

2016

# Multi-Type Display Calculus for Dynamic Epistemic Logic

Sabine Frittella  
*Aix-Marseille Université*


Giuseppe Greco  
*TU Delft*

Alexander Kurz  
*Chapman University, akurz@chapman.edu*

Alessandra Palmigiano  
*TU Delft*

Vlasta Sikimić  
*University of Belgrade*

Follow this and additional works at: [https://digitalcommons.chapman.edu/engineering\\_articles](https://digitalcommons.chapman.edu/engineering_articles)

 Part of the [Algebra Commons](#), [Logic and Foundations Commons](#), [Other Computer Engineering Commons](#), [Other Computer Sciences Commons](#), and the [Other Mathematics Commons](#)

---

## Recommended Citation

S. Frittella, G. Greco, A. Kurz, A. Palmigiano, and V. Sikimić, "Multi-type display calculus for dynamic epistemic logic," *Journal of Logic and Computation*, vol. 26, no. 6, pp. 2017–2065, Dec. 2016. DOI: 10.1093/logcom/exu068

This Article is brought to you for free and open access by the Fowler School of Engineering at Chapman University Digital Commons. It has been accepted for inclusion in Engineering Faculty Articles and Research by an authorized administrator of Chapman University Digital Commons. For more information, please contact [laughtin@chapman.edu](mailto:laughtin@chapman.edu).

---

# Multi-Type Display Calculus for Dynamic Epistemic Logic

## **Comments**

This is a pre-copy-editing, author-produced PDF of an article accepted for publication in *Journal of Logic and Computation* following peer review. The definitive publisher-authenticated version

S. Frittella, G. Greco, A. Kurz, A. Palmigiano, and V. Sikimić, “Multi-type display calculus for dynamic epistemic logic,” *Journal of Logic and Computation*, vol. 26, no. 6, pp. 2017–2065, Dec. 2016.

is available online at DOI: [10.1093/logcom/exu068](https://doi.org/10.1093/logcom/exu068)

## **Copyright**

Oxford University Press

# Multi-type Display Calculus for Dynamic Epistemic Logic

Sabine Frittella\* Giuseppe Greco<sup>†</sup> Alexander Kurz<sup>‡</sup>  
Alessandra Palmigiano<sup>§</sup> Vlasta Sikimić<sup>¶</sup>

May 22, 2018

## Abstract

In the present paper, we introduce a multi-type display calculus for dynamic epistemic logic, which we refer to as Dynamic Calculus. The display-approach is suitable to modularly chart the space of dynamic epistemic logics on weaker-than-classical propositional base. The presence of types endows the language of the Dynamic Calculus with additional expressivity, allows for a smooth proof-theoretic treatment, and paves the way towards a general methodology for the design of proof systems for the generality of dynamic logics, and certainly beyond dynamic epistemic logic. We prove that the Dynamic Calculus adequately captures Baltag-Moss-Solecki's dynamic epistemic logic, and enjoys Belnap-style cut elimination.

*Keywords:* display calculus, dynamic epistemic logic, modularity, multi-type system.

*Math. Subject Class. 2010:* 03B42, 03B20, 03B60, 03B45, 03F03, 03G10, 03A99.

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Preliminaries</b>	<b>4</b>
2.1	The logic of epistemic actions and knowledge . . . . .	4
2.2	Intuitionistic EAK . . . . .	7
2.3	Display calculi . . . . .	8
2.4	A single-type display calculus for EAK . . . . .	10

\*Laboratoire d'Informatique Fondamentale de Marseille (LIF) - Aix-Marseille Université.

<sup>†</sup>Department of Values, Technology and Innovation - TU Delft.

<sup>‡</sup>Department of Computer Science - University of Leicester.

<sup>§</sup>Department of Values, Technology and Innovation - TU Delft. The research of the second and fourth author has been made possible by the NWO Vidi grant 016.138.314, by the NWO Aspasia grant 015.008.054, and by a Delft Technology Fellowship awarded in 2013 to the fourth author.

<sup>¶</sup>The Institute of Philosophy of the Faculty of Philosophy - University of Belgrade.

<b>3</b>	<b>Multi-type calculi, and cut elimination metatheorem</b>	<b>13</b>
3.1	Multi-type calculi . . . . .	13
3.2	Quasi-properly displayable multi-type calculi . . . . .	14
3.3	Belnap-style metatheorem for multi-types . . . . .	16
<b>4</b>	<b>The Dynamic Calculus for EAK</b>	<b>19</b>
<b>5</b>	<b>Soundness</b>	<b>30</b>
<b>6</b>	<b>Completeness and cut elimination</b>	<b>33</b>
6.1	Derivable rules and completeness . . . . .	33
6.2	Belnap-style cut-elimination, and subformula property . . . . .	36
<b>7</b>	<b>Conservativity</b>	<b>37</b>
<b>8</b>	<b>Conclusions and further directions</b>	<b>41</b>
<b>9</b>	<b>Appendix</b>	<b>43</b>
9.1	Cut elimination . . . . .	43
9.2	Completeness . . . . .	45
<b>10</b>	<b>Cut Elimination for the Dynamic Calculus for EAK</b>	<b>52</b>

## 1 Introduction

**Motivation.** The range of nonclassical logics has been rapidly expanding, driven by influences from other fields which have opened up new opportunities for applications. The logical formalisms which have been developed as a result of this interaction have attracted the interest of a research community wider than the logicians, and their theory has been intensively investigated, especially w.r.t. their semantics and computational complexity.

However, most of these logics lack a comparable proof-theoretic development. More often than not, the hurdles preventing a standard proof-theoretic development for these logics are due precisely to the very features which make them suitable for applications, such as e.g. their not being closed under uniform substitution, or the existence of certain interactions between logical connectives, which cannot be expressed within the language itself.

A case in point is Baltag-Moss-Solecki’s logic of epistemic actions and knowledge (EAK), which is the main focus of the present paper. The Hilbert-style presentation of EAK prominently features non schematic axioms such as

$$[\alpha]p \leftrightarrow (Pre(\alpha) \rightarrow p),$$

where the variable  $p$  ranges over atomic propositions, and  $Pre(\alpha)$  is a meta-linguistic abbreviation for an arbitrary formula, and axioms such as

$$[\alpha][a]A \leftrightarrow (Pre(\alpha) \rightarrow \bigwedge \{[a][\beta]A \mid \alpha a \beta\}),$$

in which the extra-linguistic label  $\alpha a \beta$  expresses the fact that actions  $\alpha$  and  $\beta$  are indistinguishable for agent  $a$ .

Difficulties posed by features such as these caused the existing proposals of calculi in the literature to be often ad hoc, not easily generalizable e.g. to other logics, and more in general lacking a smooth proof-theoretic behaviour. In particular, the difficulty in smoothly transferring results from one logic to another is a problem in itself, since logics such as EAK typically come in large families. Hence, proof-theoretic approaches which uniformly apply to each logic in a given family are in high demand (for an expanded discussion of the existing proof systems for dynamic epistemic logics, see [15, Section 3]).

The problem of the transfer of results, tools and methodologies has been addressed in the proof-theoretic literature for the families of substructural and modal logics, and has given rise to the development of several generalizations of Gentzen sequent calculi (such as hyper-, higher level-, display- or labelled-sequent calculi).

**Contribution.** The present paper focuses on the core technical aspects of a proof-theoretic methodology and set-up closely linked to Belnap’s display calculi [3]. Specifically, our main contribution is the introduction of a methodology for the design of display calculi based on *multi-type languages*. In the case study provided by EAK, we start by observing that having to resort to the label  $\alpha a \beta$  is symptomatic of the fact that the language of EAK lacks the necessary expressivity to autonomously capture the piece of information encoded in the label.

In order to provide the desired additional expressivity, we introduce a language in which not only formulas are generated from formulas and actions (as it happens in the symbol  $\langle a \rangle A$ ) and formulas are generated from formulas and agents (as it happens in the symbol  $\langle a \rangle A$ ), but also *actions* are generated from the interaction between *agents* and *actions*, which is precisely what the label  $\alpha a \beta$  is about.

In the multi-type language for EAK introduced in the present paper, each generation step mentioned above is explicitly accounted for via special connectives taking arguments of *different types*. In principle, more than one alternative is possible in this respect; our choice for the present setting consists of the following types: **Ag** for agents, **Fnc** for functional actions, **Act** for actions, and **Fm** for formulas. Hence, the present setting introduces a separation between functional, i.e. deterministic actions, of type **Fnc**, and possibly nondeterministic actions, of type **Act** (see discussion at the end of section 4).

The proposed calculus provides an interesting and in our opinion very promising *methodological* platform towards the uniform development of a general proof-theoretic account of all dynamic logics, and also, from a purely structurally proof-theoretic viewpoint, for clarifying and sharpening the formulation of criteria lead-

ing to the statement and proof of meta-theoretic results such as Belnap-style cut-elimination (see Section 8).

**Structure of the paper.** In Section 2, we collect the relevant preliminaries on EAK, display calculi, and the (single-type) display calculus  $D'$ .EAK. In Section 3.1, we sketch the general features of the environment of multi-type display calculi, extend Wansing's definition of properly displayable calculi to the multi-type setting, and prove the corresponding extension of Belnap's cut elimination metatheorem. In Section 4, we propose a novel display calculus for EAK, which we refer to as *Dynamic Calculus*, and which concretely exemplifies the notion of multi-type display calculus. In Sections 5-7, we prove that the Dynamic Calculus adequately captures EAK, and enjoys Belnap-style cut elimination. In Section 8, we collect some conclusions and indicate further directions. The routine proofs and derivations are collected in Section 9, the appendix.

## 2 Preliminaries

In the present section, we collect the needed preliminaries: in 2.1, we review the logic of epistemic actions and knowledge. Our presentation slightly departs from [2], and closely follows [20, 18].<sup>1</sup> In 2.2, we briefly review the intuitionistic version of EAK, the axiomatization of which is directly captured in the rules of the calculus introduced in Section 4. In 2.3, we sketch the main relevant features of display calculi. In 2.4, we briefly report on the (single-type) display calculus for EAK introduced in [15].

### 2.1 The logic of epistemic actions and knowledge

The logic of epistemic actions and knowledge (further on EAK) is a logical framework which combines a multi-modal classical logic with a dynamic-type propositional logic. Static modalities in EAK are parametrized with agents, and their intended interpretation is epistemic, that is,  $\langle a \rangle A$  intuitively stands for 'agent  $a$  thinks that  $A$  might be the case'. Dynamic modalities in EAK are parametrized with epistemic *action-structures* (defined below) and their intended interpretation is analogous to that of dynamic modalities in e.g. Propositional Dynamic Logic. That is,  $\langle \alpha \rangle A$  intuitively stands for 'the action  $\alpha$  is executable, and after its execution  $A$  is the case'. Informally, action structures loosely resemble Kripke models, and encode information about epistemic actions such as e.g. public announcements, private announcements to a group of agents, with or without (actual or suspected) wiretapping, etc. Action structures consist of a finite nonempty domain of action-states, a designated state, binary relations on the domain for each agent, and a

---

<sup>1</sup>The account of EAK developed in [20, 18] is specifically tailored to facilitate the dual characterization at the base of the definition of the intuitionistic counterparts of EAK, which the calculus introduced in Section 4 takes as basic. So for the sake of a tighter presentation we include it here.

precondition map. Each state in the domain of an action structure  $\alpha$  represents the possible appearance of the epistemic action encoded by  $\alpha$ . The designated state represents the action actually taking place. Each binary relation of an action structure represents the type, or degree, of uncertainty entertained by the agent associated with the given binary relation about the action taking place; for instance, the agents' knowledge, ignorance, suspicions. Finally, the precondition function maps each state in the domain to a formula, which is intended to describe the state of affairs under which it is possible to execute the (appearing) action encoded by the given state. This formula encodes the *preconditions* of the action-state. The reader is referred to [2] for further intuition and concrete examples.

Let  $\text{AtProp}$  be a countable set of atomic propositions, and  $\text{Ag}$  be a nonempty set (of agents). The set  $\mathcal{L}$  of formulas  $A$  of the logic of epistemic actions and knowledge (EAK), and the set  $\text{Act}(\mathcal{L})$  of the *action structures over  $\mathcal{L}$*  are defined simultaneously as follows:

$$A := p \in \text{AtProp} \mid \neg A \mid A \vee A \mid \langle a \rangle A \mid \langle \alpha \rangle A \quad (\alpha \in \text{Act}(\mathcal{L}), a \in \text{Ag}),$$

where an *action structure over  $\mathcal{L}$*  is a tuple  $\alpha = (K, k, (\alpha_a)_{a \in \text{Ag}}, \text{Pre}_\alpha)$ , such that  $K$  is a finite nonempty set,  $k \in K$ ,  $\alpha_a \subseteq K \times K$  and  $\text{Pre}_\alpha : K \rightarrow \mathcal{L}$ .

The symbol  $\text{Pre}(\alpha)$  stands for  $\text{Pre}_\alpha(k)$ . For each action structure  $\alpha$  and every  $i \in K$ , let  $\alpha_i := (K, i, (\alpha_a)_{a \in \text{Ag}}, \text{Pre}_\alpha)$ . Intuitively, the family of action structures  $\{\alpha_i \mid k\alpha_a i\}$  encodes the uncertainty of agent  $a$  about the action  $\alpha = \alpha_k$  that is actually taking place. Perhaps the best known epistemic actions are *public announcements*, formalized as action structures  $\alpha$  such that  $K = \{k\}$ , and  $\alpha_a = \{(k, k)\}$  for all  $a \in \text{Ag}$ . The logic of public announcements (PAL) [22] can then be subsumed as the fragment of EAK restricted to action structures of the form described above. The connectives  $\top$ ,  $\perp$ ,  $\wedge$ ,  $\rightarrow$  and  $\leftrightarrow$  are defined as usual.

Standard models for EAK are relational structures  $M = (W, (R_a)_{a \in \text{Ag}}, V)$  such that  $W$  is a nonempty set,  $R_a \subseteq W \times W$  for each  $a \in \text{Ag}$ , and  $V : \text{AtProp} \rightarrow \mathcal{P}(W)$ . The interpretation of the static fragment of the language is standard. For every Kripke frame  $\mathcal{F} = (W, (R_a)_{a \in \text{Ag}})$  and each action structure  $\alpha$ , let the Kripke frame  $\coprod_\alpha \mathcal{F} := (\coprod_K W, ((R \times \alpha)_a)_{a \in \text{Ag}})$  be defined as follows:  $\coprod_K W$  is the  $|K|$ -fold coproduct of  $W$  (which is set-isomorphic to  $W \times K$ ), and  $(R \times \alpha)_a$  is a binary relation on  $\coprod_K W$  defined as

$$(w, i)(R \times \alpha)_a(u, j) \quad \text{iff} \quad wR_a u \text{ and } i\alpha_a j.$$

For every model  $M$  and each action structure  $\alpha$ , let

$$\coprod_\alpha M := (\coprod_\alpha \mathcal{F}, \coprod_K V)$$

be such that  $\coprod_\alpha \mathcal{F}$  is defined as above, and  $(\coprod_K V)(p) := \coprod_K V(p)$  for every  $p \in \text{AtProp}$ . Finally, let the *update* of  $M$  with the action structure  $\alpha$  be the submodel  $M^\alpha := (W^\alpha, (R_a^\alpha)_{a \in \text{Ag}}, V^\alpha)$  of  $\coprod_\alpha M$  the domain of which is the subset

$$W^\alpha := \{(w, j) \in \coprod_K W \mid M, w \Vdash \text{Pre}_\alpha(j)\}.$$

Given this preliminary definition, formulas of the form  $\langle \alpha \rangle A$  are interpreted as follows:

$$M, w \Vdash \langle \alpha \rangle A \quad \text{iff} \quad M, w \Vdash \text{Pre}_\alpha(k) \text{ and } M^\alpha, (w, k) \Vdash A.$$

The model  $M^\alpha$  is intended to encode the (factual and epistemic) state of affairs after the execution of the action  $\alpha$ . Summing up, the construction of  $M^\alpha$  is done in two stages: in the first stage, as many copies of the original model  $M$  are taken as there are ‘epistemic potential appearances’ of the given action (encoded by the action states in the domain of  $\alpha$ ); in the second stage, states in the copies are removed if their associated original state does not satisfy the preconditions of their paired action-state.

A complete axiomatization of EAK consists of copies of the axioms and rules of the minimal normal modal logic K for each modal operator, either epistemic or dynamic, plus the following (interaction) axioms:

$$\begin{aligned} (2.1) \quad & \langle \alpha \rangle p \leftrightarrow (\text{Pre}(\alpha) \wedge p); \\ (2.2) \quad & \langle \alpha \rangle \neg A \leftrightarrow (\text{Pre}(\alpha) \wedge \neg \langle \alpha \rangle A); \\ (2.3) \quad & \langle \alpha \rangle (A \vee B) \leftrightarrow (\langle \alpha \rangle A \vee \langle \alpha \rangle B); \\ (2.4) \quad & \langle \alpha \rangle \langle a \rangle A \leftrightarrow (\text{Pre}(\alpha) \wedge \bigvee \{ \langle a \rangle \langle \alpha_i \rangle A \mid k\alpha_i \}). \end{aligned}$$

The interaction axioms above can be understood as attempts at defining the meaning of any given dynamic modality  $\langle \alpha \rangle$  in terms of its interaction with the other connectives. In particular, while axioms (2.2) and (2.3) occur also in other dynamic logics such as PDL, axioms (2.1) and (2.4) capture the specific behaviour of epistemic actions. Specifically, axiom (2.1) encodes the fact that epistemic actions do not change the factual state of affairs, and axiom (2.4) plausibly rephrases the fact that ‘after the execution of  $\alpha$ , agent  $a$  thinks that  $A$  might be the case’ in terms of ‘there being some epistemic appearance of  $\alpha$  to  $a$  such that  $a$  thinks that, after its execution,  $A$  is the case’. An interesting aspect of these axioms is that they work as rewriting rules which can be iteratively used to transform any EAK-formula into an equivalent one free of dynamic modalities. Hence, the completeness of EAK follows from the completeness of its static fragment, and EAK is not more expressive than its static fragment. However, and interestingly, there is an exponential gap in succinctness between equivalent formulas in the two languages [19].

