First order logic properly displayed

Samuel Balco, Giuseppe Greco, Alexander Kurz, Andrew Moshier, Alessandra Palmigiano*, and Apostolos Tzimoulis

1Vrije Universiteit Amsterdam
2University of Leicester
3University of Johannesburg
4Chapman University

Abstract

We introduce a proper display calculus for first-order logic, of which we prove soundness, completeness, conservativity, subformula property and cut elimination via a Belnap-style metatheorem. All inference rules are closed under uniform substitution and are without side conditions.

Keywords: First-order classical logic, proper display calculi, properly displayable logics, multi-type calculi, multi-type algebras.

Math. Subject Class. 03B10, 03B35, 03B45, 03B47, 03F03, 03F05, 03F07, 03G05, 03G10, 03G15, 06A06, 06A11, 06D10, 06D50, 06E15.

1 Introduction

In the proof-theoretic literature, the treatment of quantifiers in first-order logic typically follows the lines of the original Gentzen’s sequent calculi, and correspondingly, the introduction rules for first-order quantifiers are mostly additive:

\[
\begin{align*}
\forall_L & \quad A[t/x], \Gamma \vdash \Delta \\
& \quad \forall x A, \Gamma \vdash \Delta \\
& \quad \forall_R \\

\exists_L & \quad A[y/x], \Gamma \vdash \Delta \\
& \quad \exists x A, \Gamma \vdash \Delta \\
& \quad \exists_R
\end{align*}
\]

This is reflected also in the early display calculi literature, where quantifiers are not paired with structural connectives, up to Wansing [37, 33, 3], who introduces a display calculus for fragments of first-order logic based on the idea that

*The research of the second and fifth author has been funded in part by the NWO grant KIVI.2019.001.

1We use the standard notation from [29] where \(A[t/x]\) signifies that the free occurrences of the variable \(x\) in the formula \(A\) are substituted with the term \(t\) (see Section 2 for the formal definition of substitutions). We also assume the usual eigenvariable conditions, namely \(y\) must not occur free anywhere in the conclusion of the rules \(\forall_R\) and \(\exists_L\). Notice that the notation \(A[y/x]\) in the premises of \(\forall_R\) and \(\exists_L\) is more restrictive and signifies that the formula has been derived for an arbitrary variable \(y\) (not just a term \(t\)).
quantifiers can be treated as modal operators in modal logic, and correspondingly defines introduction rules for quantifiers which involve their structural counterparts.

The key observation motivating this approach is that the existential quantifier behaves similarly to a modal diamond operator, and dually, the universal quantifier behaves similarly to a modal box operator. What underlies these similarities (observed and exploited in [22, 31, 28, 35, 37, 36]), and semantically supports the requirement that the resulting calculus enjoys the display property (see e.g. [2, 36, 12] for the formal definition of this notion), is the order-theoretic notion of adjunction: indeed, the set-theoretic interpretations of the existential and universal quantifiers are the left and right adjoint of the inverse projection map respectively, and more generally, in categorical semantics, the left and right adjoint of the pullbacks along projections [23, 10, Chapter 15].

The display calculus of [37] contains rules with side conditions on the free and bound variables of formulas, similar to the ones of the original Gentzen sequent calculus. This implies that the introduction rules for first-order quantifiers are not closed under uniform substitution, i.e. the display calculus introduced in [37, 36] is not proper [36, Section 4.1].

In this paper we overcome this difficulty and introduce a proper display calculus for first-order logic. The main idea is that, as was the case of other logical frameworks (cf. e.g. [7, 9, 16]), a suitable multi-type presentation makes it possible to encode the side conditions into analytic (structural) rules involving different types. Wansing’s insight that quantifiers can be treated proof-theoretically as modal operators naturally embeds into the multi-type approach by simply regarding (∀x) and (∃x) as modal operators bridging different types (i.e. as heterogeneous operators). Following Lawvere [23, 25, 24] and Halmos [18], this requires to consider as many types as there are finite sets of free variables; that is, two first-order formulas have the same type if and only if they have exactly the same free variables.

Following these ideas, we introduce a proper multi-type display calculus for classical first-order logic, and show its soundness, completeness, conservativity, cut elimination and subformula property. The paper is organized as follows. In Section 2, we gather preliminary notions, definitions and notation for first-order logic. In Section 3, we recast the models of first-order logic in a framework amenable to support the semantics of the multi-type calculus for first-order logic, especially regarding the adjunction properties of quantification and substitution. From this semantic framework, we extract the defining conditions of the algebraic multi-type semantics for the calculus. In Section 4, we introduce the multi-type language of first-order logic. In Section 5, we introduce the display calculus for first-order classical logic. In Section 6, we prove its soundness, completeness, conservativity, cut elimination and subformula property. In Section 7, we summarize the results of this article and collect further research directions.

2 Preliminaries on first-order logic

In this section we collect definitions and basic facts about first-order logic and introduce the notation that will be used throughout the paper.
Language. Let $\text{Var} = \{v_1, \ldots, v_n, \ldots\}$ be a countable set of variables. Throughout the paper we let $\mathcal{P}_n(\text{Var})$ denote the set of finite subsets of $\text{Var}$. A first-order logic $\mathcal{L}$ over $\text{Var}$ consists of a set of relation symbols $(R_i)_{i \in I}$ each of finite arity $n_i$; a set of function symbols $(f_j)_{j \in J}$ each of finite arity $n_j$; and a set of constant symbols $(c_k)_{k \in K}$ (0-ary functions). The language of first-order logic is built up from terms defined recursively as follows:

$$\text{Trm} \ni t ::= v_m \mid c_k \mid f_j(t_1, \ldots, t)$$

The formulas of first-order logic are defined recursively as follows:

$$\mathcal{L} \ni A ::= R_i(\bar{t}) \mid t_1 = t_2 \mid \bot \mid A \land A \mid A \lor A \mid A \to A \mid \forall x A \mid \exists x A$$

where $\bar{t}$ is a sequence of terms of arity $n_1$ and negation $\neg A$ is defined as usual as $A \to \bot$.

For any term $t$ and formula $A$ we let $\text{FV}(t) \in \mathcal{P}_n(\text{Var})$ and $\text{FV}(A) \in \mathcal{P}_n(\text{Var})$ denote the sets of free variables of $t$ and $A$ respectively, recursively defined as follows:

- $\text{FV}(v_m) = \{v_m\}$
- $\text{FV}(c_k) = \emptyset$
- $\text{FV}(f_j(t_1, \ldots, t_n)) = \bigcup_{1 \leq i \leq n_j} \text{FV}(t_i)$
- $\text{FV}(\top) = \emptyset$
- $\text{FV}(\bot) = \emptyset$
- $\text{FV}(t_1 = t_2) = \text{FV}(t_1) \cup \text{FV}(t_2)$
- $\text{FV}(R_i(t_1, \ldots, t_n)) = \bigcup_{1 \leq i \leq n_i} \text{FV}(t_i)$
- $\text{FV}(A \land B) = \text{FV}(A) \cup \text{FV}(B)$
- $\text{FV}(A \lor B) = \text{FV}(A) \cup \text{FV}(B)$
- $\text{FV}(A \to B) = \text{FV}(A) \cup \text{FV}(B)$
- $\text{FV}(\forall x A) = \text{FV}(A) \setminus \{x\}$
- $\text{FV}(\exists x A) = \text{FV}(A) \setminus \{x\}$.

In what follows, we will often identify sequences of variables and terms with the set containing the elements of the sequence. Hence, given a sequence of terms $\bar{t}$ we define $\text{FV}(\bar{t})$ as the union of the sets of free variables of the elements of the sequence. Finally, if $\bar{t}$ is a sequence of terms and $s$ is a term, we let $s, \bar{t}$ denote the sequence obtained by prefixing $s$ to $\bar{t}$. We will also abuse notation and write e.g. $\bar{x} \supseteq \text{FV}(s)$ to signify that every free variable in $s$ occurs in the sequence $\bar{x}$. In what follows we will conflate notation and use $\mathcal{L}$ to denote signature, formulas and set of theorems.

Substitution. In our treatment of first-order logic we assume that substitution happens simultaneously for all free variables of a term or a formula. If $\bar{x} \supseteq \text{FV}(s)$ and $\bar{y} \supseteq \text{FV}(A)$ and for each $v_m \in \bar{x}$ we let $t_{v_m}$ denote the corresponding term which will be substituted for $v_m$, then $t_{\bar{x}}$ denotes a sequence of terms over $\mathcal{L}((\bar{x}/\bar{y}))$ denotes the simultaneous substitution of each $v_m$ with $t_{v_m}$ and we define $s((\bar{x}/\bar{y}))$ and $A((\bar{x}/\bar{y}))$ recursively as follows:

That is, $t_{\bar{x}} \in \text{Trm}^\mathcal{L}$ and $t_{\bar{x}}(v_m) = t_{v_m}$.
\[
\begin{align*}
v_n(t_\tau/\tau) & = t_v_n \\
c_k(t_\tau/\tau) & = c_k \\
f(s_1, \ldots, s_n)(t_\tau/\tau) & = f(s_1, t_\tau/\tau, \ldots, s_n, t_\tau/\tau) \\
\top(t_\tau/\tau) & = \top \\
\bot(t_\tau/\tau) & = \bot \\
(s_1 = s_2)(t_\tau/\tau) & = s_1(t_\tau/\tau) = s_2(t_\tau/\tau) \\
R(s_1, \ldots, s_n)(t_\tau/\tau) & = R(s_1, t_\tau/\tau, \ldots, s_n, t_\tau/\tau) \\
(A \land B)(t_\tau/\tau) & = A(t_\tau/\tau) \land B(t_\tau/\tau) \\
(A \lor B)(t_\tau/\tau) & = A(t_\tau/\tau) \lor B(t_\tau/\tau) \\
(A \rightarrow B)(t_\tau/\tau) & = A(t_\tau/\tau) \rightarrow B(t_\tau/\tau) \\
(\forall y)A(t_\tau/\tau) & = \forall z(A(t_\tau/\tau)), \ z \not\in \text{FV}(t_\tau) \cup \text{FV}(\forall y)A \\
(\exists y)A(t_\tau/\tau) & = \exists z(A(t_\tau/\tau)), \ z \not\in \text{FV}(t_\tau) \cup \text{FV}(\exists y)A
\end{align*}
\]

Models. The models of a first-order logic \(\mathcal{L}\) are tuples
\[
M = (D, (R_t^D)_{t \in T}, (f_j^D)_{j \in J}, (c_k^D)_{k \in K})
\]
where \(D\) is a non-empty set (the domain of the model) and \(R_t^D, f_j^D, c_k^D\) are concrete \(n_t\)-ary relations over \(D\), \(n_j\)-ary functions on \(D\) and elements of \(D\) respectively interpreting the symbols of the language in the model \(M\). The interpretation of a formula in \(M\) is facilitated by variable interpretation maps \(\nu : \text{Var} \rightarrow D\) and is given recursively as follows:

\[
\begin{align*}
(v_m)_D & = \nu(v_m) \\
(f_j(t_1, \ldots, t_n))^D & = (f_j^D(t_1^D, \ldots, t_n^D)) \\
M, \nu \models \top & \iff \text{Always} \\
M, \nu \models \bot & \iff \text{Never} \\
M, \nu \models R_t(t_\tau) & \iff R_t^D(x_1^D) \\
M, \nu \models t_1 = t_2 & \iff t_1^D = t_2^D \\
M, \nu \models A \land B & \iff M, \nu \models A \text{ and } M, \nu \models B \\
M, \nu \models A \lor B & \iff M, \nu \models A \text{ or } M, \nu \models B \\
M, \nu \models A \rightarrow B & \iff M, \nu \models A \text{ implies } M, \nu \models B \\
M, \nu \models \forall y A & \iff M, \nu' = A \text{ for all } \nu' \text{ such that } \nu'(x) = \nu(x) \text{ for all } x \neq y \\
M, \nu \models \exists y A & \iff M, \nu' = A \text{ for some } \nu' \text{ such that } \nu'(x) = \nu(x) \text{ for all } x \neq y
\end{align*}
\]

Axiomatic system. The following deductive system [5], denoted with \(\vdash_{\text{FO}}\), is sound and complete w.r.t. the models mentioned above:

1. Propositional tautologies;
2. \(\forall x (B \rightarrow C) \rightarrow (\forall x B \rightarrow \forall x C)\);
3. \(B \rightarrow \forall x B\), where \(x \notin \text{FV}(B)\);
4. \(\forall x B \rightarrow B(t/x)\);
5. Equality axioms:
   \begin{align*}
   (a) & \quad t = t; \\
   (b) & \quad s = t \rightarrow (A(s/x) \rightarrow A(t/x))
   \end{align*}
6. All universal closures of instances of the above;
7. Modus ponens.
3 Semantic analysis

In the present section we use the same notational conventions we will adopt in the calculus D.FO and D.FO*. The letter symbolizing a function could occur in round, angular or square brackets. Angular (alluding to diamonds) for left adjoints, Round for functions that are simultaneously both left and right adjoints, and square (alluding to boxes) for right adjoints denote functions here and not function symbols.

