K_j\psi)$ First, we prove the left-to-right direction.

Suppose that $\mathcal{M}_{\mathrm{EL}}, w \models K_j\psi$. Then, for all v such that wR_jv, we have $\mathcal{M}_{\mathrm{EL}}, v \models \psi$. We divide the proof into two cases depending on whether such a world $v \in W$ exists.

(i) If $v \in W$ reachable by R_j from w exists, take arbitrary one. Then, we have $\mathcal{M}_{\mathrm{EL}}, v \models \varphi$. By the induction hypothesis, especially $\mathcal{M}^{\alpha}_{\mathrm{AK}}, (j, v) \models T(\varphi)$. Since we took v arbitrarily, it follows that $\mathcal{M}^{\alpha}_{\mathrm{AK}}, (j, w) \models \Box_K T(\varphi)$. By the definition of V^{α}, we finally get that $\mathcal{M}^{\alpha}_{\mathrm{AK}}, (i, w) \models @_{T(j)}\Box_K T(\varphi)$ for all $i \in A$.

(ii) If there is no $v \in W$ such that wR_jv, we straightforwardly get that $\mathcal{M}^{\alpha}_{\mathrm{AK}}, (j, w) \models \Box_K T(\varphi)$. In the same way as in the former case, we have $\mathcal{M}^{\alpha}_{\mathrm{AK}}, (i, w) \models @_{T(j)}\Box_K T(\varphi)$ for all $i \in \mathbf{A}$.

In both cases, we can reach the result $\mathcal{M}^{\alpha}_{\mathrm{AK}}, (i, w) \models @_{T(j)}\Box_K T(\varphi)$ for all $i \in \mathbf{A}$. Therefore, we have $\mathcal{M}^{\alpha}_{\mathrm{AK}}, (i, w) \models T(K_j\varphi)$.

Next, we prove the other direction. Take one $i \in A$ and suppose that $\mathcal{M}^{\alpha}_{\mathrm{AK}}, (i, w) \models T(K_j\psi)$. It means that for all v such that wR_jv, $\mathcal{M}^{\alpha}_{\mathrm{AK}}, (j, v) \models T(\psi)$ holds. Take one v such that wR_jv (if we cannot, then $\mathcal{M}_{\mathrm{EL}}, w \models K_j\psi$ is straightforward). By the induction hypothesis, we have $\mathcal{M}_{\mathrm{EL}}, v \models \psi$. Since we took v arbitrarily, it follows that $\mathcal{M}_{\mathrm{EL}}, w \models K_j\psi$. □

DEFINITION 4.6. Given an AK model $\mathcal{M}_{\mathrm{AK}} = (W_A \times W_K, (R_y)_{y \in W_K}, (S_x)_{x \in W_A}, V)$, the induced EL model $\mathcal{M}^{\beta}_{\mathrm{EL}}$ is defined as follows:
$\mathcal{M}^{\beta}_{\mathrm{EL}} = (W_K, (S^{\beta}_i)_{i \in \mathbf{A}}, V^{\beta})$, where:

- $\mathbf{A}$ is the set used in Definition 2.1.
- $yS^{\beta}_i z$ in $\mathcal{M}^{\beta}_{\mathrm{EL}}$ iff $yS_{T(i)^V} z$ in $\mathcal{M}_{\mathrm{AK}}$.
- $V^{\beta}(p) = V(T(p))$.

Let us consider a function $\beta : W_A \to \mathbf{A}$ such that $\beta(T(i)^V) = i$ for all $i \in \mathbf{A}$. It expresses the correspondence between an agent in W_A and an agent in $\mathbf{A}$. The illustration of this condition in Figure 2 may help your understanding.

$$\begin{tikzcd}
\mathbf{A} \ni i \arrow[r, mapsto, "T"] & a \in \mathbf{Nom}_1 \arrow[d, mapsto, "(\cdot)^V"] \\
\mathbf{A} \ni i \arrow[u, mapsto, dashed, "\mathrm{id}_{\mathbf{A}}"] & x \in W_A \arrow[l, mapsto, "\beta"']
\end{tikzcd}$$

Figure 2: The condition β satisfies ($\mathrm{id}_{\mathbf{A}}$ is the identity on $\mathbf{A}$).

Lemma 4.7. *For any $\varphi \in \mathcal{L}_{EL}$ and for any $x \in W_A$,*

$$\mathcal{M}_{AK}, (x, y) \models T(\varphi) \iff \mathcal{M}^{\beta}_{EL}, y \models \varphi.$$

Proof: By induction on the complexity of φ.

($\varphi = p$) For all $x \in W^A$,

$$\begin{aligned}
\mathcal{M}_{\mathrm{AK}}, (x, y) \models T(p) &\iff y \in V(T(p)) \\
&\iff y \in V^{\beta}(p) \\
&\iff \mathcal{M}^{\beta}_{\mathrm{EL}}, y \models p.
\end{aligned}$$

($\varphi = \neg\psi, \psi \wedge \chi$) Straightforward.

($\varphi = K_j\psi$) First, we prove the left-to-right direction.

Suppose that $\mathcal{M}_{\mathrm{AK}}, (x, y) \models T(K_j\psi)$. That is, we assume that $\mathcal{M}_{\mathrm{AK}}, (x, y) \models @_{T(j)}\Box_K T(\psi)$. Then, for all z such that $yS_{T(j)^V}z$, we have $\mathcal{M}_{\mathrm{AK}}, (T(j)^V, z) \models T(\psi)$. Bearing in mind the definition of S^{β}_i, it suffices to pick up one $z \in W_K$ such that $yS^{\beta}_j z$ (if we cannot, it is straightforward that $\mathcal{M}^{\beta}_{\mathrm{EL}}, y \models K_j\psi$ holds). By the assumption, we have $\mathcal{M}_{\mathrm{AK}}, (T(j)^V, z) \models T(\psi)$. By the induction hypothesis, $\mathcal{M}^{\beta}_{\mathrm{EL}}, z \models \psi$. Since we picked up z arbitrarily, we have $\mathcal{M}^{\beta}_{\mathrm{EL}}, y \models K_j\psi$.

Next, we prove the other direction. Suppose that $\mathcal{M}^{\beta}_{\mathrm{EL}}, y \models K_j\psi$. It means that for all $z \in W_K$ such that $yS^{\beta}_j z$, $\mathcal{M}^{\beta}_{\mathrm{EL}}, z \models \psi$ holds. Now, pick $z \in W_A$ such that $yS_{T(j)^V}z$ arbitrarily (if we cannot, we have $\mathcal{M}_{\mathrm{AK}}, (x, y) \models T(K_j\psi)$ for all $x \in W_A$), and we have $yS^{\beta}_j z$. Then, we have $\mathcal{M}_{\mathrm{AK}}, z \models \psi$. By the induction hypothesis, $\mathcal{M}_{\mathrm{AK}}, (T(j)^V, z) \models T(\psi)$. Since we pick up z arbitrarily, it follows that $\mathcal{M}_{\mathrm{AK}}, (x, y) \models @_{T(j)}\Box_K T(\psi)$ for any $x \in W_A$, which means $\mathcal{M}_{\mathrm{AK}}, (x, y) \models T(K_j\psi)$. $\Box$

Now, we are ready to prove the main theorem, Theorem 4.3. Here is the proof.