Action structures are one among many possible ways to represent actions. Following [17], we prefer to keep a black-box perspective on actions, and to identify agents  $a$  with the indistinguishability relation they induce on actions; so, in the remainder of the article, the role of the action-structures  $\alpha_i$  for  $k\alpha_i$  will be played by actions  $\beta$  such that  $\alpha a \beta$ , allowing us to reformulate (2.4) as

$$\langle \alpha \rangle \langle a \rangle A \leftrightarrow (\text{Pre}(\alpha) \wedge \bigvee \{ \langle a \rangle \langle \beta \rangle A \mid \alpha a \beta \}).$$



### Axioms

$$\begin{aligned}
& A \rightarrow (B \rightarrow A) \\
& (A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C)) \\
& A \rightarrow (B \rightarrow A \wedge B) \\
& A \wedge B \rightarrow A \\
& A \wedge B \rightarrow B \\
& A \rightarrow A \vee B \\
& B \rightarrow A \vee B \\
& (A \rightarrow C) \rightarrow ((B \rightarrow C) \rightarrow (A \vee B \rightarrow C)) \\
& \perp \rightarrow A \\
& [a](A \rightarrow B) \rightarrow ([a]A \rightarrow [a]B) \\
& \langle a \rangle(A \vee B) \rightarrow \langle a \rangle A \vee \langle a \rangle B \\
& \neg \langle a \rangle \perp \\
\text{FS1} & \langle a \rangle(A \rightarrow B) \rightarrow ([a]A \rightarrow \langle a \rangle B) \\
\text{FS2} & \langle \langle a \rangle A \rightarrow [a]B \rangle \rightarrow [a](A \rightarrow B)
\end{aligned}$$

### Inference Rules

$$\begin{aligned}
\text{MP} & \text{ if } \vdash A \rightarrow B \text{ and } \vdash A, \text{ then } \vdash B \\
\text{Nec} & \text{ if } \vdash A, \text{ then } \vdash [a]A
\end{aligned}$$

Table 1: Axioms and rules of the intuitionistic modal logic m-IK

## 2.2 Intuitionistic EAK

The (single-agent version of the) intuitionistic logic of epistemic actions and knowledge (IEAK) has been introduced in [18]. In the present subsection we report on its multi-agent version. The reason for mentioning this logic in the preliminaries is that the calculus introduced in Section 4 takes the Hilbert-style axiomatization of IEAK—rather than that of its Boolean counterpart—as basic, and many of its rules are motivated by axioms which define the intuitionistic setting (see Section 4 for further details on this topic).<sup>2</sup>

Let  $\text{AtProp}$  be a countable set of atomic propositions, and let  $\text{Ag}$  be a nonempty set (of agents). The set  $\mathcal{L}(\text{m-IK})$  of the formulas  $A$  of the multi-modal version m-IK of Fischer Servi intuitionistic modal logic IK (cf. [26]) are inductively defined as follows:

$$A := p \in \text{AtProp} \mid \perp \mid A \vee A \mid A \wedge A \mid A \rightarrow A \mid \langle a \rangle A \mid [a]A \quad (a \in \text{Ag})$$

Let  $\neg A$  abbreviate as usual  $A \rightarrow \perp$ . The Hilbert-style presentation of m-IK is reported in Table 1.

---

<sup>2</sup>The Boolean setting is captured by adding the so-called *Grishin rules* (see page 26) to the basic framework.

### Interaction Axioms

$$\begin{aligned}
\langle \alpha \rangle p &\leftrightarrow Pre(\alpha) \wedge p \\
[\alpha] p &\leftrightarrow Pre(\alpha) \rightarrow p \\
\langle \alpha \rangle \perp &\leftrightarrow \perp \\
\langle \alpha \rangle \top &\leftrightarrow Pre(\alpha) \\
[\alpha] \top &\leftrightarrow \top \\
[\alpha] \perp &\leftrightarrow \neg Pre(\alpha) \\
[\alpha](A \wedge B) &\leftrightarrow [\alpha]A \wedge [\alpha]B \\
\langle \alpha \rangle(A \wedge B) &\leftrightarrow \langle \alpha \rangle A \wedge \langle \alpha \rangle B \\
\langle \alpha \rangle(A \vee B) &\leftrightarrow \langle \alpha \rangle A \vee \langle \alpha \rangle B \\
[\alpha](A \vee B) &\leftrightarrow Pre(\alpha) \rightarrow (\langle \alpha \rangle A \vee \langle \alpha \rangle B) \\
\langle \alpha \rangle(A \rightarrow B) &\leftrightarrow Pre(\alpha) \wedge (\langle \alpha \rangle A \rightarrow \langle \alpha \rangle B) \\
[\alpha](A \rightarrow B) &\leftrightarrow \langle \alpha \rangle A \rightarrow \langle \alpha \rangle B \\
\langle \alpha \rangle \langle a \rangle A &\leftrightarrow Pre(\alpha) \wedge \bigvee \{ \langle a \rangle \langle \beta \rangle A \mid \alpha a \beta \} \\
[\alpha] \langle a \rangle A &\leftrightarrow Pre(\alpha) \rightarrow \bigvee \{ \langle a \rangle \langle \beta \rangle A \mid \alpha a \beta \} \\
[\alpha] [a] A &\leftrightarrow Pre(\alpha) \rightarrow \bigwedge \{ [a] [\beta] A \mid \alpha a \beta \} \\
\langle \alpha \rangle [a] A &\leftrightarrow Pre(\alpha) \wedge \bigwedge \{ [a] [\beta] A \mid \alpha a \beta \}
\end{aligned}$$

### Inference Rules

$$vNec \quad \text{if } \vdash A, \text{ then } \vdash [\alpha]A$$

Table 2: Axioms and rules of the intuitionistic epistemic logic IEAK

To define the language of IEAK, let  $AtProp$  be a countable set of atomic propositions, and let  $Ag$  be a nonempty set. The set  $\mathcal{L}(IEAK)$  of formulas  $A$  of the intuitionistic logic of epistemic actions and knowledge (IEAK), and the set  $Act(\mathcal{L})$  of the *action structures*  $\alpha$  over  $\mathcal{L}$  are defined simultaneously as follows:

$$A := p \in AtProp \mid \perp \mid A \rightarrow A \mid A \vee A \mid A \wedge A \mid \langle a \rangle A \mid [a]A \mid \langle \alpha \rangle A \mid [\alpha]A,$$

where  $a \in Ag$ , and an *action structure*  $\alpha$  over  $\mathcal{L}(IEAK)$  is defined in a completely analogous way as action structures in the classical case, the only difference lying in the codomain of  $Pre_\alpha$ . Then, the logic IEAK is defined in a Hilbert-style presentation which includes the axioms and rules of m-IK plus the axioms and rules in Table 2.

## 2.3 Display calculi

The first display calculus appears in Belnap's paper [3], as a sequent system augmenting and refining Gentzen's basic design of sequent calculi, which admit two types of rules: the structural, and the operational. Belnap's refinement is based on the introduction of a special syntax for the constituents of each sequent, which includes *structural connectives* along with *logical*, or *operational connectives*. For an expanded discussion of these ideas, the reader is referred to [15, 28, 25].

**Structures and display property.** *Structures* are built up much in the same way as formulas, taking formulas as atomic components, and applying structural connectives (which are typically 0-ary, unary and binary) so that each structure can be uniquely associated with and identified by its generation tree. Every node of such a generation tree defines a *substructure* of the given structure.

**Definition 1.** (cf. [3, Section 3.2]) A proof system enjoys the *full display property* iff for every sequent  $X \vdash Y$  and every substructure  $Z$  of either  $X$  or  $Y$ , the sequent  $X \vdash Y$  can be transformed, using the rules of the system, into a logically equivalent sequent which is either of the form  $Z \vdash W$  or of the form  $W \vdash Z$ , for some structure  $W$ . In the first case,  $Z$  is *displayed in precedent position*, and in the second case,  $Z$  is *displayed in succedent position*. The rules enabling this equivalent rewriting are called *display postulates*.

In what follows, we will sometimes write e.g.  $(X \vdash Y)[Z]^{pre}$  (resp.  $(X \vdash Y)[Z]^{suc}$ ) to indicate that  $Z$  occurs as a substructure in precedent (resp. succedent) position within the sequent  $X \vdash Y$ . Thanks to the fact that the display postulates are based on adjunction and residuation, it can be proved that exactly one of the two alternatives mentioned in the definition above occurs. In other words, in a system enjoying the display property, any substructure of any sequent  $X \vdash Y$  is always displayed either only in precedent position or only in succedent position. This is why we can talk about occurrences of substructures in precedent or in succedent position, even if they are nested deep within a given sequent.

**Uniform strategy for cut-elimination.** In [3], a meta-theorem is proven, which gives sufficient conditions in order for a sequent calculus to enjoy cut elimination. This meta-theorem captures the essentials of the cut-elimination procedure Gentzen-style, and is the main technical motivation for the design of Display Logic. Belnap's meta-theorem gives a set of eight conditions on sequent calculi, most of which are verified by inspection on the shape of the rules. Together, these conditions guarantee that the cut rule is eliminable in the given sequent calculus, and that the system enjoys the subformula property. When Belnap's meta-theorem can be applied, it provides a much smoother and more modular route to cut elimination than the Gentzen-style proofs. Belnap's original meta-theorem has been generalized and refined by various authors (cf. [23, 25, 28]). Particularly relevant to us is the notion of *properly displayable calculus*, introduced in [28, Section 4.1], a generalization of which has been proposed in [15], which in its turn is further generalized in Section 3.2.

**Relativized display property.** The full display property is a key ingredient in the proof of the cut-elimination metatheorem. For instance, it enables a system enjoying it to meet Belnap's condition  $C_8$  for the cut-elimination metatheorem. However, it turns out that an analogously good behaviour can be guaranteed of any sequent calculus enjoying the following weaker property:

**Definition 2.** A proof system enjoys the *relativized display property* iff for every *derivable* sequent  $X \vdash Y$  and every substructure  $Z$  of either  $X$  or  $Y$ , the sequent  $X \vdash Y$  can be transformed, using the rules of the system, into a logically equivalent sequent which is either of the form  $Z \vdash W$  or of the form  $W \vdash Z$ , for some structure  $W$ .

The calculus defined in Section 4 does not enjoy the full display property, but does enjoy the relativized display property above (more about this in Sections 4 and 7), which enables it to verify the condition  $C'_8$  (see Section 3.2). More details about it are collected in Section 9.1. Finally, notice that the definition of substructures in precedent or succedent position within each sequent can be given in a way which does not rely on the full display property. It is enough to rely on the polarity of the coordinates of each structural connective: if these polarities are assigned, then for any sequent  $X \vdash Y$ , if  $Z$  is a substructure of  $X$ , then  $Z$  is in precedent (resp. succedent) position if, in the generation tree of  $X$ , the path from  $Z$  to the root goes through an even (resp. odd) number of coordinates with negative polarity. If  $Z$  is a substructure of  $Y$ , then  $Z$  is in succedent (resp. precedent) position if, in the generation tree of  $Y$ , the path from  $Z$  to the root goes through an even (resp. odd) number of coordinates with negative polarity.

## 2.4 A single-type display calculus for EAK

In [15], a display calculus is introduced for EAK, which is shown to be sound w.r.t. the final coalgebra semantics, syntactically complete w.r.t. EAK and to enjoy cut-elimination Belnap-style. In the present subsection we briefly report on it, not only for the sake of providing a relevant example of display calculus, but above all because a translation can be established between the operational language of  $D'.EAK$  and of the Dynamic Calculus (cf. Section 4). This translation is important for the further treatment of Sections 5 and 7.

The structural and operational languages of  $D'.EAK$  are expansions of the standard structural and operational propositional languages with the following (structural and operational) modal operators, indexed by agents  $a$  and actions  $\alpha$ , and (structural and operational) constant symbols:

Structural symbols	$\{a\}$	$\overline{a}$	$\{\alpha\}$	$\overline{\alpha}$	$\Phi_\alpha$
Operational symbols	$\langle a \rangle$   $[a]$	$(\widehat{a})$   $(\overline{a})$	$\langle \alpha \rangle$   $[\alpha]$	$\widehat{\alpha}$   $\overline{\alpha}$	$1_\alpha$

The structural connectives  $\{\alpha\}$  and  $\overline{\alpha}$  correspond to diamond-type modalities when occurring in precedent position, and to box-type modalities when occurring in succedent position. The structural and operational constants  $\Phi_\alpha$  and  $1_\alpha$  are used to capture the proof-theoretic behaviour of the metalinguistic abbreviation  $Pre(\alpha)$  at the object-level. In the rules below, the structural connective  $\Phi_\alpha$  can occur only in precedent position. Hence, the structural constant symbol  $\Phi_\alpha$  can never be interpreted as anything else than  $1_\alpha$ . However, a natural way to extend  $D'.EAK$  would

be to introduce an operational constant symbol  $0_\alpha$ , intuitively standing for the postconditions of  $\alpha$  for each action  $\alpha$ , and dualize the relevant rules so as to capture the behaviour of postconditions.

The connectives  $\widehat{\alpha}$  and  $\overline{\alpha}$  occur within brackets since they are not actually part of the logical language of D'.EAK, but point at the fact that the structural connective  $\overline{\alpha}$  is interpreted in the final coalgebra as the diamond (resp. box) associated with the *converse* of the relation associated with the epistemic action  $\alpha$  (for an expanded discussion on this, the reader is referred to [15, Section 5.2]). The key aspect of the final coalgebra as a semantic environment for EAK is that it makes it possible to see the dynamic connectives  $[\alpha]$  and  $\langle\alpha\rangle$  as parts of *adjoint pairs*, precisely involving the additional modalities  $\widehat{\alpha}$  and  $\overline{\alpha}$ . Specifically, we have the following (syntactic) adjunction relations  $\langle\alpha\rangle \dashv \overline{\alpha}$  and  $\widehat{\alpha} \dashv [\alpha]$ : for all formulas  $A, B$ ,

$$(2.5) \quad \langle\alpha\rangle A \vdash B \text{ iff } A \vdash \overline{\alpha} B \quad \widehat{\alpha} A \vdash B \text{ iff } A \vdash [\alpha] B$$

The reader is referred to [15, Section 5] for a detailed discussion. The two tables below introduce the structural rules for the dynamic modalities which have the same shape as those for the agent-indexed modalities, here omitted.

### Structural Rules

$$\begin{array}{c}
\begin{array}{c}
nec_L^{dyn} \frac{I \vdash X}{\{\alpha\} I \vdash X} \quad \frac{X \vdash I}{X \vdash \{\alpha\} I} nec_R^{dyn} \\
dyn nec_L \frac{I \vdash X}{\overline{\alpha} I \vdash X} \quad \frac{X \vdash I}{X \vdash \overline{\alpha} I} dyn nec_R \\
FS_L^{dyn} \frac{\{\alpha\} Y > \{\alpha\} Z \vdash X}{\{\alpha\} (Y > Z) \vdash X} \quad \frac{Y \vdash \{\alpha\} X > \{\alpha\} Z}{Y \vdash \{\alpha\} (X > Z)} FS_R^{dyn} \\
mon_L^{dyn} \frac{\{\alpha\} X; \{\alpha\} Y \vdash Z}{\{\alpha\} (X; Y) \vdash Z} \quad \frac{Z \vdash \{\alpha\} Y; \{\alpha\} X}{Z \vdash \{\alpha\} (Y; X)} mon_R^{dyn} \\
dyn FS_L \frac{\overline{\alpha} Y > \overline{\alpha} X \vdash Z}{\overline{\alpha} (Y > X) \vdash Z} \quad \frac{Y \vdash \overline{\alpha} X > \overline{\alpha} Z}{Y \vdash \overline{\alpha} (X > Z)} dyn FS_R \\
dyn mon_L \frac{\overline{\alpha} X; \overline{\alpha} Y \vdash Z}{\overline{\alpha} (X; Y) \vdash Z} \quad \frac{Z \vdash \overline{\alpha} Y; \overline{\alpha} X}{Z \vdash \overline{\alpha} (Y; X)} dyn mon_R \\
conj \frac{\{a\} (X; \overline{a} Y) \vdash Z}{\{a\} X; Y \vdash Z} \quad \frac{X \vdash \{a\} (Y; \overline{a} Z)}{X \vdash \{a\} Y; Z} conj \\
conj \frac{\overline{a} (X; \{a\} Y) \vdash Z}{\overline{a} X; Y \vdash Z} \quad \frac{X \vdash \overline{a} (Y; \{a\} Z)}{X \vdash \overline{a} Y; Z} conj
\end{array}
\end{array}$$

The *conj*-rules and the *FS*-rules can be shown to be interderivable thanks to the following display postulates.

### Display Postulates

$$(\{a\}, \overline{a}) \frac{\frac{\{a\} X \vdash Y}{X \vdash \overline{a} Y}}{\overline{a} Y \vdash X} \quad \frac{Y \vdash \{a\} X}{\overline{a} Y \vdash X} (\overline{a}, \{a\})$$

The display postulates above are direct translations of the adjunction relations (2.5). Next, we report on the structural rules which are to capture the specific behaviour of epistemic actions:

### Atom

$$\frac{}{\Gamma p \vdash \Delta p} \text{atom}$$

where  $\Gamma$  and  $\Delta$  are arbitrary finite sequences of the form  $(\alpha_1) \dots (\alpha_n)$  (possibly of different length), such that each  $(\alpha_j)$  is of the form  $\{\alpha_j\}$  or of the form  $\widetilde{\alpha_j}$ , for  $1 \leq j \leq n$ . Intuitively, the *atom rules* capture the requirement that epistemic actions do not change the factual state of affairs (in the Hilbert-style presentation of EAK, this is encoded in the axiom (2.1) in Section 2.1).

### Structural Rules for Epistemic Actions

$$\begin{array}{c} \frac{X \vdash Y}{\{\alpha\}X \vdash \{\alpha\}Y} \text{balance} \\ \\ \text{comp}_L^\alpha \frac{\{\alpha\} \widetilde{\alpha} X \vdash Y}{\Phi_\alpha; X \vdash Y} \quad \frac{X \vdash \{\alpha\} \widetilde{\alpha} Y}{X \vdash \Phi_\alpha > Y} \text{comp}_R^\alpha \\ \\ \text{reduce}_L \frac{\Phi_\alpha; \{\alpha\}X \vdash Y}{\{\alpha\}X \vdash Y} \quad \frac{Y \vdash \Phi_\alpha > \{\alpha\}X}{Y \vdash \{\alpha\}X} \text{reduce}_R \\ \\ \text{swap-in}_L \frac{\{\alpha\}\{\mathbf{a}\}X \vdash Y}{\Phi_\alpha; \{\mathbf{a}\}\{\beta\}_{\alpha\mathbf{a}\beta} X \vdash Y} \quad \frac{Y \vdash \{\alpha\}\{\mathbf{a}\}X}{Y \vdash \Phi_\alpha > \{\mathbf{a}\}\{\beta\}_{\alpha\mathbf{a}\beta} X} \text{swap-in}_R \\ \\ \text{swap-out}_L \frac{(\{\mathbf{a}\}\{\beta\} X \vdash Y \mid \alpha\mathbf{a}\beta)}{\{\alpha\}\{\mathbf{a}\}X \vdash \mathbin{\text{;}}(Y \mid \alpha\mathbf{a}\beta)} \quad \frac{(Y \vdash \{\mathbf{a}\}\{\beta\} X \mid \alpha\mathbf{a}\beta)}{\mathbin{\text{;}}(Y \mid \alpha\mathbf{a}\beta) \vdash \{\alpha\}\{\mathbf{a}\}X} \text{swap-out}_R \end{array}$$

The *swap-in* rules are unary and should be read as follows: if the premise holds, then the conclusion holds relative to any action  $\beta$  such that  $\alpha\mathbf{a}\beta$ . The *swap-out* rules do not have a fixed arity; they have as many premises<sup>3</sup> as there are actions  $\beta$  such that  $\alpha\mathbf{a}\beta$ . In their conclusion, the symbol  $\mathbin{\text{;}}(Y \mid \alpha\mathbf{a}\beta)$  refers to a string  $(\dots(Y; Y); \dots; Y)$  with  $n$  occurrences of  $Y$ , where  $n = |\{\beta \mid \alpha\mathbf{a}\beta\}|$ . The *swap-in* and *swap-out* rules encode the interaction between dynamic and epistemic modalities as it is captured by the interaction axioms in the Hilbert style presentation of EAK (cf. (2.4) in Section 2.1 and similarly in Section 2.2). The *reduce* rules encode well-known EAK validities such as  $\langle \alpha \rangle A \rightarrow (Pre(\alpha) \wedge \langle \alpha \rangle A)$ .