In order to develop a multi-type environment in which the insights discussed in the introduction can be carried out, we need as many types as there are finite sets of variables. Therefore, types $F$ correspond to elements of $\mathcal{P}_\omega(\text{Var})$. On the semantic side, each such type interprets formulas $A$ such that $\text{FV}(A) = F$, and hence, being closed under all propositional connectives, each type is naturally endowed with the structure of a Boolean algebra. Thus, given a first-order logic $\mathcal{L}$ over a countable set of variables $\text{Var}$, and letting $\text{Var}_x := \text{Var} \setminus \{x\}$ for every $x \in \text{Var}$, the semantic environment that supports the multi-type presentation of $\mathcal{L}$ is based on structures $\mathbb{H} = (A, Q, S)$, where

- $A = \{A_F \mid F \in \mathcal{P}_\omega(\text{Var})\}$;
- $Q = \{(x)_T, [x]_F, \langle x \rangle_F \mid x \in F \in \mathcal{P}_\omega(\text{Var}) \text{ and } T \in \mathcal{P}_\omega(\text{Var}_x)\}$;
- $S = \{(t)_F, [t]_F, \langle t \rangle_F \mid F \in \mathcal{P}_\omega(\text{Var}) \text{ and } t_F \in \text{Trm}^F \}$ (see Footnote 2),

where every $A_F$ is a Boolean algebra, and the following adjunctions hold for every finite set of free variables $F \in \mathcal{P}_\omega(\text{Var})$, every $x \in F$, and every $T \in \mathcal{P}_\omega(\text{Var}_x)$:

\[
\begin{align*}
(x)_{T \cup \{x\}} & \quad \vdash \quad A_T \cup \{x\} \\
(x)_{T \cup \{x\}} & \quad \vdash \quad A_T \\
(x)_{T \cup \{x\}} & \quad \vdash \quad (x)_{T \cup \{x\}} \\
\end{align*}
\]

Notice that in $(x)_T, [x]_F, \langle x \rangle_F$ the subscripts denote the domain of the functions. In practice we will always omit the subscript since it will be obvious from the context.

The adjunctions illustrated above justify the soundness of the following rules (see Section 5 for the definition of the language and the multi-type display calculi D.FO and D.FO*):

\[
\begin{align*}
& (\hat{x}) Y \vdash Y \\
& X \vdash (\hat{x}) Y \\
& (\hat{x}) Y \vdash X \\
& (\check{x}) X \vdash (\hat{x}) X \\
& (\check{x}) X \vdash X \\
& (\check{x}) Y \vdash Y \\
& (\check{x}) Y \vdash X \\
\end{align*}
\]

where $(\hat{x})$ (resp. $(\hat{x})$) is the structural connective corresponding to existential (resp. universal) quantification, and $(\check{x})$ is the structural connective corresponding to the inverse projection map:

\[
\begin{align*}
& (\check{t}) X \vdash X \\
& X \vdash (\check{t}) Y \\
& (\check{t}) Y \vdash Y \\
& (\check{t}) Y \vdash X \\
\end{align*}
\]
where \((\triangleright_{\tau})\) is the structural connective corresponding to the uniform substitution of \(F\) with \(t_F\), and \((t_F|\) (resp. \((t_F)\)) is its right (resp. left) adjoint.

In order to identify the relevant properties which we need to impose on the multi-type environment outlined above, in what follows, we establish a systematic connection between the standard \(\mathcal{L}\)-models and a subclass of the structures described above.

Let \(M\) be a model of some fixed but arbitrary first-order language \(\mathcal{L}\). Let \(S, T\) be finite subsets of variables and let \(M^S\) and \(M^T\) denote the sets of functions into \(M\) with domains \(S\) and \(T\) respectively. If \(f : M^S \to M^T\) is a map, then \(\text{graph}(f)\) induces a complete Boolean algebra homomorphism \(\text{graph}(f) : \mathcal{P}(M^T) \to \mathcal{P}(M^S)\), between the Boolean algebras \(\mathcal{P}(M^T)\) and \(\mathcal{P}(M^S)\) of the potential interpretations of \(T\)-predicates and the potential interpretations of \(S\)-predicates, defined as follows: for any \(B \subseteq M^T\),

\[
\langle f \rangle(B) = f^{-1}[B] = \{ m \in M^S \mid f(m) \in B \}.
\]

Hence \((f)\) has both a right-adjoint \([f]\) and a left-adjoint \(\langle f \rangle\), as in the following picture:

\[
\begin{array}{ccc}
\mathcal{P}(M^T) & \xrightarrow{\langle f \rangle} & \mathcal{P}(M^S) \\
\downarrow & & \downarrow \\
\mathcal{P}(M^S) & \xleftarrow{[f]} & \mathcal{P}(M^T)
\end{array}
\]

which are defined by the following assignments: for any \(A \subseteq M^S\),

\[
\langle f \rangle(A) = \{ n \in M^T \mid \forall m \in M^S. f(m) = n \Rightarrow m \in A \}
\]

\([f](A) = \{ n \in M^T \mid \exists m \in M^S. f(m) = n \& m \in A \}.
\]

Notice in particular that if \(A = \{ m \} \in J^\infty(\mathcal{P}(M^S))\) for some \(m : S \to M\), then \(\langle f \rangle(\{ m \}) = \{ f(m) \}\).

There are two types of relevant instances of maps \(f : M^S \to M^T\). The first is given for \(S := T \cup \{ x \}\) for some \(x \notin T\), and \(f := \pi_x : M^{T\cup\{x\}} \to M^T\) is the canonical projection along \(T\). In this case, \((\pi_x)(A)\) is the cylindrification (cf. [19]) of \(A\) over the \(x\)-coordinate, and the adjoint maps \([\pi_x]\) and \(\langle \pi_x \rangle\) are the semantic counterparts of the usual first-order logic quantifiers \(\forall x\) and \(\exists x\). The second type of instance is given for \(f\) arising from a simultaneous substitution \((t/\tau)\).

In this case, \(S := \text{FV}(t_\tau), T := \tau = \{ x_1, \ldots, x_n \}\), and \(f := t_\tau : M^S \to M^T\) is defined by the assignment: for any \(m \in M^S\),

\[
t_\tau(m) = (t_{x_1}(m), \ldots, t_{x_n}(m)).
\]

In this case, the corresponding \((t/\tau)\) is such that, if \(B \subseteq M^T\) is the semantic interpretation of a \(T\)-predicate \(P\), then \((t_\tau)(B)\) is the semantic interpretation of

\[\text{FV}(P) = T.\]
the $S$-predicate $(t_\pi \pi)^P$ resulting from applying the simultaneous substitution $(t_\pi \pi)$ to $P$. That is, in this case, $(t_\pi)$ is the semantic interpretation of the substitution from which $t_\pi$ arises.

**Proposition 1.** For every model $M$ for $\mathcal{L}$ and for every $f : M^S \rightarrow M^T$ the following inclusions hold:

$$
\begin{align*}
[f](A \cup [f](B)) & \subseteq [f](A \cup B) \\
[f](A \cap [f](B)) & \subseteq [f](A \cap [f](B)) \\
[f](A \Rightarrow [f](B)) & \subseteq [f](A \Rightarrow [f](B)) \\
[f](A \supseteq [f](B)) & \subseteq [f](A \supseteq [f](B))
\end{align*}
$$

**Proof.** The inclusions in the first row are a consequence of the monotonicity of $[f]$ and $(f)$. As to proving the left hand inclusion of the second row, by adjunction and residuation it is enough to show that

$$
A \cap (f)((f)(A) \Rightarrow [f](B)) \subseteq B.
$$

Since $A \subseteq (f)((f)(A))$ it is enough to show that

$$(f)((f)(A)) \cap (f)((f)(A) \Rightarrow [f](B)) \subseteq B.
$$

Since $(f)$ is meet-preserving this is equivalent to

$$(f)((f)(A) \cap (f)(A) \Rightarrow [f](B))) \subseteq B
$$

which, by adjunction, is equivalent to

$$(f)(A) \cap ((f)(A) \Rightarrow [f](B)) \subseteq [f](B)
$$

which holds by the definition of implication (also when the implication is defined intuitionistically). The proof of the right hand side inclusion of the second row is dual and uses the join-preservation of $(f)$. \hfill \square

Summing up, we are justified in introducing the following

**Definition 2.** For every model $M$ for $\mathcal{L}$, the heterogeneous powerset $\mathcal{L}$-model associated with $M$ is the tuple $(\mathbb{H}_M, V)$ such that $\mathbb{H}_M = (A_M, Q_M, S_M)$ is defined as follows:

- $A_M = \{P(M^F) \mid F \in P_\omega(Var)\}$;
- $Q_M = \{(\pi_x), (\pi_x), (\pi_x) \mid F \in P_\omega(Var_x) \text{ and } \pi_x : M^{F \cup \{x\}} \rightarrow M^F\}$;
- $S_M = \{(t_F), (t_F), (t_F) \mid F \in P_\omega(Var) \text{ and } t_F \in Trm^F\}$,

and $V$ is such that $V(A) \in P(M^F)$ for every atomic formula $A$ of $\mathcal{L}$ with $\text{FV}(A) = F$. In Section\footnote{The connective $\supseteq$ denotes the left residual of join and, in the case of Boolean algebras, $A \supseteq B = \neg A \land B$.} we formally introduce the multi-type propositional language $\mathcal{L}_{MT}$ which is naturally interpreted on heterogeneous models such as those above. Notice that, in the case of heterogeneous powerset $\mathcal{L}$-models, $(\pi_x), (\pi_x), (\pi_x)$ are the concrete functions arising from $\pi_x$ as described above.

**Proposition 3.** For every $\mathcal{L}$-model $M$, the heterogeneous $\mathcal{L}$-model associated with $M$ satisfies the following identities:

\footnote{In category-theoretic parlance, this can be seen as a product-preserving functor from the category of terms to the category of Boolean algebras.}
\[(f)(A \cap B) = (f)(A) \cap (f)(B) \quad \text{(a)}\]
\[(f)(A \supset B) = (f)(A) \supset (f)(B) \quad \text{(b)}\]
\[(\pi_x)(\pi_y | A) = | \pi_y | (\pi_x | A) \quad \text{(c)}\]
\[(t_F)(\pi_y | A) = | \pi_y | (z_y, t_F | A) \quad \text{(d)}\]
\[(s(t_F/\pi_F))(\pi_y) = (\pi_y)(s_{t_F}) \quad \text{(e)}\]
\[(t_F)(\pi_y \cdots \pi_y | A) = (\pi_y)(s_{t_F} \cdot t_F(A)) \quad \text{(f)}\]

where \(x \neq y, z \notin \text{FV}(t_F) \cup F\) and \(\{z_1, \ldots, z_k\} = \text{FV}(s) \setminus \text{FV}(t_F)\) and \(f \in \{\pi_x, t_F\}\).

\textbf{Proof.} The identities in (a) and (b) immediately follow from \((\pi_x)\) being a Boolean algebra homomorphism. Notice that the remaining identities are all Sahlqvist, so, in what follows, we will show that their correspondents computed with ALBA hold in the model.

As to the identities in (c), they both have the same ALBA-reduct, namely
\[(\pi_x)(\pi_y) = (\pi_y)(\pi_x) | j,\]
where \(j \in J^\infty(\mathcal{P}(M^{F \cup \{x\}}))\), that is, \(j\) can be identified with a map \(j: F \cup \{x\} \rightarrow M\). As discussed above, \((\pi_x)\) takes any such \(j\) in the appropriate type to the \(y\)-cylindrification of \(j\), and \((\pi_x)\) forgets the \(x\)-coordinate, then it is clear that these two operations commute when \(x \neq y\).

The identities in (d) have the following ALBA-reducts respectively:
\[(z_y, t_F)(\pi_y) = (\pi_y)(t_F) | j \quad \text{and} \quad (t_F)(\pi_y) = (\pi_y)(z_y, t_F) | j,\]
where \(j \in J^\infty(\mathcal{P}(M^{F \cup \{t_F\}}))\), that is, we can write \(j: \text{FV}(t_F) \rightarrow M\). Notice that \((\pi_x)\) takes \(j\) to its \(z\)-cylindrification in \(\mathcal{P}(M^{F \cup \{t_F\}})\), and \((z_y, t_F)\) transforms any element \(j'\) of this cylindrification by sending any coordinate different from \(z\) to the corresponding \(t(j')\) and renaming the element in the \(z\)-coordinate by declaring it the \(y\)-coordinate; moreover \((t_F)\) transforms \(j\) into \(t(j)_{t_F}\), and \((\pi_y)\) cylindrifies it by adding the \(y\)-coordinate. It is clear that these two compositions yield the same outcome. The second identity is argued analogously.