Proof: We prove it by showing the contraposition. To prove the left-to-right direction, suppose that we have some φ such that $\not\models_{\mathrm{AK}} T(\varphi)$. Then, there is a model $\mathcal{M}_{\mathrm{AK}}$ and its pair of points (x, y) such that $\mathcal{M}_{\mathrm{AK}}, (x, y) \models \neg T(\varphi)$, which means that $\mathcal{M}_{\mathrm{AK}}, (x, y) \models T(\neg\varphi)$. Then, by Lemma 4.7, we have $\mathcal{M}^{\beta}_{\mathrm{EL}}, y \models \neg\varphi$, which leads us to the conclusion that $\not\models_{\mathrm{EL}} \varphi$. The case of the other direction can be done by using Lemma 4.5. $\Box$

We usually treat binary relations of EL models as equivalence relations. Moreover, once we want to deal with beliefs by means of a modal operator, we impose yet another condition on accessibility relations. The following corollary shows how embedding can reflect these restrictions.

Proposition 4.8. We have the following properties:

(i) For every $i \in \mathbf{A}$, if R_i in $\mathcal{M}_{\mathrm{EL}}$ is reflexive (or serial, symmetric, transitive, Euclidean), then so is R_i in $\mathcal{M}^{\alpha}_{\mathrm{AK}}$.

(ii) For every $x \in W_A$, if S_x in $\mathcal{M}_{\mathrm{AK}}$ is reflexive (or serial, symmetric, transitive, Euclidean), then so is S^{β}_i in $\mathcal{M}^{\beta}_{\mathrm{EL}}$.

Proof: The former is obvious, and the latter is straightforward from the definition of S^{β}_i. $\Box$

5. Proof System

In this section, we introduce a tableau calculus as a proof system.

In constructing a tableau calculus for agent-knowledge logic, we have made significant use of that for hybrid logic. The primary reference is the work of Bolander and Blackburn [3]. We also refer to Nishimura [13], who studies tableau calculi for some two-dimensional hybrid logics.

For simplicity, this section deals only with the negation normal form (NNF, in short) of formulas. Moreover, if we write $\neg\varphi$, we assume it as its NNF. For the satisfaction operators, a formula $\neg @_a \varphi$ is equivalent to $@_a \neg\varphi$. That is, for any model and its possible world (x, y), a formula φ, and a nominal $a \in \mathbf{Nom}_A$, we have

$$\mathcal{M}_{\mathrm{AK}}, (x, y) \models @_a \neg\varphi \iff \mathcal{M}_{\mathrm{AK}}, (x, y) \models \neg @_a \varphi.$$

The same equivalence holds for the case of $k \in \mathbf{Nom}_K$. Transformations to the NNF involving Boolean and modal operators can be done in the usual way.

5.1. Tableau Calculus

Here we provide a tableau calculus of agent-knowledge logic, denoted by $\mathbf{T}_{\mathrm{AK}}$.

Definition 5.1. A *tableau* is a well-founded tree constructed in the following way:

- Start with a formula of the form $@_a @_k \varphi$ (called the *root formula*), where φ is a formula of agent-knowledge logic and $a \in \mathbf{Nom}_A, k \in \mathbf{Nom}_K$ does not occur in φ.
- For each branch, extend it by applying rules (see Definition 5.3) to all nodes as often as possible. However, we can no longer add any formula in a branch if at least one of the following conditions is satisfied:
  - (i) Every new formula generated by applying any rule already exists in the branch.
  - (i) The branch is closed (see Definition 5.2).

Here, a *branch* means a maximal path of a tableau. If a formula φ occurs in a branch Θ, we write $\varphi \in \Theta$.

DEFINITION 5.2. A branch of a tableau Θ is *closed* if one of the following conditions holds:

(ii) There are $a \in \mathbf{Nom}_A$, $k, l \in \mathbf{Nom}_K$, and $s \in \mathbf{Prop}_A \cup \mathbf{Nom}_A$ such that $@_a@_k s, @_a@_l\neg s \in \Theta$.

(ii) There are $a, b \in \mathbf{Nom}_A$, $k \in \mathbf{Nom}_K$, and $t \in \mathbf{Prop}_K \cup \mathbf{Nom}_K$ such that $@_a@_k t, @_b@_k\neg t \in \Theta$.

We say that Θ is *open* if it is not closed. A tableau is called *closed* if all branches in the tableau are closed.

DEFINITION 5.3. We provide the rules of $\mathbf{T}_{\mathrm{AK}}$ in Figure 3.

In these rules, the formulas above the line show the formulas that have already occurred in the branch, and the formulas below the line show the formulas that will be added to the branch. The vertical line in the $[\vee]$ means that the branch splits to the left and right.

DEFINITION 5.4 (provability). Given a formula φ, we say that φ is *provable* in $\mathbf{T}_{\mathrm{AK}}$ if there is a closed tableau whose root formula is $@_a@_k\neg\varphi$, where $a \in \mathbf{Nom}_A$ and $k \in \mathbf{Nom}_K$ do not occur in φ.

5.2. Termination

A tableau calculus has *the termination property* if, for any tableau constructed in the system, all branches have a finite length. We first prove that the tableau calculus $\mathbf{T}_{\mathrm{AK}}$ introduced above has the termination property. This proof is based on the termination proof in [3]; however, it is more complicated because we are now dealing with two-dimensional circumstances.

DEFINITION 5.5. Let $@_a@_k\varphi, @_b@_l\psi$ $(a, b \in \mathbf{Nom}_A,\ k, l \in \mathbf{Nom}_K)$ be formulas. We say that $@_a@_k\varphi$ is a *prefixed subformula* of $@_b@_l\psi$ if φ is a subformula of ψ.

$$\frac{@_a @_k \neg b}{@_b @_k b}\ [\neg_A]^{*1} \qquad \frac{@_a @_k \neg l}{@_a @_l l}\ [\neg_K]^{*1}$$

$$\frac{@_a @_k (\varphi \wedge \psi)}{\begin{array}{c} @_a @_k \varphi \\ @_a @_k \psi \end{array}}\ [\wedge] \qquad \frac{@_a @_k (\varphi \vee \psi)}{@_a @_k \varphi \mid @_a @_k \psi}\ [\vee]$$

$$\frac{@_a @_k \Diamond_A \varphi}{\begin{array}{c} @_a @_k \Diamond_A b \\ @_b @_k \varphi \end{array}}\ [\Diamond_A]^{*1,*2,*3} \qquad \frac{@_a @_k \Diamond_K \varphi}{\begin{array}{c} @_a @_k \Diamond_K l \\ @_a @_l \varphi \end{array}}\ [\Diamond_K]^{*1,*2,*4}$$

$$\frac{\begin{array}{c} @_a @_k \Box_A \varphi \\ @_a @_k \Diamond_A b \end{array}}{@_b @_k \varphi}\ [\Box_A]^{*5} \qquad \frac{\begin{array}{c} @_a @_k \Box_K \varphi \\ @_a @_k \Diamond_K l \end{array}}{@_a @_l \varphi}\ [\Box_K]^{*5}$$

$$\frac{@_a @_k @_b \varphi}{@_b @_k \varphi}\ [@_A] \qquad \frac{@_a @_k @_l \varphi}{@_a @_l \varphi}\ [@_K] \qquad \frac{\begin{array}{c} @_a @_k \varphi \\ @_a @_k b \end{array}}{@_b @_k \varphi}\ [Id_A]^{*2} \qquad \frac{\begin{array}{c} @_a @_k \varphi \\ @_a @_k l \end{array}}{@_a @_l \varphi}\ [Id_K]^{*2}$$