Finally, the operational rules for  $\langle \alpha \rangle$ ,  $[\alpha]$ , and  $1_\alpha$  are reported below:

<sup>3</sup>The *swap-out* rule could indeed be infinitary if action structures were allowed to be infinite, which in the present setting, as in [2], is not the case.

### Operational Rules

$$\begin{array}{c}
 \langle \alpha \rangle_L \frac{\{\alpha\}A \vdash X}{\langle \alpha \rangle A \vdash X} \quad \frac{X \vdash A}{\{\alpha\}X \vdash \langle \alpha \rangle A} \langle \alpha \rangle_R \\
 [\alpha]_L \frac{A \vdash X}{[\alpha]A \vdash \{\alpha\}X} \quad \frac{X \vdash \{\alpha\}A}{X \vdash [\alpha]A} [\alpha]_R \\
 l_{\alpha L} \frac{\Phi_{\alpha} \vdash X}{l_{\alpha} \vdash X} \quad \frac{}{\Phi_{\alpha} \vdash l_{\alpha}} l_{\alpha R}
 \end{array}$$

## 3 Multi-type calculi, and cut elimination metatheorem

The present section is aimed at introducing the environment of multi-type display calculi. Our treatment will be very general, and in particular, no signature will be specified. However, the calculus introduced in Section 4 is a concrete instantiation of this abstract description.

### 3.1 Multi-type calculi

Our starting point is a propositional language, the terms of which form  $n$  pairwise disjoint types  $T_1 \dots T_n$ , each of which with its own signature. We will use  $a, b, c$  and  $x, y, z$  to respectively denote operational and structural terms of unspecified (possibly different) type. Further, we assume that operational connectives and structural connectives are given *both* within each type *and* also between different types, so that the display property holds.

In the applications we have in mind, the need will arise to support types that are semantically ordered by inclusion. For example, in Section 4 we will introduce, beside the type  $Fm$  of formulas, two types  $Fnc$  and  $Act$  of functional and general actions, respectively. The need for enforcing the distinction between functional and general actions in the specific situation of Section 4 arises because of the presence of the rule *balance* (see page 30 for more details on this topic). The semantic point of view suggests to treat  $Fnc$  as a proper subset of  $Act$ , but our syntactic stipulations, although will be sound w.r.t. this state of affairs, will be tuned for the more general situation in which the sets  $Fnc$  and  $Act$  are disjoint. This is convenient as each term can be assigned a unique type *unambiguously*. This is a crucial requirement for the Belnap-style cut elimination theorem of the next section, and will be explicitly stated in condition  $C'_2$  below.

**Definition 3.1.** A sequent  $x \vdash y$  is *type-uniform* if  $x$  and  $y$  are of the same type  $T$ . In this case, we will say that  $x \vdash y$  is of type  $T$ .

A fundamental and very natural desideratum for rules in a multi-type display calculus is that they preserve type-uniformity, that is, each rule should be such that if all the premises are type uniform, then the conclusion is type uniform. As we will see, all rules in the multi-type calculus introduced in Section 4 preserve type uniformity.

Finally, in a display calculus, the cut rule is typically of the following form:

$$\frac{X \vdash A \quad A \vdash Y}{X \vdash Y} \textit{Cut}$$

where  $X, Y$  are structures and  $A$  is a formula. This translates straightforwardly to the multi-type environment, by the stipulation that cut rules of the form

$$\frac{x \vdash a \quad a \vdash y}{x \vdash y} \textit{Cut}$$

are allowed in the given multi-type system for each type. These cut rules will be asked to satisfy the following additional requirement:

**Definition 3.2.** A rule is *strongly type-uniform* if its premises and conclusion are of the same type.

### 3.2 Quasi-properly displayable multi-type calculi

In [15], to show that Belnap-style cut elimination holds for the display calculus  $D'$ .EAK, the definition of quasi-properly displayable calculi is given (generalizing Wansing's definition of properly displayable calculi [28, Section 4.2]), and its corresponding Belnap style meta-theorem is discussed. We are working towards the proof that the multi-type display calculus introduced in Section 4 enjoys cut elimination Belnap-style. The aim of the present subsection is then to extend the notion of quasi-properly displayable calculi to the multi-type environment. Let a *quasi-properly displayable multi-type calculus* be any displaycalculus in a multi-type language satisfying the following list of conditions<sup>4</sup>:

**C<sub>1</sub>: Preservation of operational terms.** Each operational term occurring in a premise of an inference rule *inf* is a subterm of some operational term in the conclusion of *inf*.

**C<sub>2</sub>: Shape-alikeness of parameters.** Congruent parameters<sup>5</sup> are occurrences of the same structure.

**C'<sub>2</sub>: Type-alikeness of parameters.** Congruent parameters have exactly the same type. This condition bans the possibility that a parameter changes type along its history.

<sup>4</sup>See [15] for a discussion on  $C'_5$  and  $C''_5$ .

<sup>5</sup>The congruence relation is an equivalence relation which is meant to identify the different occurrences of the same formula or substructure along the branches of a derivation [3, section 4], [25, Definition 6.5]. Condition  $C_2$  can be understood as a condition on the *design* of the rules of the system if the congruence relation is understood as part of the specification of each given rule; that is, each rule of the system should come with an explicit specification of which elements are congruent to which (and then the congruence relation is defined as the reflexive and transitive closure of the resulting relation). In this respect,  $C_2$  is nothing but a sanity check, requiring that the congruence is defined in such a way that indeed identifies the occurrences which are intuitively "the same".



**C’<sub>3</sub>: Restricted non-proliferation of parameters.** Each parameter in an inference rule *inf* is congruent to at most one constituent in the conclusion of *inf*. This restriction does not need to apply to parameters of any type  $\mathsf{T}$  such that the only applications of cut with cut terms of type  $\mathsf{T}$  are of the following shapes:

$$\frac{\begin{array}{c} \vdots \\ X \vdash a \end{array} \quad a \vdash a}{X \vdash a} \quad \frac{a \vdash a \quad \begin{array}{c} \vdots \\ a \vdash Y \end{array}}{a \vdash Y}$$

**C<sub>4</sub>: Position-alikeness of parameters.** Congruent parameters are either all antecedent or all succedent parts of their respective sequents.

**C’<sub>5</sub>: Quasi-display of principal constituents.** If an operational term  $a$  is principal in the conclusion sequent  $s$  of a derivation  $\pi$ , then  $a$  is in display, unless  $\pi$  consists only of its conclusion sequent  $s$  (i.e.  $s$  is an axiom).

**C’’<sub>5</sub>: Display-invariance of axioms.** If a display rule can be applied to an axiom  $s$ , the result of that rule application is again an axiom.

**C’<sub>6</sub>: Closure under substitution for succedent parts within each type.** Each rule is closed under simultaneous substitution of arbitrary structures for congruent operational terms occurring in succedent position, *within each type*.

**C’<sub>7</sub>: Closure under substitution for precedent parts within each type.** Each rule is closed under simultaneous substitution of arbitrary structures for congruent operational terms occurring in precedent position, *within each type*.

Condition  $C_6$  (and likewise  $C’_7$ ) ensures, for instance, that if the following inference is an application of the rule  $R$ :

$$\frac{(x \vdash y)([a]_i^{suc} \mid i \in I)}{(x' \vdash y')[a]^{suc}} R$$

and  $([a]_i^{suc} \mid i \in I)$  represents all and only the occurrences of the operational term  $a$  in the premiss which are congruent to the occurrence of  $a$  in the conclusion<sup>6</sup>, then also the following inference is an application of the same rule  $R$ :

$$\frac{(x \vdash y)([z/a]_i^{suc} \mid i \in I)}{(x' \vdash y')[z/a]^{suc}} R$$

where the structure  $z$  is substituted for  $a$ , and  $z$  and  $a$  have the same type.

<sup>6</sup>Clearly, if  $I = \emptyset$ , then the occurrence of  $a$  in the conclusion is congruent to itself.

**C’<sub>8</sub>: Eliminability of matching principal constituents.** This condition requests a standard Gentzen-style checking, which is now limited to the case in which both cut formulas are *principal*, and hence each of them has been introduced with the last rule application of each corresponding subdeduction. In this case, analogously to the proof Gentzen-style, condition C’<sub>8</sub> requires being able to transform the given deduction into a deduction with the same conclusion in which either the cut is eliminated altogether, or is transformed in one or more applications of the cut rule, involving proper subterms of the original operational cut-term. In addition to this, specific to the multi-type setting is the requirement that the new application(s) of the cut rule be also *strongly type-uniform* (cf. condition C<sub>10</sub> below).

**C’’<sub>8</sub>: Closure of axioms under cut.** If  $x \vdash a$  and  $a \vdash y$  are axioms, then  $x \vdash y$  is again an axiom.

**C<sub>9</sub>: Type-uniformity of derivable sequents.** Each derivable sequent is type-uniform.

**C<sub>10</sub>: Strong type-uniformity of cut rules.** All cut rules are strongly type-uniform (cf. Definition 3.2).

### 3.3 Belnap-style metatheorem for multi-types

In the present subsection, we state and prove the Belnap-style metatheorem which we will appeal to when establishing the cut elimination Belnap-style for the calculus we will introduce in the next section.

**Theorem 3.3.** *Any multi-type display calculus satisfying  $C_2, C'_2, C'_3, C_4, C'_5, C''_5, C'_6, C'_7, C'_8, C''_8, C_9$  and  $C_{10}$  is cut-admissible. If also  $C_1$  is satisfied, then the calculus enjoys the subformula property.*

*Proof.* This is a generalization of the proof in [29, Section 3.3, Appendix A]. For the sake of conciseness, we will expand only on the parts of the proof which depart from that treatment. As usual, the proof is done by induction on the ordered pair of parameters given by the complexity of the cut term and the height of the cut. Our original derivation is

$$\frac{\begin{array}{c} \vdots \pi_1 \\ x \vdash a \end{array} \quad \begin{array}{c} \vdots \pi_2 \\ a \vdash y \end{array}}{x \vdash y}$$

**Principal stage: both cut formulas are principal.** There are three subcases.

If the end sequent  $x \vdash y$  is identical to the conclusion of  $\pi_1$  (resp.  $\pi_2$ ), then we can eliminate the cut simply replacing the derivation above with  $\pi_1$  (resp.  $\pi_2$ ).

If the premises  $x \vdash a$  and  $a \vdash y$  are axioms, then, by  $C'_8$ , the conclusion  $x \vdash y$  is an axiom, therefore the cut can be eliminated by simply replacing the original derivation with  $x \vdash y$ .

If one of the two premises of the cut in the original derivation is not an axiom, then, by  $C'_8$ , there is a proof of  $x \vdash y$  which uses the same premise(s) of the original derivation and which involves only strongly uniform cuts on proper subterms of  $a$ .

**Parametric stage: at least one cut term is parametric.** There are two subcases: either one cut term is principal or they are both parametric.

Consider the subcase in which one cut term is principal. W.l.o.g. we assume that the cut-term  $a$  is principal in the left-premise  $x \vdash a$  of the cut in the original proof (the other case is symmetric). We can assume w.l.o.g. that the conclusion of the cut is different from either of its premises. Then, conditions  $C_2$  and  $C'_3$  make it possible to trace the history-tree of the occurrences of the cut-term  $a$  in  $\pi_2$  (cf. [15, Remark 1]), and by conditions  $C'_2$  and  $C_4$ , any ancestor of  $a$  is of the same type and in the same position (that is, is in precedent position). The situation can be pictured as follows:

$$\begin{array}{c}
 \begin{array}{ccc}
 \vdots \pi_{2,i} & \vdots \pi_{2,j} & \vdots \pi_{2,k} \\
 \underline{a_i} \vdash y_i & (x_j \vdash y_j)[\underline{a_j}]^{pre} & (x_k \vdash y_k)[\bar{a}_k]^{pre} \\
 \ddots & \vdots & \ddots \\
 & \ddots \vdots \pi_2 & \\
 \vdots \pi_1 & & \\
 \hline
 x \vdash a & & a \vdash y \\
 \hline
 & & x \vdash y
 \end{array}
 \end{array}$$

where, for  $i, j, k \in \{1, \dots, n\}$ , the nodes

$$\underline{a_i} \vdash y_i, \quad (x_j \vdash y_j)[\underline{a_j}]^{pre}, \quad \text{and} \quad (x_k \vdash y_k)[\bar{a}_k]^{pre}$$

represent the three ways in which the leaves  $a_i$ ,  $a_j$  and  $a_k$  in the history-tree of  $a$  in  $\pi_2$  can be introduced, and which will be discussed below. The notation  $\underline{a}$  (resp.  $\bar{a}$ ) indicates that the given occurrence is principal (resp. parametric). Notice that condition  $C_4$  guarantees that all occurrences in the history of  $a$  are in precedent position in the underlying derivation tree, and condition  $C'_2$  guarantees that the type of  $a$  never changes along its history. Let  $a_l$  be introduced as a parameter (as represented in the picture above in the conclusion of  $\pi_{2,k}$  for  $a_l = a_k$ ). Assume that  $(x_k \vdash y_k)[\bar{a}_k]^{pre}$  is the conclusion of an application *inf* of the rule *Ru* (for instance, in the calculus of Section 4, this situation arises if  $a_k$  is of type *Fm* and has been introduced with an application of Weakening, or if  $a_k$  is of type *Fnc* and

has been introduced with an application of Atom, or Balance). Since  $a_k$  is a leaf in the history-tree of  $a$ , we have that  $a_k$  is congruent only to itself in  $x_k \vdash y_k$ . Notice that the assumption that every derivable sequent is type-uniform ( $C_9$ ), and the type-alikeness of parameters ( $C'_2$ ) imply that the sequent  $a_1, a_k$  and  $x$  have the same type. Hence,  $C'_7$  implies that it is possible to substitute  $x$  for  $a_k$  by means of an application of the same rule  $Ru$ . That is,  $(x_k \vdash y_k)[\bar{a}_k]$  can be replaced by  $(x_k \vdash y_k)[\bar{x}/\bar{a}_k]$ .

Let  $a_l$  be introduced as a principal formula. The corresponding subcase in [29] splits into two subsubcases: either  $a_l$  is introduced in display or it is not.

If  $a_l$  is in display (as represented in the picture above in the conclusion of  $\pi_{2,i}$  for  $a_l = a_i$ ), then we form a subderivation using  $\pi_1$  and  $\pi_{2,i}$  and applying cut as the last rule. The assumptions that the original cut is strongly type-uniform ( $C_{10}$ ), that every derivable sequent is type-uniform ( $C_9$ ), and the type-alikeness of parameters ( $C'_2$ ) imply that the sequent  $\underline{a}_i \vdash y_i$  is of the same type as the sequents  $x \vdash a$  and  $a \vdash y$ . Hence, the new cut is strongly type-uniform.

If  $a_l$  is not in display (as represented in the picture above in the conclusion of  $\pi_{2,j}$  for  $a_l = a_j$ ), then condition  $C'_5$  implies that  $(x_j \vdash y_j)[\underline{a}_j]^{pre}$  is an axiom, and  $C''_5$  implies that some axiom  $\underline{a}_j \vdash y'_j$  exists, which is display-equivalent to the first axiom, and in which  $a_j$  occurs in display. Let  $\pi'$  be the derivation which transforms  $\underline{a}_j \vdash y'_j$  into  $(x_j \vdash y_j)[\underline{a}_j]^{pre}$ . We form a subderivation using  $\pi_1$  and  $\underline{a}_j \vdash y'_j$  and joining them with a cut application, then attaching  $\pi'[x/a_j]^{pre}$  below the new cut.

The transformations just discussed explain how to transform the leaves of the history tree of  $a$ . Finally, since, as discussed above,  $x$  has the same type of  $a$ , condition  $C'_7$  implies that substituting  $x$  for each occurrence of  $a$  in the history tree of the cut term  $a$  in  $\pi_2$  (and in each occurring  $\pi'$  as above) gives rise to an admissible derivation  $\pi_2[x/a]^{pre}$  (use  $C'_6$  for the symmetric case).

Summing up, this procedure generates the following proof tree:

$$\begin{array}{c}
\vdots \pi_1 \\
x \vdash a \quad \underline{a}_j \vdash y' \\
\hline
x \vdash y' [a]^{suc} \\
\vdots \pi' [x/a]^{pre} \quad \vdots \pi_{2,k} \\
(x_j \vdash y_j) [x/a_j]^{pre} \quad (x_k \vdash y_k) [\bar{x}/\bar{a}_k]^{pre} \\
\vdots \quad \ddots \\
\ddots \vdots \ddots \pi_2 [x/a]^{pre} \\
x \vdash y
\end{array}$$

We observe that in each newly introduced application of the cut rule, both cut terms are principal. Hence, we can apply the procedure described in the Principal stage and transform the original derivation in a derivation in which the cut terms of the

newly introduced cuts have strictly lower complexity than the original cut terms. When the newly introduced applications of cut are of lower height than the original one, we do not need to resort to the Principal stage.<sup>7</sup>

Finally, as to the subcase in which both cut terms are parametric, consider a proof with at least one cut. The procedure is analogous to the previous case. Namely, following the history of one of the cut terms up to the leaves, and applying the transformation steps described above, we arrive at a situation in which, whenever new applications of cuts are generated, in each such application at least one of the cut formulas is principal. To each such cut, we can apply (the symmetric version of) the Parametric stage described so far.

□

## 4 The Dynamic Calculus for EAK

As mentioned in the introduction, the key idea is to introduce a language in which not only formulas are generated from formulas and actions (as it happens in the symbol  $\langle a \rangle A$ ) and formulas are generated from formulas and agents (as it happens in the symbol  $\langle a \rangle A$ ), but also *actions* are generated from the interactions between *agents* and *actions*.