The left-hand identity in (e) has the following ALBA-reduct:
\[(\pi_x)(\pi_y) = (\pi_y)(\pi_x) | j,\]
Recalling that, with a bit of notational abuse, \((f)(j) = f(j)\), the assumption that \(x \neq y\) guarantees that the order in which the projections are applied does not matter.

The right-hand identity in (e) has the following ALBA-reduct:
\[(s(t_F/\pi_F))(\pi_y) = (s_{t_F})(t_F) | j,\]
This identity expresses that applying \(t_F\) to \(j\) and then \(s_{t_F}\) to \(t(j)_{t_F}\) is the same as applying the compounded substitution map \(s(t_F/\pi_F)\) to \(j\), which is trivially true.

Finally, the identity (f) has the following ALBA-reduct:
\[(t_F)(\pi_y \cdots \pi_y | A) = (\pi_y)(s_{t_F} \cdot t_F | A) | j,\]
This identity expresses that applying \(t_F\) after having forgotten the coordinates \(z_1, \ldots, z_n\) is the same as applying \(t_F\) in parallel with any substitution \(s_{t_F}\) with \(\{z_1, \ldots, z_n\} = \text{FV}(s) \setminus \text{FV}(t_F)\) and \(y \notin F\), and then forgetting the \(y\)-coordinate. This is again immediately true. \(\square\)
Remark 4. Notice that while the identities in \((a)\) and \((b)\) in the proposition above hold for any function \(f\), the remaining identities hold true because of the particular functions involved, and hence by choosing different functions, corresponding to different notions of quantifiers and substitutions, these properties might change.

Proposition\(^3\) provides the guidelines for completing the definition of heterogeneous \(L\)-algebras. Namely, we are going to use the identities \((a)-(f)\) as parts of the following

Definition 5. For any first-order logic \(L\) over a denumerable set of individual variables \(\text{Var}\), a heterogeneous \(L\)-algebra is a tuple \(H = (A, Q, S)\), such that

- \(A = \{A_F \mid F \in \mathcal{P}_\omega(\text{Var})\}\);
- \(Q = \{(x)_T, (x)_F, (x)_F, x \in F \in \mathcal{P}_\omega(\text{Var})\) and \(T \in \mathcal{P}_\omega(\text{Var}_x)\}\);
- \(S = \{(t_F), (t_F), (t_F) \mid F \in \mathcal{P}_\omega(\text{Var})\) and \(t_F \in \text{Trm}_F\}\).

where for every \(F \in \mathcal{P}_\omega(\text{Var})\), \(A_F\) is a Boolean algebra and

\[
\begin{array}{c}
\xymatrix{ & A_{T \cup \{x\}} \ar[rd] - T \\
A_T \ar[ru] - T & & A_F \\
(x)_{T \cup \{x\}} \ar[ru] - T & & (x)_T \\
\end{array}
\]

\[
\begin{array}{c}
\xymatrix{ & A_{T \cup \{x\}} \ar[rd] - T \\
A_T \ar[ru] - T & & A_F \\
(t_F) \ar[ru] - T & & (t_F) \\
\end{array}
\]

such that \((x)_T\) is an order embedding and the following equations hold (we omit mentioning the types):

\[
\begin{align*}
\ast\ast(a \land b) &= \ast\ast(a) \land \ast\ast(b) & \ast\ast(a \lor b) &= \ast\ast(a) \lor \ast\ast(b) & (a) \\
\ast\ast(a \rightarrow b) &= \ast\ast(a) \rightarrow \ast\ast(b) & \ast\ast(a \rightarrow b) &= \ast\ast(a) \rightarrow \ast\ast(b) & (b) \\
\ast\ast[y](a) &= \ast\ast[y](a) & \ast\ast[y](a) &= \ast\ast[y](a) & (c) \\
\ast\ast[t_F](y)(a) &= \ast\ast[z](z_F, t_F)(a) & \ast\ast[t_F](y)(a) &= \ast\ast[z](z_F, t_F)(a) & (d) \\
\ast\ast(y)(a) &= \ast\ast(y)(a) & \ast\ast[t_F](y)(a) &= \ast\ast[z](z_F, t_F)(a) & (e) \\
\ast\ast[t_F](y)(a) &= \ast\ast[z](z_F, t_F)(a) & (f) \\
\end{align*}
\]

where \(x \neq y, z \notin \text{FV}(t_F) \cup F\), \((\ast) \in \{(x), (t_F)\}\) and \(\{z_1, \ldots, z_k\} = \text{FV}(s) \setminus \text{FV}(t_F)\).

A heterogeneous \(L\)-algebra is perfect if every \(A_F\) is\(^7\)

Clearly, even the class of perfect heterogeneous \(L\)-algebras is much larger than the class of those arising from \(L\)-models. However, as we will discuss in the next section, there is a way in which \(L\) is sound with respect to this larger class of models.

4 Multi-type presentation of first-order logic

Let \(L\) be a first-order language over a denumerable set of individual variables \(\text{Var}\). The notion of heterogeneous \(L\)-algebras (cf. Definition\(^5\) naturally

\(^7\)A Boolean algebra is perfect if it is complete and completely join-generated by its atoms. Equivalently, perfect Boolean algebras are isomorphic to powerset algebras.
comes with the following multi-type propositional language $\mathcal{L}_{\mathbb{MT}}$ canonically interpreted in it. Types in this language bijectively correspond to elements $F \in \mathcal{P}_\omega(\text{Var})$. The sets $\mathcal{L}_F$ of $F$-formulas, for each such $F$, are defined by simultaneous induction as follows:

\[
\mathcal{L}_F \ni A^F \quad ::= \quad R(t_{\pi}) \mid t = s \mid 
\top^F \mid \bot^F \mid A^F \land A^F \mid A^F \lor A^F \mid A^F \to A^F \mid A^F \supset A^F \mid 
[y]A^{F \cup \{y\}} \mid (x)A^{F \setminus \{x\}} \mid (t_{\gamma})A^F
\]

where $\text{FV}(t_{\gamma}) = F$, $y \notin F$, $x \in F$, and $R$ is any relation symbol of $\mathcal{L}$, and $t_{\gamma} : F' \to \text{Trm}$. The symbol $(t_{\gamma})$ denotes the simultaneous substitution of $t_{v_m}$ for $v_m$ for every $v_m \in F'$.

**Definition 6.** A heterogeneous algebraic $\mathcal{L}$-model is a tuple $(\mathbb{H}, V)$ such that $\mathbb{H}$ is a heterogeneous $\mathcal{L}$-algebra and $V$ maps atomic propositions $R(t_{\gamma})$ and $t = s$ in $\mathcal{L}_F$ to elements of $\mathbb{H}_F$ so that for any terms $r, s$ and $t_{\pi}$ and any relation symbol $P$ (also including equality and $\top$):

1. $V(P(t_{\pi})) = (\tau) V(P(t_{\pi}))$
2. $V(t = s) = \top$
3. $(z_1, \ldots, z_n) V(r = s) \land (y_1, \ldots, y_m) V(P(r, t_{\pi})) \leq (z_1, \ldots, z_k) V(P(s, t_{\pi}))$

where $\{x_1, \ldots, x_n\} = \text{FV}(t_{\gamma}) \setminus \text{FV}(r, s)$, $\{y_1, \ldots, y_m\} = \text{FV}(s) \setminus \text{FV}(r, t_{\gamma})$ and $\{z_1, \ldots, z_k\} = \text{FV}(r) \setminus \text{FV}(s, t_{\gamma})$.

The definition of the interpretation $(\mathbb{H}, V) \models A^F$ of $\mathcal{L}_{\mathbb{MT}}$-formulas into heterogeneous $\mathcal{L}$-algebras straightforwardly generalizes the definition of the interpretation of propositional languages in algebras of compatible signature.

The discussion of the previous section also motivates the definition of the following translation:

**Definition 7.** Define the translation $(\cdot)^{\top} : \mathcal{L}_{\mathbb{MT}} \to \mathcal{L}$ as follows:

\[
R(t_{\gamma})^{\top} = R(t_{\gamma}) \quad \top^{\top} = \top \quad t = s^{\top} = t = s \quad \bot^{\top} = \bot \quad (A \land B)^{\top} = A^{\top} \land B^{\top} \quad (A \lor B)^{\top} = A^{\top} \lor B^{\top} \quad (A \supset B)^{\top} = A^{\top} \supset B^{\top} \quad (\exists yA^{\top}) = \exists yA^{\top} \quad (\forall yA^{\top}) = \forall yA^{\top} \quad ((t_{\gamma})A)^{\top} = ((t_{\gamma})A)^{\top}
\]

**Lemma 8.** Let $(\mathbb{H}, V)$ be a heterogeneous algebraic $\mathcal{L}$-model. Let $P$ be a relation symbol of $\mathcal{L}$ or the equality. If

\[
((t_{\gamma_1}) \cdots (t_{\gamma_n}) P(t_{\pi}))^{\top} = ((s_{\gamma_1}) \cdots (s_{\gamma_m}) P(s_{\pi}))^{\top}
\]

then

\[
(t_{\gamma_1}) \cdots (t_{\gamma_n}) V(P(t_{\pi})) = (s_{\gamma_1}) \cdots (s_{\gamma_m}) V(R(s_{\pi})).
\]

**Proof.** Let

\[
((t_{\gamma_1}) \cdots (t_{\gamma_n}) R(t_{\pi}))^{\top} = ((s_{\gamma_1}) \cdots (s_{\gamma_m}) R(s_{\pi}))^{\top}.
\]

(4.1)

Let us assume that $(t_{\gamma_1}) \cdots (t_{\gamma_n}) R(t_{\pi}) = R(t_{\pi})$, that is

\[
((t_{\gamma_1}) \cdots (t_{\gamma_n}) (t_{\gamma}) R(t_{\pi}))^{\top} = R(t_{\pi})
\]

10
therefore the composition of \((t_{\tau_1}) \cdots (t_{\tau_k})(\tau)\) is \((r \tau)\). By applying the right-hand
equality of \((e)\) of Definition 5 a finite number of times we obtain

\[
(t_{\tau_1}) \cdots (t_{\tau_k})(\tau)V(R(\tau)) = (r \tau)V(R(\tau)) = V(R(r \tau))
\]  

(4.2)

the second equality holding by item 1 of Definition 6. By assumption (4.1),
equality of \(\text{Lemma 9.} \)

\[
(s_{\tau_1}) \cdots (s_{\tau_m})R(s_{\tau}) = R(r_{\tau}).
\]

Similar reasoning as above yields

\[
(s_{\tau_1}) \cdots (s_{\tau_m})(s \tau)V(R(\tau)) = V(R(r_{\tau})).
\]  

(4.3)

By (4.2) and (4.3) we get

\[
(t_{\tau_1}) \cdots (t_{\tau_k})(\tau)V(R(\tau)) = (s_{\tau_1}) \cdots (s_{\tau_m})(s \tau)V(R(\tau)).
\]

Since \((\tau)V(R(\tau)) = V(R(t \tau))\) and \((s \tau)V(R(\tau)) = V(R(s_{\tau}))\) by Definition 6,

\[
(t_{\tau_1}) \cdots (t_{\tau_k})V(R(t \tau)) = (s_{\tau_1}) \cdots (s_{\tau_m})V(R(s_{\tau})),
\]

which completes the proof. \(\square\)

**Lemma 9.** Let \(A\) be a formula with free variable \(x\). Then

\[
(x_1) \cdots (x_n)V(r = s) \land (y_1) \cdots (y_m)(r \land t_F)V(A) \leq (z_1) \cdots (z_k)(s \land t_F)V(A),
\]

where \(\{x_1, \ldots, x_n\} = \text{FV}(t_F) \setminus \text{FV}(r, s)\), \(\{y_1, \ldots, y_m\} = \text{FV}(s) \setminus \text{FV}(r, t_F)\) and
\(\{z_1, \ldots, z_k\} = \text{FV}(r) \setminus \text{FV}(s, t_F)\).

**Proof.** We proceed by induction on the complexity of \(A\). The base case follows immediately from 1 and 3 of Definition 6. Since \((x)\) and \((s \tau)\) are Boolean algebra homomorphisms the cases when \(A\) is \(B \land C\) and \(\perp\) follow easily. The case when \(A\) is \(B \rightarrow C\) uses the fact that \((x)\) and \((s \tau)\) are Boolean algebra homomorphisms and utilizes the following reasoning: Assume that \(x \land b_1 \leq c_1\) and \(x \land c_2 \leq b_2\).