*1: This rule can be applied only one time per formula.
*2: The formula above the line is not an accessibility formula. Here, an *accessibility formula* is the formula of the form $@_a @_k \Diamond_A b$ ($@_a @_k \Diamond_K l$) generated by $[\Diamond_A]$ ($[\Diamond_K]$), where b (l) is a new nominal.
*3: $b \in \mathbf{Nom}_A$ does not occur in the branch.
*4: $l \in \mathbf{Nom}_K$ does not occur in the branch.
*5: The second formula above the line is an accessibility formula.

Figure 3: The rules of $\mathbf{T}_{\mathrm{AK}}$

Lemma 5.6. *For any formula* $@_a@_k\varphi$ *occurring in a branch* Θ *of a tableau, at least one of the following conditions holds:*

- $@_a@_k\varphi$ *is a prefixed subformula of the root formula.*
- $@_a@_k\varphi$ *is an accessibility formula.*

Proof: By induction on the length of Θ. □

Definition 5.7. Let Θ be a branch. For every pair (a, k) $(a \in \mathbf{Nom}_A, k \in \mathbf{Nom}_K)$ of nominals occurring in Θ, the set $T^{\Theta}((a, k))$ is defined as follows:

$$T^{\Theta}((a,k))=\{\varphi \,|\, @_a@_k\varphi\in\Theta \text{ is a prefixed subformula of the root formula}\}.$$

Since the number of prefixed subformulas is finite for any formula $@_a@_k\varphi$, the set $T^{\Theta}((a, k))$ is finite for any pair (a, k) of nominals.

Definition 5.8. Let Θ be a branch of a tableau, and let $a, b \in \mathbf{Nom}_A$ and $k, l \in \mathbf{Nom}_K$ be nominals occurring in Θ. A pair (b, l) of nominals is *generated* by (a, k) in Θ (denoted by $(a, k) \prec_\Theta (b, l)$) if one of the following conditions holds:

(i) $k = l$ and b is introduced by applying $[\Diamond_A]$ to $@_a@_k\Diamond_A\varphi$.

(ii) $a = b$ and l is introduced by applying $[\Diamond_K]$ to $@_a@_k\Diamond_K\varphi$.

Observe that the following equivalences hold: $(a, k) \prec_\Theta (b, l)$ if and only if one of the following conditions holds:

(i) $k = l$ and there is an accessibility formula $@_a@_k\Diamond_A b \in \Theta$.

(ii) $a = b$ and there is an accessibility formula $@_a@_k\Diamond_K l \in \Theta$.

Thus far, the discussion has proceeded in nearly the same manner as in previous studies such as [3]. However, the situation changes from the following lemma.

Lemma 5.9. *Given a branch* Θ *of a tableau, we define a structure* $G^{\Theta} = (N^{\Theta}, \prec_{\Theta})$ *where:*

- $N^{\Theta} = \{(a,k) \mid \textit{there is a } \varphi \textit{ such that } @_a @_k \varphi \in \Theta\}$,
- $\prec_{\Theta}$ *is the relation defined in Definition 5.8.*

Then G^{Θ} *is a disjoint union of well-founded and finitely branching trees.*

Proof: Properties that G is well-founded and finitely branching are proved in a similar way to the proof of [3, Lemma 4.2]. Then the rest of the proof is to show that G is a disjoint union of trees.

Suppose there are pairs (a,k) and (a',k') of nominals such that both $(a,k) \prec_{\Theta} (b,l)$ and $(a',k') \prec_{\Theta} (b,l)$. This proof is split into two cases.

(a) Suppose that $k = k' = l$ and there are accessibility formulas $@_a @_k \Diamond_A b$, $@'_a @_k \Diamond_A b \in \Theta$. However, it never occurs since we have to introduce a new nominal whenever we apply $[\Diamond_A]$ or $[\Diamond_K]$ to a branch.

(b) Suppose that $k = l$, $a' = b$ and there are accessibility formulas $@_a @_k \Diamond_A b, @_{a'} @_{k'} \Diamond_K l \in \Theta$. If $@_a @_k \Diamond_A b$ appears first, then the occurrence $@_{a'} @_{k'} \Diamond_K l$ in Θ contradicts the restriction of $[\Diamond_K]$ since $l = k$ already exists in Θ before adding $@_{a'} @_{k'} \Diamond_K l$. $\square$

Note that, unlike [3, Lemma 6.4], the lemma shown above does not claim that there are only finitely many trees. In the tableau for one-dimensional hybrid logic, there can be only finitely many nominals that serve as roots in G^{Θ}; they are restricted to those occurring in the root formula. Consequently, if N^{Θ} is infinite, we immediately obtain an infinite sequence of $\prec_{\Theta}$ by König's Lemma. In the present case, however, if at least one member of the pair (a,k) is a nominal from the root formula, it has the potential to become a root of G^{Θ}.

While this issue must be addressed, we first define a function m_{Θ} for a pair of nominals (a,k) that returns the maximum length of formulas labeled by it, and then show that this value decreases along the transitions of $\prec_{\Theta}$.

Definition 5.10. Let Θ be a branch of a tableau. For any pair (a, k) of nominals occurring in Θ. We define a function $m_\Theta : \mathbf{Nom}_A \times \mathbf{Nom}_K \to \mathbb{N}$ as follows:

$$m_\Theta((a, k)) = \max\{|\varphi| \mid @_a @_k \varphi \in \Theta\}.$$

Then, we want to show that $(a, k) \prec_\Theta (b, l)$ implies $m_\Theta((a, k)) > m_\Theta((b, l))$. However, directly proving it does not work well. Thus, we show an extended result as a lemma.

Lemma 5.11. *Let Θ be a branch of a tableau.*

(i) If $(a, k) \prec_\Theta (b, k)$, then for all $l \in \mathbf{Nom}_K$, we have $m_\Theta((a, k)) > m_\Theta((b, l))$.

(ii) If $(a, k) \prec_\Theta (a, l)$, then for all $b \in \mathbf{Nom}_A$, we have $m_\Theta((a, k)) > m_\Theta((b, l))$.

Proof: We only show (i). The other part of the lemma can be shown similarly.

Assume $(a, k) \prec_\Theta (b, k)$. Take a nominal $l \in \mathbf{Nom}_K$ and a formula φ such that $m_\Theta((b, l)) \geq m_\Theta((b, l'))$ for all $l' \in \mathbf{Nom}_K$ and $|\varphi| = m_\Theta((b, l))$. We show that $m_\Theta((a, k)) > |\varphi|$ or contradiction by dividing it into cases depending on which rule $@_b @_l \varphi$ was introduced by.