**An algebraically motivated introduction.** In the present section, we define a *multi-type language* into which the language of (I)EAK translates, and in which each generation step mentioned above is explicitly accounted for via special binary connectives taking arguments of *different types*. More than one alternative is possible in this respect; our choice for the present setting consists of the following types: **Ag** for agents, **Fnc** for functional actions, **Act** for actions, and **Fm** for formulas. We also stipulate that **Ag**, **Act**, **Fm** and **Fnc** are pairwise disjoint. The new connectives, and their types, are:

$$(4.1) \quad \Delta_0, \blacktriangle_0 : \text{Fnc} \times \text{Fm} \rightarrow \text{Fm}$$

$$(4.2) \quad \Delta_1, \blacktriangle_1 : \text{Act} \times \text{Fm} \rightarrow \text{Fm}$$

$$(4.3) \quad \Delta_2, \blacktriangle_2 : \text{Ag} \times \text{Fm} \rightarrow \text{Fm}$$

$$(4.4) \quad \Delta_3, \blacktriangle_3 : \text{Ag} \times \text{Fnc} \rightarrow \text{Act}$$

We stipulate that the interpretations of the connectives are maps preserving existing joins in each coordinate (see below) with algebras as domains and codomains suitable to interpret (functional) actions, formulas, and agents respectively. For instance, suitable choices for domains of interpretation for formulas can be complete

---

<sup>7</sup> This is for instance the case if, in the original derivation, the history-tree of the cut term  $a$  (in the right-hand-side premise of the given cut application) contains at most one leaf  $a_i$  which is principal. However, the procedure described above in the Parametric stage does not always produce cuts of lower height. For instance, in the calculus introduced in Section 4, this situation may arise when two ancestors of a cut term of type **Fm** are introduced as principal along the same branch, and then are identified via an application of the rule Contraction.

atomic Boolean algebras or perfect Heyting algebras (cf. [18]); in the setting of e.g. epistemic action logic (cf. [27]), following [1], the domain of interpretation for actions can be a quantale or a relation algebra (of which the functional actions can be a sub-monoid). In the setting of EAK, in which no algebraic structure is required of actions and agents, a suitable domain of interpretation can be a complete join-semilattice, which is completely join-generated by a given subset (interpreting the functional actions), and the domain of interpretation of agents can be a set.<sup>8</sup>

In Section 5, the final coalgebra  $Z$  (cf. [15, Section 5]) is taken as semantic environment for the Dynamic Calculus. In this setting, the boolean algebra  $\mathcal{P}Z$  is taken as the domain of interpretation for Fm-type terms, Fnc-type terms are interpreted as graphs of partial functions on  $Z$ , subject to certain restrictions, and the domain of interpretation of Act-type terms is the complete  $\cup$ -semilattice generated by the domain of interpretation of Fnc.

In all the domains of interpretation which are complete lattices (i.e. the algebras interpreting terms of type Fm and Act), the fact that the interpretation of each connective  $\Delta$  and  $\blacktriangle$  is completely join-preserving in its second coordinate implies that it has a right adjoint in its second coordinate. These right adjoints provide natural interpretation for the following additional connectives:

$$(4.5) \quad \rightarrow_0, \dashv_0 \quad : \quad \text{Fnc} \times \text{Fm} \rightarrow \text{Fm}$$

$$(4.6) \quad \rightarrow_1, \dashv_1 \quad : \quad \text{Act} \times \text{Fm} \rightarrow \text{Fm}$$

$$(4.7) \quad \rightarrow_2, \dashv_2 \quad : \quad \text{Ag} \times \text{Fm} \rightarrow \text{Fm}.$$

The assumptions above imply that  $\Delta_1$  and  $\blacktriangle_1$  have right adjoints also in their first coordinate. Hence, each of the following connectives can be naturally interpreted, in the setting above, as the right adjoint of  $\Delta_1$  and  $\blacktriangle_1$  respectively:

$$(4.8) \quad \blacktriangleleft_1, \triangleleft_1 \quad : \quad \text{Fm} \times \text{Fm} \rightarrow \text{Act}.$$

Intuitively, for all formulas  $A, B$ , the term  $B \blacktriangleleft_1 A$  denotes the weakest epistemic action  $\gamma$  such that, if  $A$  was true before  $\gamma$  was performed, then  $B$  is true after any successful execution of  $\gamma$ . This is also related to Vaughn Pratt's notion of weakest preserver (cf. [24, Section 4.2]) However, we cannot assume that more adjoints exist, which would provide semantic interpretation for the following symbols:

$$\blacktriangleleft_0, \triangleleft_0 \quad : \quad \text{Fm} \times \text{Fm} \rightarrow \text{Fnc}$$

$$\blacktriangleleft_2, \triangleleft_2 \quad : \quad \text{Fm} \times \text{Fm} \rightarrow \text{Ag}$$

$$\blacktriangleleft_3, \triangleleft_3 \quad : \quad \text{Act} \times \text{Fnc} \rightarrow \text{Ag}$$

$$\blacktriangleright_3, \triangleright_3 \quad : \quad \text{Ag} \times \text{Act} \rightarrow \text{Fnc}.$$

---

<sup>8</sup>Notice also that for other dynamic logics the domain of interpretation of agents might be endowed with some algebraic structure; for instance, in the case of game logic (cf. [21]), the set of agents consists of two elements, on which a negation-type operation can be assumed.

**Virtual adjoints.** We adopt the following notational convention about the three different shapes of arrows introduced so far. Arrows with straight tails ( $\rightarrow$  and  $\rightarrow$ ) stand for connectives which have a semantic counterpart and which are included in the language of the Dynamic Calculus (see the grammar of operational terms on page 23); arrows with no tail (e.g.  $\blacktriangleleft$  and  $\triangleleft$ ) do have a semantic interpretation but are *not* included in the language at the operational level, and arrows with squiggly tails ( $\rightsquigarrow$ ,  $\triangleleft\rightsquigarrow$ ,  $\rightsquigarrow$  and  $\blacktriangleleft\rightsquigarrow$ ) stand for syntactic objects, called *virtual adjoints*, which do not have a semantic interpretation, but will play an important role, namely guaranteeing the Dynamic Calculus to enjoy the relativized display property (cf. Definition 2). In what follows, virtual adjoints will be introduced *only* as structural connectives. That is, they will not correspond to any operational connective, and they will not appear actively in any rule schema other than the display postulates (cf. Definition 1). As will be shown in Section 7, these limitations keep the calculus sound even if virtual adjoints do not have an independent semantic interpretation.

The  $\Delta \dashv \rightarrow$  and  $\blacktriangle \dashv \rightarrow$  adjunction relations stipulated above translate into the following clauses for every agent  $a$ , every functional action  $\alpha$ , every action  $\gamma$ , and every formula  $A$ :

$$(4.9) \quad \alpha \Delta_0 A \leq B \text{ iff } A \leq \alpha \rightarrow_0 B \quad \alpha \blacktriangle_0 A \leq B \text{ iff } A \leq \alpha \rightarrow_0 B$$

$$(4.10) \quad \gamma \Delta_1 A \leq B \text{ iff } A \leq \gamma \rightarrow_1 B \quad \gamma \blacktriangle_1 A \leq B \text{ iff } A \leq \gamma \rightarrow_1 B$$

$$(4.11) \quad a \Delta_2 A \leq B \text{ iff } A \leq a \rightarrow_2 B \quad a \blacktriangle_2 A \leq B \text{ iff } A \leq a \rightarrow_2 B.$$

The adjunction relations  $\Delta_1 \dashv \blacktriangleleft_1$  and  $\blacktriangle_1 \dashv \triangleleft_1$  translate into the following clauses for every action  $\gamma$  and every formula  $A$ :

$$(4.12) \quad \gamma \Delta_1 A \leq B \text{ iff } \gamma \leq B \blacktriangleleft_1 A \quad \gamma \blacktriangle_1 A \leq B \text{ iff } \gamma \leq B \triangleleft_1 A.$$

As we will see, the display postulates corresponding to triangle- and arrow-shaped connectives are modelled over the conditions (4.9)-(4.12) above. Also the display postulates involving virtual adjoints are shaped in the same way, which explains their name.

**Translating D'.EAK into the multi-type setting.** The intended link between the language of D'.EAK (cf. Section 2.4) and the language of the Dynamic Calculus is illustrated in the following table:

$\langle \alpha \rangle A$	becomes	$\alpha \Delta_0 A$		$\widehat{\alpha} A$	becomes	$\alpha \blacktriangle_0 A$
$\langle a \rangle A$	becomes	$a \Delta_2 A$		$\widehat{a} A$	becomes	$a \blacktriangle_2 A$
$[\alpha] A$	becomes	$\alpha \rightarrow_0 A$		$\overline{\alpha} A$	becomes	$\alpha \rightarrow_0 A$
$[a] A$	becomes	$a \rightarrow_2 A$		$\overline{a} A$	becomes	$a \rightarrow_2 A$
$1_\alpha$	becomes	$\alpha \Delta_0 \top$				

The table above can be extended to the definition of a formal translation between the operational language of D'.EAK and that of the Dynamic Calculus, simply

by preserving the non modal propositional fragment. We omit the details of this straightforward inductive definition. In Section 5, this translation will be elaborated on, and the interpretation of the language of the Dynamic Calculus in the final coalgebra will be defined so that the translation above preserves the validity of sequents. In the light of this translation, the adjunction conditions in clauses (4.9) correspond to the adjunction conditions (2.5) in  $D'.EAK$ , which, in their turn, motivate the display postulates reported on in Section 2.4:

$$\langle \alpha \rangle \dashv \overline{\alpha} \quad \widehat{\alpha} \dashv [\alpha].$$

The connectives  $\Delta_3$  and  $\blacktriangle_3$  have no counterpart in the language of  $D'.EAK$ , but the introduction of  $\blacktriangle_3$  is exactly what brings the additional expressiveness we need in order to eliminate the label. Indeed, we stipulate that for every  $a$  and  $\alpha$  as above,

$$a \blacktriangle_3 \alpha = \bigvee \{ \beta \mid \alpha a \beta \}. \quad (4.13)$$

A way to understand this stipulation is in the light of the discussion in [15, Section 4.3] after clause (8). There, in the context of a discussion about the proof system in [1], the link between the semantic condition  $f_A^M(m \star q) \leq f_A^M(m) \star f_A^Q(q)$  (cf. [1, Definitions 2.2(2) and 2.3]) and the axiom (2.4)—which in [1] was left implicit—is made more explicit, by understanding the action  $f_A^Q(q)$  as the join, taken in  $Q$ , of all the actions  $q'$  which are indistinguishable from  $q$  for the agent  $A$ . In the present setting, the stipulation (4.13) says that  $a \blacktriangle_3 \alpha$  encodes exactly the same information encoded in  $f_A^Q(q)$ , namely, the nondeterministic choice between all the actions that are indistinguishable from  $\alpha$  for the agent  $a$ .

**Additional conditions.** As was the case in the setting of  $D'.EAK$ , in order to express in this new language that e.g.  $\langle \alpha \rangle$  and  $[\alpha]$  are “interpreted over the same relation”, Sahlqvist correspondence theory (cf. e.g. [8, 9, 7] for a state-of-the-art-treatment) provides us with two alternatives: one of them is that we impose the following Fischer Servi-type conditions to hold for every  $a \in \text{Ag}$ ,  $\alpha \in \text{Fnc}$ ,  $\gamma \in \text{Act}$  and  $A, B \in \text{Fm}$ :

$$\begin{array}{ll} (\alpha \Delta_0 A) \rightarrow (\alpha \rightarrow_0 B) \leq \alpha \rightarrow_0 (A \rightarrow B) & (\alpha \blacktriangle_0 A) \rightarrow (\alpha \rightarrow_0 B) \leq \alpha \rightarrow_0 (A \rightarrow B) \\ (\gamma \Delta_1 A) \rightarrow (\gamma \rightarrow_1 B) \leq \gamma \rightarrow_1 (A \rightarrow B) & (\gamma \blacktriangle_1 A) \rightarrow (\gamma \rightarrow_1 B) \leq \gamma \rightarrow_1 (A \rightarrow B) \\ (a \Delta_2 A) \rightarrow (a \rightarrow_2 B) \leq a \rightarrow_2 (A \rightarrow B) & (a \blacktriangle_2 A) \rightarrow (a \rightarrow_2 B) \leq a \rightarrow_2 (A \rightarrow B). \end{array}$$

$$\begin{array}{ll} \alpha \Delta_0 (A \succ B) \leq (\alpha \rightarrow_0 A) \succ (\alpha \Delta_0 B) & \alpha \blacktriangle_0 (A \succ B) \leq (\alpha \rightarrow_0 A) \succ (\alpha \blacktriangle_0 B) \\ \gamma \Delta_1 (A \succ B) \leq (\gamma \rightarrow_1 A) \succ (\gamma \Delta_1 B) & \gamma \blacktriangle_1 (A \succ B) \leq (\gamma \rightarrow_1 A) \succ (\gamma \blacktriangle_1 B) \\ a \Delta_2 (A \succ B) \leq (a \rightarrow_2 A) \succ (a \Delta_2 B) & a \blacktriangle_2 (A \succ B) \leq (a \rightarrow_2 A) \succ (a \blacktriangle_2 B). \end{array}$$

To see that the conditions above correspond to the usual Fischer Servi axioms in standard modal languages, one can observe that the conditions in the first and third line above are images, under the translation discussed above, of the Fischer



Servi axioms reported on in Section 2.2). The second alternative is to impose that, for every  $0 \leq i \leq 2$ , the connectives  $\Delta_i$  and  $\blacktriangle_i$  yield *conjugated diamonds* (cf. discussion in [15, Section 6.2]); that is, the following inequalities hold for all  $\mathbf{a} \in \mathbf{Ag}$ ,  $\alpha, \beta \in \mathbf{Fnc}$ , and  $A, B \in \mathbf{Fm}$ :

$$\begin{array}{ll} (\alpha \Delta_0 A) \wedge B \leq \alpha \Delta_0 (A \wedge \alpha \blacktriangle_0 B) & (\alpha \blacktriangle_0 A) \wedge B \leq \alpha \blacktriangle_0 (A \wedge \alpha \Delta_0 B) \\ (\gamma \Delta_1 A) \wedge B \leq \gamma \Delta_1 (A \wedge \gamma \blacktriangle_1 B) & (\gamma \blacktriangle_1 A) \wedge B \leq \gamma \blacktriangle_1 (A \wedge \gamma \Delta_1 B) \\ (\mathbf{a} \Delta_2 A) \wedge B \leq \mathbf{a} \Delta_2 (A \wedge \mathbf{a} \blacktriangle_2 B) & (\mathbf{a} \blacktriangle_2 A) \wedge B \leq \mathbf{a} \blacktriangle_2 (A \wedge \mathbf{a} \Delta_2 B). \end{array}$$

$$\begin{array}{ll} \alpha \rightarrow_0 (A \vee \alpha \rightarrow_0 B) \leq (\alpha \rightarrow_0 A) \vee B & \alpha \rightarrow_0 (A \vee \alpha \rightarrow_0 B) \leq (\alpha \rightarrow_0 A) \vee B \\ \gamma \rightarrow_1 (A \vee \gamma \rightarrow_1 B) \leq (\gamma \rightarrow_1 A) \vee B & \gamma \rightarrow_1 (A \vee \gamma \rightarrow_1 B) \leq (\gamma \rightarrow_1 A) \vee B \\ \mathbf{a} \rightarrow_2 (A \vee \mathbf{a} \rightarrow_2 B) \leq (\mathbf{a} \rightarrow_2 A) \vee B & \mathbf{a} \rightarrow_2 (A \vee \mathbf{a} \rightarrow_2 B) \leq (\mathbf{a} \rightarrow_2 A) \vee B. \end{array}$$

The conditions in the first and third line above are images, under the translation discussed above, of the conjugation conditions reported on in [15, Section 6.2].

**The operational language, formally.** Let us introduce the operational terms of the multi-type language by the following simultaneous induction, based on sets  $\mathbf{AtProp}$  of atomic propositions,  $\mathbf{Fnc}$  of functional actions, and  $\mathbf{Ag}$  of agents:

$$\begin{array}{l} \mathbf{Fm} \ni A ::= p \mid \perp \mid \top \mid A \wedge A \mid A \vee A \mid A \rightarrow A \mid A \succ A \mid \\ \quad \alpha \Delta_0 A \mid \alpha \rightarrow_0 A \mid \gamma \Delta_1 A \mid \gamma \rightarrow_1 A \mid \mathbf{a} \Delta_2 A \mid \mathbf{a} \rightarrow_2 A \mid \\ \quad \alpha \blacktriangle_0 A \mid \alpha \rightarrow_0 A \mid \gamma \blacktriangle_1 A \mid \gamma \rightarrow_1 A \mid \mathbf{a} \blacktriangle_2 A \mid \mathbf{a} \rightarrow_2 A \\ \mathbf{Fnc} \ni \alpha ::= \alpha \\ \mathbf{Act} \ni \gamma ::= \mathbf{a} \blacktriangle_3 \alpha \mid \mathbf{a} \Delta_3 \alpha \\ \mathbf{Ag} \ni \mathbf{a} ::= \mathbf{a} \end{array}$$

The fundamental difference between the language above and the language of D'.EAK is that, in D'.EAK, agents and actions are *parametric indexes* in the construction of formulas, which are the only first-class citizens. In the present setting, however, each type lives on a par with any other. Because of the relative simplicity of the EAK setting, two of the four types are attributed no algebraic structure at the operational level. However, it is not difficult to enrich the algebraic structure of those types with sensible and intuitive operations: for instance, the skip and crash actions are functional, and parallel and sequential composition and iteration on functional actions preserve functionality, hence can be added to the array of constructors for  $\mathbf{Fnc}$ . As a consequence of the fact that each type is a first-class citizen, as we will see shortly, four types of structures will be defined, and the turnstile symbol in the sequents of this calculus will be interpreted in the appropriate domain.

**On the meta-linguistic labels  $\alpha\mathbf{a}\beta$ .** Let us illustrate how the label  $\alpha\mathbf{a}\beta$  can be subsumed when translating D'.EAK-formulas in the multi-type language. Con-

sider for example (the intuitionistic counterparts of) the following axiom (cf. (2.4)):

$$(Pre(\alpha) \rightarrow \bigwedge \{[a][\beta]A \mid \alpha a \beta\}) \rightarrow [\alpha][a]A$$

By applying the translation above we get:

$$(\alpha \Delta_0 \top \rightarrow \bigwedge \{a \rightarrow_2 (\beta \rightarrow_0 A) \mid \alpha a \beta\}) \rightarrow \alpha \rightarrow_0 (a \rightarrow_2 A).$$

Since (the semantic interpretation of)  $\rightarrow_2$  is completely meet-preserving in the second coordinate, the clause above is semantically equivalent to the following one:

$$(\alpha \Delta_0 \top \rightarrow [a \rightarrow_2 \bigwedge \{\beta \rightarrow_0 A \mid \alpha a \beta\}]) \rightarrow \alpha \rightarrow_0 (a \rightarrow_2 A).$$

The next step is the only place of the chapter in which we will need to assume that (the domains of interpretation of  $\mathbf{Fcn}$  and  $\mathbf{Act}$  are such that)  $\mathbf{Fcn} \subseteq \mathbf{Act}$ . Under this assumption,  $\rightarrow_0$  can be taken as the restriction of  $\rightarrow_1$ . By general order-theoretic facts (see e.g. [10]), the latter is completely join-reversing in its first coordinate. Hence, we can equivalently rewrite the clause above as follows:

$$(\alpha \Delta_0 \top \rightarrow [a \rightarrow_2 (\bigvee \{\beta \mid \alpha a \beta\} \rightarrow_1 A)]) \rightarrow \alpha \rightarrow_0 (a \rightarrow_2 A).$$

Now we apply the stipulation (4.13) and get the following :

$$(\alpha \Delta_0 \top \rightarrow [a \rightarrow_2 ((a \blacktriangle_3 \alpha) \rightarrow_1 A)]) \rightarrow \alpha \rightarrow_0 (a \rightarrow_2 A). \quad (4.14)$$

An analogous argument justifies that the following axiom:

$$\langle \alpha \rangle \langle a \rangle A \rightarrow (Pre(\alpha) \wedge \bigvee \{\langle a \rangle \langle \beta \rangle A \mid \alpha a \beta\})$$

corresponds to:

$$\alpha \Delta_0 (a \Delta_2 A) \rightarrow (\alpha \Delta_0 \top \wedge a \Delta_2 [(a \blacktriangle_3 \alpha) \Delta_1 A]). \quad (4.15)$$

Without appealing to  $\mathbf{Fcn} \subseteq \mathbf{Act}$ , we could take the correspondences above as primitive stipulations.