We have:

\[
x \land b_1 \leq c_1 \\
x \land b_2 \land (b_2 \rightarrow b_1) \leq c_1 \\
x \land c_2 \land (b_2 \rightarrow b_1) \leq c_1 \\
x \land (b_2 \rightarrow b_1) \leq c_2 \land c_1.
\]

If \(A\) is \((y)B\), \((y)B\), \((y)B\) we use the equalities \((c)-(f)\) of Definition 6. \(\square\)

**Corollary 10.** Let \(A\) be a formula with free variable \(x\). If

\[
(x_1) \cdots (x_n)V(r = s) \leq (z_1) \cdots (z_k)(s \land t_F)V(A)
\]

then

\[
(x_1) \cdots (x_n)V(r = s) \leq (z_1) \cdots (z_k)(s \land t_F)V(A)
\]

where \(\{x_1, \ldots, x_n\} = \text{FV}(t_F) \setminus \text{FV}(r, s)\), \(\{y_1, \ldots, y_m\} = \text{FV}(s) \setminus \text{FV}(r, t_F)\) and
\(\{z_1, \ldots, z_k\} = \text{FV}(r) \setminus \text{FV}(s, t_F)\).

Notice that this proof works also for Heyting algebras.
Proof. Immediately from Lemma 9. □

Proposition 11. For every $\mathcal{L}$-model $M$ and every $A \in \mathcal{L}_{MT}$,

$$M \models A^* \text{ iff } (\mathbb{H}_M, V) \models A.$$  

Proof. By straightforward induction on $A$. □

The identities (a)-(f) of Proposition 5 have the following syntactic counterparts in the language $\mathcal{L}_{MT}$:

$$(*) (A \land B) \vdash (*) A \land (*) B \quad L \vdash (*) A \lor (*) B \quad L \vdash (*) A \leftrightarrow (*) B \quad (a)$$

$$(*) (A \to B) \vdash (*) A \to (*) B \quad (x)(y) A \vdash [y](x) A \quad (x)(y) A \vdash (y)(x) A \quad (c)$$

$$(x)(y) A \vdash [z](x,y) A \quad (t)(y) A \vdash (y)(t) A \quad (e)$$

$$(x)(y) A \vdash (y)(x) A \quad (t)(x,t) A \vdash (s(t)(x,t)) A \quad (f)$$

These inequalities are all analytic inductive (cf. [15, Definition 55]), and hence give rise to analytic structural rules. We will exploit this fact in the next section when introducing the rules of the calculus.

5 Multi-type calculus for first-order logic

For any first-order language $\mathcal{L}$ (cf. Section 2) the structural and operational connectives of the proper multi-type display calculi $D_{\mathcal{F}O}$ and $D_{\mathcal{F}O}^*$ are the following:

- Logical and structural homogeneous connectives for any type $F$:

| Structural symbols | $\vdash$ | $\dashv$ | $\vdash R(t_\tau)$ | $s \equiv t$ |
|--------------------|---------|---------|-----------------|-------------|
| Operational symbols | $\vdash$ | $\dashv$ | $\vdash R(t_\tau)$ | $s \equiv t$ |

| Structural symbols | $\hat{\land}$ | $\hat{\lor}$ | $\hat{\rightarrow}$ | $\hat{\leftarrow}$ |
|--------------------|----------------|----------------|-----------------|----------------|
| Operational symbols | $\hat{\land}$ | $\hat{\lor}$ | $(\hat{\rightarrow})$ | $(\hat{\leftarrow})$ |

- Logical and structural heterogeneous for each $x,y \in \mathcal{V}$:

| $\mathcal{L}_{F\setminus \{x\}} \to \mathcal{L}_F$ | $\mathcal{L}_{F\setminus \{y\}} \to \mathcal{L}_F$ | $\mathcal{L}_{F'} \to \mathcal{L}_F$ | $\mathcal{L}_F \to \mathcal{L}_{F'}$ |
|-----------------|-----------------|-----------------|-----------------|
| Structural symbols | $(x)$ | $(y)$ | $[t_\tau]$ | $(t_\tau)$ |
| Operational symbols | $(x)$ | $(y)$ | $[t_\tau]$ | $(t_\tau)$ |

In the table above, some operational symbols above $(\hat{\rightarrow}, \hat{\leftarrow}, \hat{\langle}, (t_\tau))$ and $[t_\tau]$ appear within brackets to signify that, unlike their associated structural symbols, they occur only in the language and calculus for $D_{\mathcal{F}O}^*$.

Below we define the display calculi $D_{\mathcal{F}O}^*$ and $D_{\mathcal{F}O}$. Notice that formally each type, $F$, has its own turnstile denoted by $\Gamma_F$. Below to avoid overloading the notation we sometimes omit the subscript, since the type can be directly deduced by the free variables of the structures on the left and right of the turnstile.

Definition 12. The display calculi $D_{\mathcal{F}O}^*$ and $D_{\mathcal{F}O}$ consist of the following display postulates, structural rules, and operational rules:
1. Identity and Cut:

\[
\begin{align*}
\text{Id} & : \quad R(t_{F'}) \vdash R(t_{F'}) \quad \text{Id} & : \quad s \equiv t \vdash s = t \\
X \vdash_{F} A & \quad A \vdash_{F} Y \quad \text{Cut}
\end{align*}
\]

2. Display postulates for homogeneous connectives:

\[
\begin{align*}
X \& Y & \vdash_{F} Z & \quad Y \vdash_{F} X \Rightarrow Z \\
X \& Y & \vdash_{F} Z & \quad Z \vdash_{F} X \Rightarrow Y \\
X \& Y & \vdash_{F} Z & \quad Z \vdash_{F} X \Leftrightarrow Y
\end{align*}
\]

3. Display postulates for quantifiers:

\[
\begin{align*}
X \vdash_{F \cup \{x\}} (\hat{x})Y & \quad (\hat{x})X \vdash_{F \setminus \{x\}} Y \\
\end{align*}
\]

4. Display postulates for substitutions:

\[
\begin{align*}
Y \vdash_{F'} (\check{t}F) & \quad \langle \check{t}F \rangle Y \vdash_{F'} X \\
\end{align*}
\]

5. Additional adjunction-related rules

\[
\begin{align*}
\text{cadj} & : \quad (\hat{x})(\check{x})X \vdash_{F} Y & \quad \text{sadj} & : \quad (\hat{t}F)(\check{t}F)X \vdash_{F} Y
\end{align*}
\]

6. Necessitation for quantifiers:

\[
\begin{align*}
\uparrow_{F \cup \{x\}} & \vdash_{F \cup \{x\}} X & \quad X \vdash_{F \cup \{x\}} \uparrow_{F \cup \{x\}} \\
\uparrow_{F \setminus \{x\}} & \vdash_{F \setminus \{x\}} X & \quad X \vdash_{F \setminus \{x\}} \uparrow_{F \setminus \{x\}}
\end{align*}
\]

7. Necessitation for substitution where \(\text{FV}(\check{t}F) = F\):

\[
\begin{align*}
\uparrow_{F'} & \vdash_{F} X & \quad X \vdash_{F} \uparrow_{F'} \\
\langle \check{t}F \rangle \uparrow_{F'} & \vdash_{F} X & \quad X \vdash_{F} (\check{t}F) \uparrow_{F'}
\end{align*}
\]

8. Structural rules for atomic formulas:

\[
\begin{align*}
(\hat{s}_{r_1}) \cdots (\hat{s}_{r_m})R(s_{\tau}) \vdash X & \quad (\hat{s}_{r_1}) \cdots (\hat{s}_{r_m})R(t_{\tau}) \vdash X \\
(\check{s}_{r_1}) \cdots (\check{s}_{r_m})R(t_{\tau}) & \vdash X & \quad (\check{s}_{r_1}) \cdots (\check{s}_{r_m})r_3 \equiv r_4 \vdash X \\
(\check{t}_{r_1}) \cdots (\check{t}_{r_m})r_3 \equiv r_4 & \vdash X & \quad (\check{t}_{r_1}) \cdots (\check{t}_{r_m})r_1 \equiv r_2 \vdash X
\end{align*}
\]

where the sequents are well-typed and\footnote{We will discuss these axioms in the conclusions.}

\[
(\check{t}_{r_1}) \cdots (\check{t}_{r_m})R(t_{\tau}) = (\check{s}_{r_1}) \cdots (\check{s}_{r_m})R(s_{\tau})
\]

and

\[
(\check{t}_{r_1}) \cdots (\check{t}_{r_m})r_1 = r_2 = (\check{s}_{r_1}) \cdots (\check{s}_{r_m})r_3 = r_4
\]
9. Structural rules encoding the behaviour of conjunction and disjunction:

\[
\begin{align*}
\vdash_L & : \frac{X \vdash F Y}{\top \wedge X \vdash F Y} & \frac{Y \vdash F X}{Y \vdash F X \vee \bot} \\
\vdash_R & : \frac{\top \wedge X \vdash F Y}{Z \vdash F X \vee Y} & \frac{Z \vdash F X \vee Y}{Z \vdash F Y \vee X} \\
E_L & : \frac{Y \wedge X \vdash F Z}{X \wedge Y \wedge F Z} & \frac{Z \vdash F X \vee Y}{Z \vdash F Y \vee X} \\
E_R & : \frac{Y \wedge X \wedge F Z}{Z \vdash F Y \vee X} & \frac{Y \vdash F X}{Z \vdash F Y \vee X} \\
W_L & : \frac{Y \vdash F Z}{X \wedge Y \vdash F Z} & \frac{Z \vdash F Y}{Z \vdash F Y \vee X} \\
W_R & : \frac{Z \vdash F Y}{Z \vdash F Y \vee X} & \frac{Y \vdash F X}{Y \vdash F X} \\
C_L & : \frac{X \wedge X \vdash F Y}{X \vdash F Y} & \frac{X \wedge X \vdash F Y}{Y \vdash F X} \\
C_R & : \frac{Y \vdash F X \vee X}{Y \vdash F X} & \frac{Y \vdash F X}{Z \vdash F Y \vee X} \\
A_L & : \frac{X \wedge (Y \wedge Z) \vdash F W}{(X \wedge Y) \wedge Z \vdash F W} & \frac{W \vdash F (Z \vee Y) \vee X}{W \vdash F Z \vee (Y \vee X)} \\
A_R & \end{align*}
\]

10. Structural rules for equality:

\[
\frac{t \doteq t \hat{\wedge} X \vdash Y}{X \vdash Y} \quad \frac{(\bar{z}_1) \cdots (\bar{z}_n) t \doteq s \vdash (\bar{y}_1) \cdots (\bar{y}_m)(\bar{t}_{x, r}) X}{(\bar{z}_1) \cdots (\bar{z}_n) s \doteq s \hat{\wedge} (\bar{z}_1) \cdots (\bar{z}_k)(\bar{s}_{x, r}) X}
\]

where \(x_1, \ldots, x_n = \text{FV}(r_F) \setminus \text{FV}(t, s), \{y_1, \ldots, y_m\} = \text{FV}(s) \setminus \text{FV}(t, r_F)\)
and \(\{z_1, \ldots, z_k\} = \text{FV}(t) \setminus \text{FV}(s, r_F)\).

11. Introduction rules for atomic formulas:

\[
\frac{\hat{R}(s_x) \vdash X}{R(s_x) \vdash X} \quad s \doteq t \vdash X
\]

12. Introduction rules for the propositional connectives:

\[
\begin{align*}
\vdash_L & : \frac{\bot \vdash \bot}{\bot \vdash \bot} & \frac{\top \vdash \top}{\top \vdash \top} \\
\vdash_R & : \frac{\top \vdash \top}{\top \vdash \top} & \frac{\bot \vdash \bot}{\bot \vdash \bot} \\
\wedge_L & : \frac{A \hat{\wedge} B \vdash X}{A \wedge B \vdash X} & \frac{X \vdash A \wedge B}{X \wedge Y \vdash A \wedge B} \\
\wedge_R & : \frac{A \wedge B \vdash X}{X \wedge Y \wedge A \wedge B} & \frac{X \wedge Y \wedge A \wedge B}{X \vee A \wedge B} \\
\vee_L & : \frac{A \vdash X}{A \vee B \vdash X \vee Y} & \frac{X \vdash A \vee B}{X \vee A \wedge B} \\
\vee_R & : \frac{A \vee B \vdash X}{X \wedge A \vee B \wedge A \wedge B} & \frac{X \wedge A \vee B \wedge A \wedge B}{X \vdash A \wedge B} \\
\rightarrow_L & : \frac{X \vdash A \rightarrow B}{A \rightarrow B \vdash X \rightarrow Y} & \frac{X \vdash A \rightarrow B}{X \vdash A \rightarrow B}
\end{align*}
\]

We omit the type since these rules involve only one type. In the presence of the exchange rules \(E_L\) and \(E_R\), the structural connective \(<\) and the corresponding operational connectives \(<\) and \(\leq\) are redundant.