(a) Suppose that $@_b @_l \varphi$ is introduced by $[\neg_A]$. Straightforwardly, we have $@_b @_l b \in \Theta$. This is obviously not an accessibility formula, so by Lemma 5.6, $@_b @_l b$ is a prefixed subformula of the root formula of Θ. Then, b is in the root formula, which contradicts that b is a generated nominal.

(b) Suppose that $@_b @_l \varphi$ is introduced by $[\neg_K]$. Then, we have $\varphi = l$ and there is another nominal $l \in \mathbf{Nom}_K$ such that $@_b @_{l'} \neg l \in \Theta$. However, it contradicts the maximality of l and φ.

(c) If $@_b @_l \varphi$ is introduced by $[\wedge]$, then there is another formula ψ such that $@_b @_l (\varphi \wedge \psi) \in \Theta$. However, it contradicts the maximality of φ. We can prove this in a similar way in the case of $[\vee]$.

(d) Suppose that $@_b@_l\varphi$ is introduced by $[\Diamond_A]$. This means that there is another nominal $b' \in \mathbf{Nom}_A$ (it may be different from a) such that $@_b@_l\varphi$ is introduced from $@_{b'}@_l\Diamond_A\varphi$. Then we have $(b', l) \prec_\Theta (b, l)$, which means b is introduced by some formula $@_{b'}@_l\Diamond_A\varphi$. However, since $(a, k) \prec_\Theta (b, k)$ and $G = (N^\Theta, \prec_\Theta)$ defined in Lemma 5.9 is a disjoint union of trees, we have $(b', l) = (a, k)$. Therefore, it follows that

$$m_\Theta((a, k)) = m_\Theta((b', l)) \geq |\Diamond_A\varphi| > |\varphi|.$$

We can prove this in a similar way in the case of $[\Box_A]$.

(e) Suppose that $@_b@_l\varphi$ is introduced by $[\Diamond_K]$. Then we have that there is another nominal $l' \in \mathbf{Nom}_K$ such that $@_b@_{l'}\Diamond_K\varphi \in \Theta$. However, it contradicts the maximality of l and φ. We can prove this in a similar way in the case of $[\Box_K]$.

(f) If $@_b@_l\varphi$ is introduced by $[@_A]$, then there is another nominal $b' \in \mathbf{Nom}_A$ such that $@_{b'}@_l@_b\varphi \in \Theta$. By Lemma 5.6, $@_b\varphi$ is a prefixed subformula of the root formula of Θ. Then, b is in the root formula, which contradicts that b is a generated nominal.

(g) If $@_b@_l\varphi$ is introduced by $[@_K]$, then there is another nominal $l' \in \mathbf{Nom}_K$ such that $@_b@_{l'}@_l\varphi \in \Theta$. However, it contradicts the maximality of l and φ.

(h) Suppose that $@_b@_l\varphi$ is introduced by $[Id_A]$. Then, we have another nominal $b' \in \mathbf{Nom}_A$ such that $@_{b'}@_lb \in \Theta$. Thus, by Lemma 5.6, b is in the root formula, which contradicts that b is a generated nominal.

(i) Suppose that $@_b@_l\varphi$ is introduced by $[Id_K]$. Then, we have another nominal $l' \in \mathbf{Nom}_K$ such that $@_b@_{l'}\varphi \in \Theta$. In this case, take l' instead and check which rule derives $@_b@_{l'}\varphi$. Note that this case cannot be applied infinitely. In this case, we also have $@_b@_{l'}l \in \Theta$, so l is in the root formula. However, the root formula contains only a finite number of nominals. □

Now, we address the remaining issue. As previously mentioned, G^Θ can be an infinite disjoint union of trees. Therefore, we consider constructing a single large tree by "grafting" the root of each tree onto a branch of another tree.

Definition 5.12. We define a relation $\lhd_\Theta$ over $\mathbf{Nom}_A \times \mathbf{Nom}_K$. For each root (b, l) of G^Θ, insert $(a, k) \lhd_\Theta (b, l)$, where (a, k) is the earliest occurrence of a label in Θ among those satisfying the following conditions:

(i) $@_b @_l \psi$ is derived from $@_a @_k \varphi$ by either $[\neg_M], [@_M]$, or $[Id_M]$ $(M \in \{A, K\})$ and

(ii) $m_\Theta((a, k)) \geq m_\Theta((b, l))$.

The following lemma guarantees that the definition of $\lhd_\Theta$ works well.

Lemma 5.13. *Let (b, l) be a root of G^Θ. If the root formula of Θ does not have the form $@_b @_l \varphi$, i.e., (b, l) is not a label of the root formula of Θ, then there is at least one pair (b', l') satisfying the conditions (i) and (ii) in Definition 5.12.*

Proof: For any root (b, l) of G^Θ, take a formula ψ such that $|\psi| = m_\Theta((b, l))$. If $@_b @_l \psi$ is introduced by $[\wedge]$ or $[\vee]$, then there is a formula $@_b @_l (\psi \wedge \chi)$ or $@_b @_l (\psi \vee \chi)$ in Θ, which contradicts the maximality of ψ. Moreover, if $@_b @_l \psi$ is derived by $[\Diamond_A]$, there exists a nominal $b' \in \mathbf{Nom}_A$ such that $@_{b'} @_l \Diamond_A \psi, @_{b'} @_l \Diamond_A b \in \Theta$. Then, we have $(b', l) \prec_\Theta (b, l)$, which contradicts that (b, l) is a root of G^Θ (the same applies to the cases for $[\Diamond_K]$, $[\Box_A]$, and $[\Box_K]$).

Therefore, $@_b @_l \varphi$ must be derived by either $[\neg_M]$, $[@_M]$, or $[Id_M]$ $(M \in \{A, K\})$. If $@_b @_l \psi$ is derived by $[\neg_A]$, we have $\psi = b$ and there is another nominal $b' \in \mathbf{Nom}_A$ such that $@_{b'} @_l \neg b \in \Theta$. Then, we have $m_\Theta((b', l)) \geq |\neg b| \geq |b| = |\psi|$. If $@_b @_l \psi$ is derived by $[@_A]$, there is another nominal $b' \in \mathbf{Nom}_A$ such that $@_{b'} @_l @_b \psi \in \Theta$. Then, we have $m_\Theta((b', l)) \geq |@_b \psi| \geq |\psi|$. In the case for $[Id_A]$, there is $b' \in \mathbf{Nom}_A$ such that $@_{b'} @_l \psi \in \Theta$. Then, we have $m_\Theta((b', l)) \geq |\psi|$. We leave the remainder of the proof of the cases for $[\neg_K]$, $[@_K]$, and $[Id_K]$. □

Recall that all the formulas in a tableau, except for accessibility formulas, are quasi-subformulas in the root formula. Observing the rules $[\neg_M]$, $[@_M]$, and $[Id_M]$ $(M \in \{A, K\})$, we have the following restriction:

- (R) If $(a, k) \lhd_\Theta (a, l)$ holds, l occurs in a root formula. Moreover, if $(a, k) \lhd_\Theta (b, k)$ holds, b occurs in a root formula.