**Structural language, formally.** As discussed in the preliminaries, display calculi manipulate two closely related languages: the operational and the structural. Let us introduce the structural language of the Dynamic Calculus, which as usual matches the operational language, although in the present case not in the same way as in  $\mathbf{D'.EAK}$ . We have *formula-type structures*, *functional action-type structures*, *action-type structures*, *agent-type structures*, defined by simultaneous recursion as

follows:

$$\text{FM} \ni X ::= A \mid \mathbf{I} \mid X ; X \mid X > X \mid$$

$$F \triangle_0 X \mid F \triangleright_0 X \mid \Gamma \triangle_1 X \mid \Gamma \triangleright_1 X \mid \mathbf{a} \triangle_2 X \mid \mathbf{a} \triangleright_2 X \mid$$

$$F \blacktriangle_0 X \mid F \blacktriangleright_0 X \mid \Gamma \blacktriangle_1 X \mid \Gamma \blacktriangleright_1 X \mid \mathbf{a} \blacktriangle_2 X \mid \mathbf{a} \blacktriangleright_2 X$$

$$\text{FNC} \ni F ::= \alpha \mid X \triangleleft_{\sim_0} X \mid X \blacktriangleleft_{\sim_0} X \mid \mathbf{a} \sim_{\triangleright_3} \Gamma \mid \mathbf{a} \sim_{\triangleright_3} \Gamma$$

$$\text{ACT} \ni \Gamma ::= \mathbf{a} \blacktriangle_3 F \mid \mathbf{a} \triangle_3 F \mid X \triangleleft_1 X \mid X \blacktriangleleft_1 X$$

$$\text{AG} \ni \mathbf{a} ::= \mathbf{a} \mid X \triangleleft_{\sim_2} X \mid X \blacktriangleleft_{\sim_2} X \mid \Gamma \triangleleft_{\sim_3} F \mid \Gamma \blacktriangleleft_{\sim_3} F.$$

**The propositional base.** As is typical of display calculi, each operational connective corresponds to one structural connective. In particular, the propositional base connectives behave exactly as in D'.EAK, but for the sake of self-containment, we are going to report on these rules below:

Structural symbols	<	>	;	I				
Operational symbols	◀	←	▷	→	∧	∨	⊓	⊥

#### Structural Rules

$$\begin{array}{c}
 Id \frac{}{p \vdash p} \quad \frac{X \vdash A \quad A \vdash Y}{X \vdash Y} \text{Cut} \\
 I_L^1 \frac{X \vdash Y}{\mathbf{I} \vdash Y < X} \quad \frac{X \vdash Y}{X < Y \vdash \mathbf{I}} I_R^1 \\
 I_L^2 \frac{X \vdash Y}{\mathbf{I} \vdash X > Y} \quad \frac{X \vdash Y}{Y > X \vdash \mathbf{I}} I_R^2 \\
 IW_L \frac{\mathbf{I} \vdash X}{Y \vdash X} \quad \frac{X \vdash \mathbf{I}}{X \vdash Y} IW_R \\
 W_L^1 \frac{X \vdash Z}{Y \vdash Z < X} \quad \frac{X \vdash Z}{X < Z \vdash Y} W_R^1 \\
 W_L^2 \frac{X \vdash Z}{Y \vdash X > Z} \quad \frac{X \vdash Z}{Z > X \vdash Y} W_R^2 \\
 C_L \frac{X; X \vdash Y}{X \vdash Y} \quad \frac{Y \vdash X; X}{Y \vdash X} C_R \\
 E_L \frac{Y; X \vdash Z}{X; Y \vdash Z} \quad \frac{Z \vdash X; Y}{Z \vdash Y; X} E_R \\
 A_L \frac{X; (Y; Z) \vdash W}{(X; Y); Z \vdash W} \quad \frac{W \vdash (Z; Y); X}{W \vdash Z; (Y; X)} A_R
 \end{array}$$

**Display Postulates**

$$\begin{aligned}
 (<, <) & \frac{X; Y \vdash Z}{X \vdash Z < Y} & \frac{Z \vdash X; Y}{Z < Y \vdash X} (<, ;) \\
 (>, >) & \frac{X; Y \vdash Z}{Y \vdash X > Z} & \frac{Z \vdash X; Y}{X > Z \vdash Y} (>, ;)
 \end{aligned}$$

The classical base is obtained by adding the so-called *Grishin* rules (following e.g. [16]), which encode classical, but not intuitionistic validities:

$$Gri_L \frac{X > (Y; Z) \vdash W}{(X > Y); Z \vdash W} \quad \frac{W \vdash X > (Y; Z)}{W \vdash (X > Y); Z} Gri_R$$

**Operational Rules**

$$\begin{aligned}
 \perp_L & \frac{}{\perp \vdash \mathbf{I}} & \frac{X \vdash \mathbf{I}}{X \vdash \perp} \perp_R \\
 \top_L & \frac{\mathbf{I} \vdash X}{\top \vdash X} & \frac{}{\mathbf{I} \vdash \top} \top_R \\
 \wedge_L & \frac{A; B \vdash Z}{A \wedge B \vdash Z} & \frac{X \vdash A \quad Y \vdash B}{X; Y \vdash A \wedge B} \wedge_R \\
 \vee_L & \frac{A \vdash X \quad B \vdash Y}{A \vee B \vdash X; Y} & \frac{Z \vdash A; B}{Z \vdash A \vee B} \vee_R \\
 \leftarrow_L & \frac{B \vdash Y \quad X \vdash A}{B \leftarrow A \vdash Y < X} & \frac{Z \vdash B < A}{Z \vdash B \leftarrow A} \leftarrow_R \\
 \prec_L & \frac{B < A \vdash Z}{B \prec A \vdash Z} & \frac{Y \vdash B \quad A \vdash X}{Y < X \vdash B \prec A} \prec_R \\
 \rightarrow_L & \frac{X \vdash A \quad B \vdash Y}{A \rightarrow B \vdash X > Y} & \frac{Z \vdash A > B}{Z \vdash A \rightarrow B} \rightarrow_R \\
 \succ_L & \frac{A > B \vdash Z}{A \succ B \vdash Z} & \frac{A \vdash X \quad Y \vdash B}{X > Y \vdash A \succ B} \succ_R
 \end{aligned}$$

**Rules for heterogeneous connectives.** Unlike what was the case in the setting of D'.EAK, in the present setting, each heterogeneous structural connective is associated with at most one operational connective, as illustrated in the following table: for  $0 \leq i \leq 3$  and  $j \in \{0, 2\}$ ,

Structural symbols	$\triangle_i$	$\blacktriangle_i$	$\triangleright_j$	$\blacktriangleright_j$
Operational symbols	$\Delta_i$	$\blacktriangle_i$	$\triangleright_j$	$\blacktriangleright_j$

That is, structural connectives are to be interpreted as usual in a context-sensitive way, but the present language lacks the operational connectives which would correspond to them on one of the two sides. This is of course because in the present setting we do not need them. However, in a setting in which they would turn out to be needed, it would not be difficult to introduce the missing operational connectives. We can now introduce the operational rules for heterogeneous connectives. Let  $x, y$  stand for structures of an undefined type, and let  $a, b$  denote operational terms of the appropriate type. Then, for  $0 \leq i \leq 3$ ,

$$\begin{array}{c} \Delta_{iL} \frac{a \triangle_i b \vdash z}{a \Delta_i b \vdash z} \quad \frac{x \vdash a \quad y \vdash b}{x \triangle_i y \vdash a \Delta_i b} \Delta_{iR} \\ \blacktriangle_{iL} \frac{a \blacktriangle_i b \vdash z}{a \blacktriangle_i b \vdash z} \quad \frac{x \vdash a \quad y \vdash b}{x \blacktriangle_i y \vdash a \blacktriangle_i b} \blacktriangle_{iR} \end{array}$$

and for  $0 \leq i \leq 2$ ,

$$\begin{array}{c} \rightarrow_{iL} \frac{x \vdash a \quad B \vdash Y}{a \rightarrow_i B \vdash x \triangleright_i Y} \quad \frac{Z \vdash a \triangleright_i B}{Z \vdash a \rightarrow_i B} \rightarrow_{iR} \\ \blacktriangleright_{iL} \frac{x \vdash a \quad B \vdash Y}{a \rightarrow_i B \vdash x \blacktriangleright_i Y} \quad \frac{Z \vdash a \blacktriangleright_i B}{Z \vdash a \rightarrow_i B} \blacktriangleright_{iR} \end{array}$$

where  $B, Y, Z$  are formula-type operational and structural terms. Clearly, the rules in the two tables above for  $i = 0, 2$  yield the operational rules for the dynamic and epistemic modal operators under the translation given early on. Notice that each sequent is always interpreted in one domain. However, since heterogeneous connectives take arguments of different types (which justifies their name), premises of binary rules are of course interpreted in different domains.

Axioms will be given in three types<sup>9</sup>, as follows:

$$\mathbf{a} \vdash \mathbf{a} \quad \alpha \vdash \alpha \quad p \vdash p \quad \perp \vdash \mathbf{I} \quad \mathbf{I} \vdash \top$$

where the first and second axioms from the left are of type **Ag** and **Fnc** respectively, and the remaining ones are of type **Fm**. A generalization of  $p \vdash p$  will be added below to the system (see *atom* axiom on page 29).

Further, we allow the following *strongly type-uniform* (cf. Definition 3.2) cut rules on operational terms:

$$\frac{\mathbf{a} \vdash \mathbf{a} \quad \mathbf{a} \vdash \mathbf{b}}{\mathbf{a} \vdash \mathbf{b}} \quad \frac{F \vdash \alpha \quad \alpha \vdash G}{F \vdash G} \quad \frac{\Gamma \vdash \gamma \quad \gamma \vdash \Delta}{\Gamma \vdash \Delta} \quad \frac{X \vdash A \quad A \vdash Y}{X \vdash Y}$$

Next, we give the display postulates for heterogeneous connectives. In what follows, let  $x, y, z$  stand for structures of an undefined type. Then, for  $0 \leq i \leq 2$ ,

$$(\Delta_i, \rightarrow_i) \frac{x \triangle_i y \vdash z}{y \vdash x \blacktriangleright_i z} \quad \frac{x \blacktriangle_i y \vdash z}{y \vdash x \triangleright_i z} (\blacktriangle_i, \rightarrow_i)$$

For  $i = 1$ , we also have

$$(\Delta_1, \leftarrow_1) \frac{x \triangle_1 y \vdash z}{x \vdash z \blacktriangleleft_1 y} \quad \frac{x \blacktriangle_1 y \vdash z}{x \vdash z \triangleleft_1 y} (\blacktriangle_1, \leftarrow_1)$$

<sup>9</sup>Indeed, there is no axiom schema for atomic terms of type **Act**, because the language does not admit them.

The display postulates above involve structural connectives each of which has a semantic interpretation. In the following display postulates, the squiggly arrows are not semantically justified: they are the *virtual adjoints*, informally introduced at the beginning of the present Section 4, which will be discussed in detail in Section 7. For each  $i = 0, 2, 3$ , we have:

$$(\Delta_i, \leftarrow_i) \frac{x \triangle_i y \vdash z}{x \vdash z \blacktriangleleft_i y} \quad \frac{x \blacktriangle_i y \vdash z}{x \vdash z \blacktriangleleft_i y} (\blacktriangle_i, \leftarrow_i)$$

and for  $i = 3$ ,

$$(\Delta_3, \rightarrow_3) \frac{x \triangle_3 y \vdash z}{y \vdash x \blacktriangleright_3 z} \quad \frac{x \blacktriangle_3 y \vdash z}{y \vdash x \blacktriangleright_3 z} (\blacktriangle_3, \rightarrow_3)$$

Notice that sequents occurring in each display postulate involving heterogeneous connectives are *not* of the same type. However, it is easy to see that the display postulates preserve the type-uniformity (cf. Definition 3.1); that is, if the premise of any instance of a display postulate is a type-uniform sequent, then so is its conclusion. Next, the *necessitation*, *conjugation*, *Fischer Servi*, and *monotonicity* rules: for  $0 \leq i \leq 2$ ,

$$\begin{array}{l} (nec_i \Delta) \frac{I \vdash W}{x \triangle_i I \vdash W} \quad \frac{W \vdash I}{W \vdash x \triangleright_i I} (nec_i \rightarrow) \\ (nec_i \blacktriangle) \frac{I \vdash W}{x \blacktriangle_i I \vdash W} \quad \frac{W \vdash I}{W \vdash x \blacktriangleright_i I} (nec_i \rightarrow) \\ (conj_i \Delta) \frac{x \triangle_i ((x \blacktriangle_i Y); Z) \vdash W}{Y; (x \triangle_i Z) \vdash W} \quad \frac{W \vdash x \triangleright_i ((x \blacktriangleright_i Y); Z)}{W \vdash Y; (x \triangleright_i Z)} (conj_i \rightarrow) \\ (conj_i \blacktriangle) \frac{x \blacktriangle_i ((x \triangle_i Y); Z) \vdash W}{Y; (x \blacktriangle_i Z) \vdash W} \quad \frac{W \vdash x \blacktriangleright_i ((x \triangleright_i Y); Z)}{W \vdash Y; (x \blacktriangleright_i Z)} (conj_i \rightarrow) \\ (FS_i \Delta) \frac{(x \triangleright_i Y) > (x \triangle_i Z) \vdash W}{x \triangle_i (Y > Z) \vdash W} \quad \frac{W \vdash (x \triangle_i Y) > (x \triangleright_i Z)}{W \vdash x \triangleright_i (Y > Z)} (FS_i \rightarrow) \\ (FS_i \blacktriangle) \frac{(x \blacktriangleright_i Y) > (x \blacktriangle_i Z) \vdash W}{x \blacktriangle_i (Y > Z) \vdash W} \quad \frac{W \vdash (x \blacktriangle_i Y) > (x \blacktriangleright_i Z)}{W \vdash x \blacktriangleright_i (Y > Z)} (FS_i \rightarrow) \\ (mon_i \Delta) \frac{(x \triangle_i Y); (x \triangle_i Z) \vdash W}{x \triangle_i (Y; Z) \vdash W} \quad \frac{W \vdash (x \triangleright_i Y); (x \triangleright_i Z)}{W \vdash x \triangleright_i (Y; Z)} (mon_i \rightarrow) \\ (mon_i \blacktriangle) \frac{(x \blacktriangle_i Y); (x \blacktriangle_i Z) \vdash W}{x \blacktriangle_i (Y; Z) \vdash W} \quad \frac{W \vdash (x \blacktriangleright_i Y); (x \blacktriangleright_i Z)}{W \vdash x \blacktriangleright_i (Y; Z)} (mon_i \rightarrow) \end{array}$$

Next, we introduce the rules translating the interaction axioms between dynamic and epistemic modalities. In what follows we omit the subscripts, since the reading is unambiguous.

$$\begin{array}{c}
\text{swap-out}_L \frac{(\mathbf{a} \blacktriangle F) \blacktriangle (\mathbf{a} \blacktriangle X) \vdash Y}{\mathbf{a} \blacktriangle (F \blacktriangle X) \vdash Y} \qquad \frac{X \vdash (\mathbf{a} \blacktriangle F) \blacktriangleright (\mathbf{a} \blacktriangleright Y)}{X \vdash \mathbf{a} \blacktriangleright (F \blacktriangleright Y)} \text{swap-out}_R \\
\text{swap-in}_L \frac{\mathbf{a} \blacktriangle (F \blacktriangle X) \vdash Y}{(\mathbf{a} \blacktriangle F) \blacktriangle (\mathbf{a} \blacktriangle ((F \triangle I) ; X)) \vdash Y} \qquad \frac{X \vdash \mathbf{a} \blacktriangleright (F \blacktriangleright Y)}{X \vdash (\mathbf{a} \blacktriangle F) \blacktriangleright (\mathbf{a} \blacktriangleright ((F \triangle I) > Y))} \text{swap-in}_R
\end{array}$$

The structure  $(\mathbf{a} \blacktriangle F)$  in the *swap*-rules above has absorbed the labels  $\alpha\mathbf{a}\beta$  in the corresponding *swap*-rules of D'.EAK. Moreover, new *swap-out* rules are unary, whereas the corresponding ones in D'.EAK are of a non-fixed arity.

The following *atom* axiom translates the *atom* axiom of D'.EAK:

$$F_1 \circ (F_2 \circ \dots (F_n \circ p) \dots) \vdash G_1 \triangleright (G_2 \triangleright \dots (G_m \triangleright p) \dots)$$

where  $F_1, \dots, F_n, G_1, \dots, G_m \in \text{FNC}$ ,  $\circ \in \{ \triangle_0, \blacktriangle_0 \}$ ,  $\triangleright \in \{ \triangleright_0, \blacktriangleright_0 \}$  and  $n, m \in \mathbb{N}$ . In what follows, we sometimes indicate the *atom* axiom with the shorter symbol  $\Phi p \vdash \Psi p$ . Notice the following difference between the present *atom* axiom and the one of D'.EAK (cf. [15]): the structural variables  $F$ s and  $G$ s (which are typically instantiated as operational variables  $\alpha$  and  $\beta$  of type Fnc) translate what in the *atom* axiom of D'.EAK were indexes for logical connectives, whereas in the Dynamic Calculus, the operational variables contained in any instantiation of the  $F$ s and  $G$ s are first-class citizens, on the same ground as the operational variable  $p$  of type Fm. Hence we need to stipulate whether the introduction of each of these variables is parametric or not. As is customary in the literature on display calculi (cf. [3, Definition 4.1]), we stipulate that the only principal variables in *atom* are the  $p$ s, and all the other variable occurrences are parametric.

Finally, the following *balance* rule:

$$\frac{X \vdash Y}{F \triangle_0 X \vdash F \triangleright_0 Y}$$

is sound only for  $F \in \text{FNC}$ , and cannot be extended to an arbitrary actions.<sup>10</sup> In this rule, every variable occurrence is parametric, and each occurrence of  $F$  is only congruent to itself.

<sup>10</sup>To see this, notice that this rule instantiates to

$$\frac{p \vdash p}{F \triangle_0 p \vdash F \triangleright_0 p}$$

If the rule *balance* is to be sound, the validity of the premise implies the validity of the conclusion, which is the translation of the sequent  $\langle F \rangle p \vdash [F]p$ , which is equivalent to the axiom  $\langle F \rangle p \rightarrow [F]p$ . It is a well known fact from Sahlqvist theory that the latter axiom corresponds to the condition that the binary relation associated with  $\langle F \rangle$  and  $[F]$  is the graph of a partial function.