13. Grishin rules for classical logic:
\[
\begin{align*}
\text{Gri} & \quad X \hat{\geq} (Y \hat{\land} Z) \vdash W \\
& \quad (X \hat{\geq} Y) \hat{\land} Z \vdash W \\
W & \vdash X \implies (Y \lor Z) \\
& \quad W \vdash (X \implies Y) \lor Z \quad \text{Gri}
\end{align*}
\]

14. Introduction rules for the heterogeneous connectives of D.FO:

\[
\begin{align*}
\langle x \rangle_L & \quad (\hat{x}) A \vdash_F X \\
& \quad (\hat{x}) A \vdash_F X \\
\| \langle x \rangle & \quad A \vdash_F X \\
& \quad A \vdash_F \{x\} \vdash A \\
\langle x \rangle_L & \quad (\hat{x}) A \vdash_F X \\
& \quad (\hat{x}) A \vdash_F X \\
\langle t_{p'} \rangle_L & \quad (\hat{t}_{p'}) A \vdash_F X \\
& \quad (\hat{t}_{p'}) A \vdash_F X \\
\langle t_{p'} \rangle & \quad A \vdash_F \{t_{p'}\} A \\
& \quad A \vdash_F \{t_{p'}\} A \\
\langle t_{p'} \rangle_L & \quad (\hat{t}_{p'}) A \vdash_F X \\
& \quad (\hat{t}_{p'}) A \vdash_F X \\
\langle t_{p'} \rangle & \quad A \vdash_F \{t_{p'}\} A \\
& \quad A \vdash_F \{t_{p'}\} A
\end{align*}
\]

15. Introduction rules for the heterogeneous connectives of D.FO*:

\[
\begin{align*}
\langle t_{p'} \rangle_L & \quad (\hat{t}_{p'}) A \vdash_F X \\
& \quad (\hat{t}_{p'}) A \vdash_F X \\
\| \langle t_{p'} \rangle & \quad A \vdash_F \{t_{p'}\} A \\
& \quad A \vdash_F \{t_{p'}\} A \\
\langle t_{p'} \rangle_L & \quad (\hat{t}_{p'}) A \vdash_F X \\
& \quad (\hat{t}_{p'}) A \vdash_F X \\
\langle t_{p'} \rangle & \quad A \vdash_F \{t_{p'}\} A \\
& \quad A \vdash_F \{t_{p'}\} A
\end{align*}
\]

16. Monotonicity and order embedding rules:

\[
\begin{align*}
\langle x \rangle_L & \quad X \vdash_F \{x\} Y \\
& \quad \langle \hat{x} \rangle X \vdash_F \{x\} Y \\
\langle t_{p'} \rangle & \quad X \vdash_F \{t_{p'}\} Y \\
& \quad \langle \hat{t}_{p'} \rangle X \vdash_F \{t_{p'}\} Y
\end{align*}
\]

17. Interaction between homogeneous and heterogeneous connectives:

\[
\begin{align*}
\langle \hat{z} \rangle_L & \quad (\hat{z}) X \hat{\geq} (\hat{z}) Y \vdash Z \\
& \quad (\hat{z}) (X \hat{\geq} Y) \vdash Z \\
\langle \hat{z} \rangle & \quad (\hat{z}) X \hat{\preceq} (\hat{z}) Y \vdash Z \\
& \quad (\hat{z}) (X \hat{\preceq} Y) \vdash Z \\
\langle \hat{z} \rangle & \quad (\hat{z}) X \hat{\geq} (\hat{z}) Y \vdash Z \\
& \quad (\hat{z}) (X \hat{\geq} Y) \vdash Z \\
\langle \hat{z} \rangle & \quad (\hat{z}) X \hat{\preceq} (\hat{z}) Y \vdash Z \\
& \quad (\hat{z}) (X \hat{\preceq} Y) \vdash Z \\
\langle \hat{t}_{p'} \rangle_L & \quad (\hat{t}_{p'}) X \hat{\geq} (\hat{t}_{p'}) Y \vdash Z \\
& \quad (\hat{t}_{p'}) (X \hat{\geq} Y) \vdash Z \\
\langle \hat{t}_{p'} \rangle & \quad (\hat{t}_{p'}) X \hat{\preceq} (\hat{t}_{p'}) Y \vdash Z \\
& \quad (\hat{t}_{p'}) (X \hat{\preceq} Y) \vdash Z \\
\langle \hat{t}_{p'} \rangle & \quad (\hat{t}_{p'}) X \hat{\geq} (\hat{t}_{p'}) Y \vdash Z \\
& \quad (\hat{t}_{p'}) (X \hat{\geq} Y) \vdash Z \\
\langle \hat{t}_{p'} \rangle & \quad (\hat{t}_{p'}) X \hat{\preceq} (\hat{t}_{p'}) Y \vdash Z \\
& \quad (\hat{t}_{p'}) (X \hat{\preceq} Y) \vdash Z
\end{align*}
\]

15
18. Interaction between heterogeneous connectives: In what follows, \( x \neq y, z \notin \text{FV}(t_{p'}) \cup F'' \) and \( \{z_1, \ldots, z_k\} = \text{FV}(s) \setminus \text{FV}(t_{p'}) \):

\[
\begin{align*}
\text{cQL} & \quad \frac{\tilde{\alpha}(\tilde{\beta})^!X \vdash Y}{(\tilde{\beta})^!X \vdash Y} & \frac{Y \vdash (\tilde{\beta})^!X}{Y \vdash [\tilde{\beta}]^!X} & \text{cQR} \\
\text{sqL} & \quad \frac{\tilde{\alpha}^!X \vdash Y}{(\tilde{\alpha}^!X \vdash Y} & \frac{Y \vdash (\tilde{\alpha}^!X)}{Y \vdash [\tilde{\alpha}^!]X} & \text{sqR} \\
\text{ccL} & \quad \frac{\tilde{\alpha}(\tilde{\beta})X \vdash Y}{(\tilde{\beta})X \vdash Y} & \frac{Y \vdash (\tilde{\beta})X}{Y \vdash [\tilde{\beta}]X} & \text{ccR} \\
\text{ssL} & \quad \frac{\tilde{\alpha}(\tilde{\beta})X \vdash Y}{(\tilde{\alpha}^!X \vdash Y} & \frac{Y \vdash [\tilde{\alpha}^!]X}{Y \vdash [\tilde{\alpha}^!]X} & \text{ssR} \\
\text{scL} & \quad \frac{\tilde{\alpha}(\tilde{\beta})X \vdash Y}{(\tilde{\alpha})(\tilde{\beta})X \vdash Y} & \frac{Y \vdash [\tilde{\alpha}][\tilde{\beta}]X}{Y \vdash [\tilde{\alpha}][\tilde{\beta}]} & \text{scR} \\
\end{align*}
\]

6. Properties

In the present section, we outline the proofs of soundness, completeness, conservativity, cut elimination and subformula property of the calculus D.FO.

6.1 Soundness

In the present subsection, we outline the verification of the soundness of the rules of D.FO w.r.t. the semantics of heterogeneous \( L \)-algebras (cf. Definition 5). As done in analogous situations [16, 14], the first step consists in interpreting structural symbols as logical symbols according to their (precedent or succedent) position as indicated in the synoptic tables of Section 3. This makes it possible to interpret sequents as inequalities, and rules as quasi-inequalities. For example, the rule on the left-hand side below is interpreted as the bi-implication on the right-hand side:

\[
\frac{(\tilde{\alpha}^!)(\tilde{\beta})^!X \vdash Y}{(\tilde{\beta})(\tilde{\alpha})X \vdash Y} \implies \forall a \forall b[(\tilde{\alpha}^!)(\tilde{\beta})^!a \leq b] \iff (\tilde{\beta})(\tilde{\alpha})^!a \leq b]
\]

Notice that the validity of the bi-implication is equivalent to the validity of the analytic inductive (in fact left-primitive) identity

\[
(\tilde{\alpha}^!)(\tilde{\beta})^!a = (\tilde{\beta})(\tilde{\alpha})^!a
\]

which holds by definition on every heterogeneous \( L \)-algebra. The soundness of the remaining unary rules is proven analogously. As to the rules for substitution (cf. Definition 12), the soundness follows from Lemma 8. Finally, the

For any sequent \( x \vdash y \), we define the signed generation trees \( +x \) and \( -y \) by labelling the root of the generation tree of \( x \) (resp. \( y \)) with the sign \( + \) (resp. \( - \)), and then propagating the sign to all nodes according to the polarity of the coordinate of the connective assigned to each node. Positive (resp. negative) coordinates propagate the same (resp. opposite) sign to the corresponding child node. Then, a substructure \( z \in x \vdash y \) is in precedent (resp. succedent) position if the sign of its root node as a subtree of \( +x \) or \( -y \) is \( + \) (resp. \( - \)).
soundness of the rules for equality (cf. Definition 12.10) follows immediately from Corollary 10.

This completes the proof that D.FO and D.FO* are sound with respect to the class of heterogeneous algebraic L-models. Since this class properly includes the structures arising from standard L-models, by Proposition 11 D.FO and D.FO* are sound with respect to standard L-models.

6.2 Translations and completeness

The aim of this subsection is to show the following

**Theorem 13.** If \( A \in \mathcal{L}_{MT} \) and \( \vdash_{FO} A' \) then \( \vdash_{D.FO} A \).

We will proceed as follows: For every \( A \in \mathcal{L} \) and \( F \in \mathcal{P}(\text{Var}) \), we will define, in two steps, a canonical \( \mathcal{L}_F \)-formula \( \kappa(F, A) \), free of explicit substitutions, and show that \( A^F \vdash \kappa(F, A) \) and \( \kappa(F, A) \vdash A^F \) are derivable sequents in D.FO for any \( A^F \in \mathcal{L}_F \) such that \( (A^F)^* = A \). Using this, we will show that for any formulas \( A, B \in \mathcal{L}_F \), if \( A^* = B^* \) then \( A \vdash B \) and \( B \vdash A \) are derivable sequents in D.FO. Thanks to this observation, to show Theorem 13 it is enough to show that for every \( \mathcal{L} \)-formula \( A \) which is a theorem of first-order logic and every \( F \in \mathcal{P}_\omega(\text{Var}) \), some \( A' \in \mathcal{L}_F \) exists such that \( \vdash_F A' \) is provable in D.FO.

Let us preliminarily show the following

**Lemma 14.** The following rules are derivable:

\[
\begin{align*}
\frac{\langle \hat{x} \rangle X \geq \langle \hat{x} \rangle Y & \vdash Z}{\langle \hat{x} \rangle (X \geq Y) \vdash Z} \\
\frac{\langle \hat{x} \rangle X \leq \langle \hat{x} \rangle Y & \vdash Z}{\langle \hat{x} \rangle (X \leq Y) \vdash Z} \\
\frac{\langle \hat{x} \rangle X \wedge \langle \hat{x} \rangle Y & \vdash Z}{\langle \hat{x} \rangle (X \wedge Y) \vdash Z}
\end{align*}
\]

where \( \langle \hat{x} \rangle \in \{\langle \hat{x}_F \rangle, \langle \hat{x} \rangle \} \) and \( \langle \hat{x} \rangle \in \{\langle \hat{x}_F \rangle, \langle \hat{x} \rangle \} \) (see also Proposition 7).

**Proof.** The proof uses the rules of Definition 12.14. We only prove one, the others being shown similarly:

\[
\begin{align*}
X \vdash \langle \hat{x} \rangle Y \rightarrow [\hat{x}]Z \\
\langle \hat{x} \rangle Y \wedge X \vdash [\hat{x}]Z \\
\langle \hat{x} \rangle (\langle \hat{x} \rangle Y \wedge X) \vdash Z \\
\langle \hat{x} \rangle Y \vee \langle \hat{x} \rangle X \vdash Z \\
\langle \hat{x} \rangle Y \vdash Z \rightarrow \langle \hat{x} \rangle X \\
Y \rightarrow Z \vdash \langle \hat{x} \rangle X \\
\langle \hat{x} \rangle X \vdash Y \rightarrow Z \\
X \vdash [\hat{x}] (Y \rightarrow Z)
\end{align*}
\]

Let us now define a function \( \sigma : \mathcal{L}_{MT} \rightarrow \mathcal{L}_{MT} \) such that for any formula \( A \in \mathcal{L}_{MT} \) the formula \( \sigma(A) \) is free of explicit substitutions. We define \( \sigma \) recursively as follows:

\[
17
\]
Lemma 15. If $\sigma(R(t_\tau)) = R(t_\tau)$;

Lemma 16. Proof

τ applying σ simply performs the substitutions. Hence applying σ is an axiom and we have

If $A_{\tau} \sigma$ is derivable in $\mathcal{L}_{\tau}$, then $\hat{R}(t_\tau) \vdash R(t_\tau)$ is an axiom and we have

as required.