Since the number of nominals occurring in the root formula is finite, we cannot have the following infinite sequence

$$(a_0, k_0) \lhd_\Theta (a_1, k_1) \lhd_\Theta (a_2, k_2) \lhd_\Theta \cdots ,$$

where $(a_i, k_i) \neq (a_j, k_j)$ $(i \neq j)$ for all $i, j \in \mathbb{N}$.

Owing to $\lhd_\Theta$, we can construct a single tree by grafting trees in G^Θ.

LEMMA 5.14. *Given a branch* Θ *of a tableau, we define a structure* $\tilde{G}^\Theta = (N^\Theta, (\prec_\Theta \cup \lhd_\Theta))$. *Then* $\tilde{G}^\Theta$ *is a finitely branching tree.*

PROOF: When a new pair of nominals emerges in N^Θ, we apply either $[\Diamond_M]$, $[\neg_M]$, $[@_M]$, or $[Id_M]$ $(M \in \{A, K\})$ to Θ. Thus, any node of $\tilde{G}^\Theta$, except for the pair of nominals that serves as a label for the root formula, has the antecedent by either $\prec_\Theta$ or $\lhd_\Theta$. Since all formulas in the tableau are derived from the root formula, any node of $\tilde{G}^\Theta$ is connected to the label of the root formula.

We have checked that G^Θ is finitely-branching. Moreover, from the fact (R) and the fact that a root formula of Θ has only a finite number of nominals, $\lhd_\Theta$ is also finitely-branching.

It is not the case that both $(a, k) \prec_\Theta (b, l)$ and $(a', k') \lhd_\Theta (b, l)$ hold—otherwise, (b, l) would be and yet not be a root of G^Θ. Also, by the definition of $\lhd_\Theta$, any antecedent of $(b, l) \in N^\Theta$ by $\lhd_\Theta$ is unique. Together with the fact that G^Θ is a disjoint union of trees, we show that $\tilde{G}^\Theta$ is a tree. □

We do not show that $\tilde{G}$ is well-founded. In fact, the following infinite ascending branch may occur:

$$(a_0, k_0) \rhd_\Theta (a_1, k_1) \rhd_\Theta (a_2, k_2) \rhd_\Theta \dots .$$

However, considering (R) and the finiteness of nominals occurring in the root formula, only a finite pair of nominals occurs in that sequence. Then, it may occur only as a finite loop. Therefore, this infinite ascending branch does not cause any problem in applying König's lemma. Now, we are ready to prove the main theorem.

THEOREM 5.15. *The tableau calculus* $\mathbf{T}_{AK}$ *has the termination property.*

PROOF: By *reductio ad absurdum.*

Suppose that there is a branch Θ of a tableau that is infinite. If only a finite number of nominals occur in Θ, then a finite number of formulas occur in Θ by Lemma 5.6, which contradicts the infinity of Θ. Thus, Θ contains infinitely many nominals, so N^{Θ} is infinite. Since $\tilde{G}^{\Theta}$ is a finitely branching tree, we find an infinite sequence

$$(a_0, k_0) \lhd_{\Theta}^{n_0} (a_1, k_1) \prec_{\Theta}^{+} (a_2, k_2) \lhd_{\Theta}^{n_1} (a_3, k_3) \prec_{\Theta}^{+} \cdots,$$

by König's Lemma (note that we cannot make an infinite ascending chain by $\lhd_{\Theta}$). Applying Lemma 5.11 and Definition 5.12, we have an infinite descent

$$m_{\Theta}((a_0, k_0)) \geq m_{\Theta}((a_1, k_1)) > m_{\Theta}((a_2, k_2)) \geq m_{\Theta}((a_3, k_3)) > \cdots,$$

which contradicts the definition of m_{Θ}. □

5.3. Completeness

The soundness of $\mathbf{T}_{\mathrm{AK}}$ can be proved in a similar way to the method introduced in [13]. Then, we proceed to prove the completeness of $\mathbf{T}_{\mathrm{AK}}$.

In preparation, we define some terminology. We say a branch Θ *saturated* if every new formula generated by applying some rules already exists in Θ. Moreover, $s \in \mathbf{Nom}_A \cup \mathbf{Nom}_K$ is a *right nominal* in a branch Θ if there are some $a \in \mathbf{Nom}_A$ and $k \in \mathbf{Nom}_K$ such that $@_a @_k s \in \Theta$. From Lemma 5.6, it is straightforward that all the right nominals in Θ occur in the root formula of Θ.

DEFINITION 5.16. Given a branch Θ of a tableau, we define binary relations $\sim_{\Theta}^{A} \subset \mathbf{Nom}_A \times \mathbf{Nom}_A$ and $\sim_{\Theta}^{K} \subset \mathbf{Nom}_K \times \mathbf{Nom}_K$ on a set of right nominals in Θ as follows.

- $a \sim_{\Theta}^{A} b$ if there is a nominal $k \in \mathbf{Nom}_K$ such that $@_a @_k b \in \Theta$.
- $k \sim_{\Theta}^{K} l$ if there is a nominal $a \in \mathbf{Nom}_A$ such that $@_a @_k l \in \Theta$.

We can show that if Θ is saturated, then both $\sim^A_\Theta$ and $\sim^K_\Theta$ are equivalence relations (see [13, Lemma 5.7]). Then, we define equivalence classes $[a]^A_\Theta$ and $[k]^K_\Theta$ of a right nominal by the relation $\sim^A_\Theta$ and $\sim^K_\Theta$, respectively. Moreover, if we arrange the elements in $\mathbf{Nom}_A$ and $\mathbf{Nom}_K$ in the order of occurrence in the branch, respectively, we can fix the minimal element $\min(A)$ for any subset A in $\mathbf{Nom}_A$ or $\mathbf{Nom}_K$. They enable us to take a representative of nominals.

Definition 5.17. Let Θ be a saturated branch and $s \in \mathbf{Nom}_A \cup \mathbf{Nom}_K$ a nominal occurring in Θ. The *urfather* of s on Θ (denoted by $u_\Theta(s)$) is defined as follows:

$$u_\Theta(s) = \begin{cases} \min([b]^A_\Theta) & \text{if } s \in \mathbf{Nom}_A \text{ and } @_s @_k b \in \Theta \\ \min([l]^K_\Theta) & \text{if } s \in \mathbf{Nom}_K \text{ and } @_a @_s l \in \Theta \\ s & \text{otherwise.} \end{cases}$$

Given an $s \in \mathbf{Nom}_A \cup \mathbf{Nom}_K$, $u_\Theta(s)$ is well-defined (see [13, Proposition 5.10]). Moreover, the following lemma holds like [13, Lemma 5.11].