**Justifying the two types of actions.** As discussed in the introduction, one of the initial aims of the present paper was introducing a formal framework expressive enough so as to capture at the object-level the information encoded in the meta-linguistic label  $\alpha\beta$ . From the order-theoretic analysis at the beginning of the present section, it emerged that the additional expressivity encoded in the connective  $\blacktriangle_3$  and its interpretation (4.13) requires a semantic environment which cannot be restricted to functional actions. The introduction of the general type  $\text{Act}$  serves this purpose. However, the fact that the rule *balance* is only sound for functional actions is the reason why both types  $\text{Fnc}$  and  $\text{Act}$  are needed in order for the Dynamic Calculus to satisfy conditions  $\text{C}'_6$  and  $\text{C}'_7$  of Section 3.2. Indeed, the distinct type  $\text{Fnc}$  allows for the rule *balance* to be formulated so that all parametric variables occur unrestricted within each type.

## 5 Soundness

In the present section, we discuss the soundness of the rules of the Dynamic Calculus and prove that those which do not involve virtual adjoints (cf. Section 4) are sound with respect to the final coalgebra semantics. In [15, Section 5], basic facts about the final coalgebra have been collected and it is explained in detail how the rules of display calculi are to be interpreted in the final coalgebra. Here we will briefly recall some basics, and refer the reader to [15, Section 5] for a complete discussion.

Structures will be translated into operational terms of the appropriate type, and operational terms will be interpreted according to their type. Specifically, each atomic proposition  $p$  is assigned to a subset  $\llbracket p \rrbracket$  of the final coalgebra  $Z$ , each agent  $a$  a binary relation  $a_Z = \llbracket a \rrbracket$  on  $Z$  representing as usual  $a$ 's uncertainty about the world, and each functional actions  $\alpha$  is assigned a functional (i.e. deterministic) relation  $\alpha_Z = \llbracket \alpha \rrbracket \subseteq Z \times Z$  subject to the restriction defining the specific feature of epistemic actions, namely, that for all  $z, z' \in Z$ , if  $z\alpha z'$ , then  $z \in \llbracket p \rrbracket$  iff  $z' \in \llbracket p \rrbracket$  for every atomic proposition  $p$ .

Further, each agent  $a$  is associated with an auxiliary binary relation  $a_{\text{Fnc}}$  on the domain of interpretation of  $\text{Fnc}$ , which is the collection of graphs of partial functions having subsets of  $Z$  as domain and range. For each agent  $a$ , the relation  $a_{\text{Fnc}}$  represents  $a$ 's uncertainty about which action takes place).

In order to translate structures as operational terms, structural connectives need to be translated as logical connectives. To this effect, non-modal structural connectives are associated with pairs of logical connectives, and any given occurrence of a structural connective is translated as one or the other, according to its (antecedent or succedent) position. The following table illustrates how to translate each propositional structural connective of type  $\text{FM}$ , in the upper row, into one or the other of the logical connectives corresponding to it on the lower row: the one on the left-hand (resp. right-hand) side, if the structural connective occurs in precedent (resp. succedent) position.



Structural symbols	<	>	;	I				
Operational symbols	◁	←	▷	→	∧	∨	⊤	⊥

Recall that, in the Boolean setting treated here, the connectives  $\triangleleft$  and  $\triangleright$  are interpreted as  $A \triangleleft B := A \wedge \neg B$  and  $A \triangleright B := \neg A \wedge B$ .

The soundness of structural and operational rules which only involve active components of type FM has been discussed in [15] and is here therefore omitted.

As to the heterogeneous connectives, their translation into the corresponding operational connectives is indicated in the table below, to be understood similarly to the one above, where the index  $i$  ranges over  $\{0, 1, 2, 3\}$  for the triangles and over  $\{0, 1, 2\}$  for the arrows.

Structural symbols	$\triangle_i$	$\blacktriangle_i$	$\triangleright_i$	$\blacktriangleright_i$
Operational symbols	$\Delta_i$	$\blacktriangleright_i$	$\rightarrow_i$	$\rightarrow_i$

The interpretation of the heterogeneous connectives involving formulas and agents corresponds to that of the well-known forward and backward modalities discussed in Section 2.4 (below on the right-hand side we recall the notation of D'.EAK):

$$\begin{aligned}
\llbracket \mathbf{a} \Delta_2 A \rrbracket &= \{z \in Z \mid \exists z'. z \mathbf{a}_Z z' \ \& \ z' \in \llbracket A \rrbracket\} && \langle \mathbf{a} \rangle A \\
\llbracket \mathbf{a} \blacktriangle_2 A \rrbracket &= \{z \in Z \mid \exists z'. z' \mathbf{a}_Z z \ \& \ z' \in \llbracket A \rrbracket\} && \widehat{\mathbf{a}} A \\
\llbracket \mathbf{a} \rightarrow_2 A \rrbracket &= \{z \in Z \mid \forall z'. z \mathbf{a}_Z z' \Rightarrow z' \in \llbracket A \rrbracket\} && [\mathbf{a}] A \\
\llbracket \mathbf{a} \blackrightarrow_2 A \rrbracket &= \{z \in Z \mid \forall z'. z' \mathbf{a}_Z z \Rightarrow z' \in \llbracket A \rrbracket\} && \overline{[\mathbf{a}]} A
\end{aligned}$$

The connectives  $\Delta_0$ ,  $\rightarrow_0$ ,  $\blacktriangle_0$ ,  $\blackrightarrow_0$ , involving formulas and functional actions, are interpreted in the same way, replacing the relation  $\mathbf{a}_Z$  with the deterministic relations  $\alpha_Z$ . From the definitions above, it immediately follows for any  $\alpha \in \text{Fnc}$ , we have  $\llbracket \alpha \Delta_0 \top \rrbracket = \text{dom}(\alpha_Z)$ , where the set

$$\text{dom}(\alpha_Z) := \{z \in Z \mid \exists z' (z' \in Z \ \& \ z \alpha_Z z')\}$$

is the *domain* of  $\alpha_Z$ .

It can also be readily verified that, after having fixed the relations interpreting all  $\alpha$ s and  $\mathbf{a}$ s, the translation of Section 4 preserves the semantic interpretation, that is,  $\llbracket A \rrbracket = \llbracket A' \rrbracket$  for any D'.EAK formula  $A$ , where  $A'$  denotes the translation of  $A$  in the language of the Dynamic Calculus.

The auxiliary relations  $\mathbf{a}_{\text{Fnc}} = \llbracket \mathbf{a} \rrbracket^{\text{Fnc}}$  are used to define the interpretations of  $\Delta_3$ - and  $\blacktriangle_3$ -operational terms. Following 4.13, we let

$$\begin{aligned}
\llbracket \mathbf{a} \blacktriangle_3 \alpha \rrbracket &= \bigcup \{G \mid \alpha_Z \mathbf{a}_{\text{Fnc}} G\}, \\
\llbracket \mathbf{a} \Delta_3 \alpha \rrbracket &= \bigcup \{G \mid G \mathbf{a}_{\text{Fnc}} \alpha_Z\}.
\end{aligned}$$

The connectives  $\Delta_1$ ,  $\rightarrow_1$ ,  $\blacktriangle_1$ ,  $\blackrightarrow_1$ , involving Act-type operational terms  $\gamma$ , are interpreted in the same way as the 0- and 2- indexed connectives, replacing the relation  $\mathbf{a}_Z$  with the interpretation of the appropriate operational term  $\gamma$  of type Act.

The soundness of all operational rules for heterogeneous connectives immediately follows from the fact that their semantic counterparts as defined above are monotone or antitone in each coordinate.

The soundness of the rule *balance* immediately follows from the fact that the functional actions are interpreted as deterministic relations (for more details cf. [15, Section 6.2]).

The soundness of the cut-rules follows from the transitivity of the inclusion relation in the domain of interpretation of each type.

The soundness of the *Atom* axioms is argued similarly to that of the *Atom* axioms of the system D'.EAK, crucially using the fact that epistemic actions do not change the factual states of affairs (cf. [15, Section 6.2]).

The display rules  $(\Delta_i, \rightarrow_i)$  and  $(\blacktriangle_i, \rightarrow_i)$  for  $0 \leq i \leq 2$ , and  $(\Delta_1, \leftarrow_1)$  and  $(\blacktriangle_1, \leftarrow_1)$  are sound as the semantics of the triangle and arrow connectives form adjoint pairs.

On the other hand, in the display rules  $(\Delta_3, \rightsquigarrow_3)$ ,  $(\blacktriangle_3, \rightsquigarrow_3)$ ,  $(\Delta_i, \leftarrow_i)$  and  $(\blacktriangle_i, \leftarrow_i)$  for  $i = 0, 2, 3$ , the arrow-connectives are what we call *virtual adjoints* (cf. Section 4), that is, they do not have a semantic interpretation. In the next section, we will account for the fact that their presence in the calculus is safe.

Soundness of necessitation, conjugation, Fischer Servi, and monotonicity rules is straightforward and proved as in [15]. In the remainder of the section, we discuss the soundness of the new rules swap-in and swap-out recalled below.

**Fact 5.1.** *The following defining clause for the interpretation of  $\blacktriangle_1$ -operational terms*

$$\llbracket \gamma \blacktriangle_1 A \rrbracket = \{z \in Z \mid \exists z'. z' \gamma_Z z \ \& \ z' \in \llbracket A \rrbracket\}$$

*immediately implies that the semantic interpretation of  $\blacktriangle_1$  is completely  $\cup$ -preserving in its first coordinate.*

*Proof.* If  $\gamma_Z = \bigcup_{i \in I} \beta_i$ , then clearly  $z' \gamma_Z z$  iff  $z' \beta_i z$  for some  $i \in I$ . □

As to the soundness of *swap-out<sub>L</sub>*, assume that the structures  $\mathfrak{a}$ ,  $F$ ,  $X$  and  $Y$  have been given the following interpretations, according to their type, as discussed above:  $\mathfrak{a}_Z \subseteq Z \times Z$ ,  $\mathfrak{a}_{\text{Fnc}}$  is a binary relation on graphs of partial functions on  $Z$ ,  $F_Z$  is a functional relation on  $Z$ , and  $X_Z, Y_Z \subseteq Z$ . Let

$$\mathfrak{a}_Z \blacktriangle_3 F_Z := \bigcup \{\beta \mid F_Z \mathfrak{a}_{\text{Fnc}} \beta\}.$$

Assume that the premise of *swap-out<sub>L</sub>* is satisfied. That is:

$$\widehat{\mathfrak{a}_Z \blacktriangle_3 F_Z} \widehat{\mathfrak{a}_Z} X_Z \subseteq Y_Z,$$

where the symbols  $\widehat{\mathfrak{a}_Z \blacktriangle_3 F_Z}$  and  $\widehat{\mathfrak{a}_Z}$  denote the semantic diamond operations associated with the converses of the relations  $\mathfrak{a}_Z \blacktriangle_3 F_Z$  and  $\mathfrak{a}_Z$  respectively. Then, the following chain of equivalences holds:

$$\begin{aligned}
\widehat{\mathbf{a}_Z \blacktriangle_3 F_Z} \widehat{\mathbf{a}_Z} X_Z \subseteq Y_Z & \text{ iff } \bigcup \{ \widehat{G} \widehat{\mathbf{a}_Z} X_Z \mid F_Z \mathbf{a}_{\text{Fnc}} G \} \subseteq Y_Z & (\text{fact 5.1}) \\
& \text{ iff } \widehat{G} \widehat{\mathbf{a}_Z} X_Z \subseteq Y_Z \text{ for every } G \text{ s.t. } F_Z \mathbf{a}_{\text{Fnc}} G \\
& \text{ iff } X_Z \subseteq [\mathbf{a}_Z][G]Y_Z \text{ for every } G \text{ s.t. } F_Z \mathbf{a}_{\text{Fnc}} G \\
& \text{ iff } X_Z \subseteq \bigcap \{ [\mathbf{a}_Z][G]Y_Z \mid F_Z \mathbf{a}_{\text{Fnc}} G \} \\
\text{hence } X_Z & \subseteq (\text{dom}(F_Z))^c \cup \bigcap \{ [\mathbf{a}_Z][G]Y_Z \mid F_Z \mathbf{a}_{\text{Fnc}} G \}.
\end{aligned}$$

Consider the new variables  $p, q, \mathbf{a}, \alpha$ , and  $\beta_i$  for each  $G_i$  such that  $F_Z \mathbf{a}_{\text{Fnc}} G_i$ . Let us stipulate that  $\llbracket p \rrbracket := X_Z$ ,  $\llbracket q \rrbracket := Y_Z$ ,  $\llbracket \mathbf{a} \rrbracket := \mathbf{a}_Z$ ,  $\llbracket \alpha \rrbracket := F_Z$ , and  $\llbracket \beta_i \rrbracket := G_i$ . Hence  $\llbracket \text{Pre}(\alpha) \rrbracket = \llbracket \alpha \Delta_0 \top \rrbracket = \text{dom}(F_Z)$ . Therefore, the computation above can continue as follows:

$$\begin{aligned}
& X_Z \subseteq (\text{dom}(F_Z))^c \cup \bigcap \{ [\mathbf{a}_Z][G]Y_Z \mid F_Z \mathbf{a}_{\text{Fnc}} G \}. \\
\text{iff } \llbracket p \rrbracket & \subseteq \llbracket \text{Pre}(\alpha) \rightarrow \bigwedge \{ [\mathbf{a}][\beta]q \mid \alpha \mathbf{a} \beta \} \rrbracket \\
\text{iff } \llbracket p \rrbracket & \subseteq \llbracket [\alpha][\mathbf{a}]q \rrbracket \\
\text{iff } X_Z & \subseteq [F_Z][\mathbf{a}_Z]Y_Z \\
\text{iff } \widehat{\mathbf{a}_Z} \widehat{F_Z} X_Z & \subseteq Y_Z
\end{aligned}$$

which completes the proof of the soundness of  $\text{swap-out}_L$ . The proof of the soundness of the remaining swap-rules is similar.

## 6 Completeness and cut elimination

In 6.1, we discuss the completeness of the Dynamic Calculus w.r.t. the final coalgebra semantics. We show that the translation (cf. Section 4) of each of the EAK axioms is derivable in the Dynamic Calculus. Our proof is indirect, and relies on the fact that EAK is complete w.r.t. the final coalgebra semantics, and that the translation preserves the semantic interpretation on the final coalgebra (as discussed in Section 5). In 6.2, we show that the Dynamic Calculus is quasi-properly displayable (cf. Section 3.2). By Theorem 3.3, this is enough to establish that the calculus enjoys cut elimination and the subformula property.

### 6.1 Derivable rules and completeness

In what follows,  $\mathbf{a}$  and  $\alpha$  are atomic variables (and also the generic operational terms) of type  $\text{Ag}$  and  $\text{Fnc}$  respectively, and  $A, B$  are generic operational terms of type  $\text{Fm}$ . Since the reading is unambiguous, in the remainder of the present paper the indexes of the heterogeneous connectives are dropped.

Under the stipulations above, the translations of the rules *reduce* from  $\text{D'.EAK}$  (cf. Section 2.4) can be derived in the Dynamic Calculus as follows.



dynamic boxes and diamonds are taken as primitive connectives. When moving to a propositional base which is weaker than the Boolean one, also the diamond/box interaction axioms such as the following one become primitive:

$$[\alpha]\langle a \rangle A \leftrightarrow 1_\alpha \rightarrow \bigvee \{ \langle a \rangle \langle \beta \rangle A \mid \alpha a \beta \}.$$

The axiom above translates as:

$$\alpha \rightarrow (a \Delta A) \leftrightarrow \alpha \Delta \top \rightarrow a \Delta ((a \blacktriangle \alpha) \Delta A).$$

$$\begin{array}{c}
\frac{\frac{\frac{\frac{a \vdash a \quad \alpha \vdash \alpha}{a \blacktriangle \alpha \vdash a \blacktriangle \alpha} \quad A \vdash A}{a \vdash a \quad (a \blacktriangle \alpha) \Delta A \vdash (a \blacktriangle \alpha) \Delta A}}{a \Delta ((a \blacktriangle \alpha) \Delta A) \vdash a \Delta ((a \blacktriangle \alpha) \Delta A)}}{(a \blacktriangle \alpha) \Delta A \vdash a \blacktriangleright (a \Delta ((a \blacktriangle \alpha) \Delta A))}}{A \vdash (a \blacktriangle \alpha) \blacktriangleright (a \blacktriangleright (a \Delta ((a \blacktriangle \alpha) \Delta A)))} \text{ swap-out}_R \\
\frac{A \vdash a \blacktriangleright (\alpha \blacktriangleright (a \Delta ((a \blacktriangle \alpha) \Delta A)))}{a \Delta A \vdash \alpha \blacktriangleright (a \Delta ((a \blacktriangle \alpha) \Delta A))}}{\frac{\alpha \vdash \alpha \quad a \Delta A \vdash \alpha \blacktriangleright (a \Delta ((a \blacktriangle \alpha) \Delta A))}{\alpha \rightarrow (a \Delta A) \vdash \alpha \blacktriangleright (\alpha \blacktriangleright (a \Delta ((a \blacktriangle \alpha) \Delta A)))}}{\alpha \blacktriangle (\alpha \rightarrow (a \Delta A)) \vdash \alpha \blacktriangleright (a \Delta ((a \blacktriangle \alpha) \Delta A))}}{\frac{I \vdash (\alpha \blacktriangle (\alpha \rightarrow (a \Delta A))) > (\alpha \blacktriangleright (a \Delta ((a \blacktriangle \alpha) \Delta A)))}{\alpha \blacktriangle (\alpha \rightarrow (a \Delta A)); I \vdash \alpha \blacktriangleright (a \Delta ((a \blacktriangle \alpha) \Delta A))}}{\text{conj}_0 \Delta \frac{\alpha \Delta (\alpha \blacktriangle (\alpha \rightarrow (a \Delta A)); I) \vdash a \Delta ((a \blacktriangle \alpha) \Delta A)}{(\alpha \rightarrow (a \Delta A)); (\alpha \Delta I) \vdash a \Delta ((a \blacktriangle \alpha) \Delta A)}}}{\frac{\alpha \Delta I \vdash (\alpha \rightarrow (a \Delta A)) > (a \Delta ((a \blacktriangle \alpha) \Delta A))}{I \vdash \alpha \blacktriangleright (\alpha \rightarrow (a \Delta A)) > (a \Delta ((a \blacktriangle \alpha) \Delta A))}}{\frac{T \vdash \alpha \blacktriangleright (\alpha \rightarrow (a \Delta A)) > (a \Delta ((a \blacktriangle \alpha) \Delta A))}{\alpha \Delta \top \vdash (\alpha \rightarrow (a \Delta A)) > (a \Delta ((a \blacktriangle \alpha) \Delta A))}}{\frac{\alpha \Delta \top \vdash (\alpha \rightarrow (a \Delta A)) > (a \Delta ((a \blacktriangle \alpha) \Delta A))}{(\alpha \rightarrow (a \Delta A)); \alpha \Delta \top \vdash a \Delta ((a \blacktriangle \alpha) \Delta A)}}}{\frac{\alpha \rightarrow (a \Delta A) \vdash \alpha \Delta \top > a \Delta ((a \blacktriangle \alpha) \Delta A)}{\alpha \rightarrow (a \Delta A) \vdash \alpha \Delta \top \rightarrow a \Delta ((a \blacktriangle \alpha) \Delta A)}}
\end{array}$$

For the other direction, recall that the counterpart of the rule *reduce* is derivable in the Dynamic Calculus 33:



As to the inductive step, one can verify by inspection that all the rules of the Dynamic Calculus preserve type-uniformity, and that the Cut rules are strongly type-uniform.  $\square$

As to condition  $C'_3$ , all parameters in all but the *swap-in* rules satisfy the condition of non-proliferation. In each *swap-in* rule, the parameters of type **Ag** and **Fnc** in the premise are congruent to *two* parameters in the conclusion. However, it is not difficult to see that in each derivation, each application of any cut rule

$$\frac{\begin{array}{c} \vdots \\ x \vdash a \end{array} \quad \begin{array}{c} \vdots \\ a \vdash y \end{array}}{x \vdash y}$$

of type **Ag** or **Fnc** must be such that the structure  $x$  reduces to the atomic term  $a$ . Indeed, because the sequent  $x \vdash a$  is derivable, by Proposition 6.1 it must be type uniform, that is, the structure  $x$  needs to be of type **AG** if  $a$  is, or of type **FNC** if  $a$  is. If  $x$  was not atomic, then its main structural connective would be a squiggly arrow  $\triangleleft\sim$  or  $\blacktriangleleft\sim$ . Because these connectives do not have any operational counterpart, such a structure cannot have been introduced by an application of an operational rule. Hence, the only remaining possibility is that it has been introduced via a display postulate. But also this case is impossible, since in display postulates introduce these connectives only in the succedent, and  $x$  is in precedent position. This finishes the verification of condition  $C'_3$ .