If $A$ is $B \land C$ then by induction hypothesis we have:

and

which yields the result since $\sigma(B \land C) = \sigma(B) \land \sigma(C)$ by definition.

If $A$ is $B \lor C$ or $B \rightarrow C$ we argue similarly.

If $A$ is $(\forall \sigma)$B then:

Notice that, while τ performs the substitutions and removes $(\sigma)$-operators, σ simply performs the substitutions. Hence applying τ after σ is the same as applying τ. This motivates the following:

Lemma 15. If $A \in \mathcal{L}_{\tau}$ then $\sigma(A) \in \mathcal{L}_{\tau}$. Furthermore $(\sigma(A))^\tau = A^\tau$.

Proof. Straightforward induction on the complexity of A. □

Lemma 16. For every $A \in \mathcal{L}_{MT}$, the sequents $A \vdash \sigma(A)$ and $\sigma(A) \vdash A$ are derivable in D.FO.

Proof. By induction on the complexity of $A$. If $A$ is $R(t_\tau)$, then $\hat{R}(t_\tau) \vdash R(t_\tau)$
which yields the result since \( \sigma((\bar{z})B) = (\bar{z})\sigma(B) \) (and similarly for the other direction).

If \( A \) is \( [x]B \) then:

\[
\frac{B \vdash \sigma(B)}{[x]B \vdash [x]\sigma(B)}
\]

which yields the result since \( \sigma([x]B) = [x]\sigma(B) \), and similarly for the other direction and for \( \langle x \rangle B \).

If \( A \) is \( (s_F)B \), notice that

\[
\frac{\langle s_F \rangle B \vdash \sigma((s_F)B)}{(s_F)B \vdash \sigma((s_F)B)} \quad \frac{\sigma((s_F)B) \vdash \langle s_F \rangle B}{\sigma((s_F)B) \vdash (s_F)B}
\]

and hence it is enough to show that \( \langle s_F \rangle B \vdash \sigma((s_F)B) \) and \( \sigma((s_F)B) \vdash \langle s_F \rangle B \) are derivable. We proceed by induction on the complexity of \( B \).

If \( B \) is \( R(t_{\bar{\eta}}) \) then by definition \( \sigma((s_F)R(t_{\bar{\eta}})) = R(t(s_{\bar{\eta}/\bar{\tau}})_{\bar{\eta}}) \) and therefore we have:

\[
\frac{\langle s_F \rangle R(t_{\bar{\eta}}) \vdash R(t(s_{\bar{\eta}/\bar{\tau}})_{\bar{\eta}})}{\langle s_F \rangle R(t_{\bar{\eta}}) \vdash R(t(s_{\bar{\eta}/\bar{\tau}})_{\bar{\eta}})} \quad \frac{\langle s_F \rangle R(t_{\bar{\eta}}) \vdash R(t_{\bar{\eta}})}{\langle s_F \rangle R(t_{\bar{\eta}}) \vdash R(t_{\bar{\eta}})}
\]

which yield the desired result.

If \( B \) is \( C \land D \) then:

\[
\frac{\langle s_F \rangle C \vdash \sigma((s_F)C)}{\langle s_F \rangle C \land (s_F)D \vdash \sigma((s_F)C) \land \sigma((s_F)D)} \quad \frac{\langle s_F \rangle D \vdash \sigma((s_F)D)}{\langle s_F \rangle C \land (s_F)D \vdash \sigma((s_F)C) \land \sigma((s_F)D)}
\]

and
\[
\begin{array}{c}
\sigma((s_P)C) \vdash \langle s_P \rangle C \\
\langle s_P \rangle \sigma((s_P)C) \vdash C \\
\langle s_P \rangle \sigma((s_P)D) \vdash D
\end{array}
\]

\[
\begin{array}{c}
\vdash (\langle s_P \rangle \sigma((s_P)C) \land (\langle s_P \rangle \sigma((s_P)D)) \vdash (\langle s_P \rangle (C \land D)) \\
\langle s_P \rangle (\langle s_P \rangle \sigma((s_P)C) \land (\langle s_P \rangle \sigma((s_P)D)) \vdash (\langle s_P \rangle (C \land D))
\end{array}
\]

\[
\begin{array}{c}
\vdash (\langle s_P \rangle \sigma((s_P)C) \land (\langle s_P \rangle \sigma((s_P)D)) \vdash (\langle s_P \rangle (C \land D)) \\
\vdash (\langle s_P \rangle \sigma((s_P)C) \land (\langle s_P \rangle \sigma((s_P)D)) \vdash (\langle s_P \rangle (C \land D))
\end{array}
\]

and if \(B\) is \(C \lor D\) then the proof is analogous.

If \(B\) is \(C \rightarrow D\) then:

\[
\begin{array}{c}
\sigma((s_P)C) \vdash \langle s_P \rangle C \\
\sigma((s_P)D) \vdash \langle s_P \rangle D \\
\sigma((s_P)C) \rightarrow \sigma((s_P)D) \vdash (\langle s_P \rangle (C \rightarrow D)) \\
\vdash (\langle s_P \rangle (C \rightarrow D)) \vdash (\langle s_P \rangle (C \rightarrow D))
\end{array}
\]

\[
\begin{array}{c}
\sigma((s_P)C) \rightarrow \sigma((s_P)D) \vdash (\langle s_P \rangle (C \rightarrow D)) \\
\sigma((s_P)C) \rightarrow \sigma((s_P)D) \vdash (\langle s_P \rangle (C \rightarrow D)) \\
\sigma((s_P)C) \rightarrow \sigma((s_P)D) \vdash (\langle s_P \rangle (C \rightarrow D))
\end{array}
\]

If \(B\) is \(\langle y \rangle C\):

\[
\begin{array}{c}
(\epsilon y \sigma((s_{\varphi})))C \vdash \sigma((s_{\varphi}, s_P)C) \\
(\epsilon y \sigma((s_{\varphi})))C \vdash (\epsilon y) \sigma((s_{\varphi}, s_P)C) \\
(\epsilon y) \sigma((s_{\varphi}, s_P)C) \vdash (\epsilon y) \sigma((s_{\varphi}, s_P)C) \\
(\epsilon y) \sigma((s_{\varphi}, s_P)C) \vdash (\epsilon y) \sigma((s_{\varphi}, s_P)C)
\end{array}
\]

and
Lemma 17. For every formula \( t \) in \( \mathcal{L}_F \), the sequents \( \kappa(F, A) \vdash \kappa(A) \) are derivable in \( \mathcal{DFO} \).

Proof. We proceed by induction on the complexity of \( A \). If \( A = R(t) \) then \( \kappa(F, A) = A \) and we are done. If \( A = B \wedge C \), notice that \( \text{FV}(A) = \text{FV}(B) = \text{FV}(C) = F \). then:

\[
\begin{align*}
\kappa(F, B) \vdash B & \quad \kappa(F, \neg A) \vdash \neg A \\
\kappa(F, B) \wedge \kappa(F, A) & \vdash B \wedge C \\
\kappa(F, \neg A) \wedge \kappa(F, A) & \vdash \neg B \wedge C \\
\kappa(F, \neg A) \wedge \kappa(F, B) & \vdash \neg B \wedge \neg A
\end{align*}
\]

We work similarly for the other direction and for \( \to \) and \( \vee \).

If \( A = \langle x \rangle B \) then \( \text{FV}(B) = F \cup \{ x \} \):
\[
\begin{align*}
\kappa(F \cup \{x\}, B^\tau) & \vdash B \\
(\exists z)\kappa(F \cup \{x\}, B^\tau) & \vdash (x)B \\
(\exists x)\kappa(F \cup \{x\}, B^\tau) & \vdash (x)B
\end{align*}
\]

and likewise for the other direction.

Finally assume that \( A \) is of the form \((x_1) \cdots (x_k)B\) for some \( k \in \omega \), where \( B \) is not of the form \((z)C\). We proceed by induction on the complexity of \( B \). We will show that \((\exists z) \cdots (\exists z_k)B \vdash \kappa(F, ((x_1) \cdots (x_k)B)^\tau)\) and \(\kappa(F, ((x_1) \cdots (x_k)B)^\tau) \vdash (\exists z) \cdots (\exists z_k)B\). Notice preliminarily that \((x_1) \cdots (x_k)B = B^\tau\) for all \( B \).

If \( B \) is \( R(\tau) \) then \( \kappa(F, B) = (x_1) \cdots (x_k)B \):
\[
\begin{align*}
\tilde{R}(\tau_x) & \vdash \kappa(F, B) \\
\kappa(F, B) & \vdash \tilde{R}(\tau_x)
\end{align*}
\]

and similarly for the other direction.

If \( B \) is \( C \land D \):
\[
\begin{align*}
(\exists x_1) \cdots (\exists x_k)C & \vdash \kappa(F, C^\tau) \\
(\exists x_1) \cdots (\exists x_k)D & \vdash \kappa(F, D^\tau)
\end{align*}
\]

and
\[
\begin{align*}
(\exists x_1) \cdots (\exists x_k)(C \land D) & \vdash \kappa(F, C^\tau \land \kappa(F, D^\tau))
\end{align*}
\]

and
\[
\begin{align*}
\kappa(F, C^\tau) & \vdash (\exists x_1) \cdots (\exists x_k)C \\
\kappa(F, D^\tau) & \vdash (\exists x_1) \cdots (\exists x_k)D \\
(\exists x_1) \cdots (\exists x_k)(C \land D) & \vdash \kappa(F, C^\tau \land \kappa(F, D^\tau))
\end{align*}
\]

cadj \times 2k
\[
\begin{align*}
(\exists x_1) \cdots (\exists x_k)(C \land D) & \vdash \kappa(F, C^\tau \land \kappa(F, D^\tau))
\end{align*}
\]

and if \( B \) is \( C \lor D \) and \( C \rightarrow D \) we argue similarly (see also proof of Lemma \[16\]).

If \( B \) is \((x)C\), let us assume without loss of generality that \( x_1 = x \)\[11\]

\[
\begin{align*}
(\exists x_1) \cdots (\exists x_k)C & \vdash \kappa(F, C^\tau) \\
(\exists x_1) \cdots (\exists x_k)C & \vdash (x)\kappa(F, C^\tau) \\
(\exists x_1) \cdots (\exists x_k)(\exists \tilde{x})C & \vdash (\tilde{x})\kappa(F, C^\tau) \\
(\exists x_1) \cdots (\exists x_k)(\exists \tilde{x})(\exists x_2)C & \vdash (\tilde{x})\kappa(F, C^\tau) \\
(\exists x_1) \cdots (\exists x_k)(\exists \tilde{x})(\exists x_2)(x)C & \vdash (\tilde{x})(x)\kappa(F, C^\tau) \\
(\exists x_1) \cdots (\exists x_k)(\exists \tilde{x})(\exists x_2)(x)C & \vdash (\tilde{x})(x)\kappa(F, C^\tau)
\end{align*}
\]

\[11\]Notice that if \( x \notin \{x_1, \ldots, x_k\} \) the last two steps of the derivations are redundant.
and

$$\kappa(F, C^\tau) \vdash (\bar{x}_2) \cdots (\bar{x}_k)C$$

\[
\begin{align*}
(\bar{x}_k) \cdots (\bar{x}_2) \kappa(F, C^\tau) & \vdash C \\
(\bar{x}_k) \cdots (\bar{x}_2) \kappa(F, C^\tau) & \vdash (x)C \\
\kappa(F, C^\tau) & \vdash (\bar{x}_2) \cdots (\bar{x}_k)C \\
\kappa(F, C^\tau) & \vdash C \\
(\bar{x})(\bar{x}_k) \cdots (\bar{x}_2) \kappa(F, C^\tau) & \vdash (x)C \\
(x) \kappa(F, C^\tau) & \vdash (\bar{x}_2) \cdots (\bar{x}_k)C \\
(\bar{x})(\bar{x}_2) \cdots (\bar{x}_k) & \vdash C \\
(\bar{x})(\bar{x}_2) \cdots (\bar{x}_k) & \vdash C
\end{align*}
\]

For $$\forall x C$$ the proof is analogous. This concludes the proof.  

\[\square\]

**Corollary 18.** For any formula $$A \in \mathcal{L}_F$$ the sequents $$A \vdash \kappa(F, A^\tau)$$ and $$\kappa(F, A^\tau) \vdash A$$ are derivable in D.FO.