Lemma 5.18. *Let Θ be a saturated branch. Then, the following properties hold:*

1. *If $a \in \mathbf{Nom}_A$ is a right nominal in Θ, then there is some $k \in \mathbf{Nom}_K$ such that $@_{u_\Theta(a)} @_k a \in \Theta$. Likely, if $k \in \mathbf{Nom}_K$ is a right nominal in Θ, then there is some $a \in \mathbf{Nom}_A$ such that $@_a @_{u_\Theta(k)} k \in \Theta$.*
2. *If $@_a @_k b \in \Theta$ ($b \in \mathbf{Nom}_A$), then $u_\Theta(a) = u_\Theta(b)$. Likely, if $@_a @_k l \in \Theta$ ($l \in \mathbf{Nom}_K$), then $u_\Theta(k) = u_\Theta(l)$.*
3. *For every prefixed subformula $@_a @_k \varphi \in \Theta$ of the root formula of Θ, we have $@_{u_\Theta(a)} @_{u_\Theta(k)} \varphi \in \Theta$.*

Definition 5.19. Given an open saturated branch Θ, a model $\mathcal{M}^\Theta_{\mathrm{AK}} = (W^\Theta_A \times W^\Theta_K, (R^\Theta_y)_{y \in W^\Theta_K}, (S^\Theta_x)_{x \in W^\Theta_A}, V^\Theta)$ generated from Θ is defined as follows:

$$\begin{aligned}
W_A^\Theta &= \{u_\Theta(a) \mid a \in \mathbf{Nom}_A \text{ occurs in } \Theta\}\\
W_K^\Theta &= \{u_\Theta(k) \mid k \in \mathbf{Nom}_K \text{ occurs in } \Theta\}\\
R_{u_\Theta(k)}^\Theta &= \{(u_\Theta(a), u_\Theta(b)) \mid \text{accessibility formula } @_a @_k \Diamond_A b \in \Theta\}\\
S_{u_\Theta(a)}^\Theta &= \{(u_\Theta(k), u_\Theta(l)) \mid \text{accessibility formula } @_a @_k \Diamond_K l \in \Theta\}\\
V^\Theta(p_A) &= \{u_\Theta(a) \mid \text{there is } k \in \mathbf{Nom}_K \text{ such that } @_a @_k p_A \in \Theta\},\\
&\quad\ \text{where } p_A \in \mathbf{Prop}_A\\
V^\Theta(p_K) &= \{u_\Theta(k) \mid \text{there is } a \in \mathbf{Nom}_A \text{ such that } @_a @_k p_K \in \Theta\},\\
&\quad\ \text{where } p_K \in \mathbf{Prop}_K\\
V^\Theta(a) &= \{u_\Theta(a)\},\ \text{where } a \in \mathbf{Nom}_A\\
V^\Theta(k) &= \{u_\Theta(k)\},\ \text{where } k \in \mathbf{Nom}_K.
\end{aligned}$$

LEMMA 5.20. *Let Θ be an open saturated branch and let $@_a @_k \varphi$ be a prefixed subformula of the root formula of Θ. Then, we have:*

$$\textit{if } @_a @_k \varphi \in \Theta, \textit{ then } \mathcal{M}_{AK}^\Theta, (u_\Theta(a), u_\Theta(k)) \models \varphi.$$

PROOF: By induction on the complexity of φ.

[$\varphi = p_A$] Suppose that $@_a @_k p_A \in \Theta$. Then, by definition, we have $u_\Theta(a) \in V^\Theta(p_A)$. Therefore, $\mathcal{M}_{\mathrm{AK}}^\Theta, (u_\Theta(a), u_\Theta(k)) \models p_A$ holds. We can do the same in the case $\varphi = p_K$.

[$\varphi = \neg p_A$] Suppose that $@_a @_k \neg p_A \in \Theta$. Since Θ is open, $@_a @_l p_A \notin \Theta$ is valid for all $l \in \mathbf{Nom}_K$. This means that $u_\Theta(a) \notin V^\Theta(p_A)$. Therefore, we have $\mathcal{M}_{\mathrm{AK}}^\Theta, (u_\Theta(a), u_\Theta(k)) \models \neg p_A$. We can do the same in the case $\varphi = \neg p_K$.

[$\varphi = b$] Suppose that $@_a @_k b \in \Theta$ ($b \in \mathbf{Nom}_A$). By Lemma 5.18, we have $u_\Theta(a) = u_\Theta(b)$. Thus, we have $\mathcal{M}_{\mathrm{AK}}^\Theta, (u_\Theta(a), u_\Theta(k)) \models b$ by the definition of $V^\Theta(b)$. We can do the same in the case $\varphi = l$ ($l \in \mathbf{Nom}_K$).

[$\varphi = \neg b$] Suppose that $@_a @_k \neg b \in \Theta$ ($b \in \mathbf{Nom}_A$). Since Θ is saturated, we also have $@_b @_k b \in \Theta$. Then by Lemma 5.18, it follows that $@_{u_\Theta(a)} @_{u_\Theta(k)} \neg b, @_{u_\Theta(b)} @_{u_\Theta(k)} b \in \Theta$. However, since Θ is open, $u_\Theta(a) \neq u_\Theta(b)$, which means $\mathcal{M}^{\Theta}_{\mathrm{AK}}, (u_\Theta(a), u_\Theta(k)) \models \neg b$. We can do the same in the case $\varphi = \neg l$ ($l \in \mathbf{Nom}_K$).

[$\varphi = \psi \wedge \chi, \psi \vee \chi$] Straightforward.

[$\varphi = \Diamond_A \psi$] If $@_a @_k \Diamond_A \psi \in \Theta$, then there is a nominal $b \in \mathbf{Nom}_A$ such that $@_a @_k \Diamond_A b, @_b @_k \psi \in \Theta$. It follows that $u_\Theta(a) R^{\Theta}_{u_\Theta(k)} u_\Theta(b)$ from the former formula. Moreover, by the latter formula and the induction hypothesis, $\mathcal{M}^{\Theta}_{\mathrm{AK}}, (u_\Theta(b), u_\Theta(k)) \models \psi$ holds. Therefore, we have $\mathcal{M}^{\Theta}_{\mathrm{AK}}, (u_\Theta(a), u_\Theta(k)) \models \Diamond_A \psi$. We can do the same in the case $\varphi = \Diamond_K \psi$.

[$\varphi = \Box_A \psi$] Suppose that $@_a @_k \Box_A \psi \in \Theta$. Now, assume that there is a nominal $b \in \mathbf{Nom}_A$ such that there is an accessibility formula $@_a @_k \Diamond_A b \in \Theta$ (If not, then there is no state reachable from $u_\Theta(a)$. Thus, $\mathcal{M}^{\Theta}_{\mathrm{AK}}, (u_\Theta(a), u_\Theta(k)) \models \Box_A \psi$ is straightforward). Then, from the definition of R^{Θ}, we have $u_\Theta(a) R^{\Theta}_{u_\Theta(k)} u_\Theta(b)$. Moreover, since Θ is saturated, we obtain $@_b @_k \psi \in \Theta$. By the induction hypothesis, $\mathcal{M}^{\Theta}_{\mathrm{AK}}, (u_\Theta(b), u_\Theta(k)) \models \psi$ holds. Since we pick up an arbitrary b, we have $\mathcal{M}^{\Theta}_{\mathrm{AK}}, (u_\Theta(a), u_\Theta(k)) \models \Box_A \psi$. We can do the same in the case $\varphi = \Box_K \psi$.

[$\varphi = @_b \psi, @_k \psi$] Straightforward. □

This lemma is called *model existence lemma*. Note that by combining it with the termination property of $\mathbf{T}_{\mathrm{AK}}$, we can show the finite model property of agent-knowledge logic as well as the completeness.