Finally, the verification steps for  $C'_8$  are collected in Section 9.1.

## 7 Conservativity

In the definition of the language of the Dynamic Calculus, we have adopted a rather inclusive policy. That is, the operational language includes almost all the logical symbols which could be assigned a natural interpretation purely on the basis of reasonable assumptions on the order-theoretic properties of the domains of interpretation of the various types of terms, the only exception being the connectives  $\blacktriangleleft_1$  and  $\triangleleft_1$ , which are excluded from the language although they are semantically justified. A very useful and powerful consequence of the fact that the Dynamic Calculus enjoys cut elimination Belnap-style is that this cut elimination is then inherited by the subcalculi corresponding to each fragment of the operational language of the Dynamic Calculus which verify as they stand the assumptions of Theorem 3.3. However, the question is still open about whether these subcalculi interact with each other in unwanted ways when their proof power is concerned: for any two such fragments  $\mathcal{L}_1 \subseteq \mathcal{L}_2$ , does the subcalculus corresponding to  $\mathcal{L}_2$  conservatively extend the one corresponding to  $\mathcal{L}_1$ ? Typically, the absence of unwanted interactions among subcalculi is deduced from having cut elimination, and soundness and completeness w.r.t. a given semantics. This way, in [15] it is also shown that the system  $D'.EAK$  conservatively extends  $EAK$ .

However, this strategy is not immediately applicable to the setting of the Dynamic Calculus, due to the structural symbols referred to as *virtual adjoints*, which are easily recognizable, since they are shaped like arrows with a squiggly tail:  $\blacktriangleleft\sim$ ,  $\sim\blacktriangleright$  etc. Virtual adjoints have no semantic justification, and hence, the rules in which they specifically occur (that is, the display postulates relative to them) cannot be justified on semantic grounds. The reason for including virtual adjoints in the language of the Dynamic Calculus is for it to enjoy the relativized display property, discussed in Section 2.3, which is key to guarantee the crucial condition C'8, requiring the existence of a way to solve the principal stage of the cut elimination theorem (cf. Section 3.3, see also Section 9.1).

When discussing virtual adjoints in Section 4, we claimed that, since they are only introduced in a derivation by way of display postulates and do not specifically intervene in any other structural rule, their presence in the calculus does not add unwanted proof power compared to D'.EAK (and hence to EAK). This is the sense in which the introduction of the virtual adjoints can be regarded as syntactically sound. The aim of the present section is to prove this claim.

A general and very powerful method for proving the conservativity of display calculi has been introduced in [5, 6] for the full intuitionistic linear logic. This method involves no less than two translations, one from the given display calculus into an intermediate shallow inference nested sequent calculus, and another one from the intermediate calculus into a deep inference nested sequent calculus. This method is very intricate, requiring the verification of hundreds of cases which account for every possible interaction between the shallow and the deep calculus. The intricacy of this proof was such that the correctness of the results in [5, 6] has been established by formalizing them in the proof assistant Isabelle/HOL, as reported in [11].

However, in the present section, a much smoother proof of conservativity is given for the Dynamic Calculus for EAK, which does not rely on any nested sequent calculus. Rather, the proof below relies on very specific and uncommon features of the design of the Dynamic Calculus for EAK. In a sense, the very fact that such a smooth proof is possible witnesses how uncommonly well behaved EAK is.

**Definition 7.1.** A sequent  $x \vdash y$  is *severe* if in the generation trees of either  $x$  or  $y$  there are occurrences of structural connectives to which no display postulates can be applied. Such occurrences will be referred to as *severe*.

Clearly, the definition above makes sense only in the context of calculi which, as is the case of the Dynamic Calculus, do not enjoy the full display property (cf. Definition 1). It can be easily verified that, in the specific case of the Dynamic Calculus, there are only two types of severe occurrences: triangle-type connectives rooting a structure in succedent position, and arrow-type connectives rooting a structure in precedent position. Examples of severe sequents then are  $(a \triangleright \alpha) \triangle A \vdash B$  and  $A \vdash a \triangle B$ .



**Lemma 7.2.** *Any rule in the Dynamic Calculus preserves the severity of sequents. That is, if a rule is applied to a severe sequent, the conclusion of that rule application is also severe.*

*Proof.* By inspection on the rules.  $\square$

**Fact 7.3.** *Let  $x \vdash y$  be a sequent of type AG or FNC, in the full language of Dynamic Calculus, which is derivable by means of a derivation  $\pi$  in which no application of Weakening, Necessitation, Balance or Atom introduce occurrences of virtual adjoints. Then  $x = a$  for some operational term  $a$  of the appropriate type.*

*Proof.* If  $x$  is not an atomic structure, then the grammar of AG and FNC prescribes that  $x$  has a virtual adjoint as a main connective. However, the assumptions imply that such structures can be introduced only by way of applications of display postulates, which introduce them in succedent position. Hence, given the assumptions, there is no way in which such a connective can be introduced in precedent position.  $\square$

**Lemma 7.4.** *Let  $X \vdash Y$  be a sequent of type FM which is derivable in the Dynamic Calculus by means of a derivation  $\pi$  in which no application of Weakening, Necessitation, Balance or Atom introduce occurrences of virtual adjoints. Then a derivation  $\pi'$  of  $X \vdash Y$  exists every node of which (hence the conclusion in particular) is free of virtual adjoints.*

*Proof.* Let  $s$  be some node/sequent in  $\pi$  where the given virtual adjoint has been introduced. Since virtual adjoints in the Dynamic Calculus are all virtual “right adjoints”, and since, by assumption, they are introduced only by way of applications of display postulates, the given virtual adjoint is the main connective in the succedent of the sequent  $s$ . Moreover, virtual adjoints are main connectives of structures of type AG and FNC. By type uniformity, this implies that the sequent  $s$  is either of type AG or FNC, and therefore  $s$  cannot be the conclusion of  $\pi$ . Some rule  $R$  must exist which takes  $s$  as a premise. It can be easily verified by inspection that  $R$  cannot coincide with any structural rule in the Dynamic Calculus which is neither a cut of the appropriate type nor a display postulate, since all structural rules different from Cut and display postulates have premises of type FM. We can also assume w.l.o.g. that  $R$  is not an application of Cut. Indeed, by Fact 7.3,  $s$  is of the form  $a \vdash x$ , with  $a$  being an operational term, and  $x$  being a non-atomic structure by assumption. Hence, if  $R$  was a Cut-application, the inference must be of the form

$$\frac{\begin{array}{c} \vdots \\ y \vdash a \end{array} \quad \begin{array}{c} \vdots \\ a \vdash x \end{array}}{y \vdash x}$$

Because Cut rules in the Dynamic Calculus are strongly type-regular, also  $y \vdash a$  would be a (derivable) sequent of type AG or FNC, hence Fact 7.3 applies to  $y \vdash a$ . That is,  $y$  must be atomic, and because  $y \vdash a$  is derivable,  $y$  must coincide

with  $a$ . Hence, the conclusion of that Cut application is again  $s$ . This shows that if  $R$  was Cut, w.l.o.g. we would be able to remove that application from the proof tree. The remaining options are that  $R$  coincides with an introduction rule of some heterogeneous logical connective. Recall that, by Fact 7.3,  $s$  is of the form  $a \vdash x$ , where  $a$  is an operational term. Then, it can be verified by inspection that no heterogeneous rule is applicable if  $x$  is not an atomic structure, which is not the case of the sequent  $s$ , as discussed above. Finally, since the left-hand side of  $s$  is atomic, no other display postulates are applicable to  $s$  but the converse direction of the same display postulate which had introduced the virtual adjoint and which makes it disappear. Therefore, the refinement  $\pi'$  of  $\pi$  consists in removing these double and redundant applications of display postulates.  $\square$

**Lemma 7.5.** *If  $\text{inf}$  is an application of Balance, Atom, Necessitation or Weakening in which some occurrence of a virtual adjoint is introduced, then the conclusion of  $\text{inf}$  is a severe sequent.*

*Proof.* As to Balance, Atom and Necessitation <sub>$i$</sub>  with  $i = 0, 2$ , notice that each of these rules introduces a structure  $x$  of type FNC or AG in precedent position. If a virtual adjoint is introduced as a substructure of  $x$ , then  $x$  is non-atomic, and it can be immediately verified by inspecting the syntax of FNC and AG that the main connective of  $x$  is an arrow-type connective, which would then be in precedent position. Hence, the resulting sequent is severe.

As to Weakening and Necessitation<sub>1</sub>, notice that these rules introduce structures  $x$  of type FM and ACT respectively. Recall that virtual adjoints root structures of type FNC or AG. Hence, if some virtual adjoint occurs in the generation tree of  $x$ , it cannot occur at the root of  $x$ . Hence, the virtual adjoint must occur in the scope of some other structural connective. Notice that the heterogeneous connectives are the only ones which can take as argument a structure rooted in a virtual adjoint. We claim that either the virtual adjoint occurs in precedent position (which would be enough to conclude that the conclusion of  $\text{inf}$  is severe), or under the scope of some structural connective to which no display postulate can be applied. Assume that the virtual adjoint occurs in succedent position. If its immediate ancestor in the generation tree of  $X$  is a triangle-type connective, then these connectives are in succedent position too, and hence no display postulate can be applied to them, which makes the conclusion of  $\text{inf}$  severe, as required. Similarly, if the immediate ancestor of the virtual adjoint is an arrow-type connective, then it can be easily checked by inspection that these connectives take structures of type FNC or AG exclusively in their antitone coordinate (that is, on the flat side of the arrow). Hence, the arrow-type connective is in precedent position, and hence no display postulate can be applied to it, as required.  $\square$

**Corollary 7.6.** *Let  $A' \vdash B'$  be a sequent of type Fm in the language of the Dynamic Calculus, such that  $A'$  and  $B'$  are, respectively, images of some  $D'$ .EAK-formulas  $A$  and  $B$  under the translation of Section 4. If  $A' \vdash B'$  is derivable in the Dynamic Calculus, then  $A \vdash B$  is derivable in  $D'$ .EAK.*

*Proof.* Let  $\pi$  be a derivation of  $A' \vdash B'$  in the Dynamic Calculus. By assumption,  $A' \vdash B'$  is not severe. Hence, no rule application in  $\pi$  can introduce severe sequents, since these, by Lemma 7.2, would then propagate till the conclusion. Hence in particular, by Lemma 7.5, in  $\pi$  there cannot be any applications of Balance, Atom, Necessitation or Weakening in which some occurrence of a virtual adjoint is introduced. Therefore, by Lemma 7.4, a derivation  $\pi'$  of  $A' \vdash B'$  exists in which no virtual adjoints occur. By the results collected in Section 5, the derivation  $\pi'$  is sound w.r.t. the final coalgebra semantics. Hence  $A' \vdash B'$  is satisfied on the final coalgebra semantics. Since, as discussed in Section 5,  $\llbracket A \rrbracket = \llbracket A' \rrbracket$  and  $\llbracket B \rrbracket = \llbracket B' \rrbracket$ , this implies that  $A \vdash B$  is satisfied. Since  $D'.EAK$  is complete w.r.t. the final coalgebra semantics, a  $D'.EAK$ -derivation of  $A \vdash B$  exists.  $\square$

## 8 Conclusions and further directions

The present paper is part of a line of research aimed at developing adequate proof-calculi for dynamic logics. These logics have proven to be very difficult to treat with standard proof-theoretic tools, due to the very features which characterize them and make them applicable to diverse fields of science, spanning from artificial intelligence to social science and economics. A central desideratum in this line of research is the development of methods which apply uniformly to different logics, and which allow a smooth transfer of results from one logic to another. The framework of display calculi has successfully met this desideratum for wide classes of logics in the family of modal and substructural logics. In particular, in the framework of display calculi it is possible to state and prove metatheorems which guarantee any given proof system to enjoy the all-important cut elimination property, provided it meets certain conditions on its design.

The main contribution of the present paper is the definition of a display calculus which smoothly encompasses the most proof-theoretically impervious features of Baltag Moss and Solecki's logic of epistemic actions and knowledge. Besides being well performing (it adequately captures EAK and enjoys Belnap-style cut elimination), this calculus provides an interesting and in our opinion very promising *methodological* platform towards the uniform development of a general proof-theoretic account of all dynamic logics, and also, from a purely structurally proof-theoretic viewpoint, for clarifying and sharpening the formulation of criteria leading to the statement and proof of meta-theoretic results such as Belnap-style cut-elimination, or conservativity issues.

**Seminal approaches.** The starting point of this methodology is to introduce enough syntactic devices, both at the operational and at the structural level, so that the parameters indexing logical connectives can be accounted for in the system as *terms* in the language of choice. This gives rise to the definition of multi-type languages, endowed with connectives which manage the interaction of the different types. This approach appears seminally in both [1] and [12]; however, in neither

paper it is fully explored: in [1] there is no theory of contexts governing the interaction of different types, and in [12], this interaction is clarified, but only at the metalinguistic level.

**Multi-type calculus for PDL.** The multi-type approach has been applied to PDL in [14]. In that setting, introducing two separate types for transitive actions and for general actions makes it possible to overcome the big hurdle given by the fact that the induction axiom features occurrences of the same formula in precedent and in succedent position, which makes it severely non amenable to the treatment in standard display calculi. Another interesting case study is given by Parikh’s game logic, which is ongoing work [13].

**Refinements of Belnap’s conditions, and type-uniformity.** The multi-type setting will hopefully prove to be conceptually advantageous to achieve a better grasp and a simpler statement of Wansing’s and Belnap’s *regularity* requirements (cf. conditions  $C_6/C_7$  in [28], [4, Section 2]) for the Belnap-style cut elimination, via the notion of type-uniformity (Definition 3.1). In [3], Belnap motivates his condition  $C_7$ <sup>11</sup> saying that “rules need not be wholly closed under substitution of structures for congruent formulas which are antecedent parts, but they must be closed enough.” Then he explains that *closed enough* refers to the closure under substitution of formulas  $A$  for structures  $X$  such that a derivation is available in the system for the sequent  $X \vdash A$ , in which the occurrence of  $A$  in the conclusion is principal. The crucial observation is that, even if a system is not defined *a priori* as multi-type, it can be regarded as a multi-type setting: indeed, the type of  $A$  can be defined as consisting of all the structures  $X$  such that the shape of derivation alluded to above exists. Then, condition  $C_6/C_7$  can be equivalently reformulated as the requirement that rules should be closed under uniform substitution within each type. Notice that, under the stipulations above, different types must be separated by at least one structural rule. For instance, in the Dynamic Calculus for EAK, the rule *balance* separates Fnc from Act. In conclusion, our conjecture is that Wansing’s and Belnap’s conditions  $C_6/C_7$  boil down to a type-uniformity requirement in a context in which types are not given explicitly. The observations above indicate that type-uniformity is a desirable design requirement for general dynamic calculi, and in particular for the development of an adequate proof theory for dynamic logics, particularly in view of a uniform path to Belnap-style cut-elimination.

**Non-proliferation.** Our analysis towards Belnap-style cut elimination led us to refine and weaken various aspects of the cut elimination metatheorem. For instance, the requirement of non-proliferation of parameters for quasi-properly displayable multi-type calculi applies only to types the grammar of which is rich

---

<sup>11</sup>Recall that Belnap’s condition  $C_7$  corresponds to Wansing’s cons-regularity for formulas occurring in precedent position. An analogous explanation holds of course for the ant-regularity condition of formulas in succedent position.

enough that allows non-trivial cut applications, that is, applications of cut the conclusion of which is different from both premises. The case study of EAK allows such a simple grammar on functional actions and agents that these two types are not subject to the restriction of non-proliferation. This in turn makes it possible to include the *swap-in* rules in the calculus, in which every occurring parameter of a type which can proliferate does indeed proliferate. Introducing some non-trivial grammar on functional actions (e.g. sequential composition) would make the restriction of non-proliferation applicable to this type, and hence would make *swap-in* not suitable anymore.

**Expanding the signature.** Notwithstanding the concerns about *swap-in*, the multi-type language provides the opportunity to consider various natural expansions of the language of actions. Early on, we argued that the connective  $\blacktriangleleft_1$  which takes formulas in both coordinate as arguments and delivers an action, has the following natural interpretation: for all formulas  $A, B$ , the term  $B \blacktriangleleft_1 A$  denotes the weakest epistemic action  $\gamma$  such that, if  $A$  was true before  $\gamma$  was performed, then  $B$  is true after any successful execution of  $\gamma$ . This connective seems particularly suited to explore epistemic capabilities and planning.

## 9 Appendix

In the following subsection, we collect the reduction steps verifying that the Dynamic Calculus verifies condition C'8; in Section 9.2, we collect the derivations which prove the syntactic completeness of the Dynamic Calculus w.r.t. IEAK (cf. Section 2.2).