**Proof.** By Lemmas \[16\] and \[17\] the sequents $$A \vdash \sigma(A)$$ and $$\sigma(A) \vdash \kappa(F, \sigma(A)^\tau)$$ are derivable in D.FO. Hence, by applying Cut, $$A \vdash \kappa(F, \sigma(A)^\tau)$$ is derivable in D.FO. Finally, by Lemma \[15\] $$\sigma(A)^\tau = A^\tau$$. This concludes the proof of the first part of the statement. The second part is argued analogously.  

\[\square\]

**Corollary 19.** If $$A, B \in \mathcal{L}_F$$ and $$A^\tau = B^\tau$$ then $$A \vdash B$$ is derivable in D.FO.

**Proof.** By Corollary \[18\] $$A \vdash \kappa(F, A^\tau)$$ and $$\kappa(F, B^\tau) \vdash B$$ are derivable. Since $$A^\tau = B^\tau$$, we derive $$A \vdash B$$ from these two sequents by applying Cut.  

The corollary above implies that in order to prove completeness in the sense specified in Theorem \[13\], it is enough to prove that, for every $$\mathcal{L}$$-formula $$A$$ which is a theorem of first-order logic and every $$F \in \mathcal{P}_u(Var)$$, there exists some $$A' \in \mathcal{L}_F$$ with $$(A')^\tau = A$$ such that $$\vdash_F A'$$ is provable in D.FO. In what follows, we will provide the required derivations.

If $$A$$ is a propositional tautology then $$\kappa(F, A)$$ is derivable using the propositional fragment of D.FO. The derivations are omitted. If $$A$$ is of the form $$\forall x(B \rightarrow C) \rightarrow (\forall xB \rightarrow \forall xC)$$ then let $$D := \kappa(F \cup \{x\}, B)$$ and $$E := \kappa(F \cup \{x\}, C)$$:

$$\frac{D \vdash D}{[x]D \vdash [x]D}$$

$$\frac{[x]D \vdash [x]D}{E \vdash E}$$

$$\frac{D \rightarrow E \vdash [\bar{x}]x[D \rightarrow E]}{\frac{[x](D \rightarrow E) \vdash [\bar{x}]x[D \rightarrow E]}{[\bar{x}]x[D \rightarrow E] \vdash [\bar{x}]x[D \rightarrow E]}}$$

$$\frac{[x]D \wedge [x](D \rightarrow E) \vdash E}{[x]D \wedge [x](D \rightarrow E) \vdash [\bar{x}]E}$$

$$\frac{[x]D \wedge [x](D \rightarrow E) \vdash [\bar{x}]E}{[x]D \wedge [x](D \rightarrow E) \vdash [\bar{x}]E}$$

$$\frac{[x](D \rightarrow E) \vdash [\bar{x}]D \rightarrow [\bar{x}]E}{[x](D \rightarrow E) \vdash [\bar{x}]D \rightarrow [\bar{x}]E}$$

$$\frac{\vdash_F [x](D \rightarrow E) \rightarrow ([x]D \rightarrow [x]E)}{\vdash_F [x](D \rightarrow E) \rightarrow ([x]D \rightarrow [x]E)}$$

23
which yields the desired result given that \((|x|(D \rightarrow E) \rightarrow (|x|D \rightarrow |x|E))^\tau = A\). In case \(x \in F\) we apply the introduction of \((\tilde{z})\) in the penultimate step.

If \(A\) is of the form \(B \rightarrow \forall xB\) where \(x \notin \text{FV}(B)\), then let \(\kappa(F \setminus \{x\}, B) = C:\)

\[
\begin{align*}
C & \vdash C \\
(\tilde{z})C & \vdash (\tilde{z})C \\
(\tilde{z})C & \vdash (x)C \\
C & \vdash [x](x)C \\
C & \vdash [x]C \\
\vdash_F C & \Rightarrow [x](x)C \\
\vdash_F C & \Rightarrow [x]C
\end{align*}
\]

which yields the desired result given that \((C \rightarrow [x](x)C)^\tau = A\). If \(A\) is of the form \(\forall xB \rightarrow B(t/x)\), where \(t\) is free for \(x\) in \(B\) then let \(C = \kappa(FV(B), B)\) and \((\tilde{t}_x, \tilde{y}_B)\) for \(y = FV(B) \setminus \{x\}\), and \((\tilde{\pi}_y)\) is the identity substitution. We proceed by cases. If \(x \in FV(B)\):

\[
\begin{align*}
C & \vdash C \\
(\tilde{t}_x, \tilde{\pi}_y)C & \vdash (\tilde{t}_x, \tilde{\pi}_y)C \\
(\tilde{t}_x, \tilde{\pi}_y)(\tilde{z})C & \vdash (\tilde{t}_x, \tilde{\pi}_y)(\tilde{z})C \\
(\tilde{z})(\tilde{t}_x, \tilde{\pi}_y)(\tilde{z})C & \vdash (\tilde{t}_x, \tilde{\pi}_y)(\tilde{z})C \\
(\tilde{z})C & \vdash [x](\tilde{t}_x, \tilde{\pi}_y)(\tilde{z})C \\
C & \vdash [x]C \\
\vdash_F C & \Rightarrow [x](\tilde{t}_x, \tilde{\pi}_y)(\tilde{z})C \\
\vdash_F C & \Rightarrow [x](\tilde{t}_x, \tilde{\pi}_y)C
\end{align*}
\]

which yields the desired result given that \((\tilde{z})(\tilde{t}_x, \tilde{\pi}_y)(\tilde{z})C \rightarrow (\tilde{t}_x, \tilde{\pi}_y)C)^\tau = A\). If \(x \notin \text{FV}(B)\):

\[
\begin{align*}
C & \vdash C \\
(\tilde{z})C & \vdash (\tilde{z})C \\
(\tilde{z})(\tilde{t}_x, \tilde{\pi}_y)(\tilde{z})C & \vdash (\tilde{t}_x, \tilde{\pi}_y)(\tilde{z})C \\
(\tilde{t}_x, \tilde{\pi}_y)(\tilde{z})C & \vdash (\tilde{t}_x, \tilde{\pi}_y)(\tilde{z})C \\
(\tilde{z})C & \vdash [x](\tilde{t}_x, \tilde{\pi}_y)(\tilde{z})C \\
C & \vdash [x]C \\
\vdash_F C & \Rightarrow [x](\tilde{t}_x, \tilde{\pi}_y)(\tilde{z})C \\
\vdash_F C & \Rightarrow [x](\tilde{t}_x, \tilde{\pi}_y)C
\end{align*}
\]

which yields the desired result given that \((\tilde{z})(\tilde{t}_x, \tilde{\pi}_y)(\tilde{z})C \rightarrow (\tilde{t}_x, \tilde{\pi}_y)C)^\tau = A\).

As for the equality axioms we have

\[
\begin{align*}
t \vdash t & \vdash t \\
\vdash_F t, t & \vdash t \vdash t \\
\vdash_F t & \vdash t \vdash t
\end{align*}
\]
Let $A$ be a formula with free variable $v$, let \( \{ x_1, \ldots, x_n \} = \text{FV}(A) \setminus \text{FV}(t, s), \)
\( \{ y_1, \ldots, y_m \} = \text{FV}(s) \setminus \text{FV}(t, A) \) and \( \{ z_1, \ldots, z_k \} = \text{FV}(t) \setminus \text{FV}(s, A) \), and let’s assume without loss of generality that $v \notin \text{FV}(s, t)$. Let

\[
B = (y_1) \cdots (y_m)(z_1) \cdots (z_k)A.
\]

Notice that $A = B^\gamma$. We have

\[
\frac{B \vdash B}{(s, \tau_1, \tau_2, \tau_3)B \vdash (s, \tau_1, \tau_2, \tau_3)B} \quad \frac{(t, \tau_1, \tau_2, \tau_3)B \vdash (s, \tau_1, \tau_2, \tau_3)B}{s_L} \quad \frac{(t, \tau_1, \tau_2, \tau_3)B \vdash (s, \tau_1, \tau_2, \tau_3)B}{s_{\text{R}}}
\]

which yields the desired result given that

\[
(s_1) \cdots (s_n)A = s \to ((s_1, \tau_1, \tau_2, \tau_3)B \to (s_2, \tau_1, \tau_2, \tau_3)B) = (s = t \to (A(s/y) \to A(t/y))).
\]

Finally using necessitation we obtain the universal closure of tautologies by repeated application of the following pattern:

\[
\frac{
\vdash_{F_U}(x) \vdash A
}{
\vdash_{F_U} \vdash A
} \quad \frac{
\vdash_{F_U} \vdash A
}{
\vdash_{F_U} \vdash [x]A
} \quad \frac{
\vdash_{F_U} \vdash [x]A
}{
\vdash_{F_U} \vdash [x]A
}
\]

This concludes the proof of completeness.

\section{Conservativity}

To argue that the calculus D.FO conservatively extend the axiomatic system $\Gamma_{\text{FO}}$, we follow the standard proof strategy discussed in [15][17]. We need to show that, for all formulas $A$ of $\mathcal{L}$, if $F_{\gamma} \vdash F \kappa(F, A)$ is derivable in D.FO, then $\Gamma_{\text{FO}} A$. This claim can be proved using the following facts: (a) the rules of D.FO are sound w.r.t. the semantics of heterogeneous $\mathcal{L}$-algebras (cf. Section 6.1); (b) The Hilbert-style system $\Gamma_{\text{FO}}$ is complete w.r.t. first-order logic models (cf. Section 2); and (c) first-order models are equivalently presented as heterogeneous algebras (cf. Section 3), so that the semantic consequence relations arising from each type of structures preserve and reflect the translation (cf. Proposition 11). Then, let $A$ be an $\mathcal{L}$-formula. If $F_{\gamma} \vdash F \kappa(F, A)$ is derivable in D.FO, then
by (a), $\mathcal{H}, V \models \top \leq_{F} \kappa(F, A)$, for every heterogeneous algebraic $\mathcal{L}$-model $(\mathcal{H}, V)$. By (c), this implies that $M \models A$, where $M$ is a first order model. By (b), this implies that $\vdash_{\text{FO}} A$, as required.

### 6.4 Cut elimination and subformula property

In the present section, we outline the proof of cut elimination and subformula property for the calculus $\text{D.FO}$ of Section 5. As discussed earlier on, the design of this calculus allows for its cut elimination and subformula property to be inferred from a metatheorem, following the strategy introduced by Belnap for display calculi. The metatheorem to which we will appeal is [6, Theorem 4.1] (cf. [6, Section 3]) for the class of multi-type calculi, of which $\text{D.FO}$ is a particularly well-behaved element, since it enjoys the full display property. For this reason, $\text{D.FO}$ satisfies the following more restricted version of condition $C''''_{5}$:

$C''''_{5}$: **Closure of axioms under cut.** If $x \vdash a$ and $a \vdash y$ are axioms, then $x \vdash y$ is again an axiom.

By this metatheorem, it is enough to verify that $\text{D.FO}$ meets the conditions listed in [6, Section 3] with $C''''_{5}$ modified as indicated above. All conditions except $C_{8}$ are readily satisfied by inspecting the rules. In what follows we verify $C_{8}$. We only treat the cases of the atomic formulas and the heterogeneous connectives:

**Atomic formulas:** We have:

\[
\begin{array}{c}
\vdash \pi_{1} \quad \vdash \pi_{1} \\
\frac{\hat{R}(t_{\pi}) \vdash R(t_{\pi}) \vdash Y}{\hat{R}(t_{\pi}) \vdash Y} \quad \frac{\hat{R}(t_{\pi}) \vdash R(t_{\pi}) \vdash Y}{\hat{R}(t_{\pi}) \vdash Y}
\end{array}
\]

**Quantifiers and their adjoint:**

\[
\begin{array}{c}
\vdash \pi_{1} \quad \vdash \pi_{2} \\
\frac{X \vdash A}{(\hat{x})X \vdash (\hat{x})A} \quad \frac{(\hat{x})A \vdash Y}{(\hat{x})A \vdash Y} \\
\frac{(\hat{x})X \vdash Y}{\vdash \pi_{2}} \quad \frac{(\hat{x})A \vdash Y}{\vdash \pi_{2}}
\end{array}
\]

\[
\begin{array}{c}
\vdash \pi_{1} \quad \vdash \pi_{2} \\
\frac{X \vdash \{\hat{x}\}A}{\vdash \pi_{2}} \quad \frac{A \vdash Y}{A \vdash Y} \\
\frac{X \vdash \{\hat{x}\}A}{\vdash \pi_{2}} \quad \frac{X \vdash \{\hat{x}\}Y}{\vdash \pi_{2}}
\end{array}
\]
The cases for \( (x) \) is done similarly to the one above.