Theorem 5.21. *The tableau calculus* $\mathbf{T}_{AK}$ *is complete for the class of all AK models.*

Proof: We show the contraposition.

Suppose that φ is not provable in $\mathbf{T}_{\mathrm{AK}}$. Then, we can find an open and saturated branch Θ with the root formula $@_a @_k \neg\varphi$, where $a \in \mathbf{Nom}_A$ and $k \in \mathbf{Nom}_K$ do not occur in φ. Then, by Lemma 5.20, we have

$\mathcal{M}^{\Theta}_{\mathrm{AK}}, (u_{\Theta}(a), u_{\Theta}(k)) \models \neg\varphi$. It means that there is an AK model and its possible world that falsifies φ. $\square$

The termination property and completeness of the tableau calculus tell us about the decidability of logic. If φ is provable, then it is provable in finite time. By contrast, if φ is unprovable, we can make a finite counterexample model. From them, the following corollary holds.

Corollary 5.22. The agent-knowledge logic is decidable.

6. Future Work and Perspective

6.1. Adding More Operators

One of the proposed future work to make agent-knowledge logic more fruitful is to add new operators.

For example, let us imitate some operators of epistemic logic. Given a group $G \subseteq \mathbf{A}$ of agents, the *everybody knows operator* E_G is defined as follows:

$$\mathcal{M}_{\mathrm{EL}}, w \models E_G\varphi \iff \mathcal{M}, w \models K_i\varphi \text{ for all } i \in G.$$

Intuitively, this formula says that everyone in the group G knows φ. Recalling that a formula $K_i\varphi$ is translated into $@_{T(i)}\square_K T(\varphi)$, we can define the everybody knows operator in agent-knowledge logic as follows, where G is a subset of $\mathbf{Nom}_A$:

$$\mathcal{M}_{\mathrm{AK}}, (x, y) \models E_G\varphi \iff \mathcal{M}, (x, y) \models @_a\square_K\varphi \text{ for all } a \in G.$$

This definition works well even if two nominals $a, b \in \mathbf{Nom}_A$ point to the same agent, but only one of a or b is in G. If they point to the same agent, then the two formulas $@_a\varphi$ and $@_b\varphi$ are equivalent for any world in any model.

Based on these definitions, we may mimic other operators used in epistemic logic, such as the operator for common knowledge C_G and the operator for distributed knowledge D_G. Can we analyze problems that have been considered in epistemic logic, such as the muddy children puzzle, using agent-knowledge logic? Research in this direction may be able to reflect various results in epistemic logic in agent-knowledge logic as well.

Additionally, there is another direction to research agent-knowledge logic, to introduce a universal operator used in hybrid logic, which may enable us to symbolize more expressions in natural language. The definition of the universal operators A_A and E_A are as follows:

$$\begin{aligned}
\mathcal{M}_{\mathrm{AK}},(x,y) \models \mathsf{A}_A\varphi &\iff \mathcal{M}_{\mathrm{AK}},(z,y) \models \varphi \text{ for all } z \in W_A,\\
\mathcal{M}_{\mathrm{AK}},(x,y) \models \mathsf{E}_A\varphi &\iff \text{there is some } z \in W_A\\
&\qquad \text{such that } \mathcal{M}_{\mathrm{AK}},(z,y) \models \varphi.
\end{aligned}$$

Owing to these operators, we can write some expressions as follows:

- $\mathsf{E}_A\Box_K p_K$: Someone knows p_K.
- $\Box_K\mathsf{A}_A\Box_K p_K$: An agent knows that all the people know p_K.
- $\mathsf{E}_A\Box_A\Box_K p_K$: There is a person all of whose friends know p_K.

6.2. Seeking More Usage

Research in agent-knowledge logic is not merely about adding new operators. In contrast, some of the languages proposed in this paper lack effective application. For example, agent names in $\mathbf{Nom}_A$ and agent-independent propositions in $\mathbf{Prop}_K$ play essential roles to embed epistemic logic into agent-knowledge logic. However, this paper does not go so far as to present practical applications for the sets $\mathbf{Prop}_A$ and $\mathbf{Nom}_K$, although it provides interpretations for the elements of them.

It might be an interesting direction to assign interpretations to the two sets other than agents and their epistemic alternative. For example, as often seen in modal logics applied to computer science, elements of W_K can be interpreted as states of a computer. In that case, W_A can be seen as a set of named computers. Under this interpretation, agent-knowledge logic might be able to describe a system where computers mutually monitor each other's behavior.

6.3. Hilbert-Style Axiomatization

In this paper, we have given the tableau calculus for agent-knowledge logic as a proof system. We can give another proof system, for example, Hilbert-style axiomatization.

Fortunately, there is already abundant prior research in related fields, such as the aforementioned Sano's work [14] on two-dimensional hybrid logic (see also [1], which focused on Facebook logic). For LHS, recent research by Chen and Li [5] gives the axiomatization.

6.4. Complexity

In this paper, we have shown the decidability of the agent-knowledge logic using tableau calculus. But what about its computational complexity? As is already known, the satisfiability problem for epistemic logic is PSPACE-complete [8]. If we want to use agent-knowledge logic as an alternative to epistemic logic, we expect it to be PSPACE-complete.

The analysis of computational complexity for a fusion in modal logic may provide a clue to solving this problem. An explanation for a fusion is in [7, p. 111]:

> Let L_1 and L_2 be two multimodal logics formulated in languages $\mathcal{L}_1$ and $\mathcal{L}_2$, both containing the language $\mathcal{L}$ of classical propositional logic, but having disjoint sets of modal operators. Denote by $\mathcal{L}_1 \otimes \mathcal{L}_2$ the union of $\mathcal{L}_1$ and $\mathcal{L}_2$. Then the *fusion* $L_1 \otimes L_2$ of L_1 and L_2 is the smallest multimodal logic L in the language $\mathcal{L}_1 \otimes \mathcal{L}_2$ containing $L_1 \cup L_2$.

From the results of Halpern and Moses [8], we can obtain that the satisfiability problem for $\mathbf{K} \otimes \mathbf{K}$ is PSPACE-complete. Since agent-knowledge logic is based on $\mathbf{K} \otimes \mathbf{K}$, we may be able to answer the question with reference to this proof.

Acknowledgements. I would like to thank Prof. Ryo Kashima and Leonardo Pacheco for their invaluable advice in writing this paper. I am also thankful for invaluable comments from Prof. Valentin Goranko, Michał Zawidzki, and Paweł Płaczek, who met in NCL'24.

References

[1] P. Balbiani, S. F. González, *Indexed Frames and Hybrid Logics*, [in:] N. Olivetti, R. Verbrugge, S. Negri, G. Sandu (eds.), **Advances in Modal Logic 13**, College Publications, Chichester (2020), pp. 56–72.

[2] P. Blackburn, B. ten Cate, *Pure Extensions, Proof Rules, and Hybrid Axiomatics*, **Studia Logica**, vol. 84(2) (2006), pp. 277–322, DOI: https://doi.org/https://doi.org/10.1007/s11225-006-9009-6.