7 Conclusions and further directions

**Contributions.** In this article we have introduced a proper multi-type display calculus for classical first-order logic and shown that it is sound, complete, conservative with respect to the standard classical first-order logic and enjoys cut elimination and the subformula property. We intended to capture first-order logic with a calculus in which quantifiers are represented also at the structural level and rules are closed under uniform substitution. We achieved this by developing an idea of Wansing’s, that the proof-theoretic treatment of quantifiers can emulate that of modal operators, in a multi-type setting in which formulas with different sets of free variables have different types. This multi-type environment is supported semantically by certain classes of heterogeneous algebras in which the maps interpreting the existential and universal quantifiers are the left and the right adjoint respectively of one and the same injective Boolean algebra homomorphism. The same adjunction pattern accounts for the semantics of substitution.

**Correspondence theory for first-order logic.** The semantic analysis in Section 3 can be regarded as a multi-type and ALBA-powered version of the correspondence results observed in [34]. Thanks to the systematic connections established between unified correspondence theory and the theory of analytic display calculi, we are now in a position to apply the results and insights of unified correspondence to the semantic environment of Section 3 and systematically exploit them for proof-theoretic purposes. This might turn out to be a useful tool in the analysis of generalized quantifiers (see discussion below).

**A modular environment.** Thanks to the fact that, in the environment of this calculus, both substitutions and quantifiers are explicitly represented as logical and structural connectives, we can now explore systematically the space of their properties and their possible interactions. For instance, the rules sqR and sqL express in a transparent and explicit way the book-keeping concerning variable capturing in first-order logic. More interestingly, this environment allows for a finer-grained analysis of fundamental interactions between quantifiers and intensional connectives. For instance, in the present calculus the rules in Definition 12.15 encode the fact that the cylindrification maps are Boolean algebra homomorphisms, which in turn captures the fact that classical propositional connectives are all extensional. If we change the propositional base to e.g. intuitionistic or bi-intuitionistic logic, or if we expand classical first-order logic with modal connectives it is desirable to allow for some additional flexibility.
For instance the constant domain axiom $\forall x (B(x) \lor A) \rightarrow ((\forall x B(x)) \lor A)$ can be captured by the analytic structural rule below on the left hand side, which is interderivable with the one on the right hand side:

$$
\frac{X \vdash [\bar{x}] (Y \lor \bar{(x)Z})}{X \vdash [\bar{x}] Y \lor Z}
\frac{(\bar{x})X \vdash (\bar{x})Y \vdash Z}{(\bar{x})(X \vdash (Y) \lor Z) \vdash Z}
$$

Indeed:

$$
\frac{X \vdash [\bar{x}] (Y \lor (x)Z)}{(\bar{x})X \vdash (\bar{x})Y \vdash Z}
\frac{(\bar{x})Z \vdash (\bar{x})X \vdash Y}{Y \vdash (\bar{x})(Z \lor (\bar{x})X)}
\frac{(\bar{x})(Z \lor (\bar{x})X) \vdash Y}{Y \vdash (\bar{x})Z \lor X}
\frac{Z \vdash [\bar{x}] Y}{X \vdash [\bar{x}] Y \lor Z}
$$

Notice that in the hypersequent calculus for bi-intuitionistic first-order logic the constant domain axiom is derivable from the Mix rule, capturing the prelinearity axiom $(A \rightarrow B) \lor (B \rightarrow A)$. In the context of this calculus the prelinearity axiom corresponds to the rule

$$
\frac{X \vdash Y \quad W \vdash Z}{\top_F \vdash (X \rightarrow Z) \lor (W \rightarrow Y)}
$$

we conjecture that in the present framework the constant domain axiom is not derivable in D.FO without the two rules in (*) and with the prelinearity rule above.

**A modular environment?** Notwithstanding the increased modularity that this calculus allows, we feel that something more can be done. For instance, the heterogeneous modal operators use individual variables and terms as parameters, much in the same way in which actions and agents were used as parameters in the first versions of the display calculus for DEL (cf. [13, 8]). While this choice is unproblematic in respect to the cut elimination and actually allows to retrieve it from the metatheorem of [6], it is also practically cumbersome, since it constrains us to admit as axioms sequents which are not useful when deriving actual formulas (indeed substitution axioms in which structural adjoints $\langle \hat{t}_p \rangle$ and $[\hat{t}_p]$ occur on both sides of the sequent cannot possibly occur in derivations that conclusions of which are pure formulas), and moreover recognizing whether such a sequent is an axiom requires lengthy calculations which should be performed internally to the calculus rather than externally. This is the focus of current investigation. Indeed having axioms and rules such as the following

$$
x \equiv x \quad c \equiv c \quad t_1 \equiv t_2 \quad t_1 \equiv t_3
t_1 \equiv s_1 \quad \ldots \quad t_n \equiv s_n \quad t \equiv s \quad t \equiv s
$$

would be much neater. This choice would also be compatible with rules such as

$$
\frac{(\bar{x})(X \hat{\lor} Y) \vdash Z}{(\bar{x})X \hat{\lor} (\bar{x})Y \vdash Z}
$$
which are completely unproblematic when \( x \) is a parameter as in the present setting but would violate the condition C5 against proliferation when considered a term. However, the metatheorem of [6] allows proliferation limited to “flat” types, i.e. types the only rules of which are identity and cut. We are working towards a solution which allows to integrate variables and terms as types.

**Generalized quantifiers.** The semantic analysis of Section 3 shows that quantifiers and substitutions correspond to a very restricted class of maps \( f : M^S \to M^T \). Therefore it is natural to ask whether different notions of quantification and substitutions can be investigated in connection with larger or different classes of such functions. This study could be relevant to the analysis of generalized quantifiers in natural language semantics [1, 20, 26, 27, 38], to dependence and independence logic [33, 11, 21] and to the analysis of different notions of substitutions [30, 31, 32].

**References**

[1] Jon Barwise and Robin Cooper. Generalized quantifiers and natural language. In *Philosophy, Language, and Artificial Intelligence*, pages 241–301. Springer, 1981.

[2] Nuel Belnap. Display logic. *Journal of Philosophical Logic*, 11:375–417, 1982.

[3] Agata Ciabattoni, Reavantha Ramanayake, and Heinrich Wansing. Hypersequent and display calculi–a unified perspective. *Studia Logica*, 102(6):1245–1294, 2014.

[4] Willem Conradie and Alessandra Palmigiano. Algorithmic correspondence and canonicity for non-distributive logics. Submitted. ArXiv: 1603.08515.

[5] H.B. Enderton. *A Mathematical Introduction to Logic*. Elsevier, 2001.

[6] S. Frittella, G. Greco, A. Kurz, A. Palmigiano, and V. Sikimić. Multi-type sequent calculi. In Janusz Kaczmarek Michal Zawidzki Andrzej Indrzejczak, editor, *Trends in Logic XIII*, pages 81–93. Lodź University Press, 2014.

[7] Sabine Frittella, Giuseppe Greco, Alexander Kurz, Alessandra Palmigiano, and Vlasta Sikimić. Multi-type display calculus for dynamic epistemic logic. *Journal of Logic and Computation*, 26(6):2017–2065, 2016.

[8] Sabine Frittella, Giuseppe Greco, Alexander Kurz, Alessandra Palmigiano, and Vlasta Sikimić. A proof-theoretic semantic analysis of dynamic epistemic logic. *Journal of Logic and Computation*, 26(6):1961–2015, 2016.

[9] Sabine Frittella, Giuseppe Greco, Alessandra Palmigiano, and Fan Yang. A multi-type calculus for inquisitive logic. In J. Väanänen, Å. Hirvonen, and R. de Queiroz, editors, *Logic, Language, Information, and Computation*, LNCS 9803, pages 215–233. Springer, 2016.

[10] Robert Goldblatt. *Topoi: the categorial analysis of logic*, volume 98. Elsevier, 2014.
[11] Erich Grädel and Jouko Väänänen. Dependence and independence. *Studia Logica*, 101(2):399–410, 2013.

[12] Giuseppe Greco, Peter Jipsen, Fei Liang, Alessandra Palmigiano, and Apostolos Tzimoulis. Algebraic proof theory for LE-logics. [arXiv:1808.04642](https://arxiv.org/abs/1808.04642), submitted, 2019.

[13] Giuseppe Greco, Alexander Kurz, and Alessandra Palmigiano. Dynamic epistemic logic displayed. In Huaxin Huang, Davide Grossi, and Olivier Roy, editors, *Proceedings of the 4th International Workshop on Logic, Rationality and Interaction (LORI-4)*, volume 8196 of *LNCS*, 2013.

[14] Giuseppe Greco, Fei Liang, M. Andrew Moshier, and Alessandra Palmigiano. Multi-type display calculus for semi De Morgan logic. *Logic, Language, Information, and Computation: 24th International Workshop, WoLLIC 2017, London, UK, July 18-21, 2017, Proceedings*, pages 199–215, 2017.

[15] Giuseppe Greco, Minghui Ma, Alessandra Palmigiano, Apostolos Tzimoulis, and Zhiguang Zhao. Unified correspondence as a proof-theoretic tool. *Journal of Logic and Computation*, 2016. doi: 10.1093/logcom/exw022. ArXiv preprint 1603.08204.

[16] Giuseppe Greco and Alessandra Palmigiano. Linear logic properly displayed. Submitted. [ArXiv preprint 1611.04181](https://arxiv.org/abs/1611.04181).

[17] Giuseppe Greco and Alessandra Palmigiano. Linear logic properly displayed. Submitted. [arXiv:1611.04181](https://arxiv.org/abs/1611.04181).

[18] Paul R Halmos. Polyadic boolean algebras. *Proceedings of the National Academy of Sciences*, 40(5):296–301, 1954.

[19] L. Henkin, J.D. Monk, and A. Tarski. *Cylindric algebras*. Number pt. 2 in Cylindric Algebras. North-Holland Publishing Company, 1971.

[20] Jaakko Hintikka. Quantifiers vs. quantification theory. *Linguistic inquiry*, 5(2):153–177, 1974.

[21] Juha Kontinen and Jouko Väänänen. Axiomatizing first-order consequences in dependence logic. *Annals of Pure and Applied logic*, 164(11):1101–1117, 2013.

[22] Steven T Kuhn. Quantifiers as modal operators. *Studia Logica*, 39(2-3):145–158, 1980.

[23] F William Lawvere. Functorial semantics of elementary theories. In *Journal of symbolic logic*, volume 31, page 294, 1966.

[24] F. William Lawvere. Adjointness in foundations. *Dialectica*, 1969.

[25] F. William Lawvere. Equality in hyperdoctrines and comprehension schema as an adjoint functor. *Proceedings of the AMS Symposium on Pure Mathematics XVII*, pages 1–14, 1970.
[26] David Lewis. Adverbs of quantification. *Formal semantics*, pages 178–188, 1975.

[27] Per Lindström. First order predicate logic with generalized quantifiers. *Theoria*, 32(3):186–195, 1966.

[28] Richard Montague. Logical necessity, physical necessity, ethics, and quantifiers. *Inquiry*, 3(1-4):259–269, 1960.

[29] Sara Negri and Jan von Plato. *Structural Proof Theory*. Cambridge University Press, Cambridge, 2001.

[30] Andrew M Pitts. *Nominal sets: Names and symmetry in computer science*. Cambridge University Press, 2013.

[31] Dana Scott. The algebraic interpretation of quantifiers. intuitionistic and classical. In Andrzej Mostowski and Foundational Studies, pages 289–312. IOS Press Amsterdam, 2008.

[32] Stanislaw Szawiel and Marek Zawadowski. Theories of analytic monads. *Mathematical Structures in Computer Science*, 24(6), 2014.

[33] Jouko Väänänen. *Dependence logic: A new approach to independence friendly logic*, volume 70. Cambridge University Press, 2007.

[34] Johan van Benthem. Modal foundations for predicate logic. *Logic Journal of the IGPL*, 5(2):259, 1997.

[35] Georg Henrik von Wright. On double quantification. *Commentationes physico-mathematicae*, 16(3), 1952.

[36] Heinrich Wansing. *Displaying modal logic*, volume 3. Springer, Trends in Logic, 1998.

[37] Heinrich Wansing. Predicate logics on display. *Studia Logica*, 62(1):49–75, 1999.

[38] Dag Westerståhl. Quantifiers in formal and natural languages. In *Handbook of philosophical logic*, pages 1–131. Springer, 1989.