[3] T. Bolander, P. Blackburn, *Termination for Hybrid Tableaus*, **Journal of Logic and Computation**, vol. 17(3) (2007), pp. 517–554, DOI: https://doi.org/https://doi.org/10.1093/logcom/exm014.

[4] T. Braüner, **Hybrid Logic and its Proof-Theory**, vol. 37, Springer Science & Business Media (2011), DOI: https://doi.org/https://doi.org/10.1007/978-94-007-0002-4.

[5] Q. Chen, D. Li, *Logic of the Hide and Seek Game: Characterization, Axiomatization, Decidability*, [in:] N. Gierasimczuk, F. R. Velázquez-Quesada (eds.), **Dynamic Logic. New Trends and Applications**, Springer Nature Switzerland (2024), pp. 20–34, DOI: https://doi.org/https://doi.org/10.1007/978-3-031-51777-8_2.

[6] R. Fagin, J. Y. Halpern, Y. Moses, M. Y. Vardi, **Reasoning about knowledge**, MIT press (1995), DOI: https://doi.org/https://doi.org/10.7551/mitpress/5803.001.0001.

[7] D. M. Gabbay, **Many-Dimensional Modal Logics: Theory and Applications**, Elsevier North Holland (2003).

[8] J. Y. Halpern, Y. Moses, *A Guide to Completeness and Complexity for Modal Logics of Knowledge and Belief*, **Artificial Intelligence**, vol. 54(3) (1992), pp. 319–379, DOI: https://doi.org/https://doi.org/10.1016/0004-3702(92)90049-4.

[9] J. Hintikka, **Knowledge and Belief: An Introduction to the Logic of the Two Notions**, 2nd ed., Cornell University Press (1962).

[10] A. Indrzejczak, *Modal Hybrid Logic*, **Logic and Logical Philosophy**, vol. 16(2-3) (2007), pp. 147–257, DOI: https://doi.org/https://doi.org/10.12775/LLP.2007.006.

[11] D. Li, S. Ghosh, F. Liu, Y. Tu, *On The Subtle Nature of a Simple Logic of The Hide and Seek Game*, [in:] **Logic, Language, Information, and Computation: 27th International Workshop, WoLLIC 2021, Virtual Event, October 5–8, 2021, Proceedings 27**, Springer (2021), pp. 201–218, DOI: https://doi.org/https://doi.org/10.1007/978-3-030-88853-4_13.

[12] D. Li, S. Ghosh, F. Liu, Y. Tu, *A Simple Logic of The Hide and Seek Game*, **Studia Logica**, vol. 111(5) (2023), pp. 821–853, DOI: https://doi.org/https://doi.org/10.1007/s11225-023-10039-4.

[13] Y. Nishimura, *Completeness of Tableau Calculi for Two-dimensional Hybrid Logics*, **Journal of Logic and Computation**, vol. 35(3) (2025), p. exae018, DOI: https://doi.org/https://doi.org/10.1093/logcom/exae018.

[14] K. Sano, *Axiomatizing Hybrid Products: How Can We Reason Many-Dimensionally in Hybrid Logic?*, **Journal of Applied Logic**, vol. 8(4) (2010), pp. 459–474, DOI: https://doi.org/https://doi.org/10.1016/j.jal.2010.08.006.

[15] K. Sano, *Paper Review: Logic in The Community*, **Journals of The Japanese Society for Artificial Intelligence**, vol. 26(6) (2011), pp. 703–707, DOI: https://doi.org/https://doi.org/10.11517/jjsai.26.6_703, (written in Japanese).

[16] J. Seligman, F. Liu, P. Girard, *Logic in the Community*, [in:] **Proceedings of the 4th Indian Conference on Logic and Its Applications**, Springer Berlin Heidelberg (2011), pp. 178–188, DOI: https://doi.org/https://doi.org/10.1007/978-3-642-18026-2_15.

[17] J. Seligman, F. Liu, P. Girard, *Facebook and the Epistemic Logic of Friendship*, [in:] **Proceedings of the 14th Conference on Theoretical Aspects of Rationality and Knowledge** (2013), pp. 229–238.

[18] J. van Benthem, *Epistemic Logic and Epistemology: The State of their Affairs*, **Philosophical Studies**, vol. 128 (2006), pp. 249–76, DOI: https://doi.org/https://doi.org/10.1007/s11098-005-4052-0.

Yuki Nishimura
Institute of Science Tokyo
School of Computing
152-8550, Ookayama 2-12-1
Meguro-ku, Tokyo, Japan
e-mail: nishimura.y.as@m.titech.ac.jp

Funding information: The research of the author was supported by JST SPRING, Grant Number JPMJSP2106 and JPMJSP2180.

Conflict of interests: None.

Ethical considerations: The Author assures of no violations of publication ethics and takes full responsibility for the content of the publication.

Declaration regarding the use of GAI tools: Not used.

Submission Guidelines

Manuscripts Papers submitted to the *BSL* should be formatted using the `BSLstyle` LaTeX class with the manuscript option loaded, which can be downloaded at https://czasopisma.uni.lodz.pl/bulletin/libraryFiles/downloadPublic/603. All prospective authors should read the "Instructions for authors" file included in the style files folder and follow the guidelines included there. Abstract and keywords are compulsory parts of each submission as they will be used in the *BSL* online search tools. Mind that an abstract should contain no references and the list of keywords should consist of at least 3 items. It is also recommended that each author having an ORCID number provides it in the `.tex` source file. Authors who are unable to comply with these requirements should contact the Editorial Office in advance.

Paper Length There is no fixed limit imposed on the length of submitted papers, however one can expect that for shorter papers, up to 18 pages long, the Editorial Board will be able to reduce the time needed for the reviewing process.

Footnotes should be avoided as much as possible, however it is not disallowed to use them if necessary.

Bibliography should be formatted using BibTeX and the `BSLbibstyle` bibliography style (to be found in the style files folder). It is essential that to each bibliography item a plain DOI number (i.e., not a full link) is attached whenever applicable. If a submitted paper is accepted for publication, the author(s) should provide the bibliography file in the `.bib` format among other source files. For more details on bibliography processing the authors are referred to the "Instructions for authors". Authors unfamiliar with BibTeX are advised to familiarize themselves with the short tutorial or video tutorial on managing bibliographies with BibTeX.

Affiliation and mailing addresses of all the authors should be included in the `\Affiliation` and `\AuthorEmail` fields, respectively, in the source `.tex` file.

Submission When the manuscript is ready, it should be submitted through our editorial platform, using the the «Make a Submission» button. If the paper is meant to be included in a special issue, the appropriate section name should be selected before submitting it. If the paper is regular, the authors can indicate the editor they would like to supervise the editorial process or leave this decision to the Editorial Office by leaving the "Comments for the Editor" section blank. For the duration of the whole editorial process of the manuscript it must not be submitted for review to any other venue.

Publication Once the manuscript has been accepted for publication and the galley proof has been revised by the authors, the article is given a DOI number and published in the *Early View* section, where articles accepted for publication and awaiting assignment to an issue are made available to the public. The authors will be notified when their article is assigned to an issue.

Copyright permission It is the authors' responsibility to obtain the necessary copyright permission from the copyright owner(s) of the submitted paper or extended abstract to publish the submitted material in the *BSL*.

ISSN 0138-0680