### 9.1 Cut elimination

Let us recall that C'8 only concerns applications of the cut rules in which both occurrences of the given cut-term are *non parametric*. Notice that non parametric occurrences of atomic terms of type Fm involve an axiom on at least one premise, thus we are reduced to the following cases (the case of the constant  $\perp$  is symmetric to the case of  $\top$  and is omitted):

$$\frac{\Phi p \vdash p \quad p \vdash \Psi p}{\Phi p \vdash \Psi p} \rightsquigarrow \Phi p \vdash \Psi p \quad \frac{\text{I} \vdash \top \quad \frac{\text{I} \vdash X}{\top \vdash X}}{\text{I} \vdash X} \rightsquigarrow \frac{\text{I} \vdash X}{\text{I} \vdash X} \rightsquigarrow \text{I} \vdash X$$

Notice that non parametric occurrences of any given (atomic) operational term  $a$  of type Fnc or Ag are confined to axioms  $a \vdash a$ . Hence:

$$\frac{a \vdash a \quad a \vdash a}{a \vdash a} \rightsquigarrow a \vdash a$$

In each case above, the cut in the original derivation is strongly uniform by assumption, and is eliminated by the transformation. As to cuts on non atomic terms, let us restrict our attention to those cut-terms the main connective of which is  $\Delta_i, \blacktriangle_i, \multimap_i, \multimap_i$  for  $0 \leq i \leq 3$  (the remaining operational connectives are straightforward and left to the reader).

$$\begin{array}{c}
\begin{array}{c} \vdots \pi_0 \quad \vdots \pi_1 \quad \vdots \pi_2 \\ \hline x \vdash a \quad y \vdash b \\ \hline x \Delta_i y \vdash a \Delta_i b \quad a \Delta_i b \vdash z \\ \hline x \Delta_i y \vdash z \end{array} \\
\rightsquigarrow \\
\begin{array}{c} \vdots \pi_1 \\ \hline y \vdash b \\ \hline y \vdash x \blacktriangleright_i z \\ \hline x \Delta_i y \vdash z \end{array}
\end{array}
\rightsquigarrow
\begin{array}{c}
\begin{array}{c} \vdots \pi_2 \\ \hline \begin{array}{c} \vdots \pi_0 \quad a \Delta_i b \vdash z \\ \hline x \vdash a \quad a \vdash z \blacktriangleleft_i b \\ \hline x \vdash z \blacktriangleleft_i b \end{array} \\ \hline \begin{array}{c} x \Delta_i b \vdash z \\ \hline b \vdash x \blacktriangleright_i z \\ \hline y \vdash x \blacktriangleright_i z \end{array} \\ \hline x \Delta_i y \vdash z
\end{array}
\end{array}$$
  

$$\begin{array}{c}
\begin{array}{c} \vdots \pi_1 \quad \vdots \pi_0 \quad \vdots \pi_2 \\ \hline y \vdash a \blacktriangleright_i b \quad x \vdash a \quad b \vdash z \\ \hline y \vdash a \multimap_i b \quad a \multimap_i b \vdash x \blacktriangleright_i z \\ \hline y \vdash x \blacktriangleright_i z \end{array} \\
\rightsquigarrow \\
\begin{array}{c} \vdots \pi_1 \\ \hline y \vdash a \blacktriangleright_i b \\ \hline x \blacktriangleleft_i y \vdash b \\ \hline x \blacktriangleleft_i y \vdash z \\ \hline y \vdash x \blacktriangleright_i z \end{array}
\end{array}
\rightsquigarrow
\begin{array}{c}
\begin{array}{c} \vdots \pi_1 \\ \hline y \vdash a \blacktriangleright_i b \\ \hline a \blacktriangleleft_i y \vdash b \\ \hline x \vdash a \quad a \vdash b \blacktriangleleft_i y \\ \hline x \vdash b \blacktriangleleft_i y \\ \hline x \blacktriangleleft_i y \vdash b \\ \hline x \blacktriangleleft_i y \vdash z \\ \hline y \vdash x \blacktriangleright_i z \end{array}
\end{array}$$
  

$$\begin{array}{c}
\begin{array}{c} \vdots \pi_0 \quad \vdots \pi_1 \quad \vdots \pi_2 \\ \hline x \vdash a \quad y \vdash b \quad a \blacktriangleleft_i b \vdash z \\ \hline x \blacktriangleleft_i y \vdash a \blacktriangleleft_i b \quad a \blacktriangleleft_i b \vdash z \\ \hline x \blacktriangleleft_i y \vdash z \end{array} \\
\rightsquigarrow \\
\begin{array}{c} \vdots \pi_1 \\ \hline y \vdash b \\ \hline y \vdash x \blacktriangleright_i z \\ \hline x \blacktriangleleft_i y \vdash z \end{array}
\end{array}
\rightsquigarrow
\begin{array}{c}
\begin{array}{c} \vdots \pi_2 \\ \hline \begin{array}{c} \vdots \pi_0 \quad a \blacktriangleleft_i b \vdash z \\ \hline x \vdash a \quad a \vdash z \blacktriangleleft_i b \\ \hline x \vdash z \blacktriangleleft_i b \end{array} \\ \hline \begin{array}{c} x \blacktriangleleft_i b \vdash z \\ \hline b \vdash x \blacktriangleright_i z \\ \hline y \vdash x \blacktriangleright_i z \end{array} \\ \hline x \blacktriangleleft_i y \vdash z
\end{array}
\end{array}$$
  

$$\begin{array}{c}
\begin{array}{c} \vdots \pi_1 \quad \vdots \pi_0 \quad \vdots \pi_2 \\ \hline y \vdash a \blacktriangleright_i b \quad x \vdash a \quad b \vdash z \\ \hline y \vdash a \multimap_i b \quad a \multimap_i b \vdash x \blacktriangleright_i z \\ \hline y \vdash x \blacktriangleright_i z \end{array} \\
\rightsquigarrow \\
\begin{array}{c} \vdots \pi_1 \\ \hline y \vdash a \blacktriangleright_i b \\ \hline a \Delta_i y \vdash b \\ \hline x \vdash a \quad a \vdash b \blacktriangleleft_i y \\ \hline x \vdash b \blacktriangleleft_i y \\ \hline x \Delta_i y \vdash b \\ \hline x \Delta_i y \vdash z \\ \hline y \vdash x \blacktriangleright_i z \end{array}
\end{array}$$

In each case above, the cut in the original derivation is strongly uniform by assumption, and after the transformation, cuts of lower complexity are introduced which can be easily verified to be strongly uniform for each  $0 \leq i \leq 3$ .

## 9.2 Completeness

To prove the completeness of the Dynamic Calculus it is enough to show that all the axioms and rules of H.IEAK are theorems and, respectively, derived or admissible rules of Dynamic Calculus. Below we show the derivations of the dynamic axioms.

- $\alpha \Delta p \dashv\vdash (\alpha \Delta \top) \wedge p$

$$\begin{array}{c}
 \frac{\frac{\frac{\alpha \vdash \alpha}{\alpha \Delta \text{I} \vdash \alpha \Delta \top} \quad \text{I} \vdash \top}{(\alpha \Delta \text{I}); (\alpha \Delta p) \vdash (\alpha \Delta \top) \wedge p}}{\alpha \Delta (\text{I}; p) \vdash (\alpha \Delta \top) \wedge p}}{\frac{\text{I}; p \vdash \blacktriangleright (\alpha \Delta \top) \wedge p}{p \vdash \blacktriangleright (\alpha \Delta \top) \wedge p}}}{\frac{\alpha \Delta p \vdash (\alpha \Delta \top) \wedge p}{\alpha \Delta p \vdash (\alpha \Delta \top) \wedge p}} \\
 \frac{\frac{\frac{\alpha \blacktriangleright p \vdash p}{\text{I} \vdash (\alpha \blacktriangleright p) > p}}{\top \vdash (\alpha \blacktriangleright p) > p}}{\alpha \vdash \alpha \quad (\alpha \blacktriangleright p); \top \vdash p}}{\alpha \Delta ((\alpha \blacktriangleright p); \top) \vdash \alpha \Delta p}}{\frac{(\alpha \Delta \top); p \vdash \alpha \Delta p}{\alpha \Delta \top \vdash \alpha \Delta p < p}}}{\frac{\alpha \Delta \top \vdash p > \alpha \Delta p}{\alpha \Delta \top; p \vdash \alpha \Delta p}}}{(\alpha \Delta \top) \wedge p \vdash \alpha \Delta p}
 \end{array}$$

- $\alpha \rightarrow p \dashv\vdash (\alpha \Delta \top) \rightarrow p$

$$\begin{array}{c}
 \frac{\frac{\frac{\alpha \vdash \alpha \quad p \vdash \blacktriangleright p}{\alpha \rightarrow p \vdash \alpha \triangleright (\alpha \blacktriangleright p)}}{\alpha \blacktriangleright \alpha \rightarrow p \vdash \blacktriangleright p}}{\frac{\text{I} \vdash (\alpha \blacktriangleright \alpha \rightarrow p) > \alpha \blacktriangleright p}{\top \vdash (\alpha \blacktriangleright \alpha \rightarrow p) > \alpha \blacktriangleright p}}}{(\alpha \blacktriangleright \alpha \rightarrow p); \top \vdash \blacktriangleright p}}{\alpha \Delta ((\alpha \blacktriangleright \alpha \rightarrow p); \top) \vdash p}}{\frac{\alpha \rightarrow p; \alpha \Delta \top \vdash p}{\alpha \Delta \top \vdash \alpha \rightarrow p > p}}}{\frac{\alpha \rightarrow p; \alpha \Delta \top \vdash p}{\alpha \Delta \top; \alpha \rightarrow p \vdash p}}}{\frac{\alpha \rightarrow p \vdash \alpha \Delta \top > p}{\alpha \rightarrow p \vdash (\alpha \Delta \top) \rightarrow p}} \\
 \frac{\frac{\frac{\alpha \vdash \alpha \quad \text{I} \vdash \top}{\alpha \Delta \text{I} \vdash \alpha \Delta \top} \quad p \vdash \alpha \triangleright p}{(\alpha \Delta \top) \rightarrow p \vdash (\alpha \Delta \text{I}) > \alpha \triangleright p}}{(\alpha \Delta \top) \rightarrow p \vdash \alpha \triangleright (\text{I} > p)}}{\alpha \blacktriangleright (\alpha \Delta \top) \rightarrow p \vdash \text{I} > p}}{\alpha \blacktriangleright (\alpha \Delta \top) \rightarrow p \vdash p}}{\frac{(\alpha \Delta \top) \rightarrow p \vdash \alpha \triangleright p}{(\alpha \Delta \top) \rightarrow p \vdash \alpha \rightarrow p}}
 \end{array}$$

- $\langle \alpha \rangle \top \dashv\vdash \text{I}_\alpha \rightsquigarrow \alpha \Delta \top \dashv\vdash \alpha \Delta \top$

$$\frac{\frac{\frac{\alpha \vdash \alpha}{\alpha \Delta \top \vdash \alpha \Delta \top}}{\alpha \Delta \top \vdash \alpha \Delta \top}}{\alpha \Delta \top \vdash \alpha \Delta \top}$$









- $\alpha \rightarrow (A \rightarrow B) \dashv\vdash \alpha \Delta A \rightarrow \alpha \Delta B$

$$\begin{array}{c}
\frac{\frac{A \vdash A}{\alpha \Delta A \vdash \alpha \triangleright A}}{\alpha \blacktriangle (\alpha \Delta A) \vdash A} \quad \frac{\frac{\alpha \vdash \alpha \quad B \vdash B}{\alpha \Delta B \vdash \alpha \Delta B}}{B \vdash \alpha \blacktriangleright (\alpha \Delta B)} \\
\frac{A \rightarrow B \vdash \alpha \blacktriangle (\alpha \Delta A) > \alpha \blacktriangleright (\alpha \Delta B)}{\alpha \vdash \alpha \quad A \rightarrow B \vdash \alpha \blacktriangleright (\alpha \Delta A > \alpha \Delta B)} \\
\frac{\alpha \rightarrow (A \rightarrow B) \vdash \alpha \triangleright (\alpha \blacktriangleright (\alpha \Delta A > \alpha \Delta B))}{\alpha \blacktriangle (\alpha \rightarrow (A \rightarrow B)) \vdash \alpha \blacktriangleright (\alpha \Delta A > \alpha \Delta B)} \\
\frac{I \vdash \alpha \blacktriangle (\alpha \rightarrow (A \rightarrow B)) > \alpha \blacktriangleright (\alpha \Delta A > \alpha \Delta B)}{\alpha \blacktriangle (\alpha \rightarrow (A \rightarrow B); I) \vdash \alpha \blacktriangleright (\alpha \Delta A > \alpha \Delta B)} \\
\frac{\alpha \Delta (\alpha \blacktriangle (\alpha \rightarrow (A \rightarrow B); I) \vdash \alpha \Delta A > \alpha \Delta B)}{\alpha \rightarrow (A \rightarrow B); (\alpha \Delta I) \vdash \alpha \Delta A > \alpha \Delta B} \text{ conj} \\
\frac{(\alpha \Delta I); \alpha \rightarrow (A \rightarrow B) \vdash \alpha \Delta A > \alpha \Delta B}{\alpha \Delta A; ((\alpha \Delta I); \alpha \rightarrow (A \rightarrow B)) \vdash \alpha \Delta B} \\
\frac{(\alpha \Delta A; (\alpha \Delta I)); \alpha \rightarrow (A \rightarrow B) \vdash \alpha \Delta B}{\alpha \Delta A; (\alpha \Delta I) \vdash \alpha \Delta B < \alpha \rightarrow (A \rightarrow B)} \\
\frac{\alpha \Delta (A; I) \vdash \alpha \Delta B < \alpha \rightarrow (A \rightarrow B)}{A; I \vdash \alpha \blacktriangleright (\alpha \Delta B < \alpha \rightarrow (A \rightarrow B))} \\
\frac{I \vdash A > (\alpha \blacktriangleright (\alpha \Delta B < \alpha \rightarrow (A \rightarrow B)))}{A \vdash \alpha \blacktriangleright (\alpha \Delta B < \alpha \rightarrow (A \rightarrow B))} \\
\frac{\alpha \Delta A \vdash \alpha \Delta B < \alpha \rightarrow (A \rightarrow B)}{\alpha \Delta A \vdash \alpha \Delta B < \alpha \rightarrow (A \rightarrow B)} \\
\frac{\alpha \Delta A \vdash \alpha \Delta B < \alpha \rightarrow (A \rightarrow B)}{\alpha \Delta A \vdash \alpha \Delta B < \alpha \rightarrow (A \rightarrow B)} \\
\frac{\alpha \Delta A; \alpha \rightarrow (A \rightarrow B) \vdash \alpha \Delta B}{\alpha \rightarrow (A \rightarrow B) \vdash \alpha \Delta A > \alpha \Delta B} \\
\frac{\alpha \rightarrow (A \rightarrow B) \vdash \alpha \Delta A \rightarrow \alpha \Delta B}{\alpha \rightarrow (A \rightarrow B) \vdash \alpha \Delta A \rightarrow \alpha \Delta B}
\end{array}
\quad
\begin{array}{c}
\frac{\alpha \vdash \alpha \quad B \vdash B}{\alpha \Delta B \vdash \alpha \triangleright B} \\
\frac{\alpha \Delta A \rightarrow \alpha \Delta B \vdash \alpha \Delta A > \alpha \triangleright B}{\alpha \Delta A \rightarrow \alpha \Delta B \vdash \alpha \triangleright (A > B)} \\
\frac{\alpha \blacktriangle (\alpha \Delta A \rightarrow \alpha \Delta B) \vdash A > B}{\alpha \blacktriangle (\alpha \Delta A \rightarrow \alpha \Delta B) \vdash A \rightarrow B} \\
\frac{\alpha \Delta A \rightarrow \alpha \Delta B \vdash \alpha \triangleright A \rightarrow B}{\alpha \Delta A \rightarrow \alpha \Delta B \vdash \alpha \rightarrow (A \rightarrow B)}
\end{array}$$





## 10 Cut Elimination for the Dynamic Calculus for EAK

Let us recall that  $C'_8$  only concerns applications of the cut rules in which both occurrences of the given cut-term are *non parametric*. Notice that non parametric occurrences of atomic terms of type Fm involve an axiom on at least one premise, thus we are reduced to the following cases (the case of the constant  $\perp$  is symmetric to the case of  $\top$  and is omitted):

$$\frac{\Phi p \vdash p \quad p \vdash \Psi p}{\Phi p \vdash \Psi p} \rightsquigarrow \Phi p \vdash \Psi p \quad \frac{\text{I} \vdash \top \quad \frac{\text{I} \vdash X}{\top \vdash X}}{\text{I} \vdash X} \rightsquigarrow \text{I} \vdash X$$

Notice that non parametric occurrences of any given (atomic) operational term  $a$  of type Fnc or Ag are confined to axioms  $a \vdash a$ . Hence:

$$\frac{a \vdash a \quad a \vdash a}{a \vdash a} \rightsquigarrow a \vdash a$$

In each case above, the cut in the original derivation is strongly uniform by assumption, and is eliminated by the transformation. As to cuts on non atomic terms, let us restrict our attention to those cut-terms the main connective of which is  $\Delta_i, \blacktriangleright_i, \rightarrow_i, \blacktriangleright_i$  for  $0 \leq i \leq 3$  (the remaining operational connectives are straightforward and left to the reader).

The following table disambiguates the various structural connectives occurring in the reduction steps represented below:

	$i = 0$	$i = 1$	$i = 2$	$i = 3$
$\triangleright$	$\triangleright_0$	$\triangleright_1$	$\triangleright_2$	$\sim\triangleright_3$
$\blacktriangleright$	$\blacktriangleright_0$	$\blacktriangleright_1$	$\blacktriangleright_2$	$\sim\blacktriangleright_3$
$\triangleleft$	$\triangleleft_{\sim 0}$	$\triangleleft_1$	$\triangleleft_{\sim 2}$	$\triangleleft_{\sim 3}$
$\blacktriangleleft$	$\blacktriangleleft_{\sim 0}$	$\blacktriangleleft_1$	$\blacktriangleleft_{\sim 2}$	$\blacktriangleleft_{\sim 3}$

$$\frac{\frac{\frac{\text{I} \vdash a \quad \text{I} \vdash b}{x \Delta_i y \vdash a \Delta_i b} \quad \frac{\text{I} \vdash a \quad \text{I} \vdash b}{a \Delta_i b \vdash z}}{x \Delta_i y \vdash z} \rightsquigarrow \frac{\frac{\text{I} \vdash a \quad \frac{\frac{\text{I} \vdash b \quad \frac{\frac{\text{I} \vdash a \quad \text{I} \vdash z \blacktriangleleft_i b}{x \vdash z \blacktriangleleft_i b}}{x \Delta_i b \vdash z}}{y \vdash b \quad b \vdash x \blacktriangleright_i z}}{y \vdash x \blacktriangleright_i z}}{x \Delta_i y \vdash z}}$$

$$\begin{array}{c}
\begin{array}{c}
\vdots \pi_1 \\
y \vdash a \triangleright_i b \\
\hline
y \vdash a \rightarrow_i b
\end{array}
\quad
\begin{array}{c}
\vdots \pi_0 \\
x \vdash a \quad b \vdash z \\
\hline
a \rightarrow_i b \vdash x \triangleright_i z
\end{array}
\quad
\begin{array}{c}
\vdots \pi_2 \\
b \vdash z \\
\hline
b \vdash z
\end{array} \\
\hline
y \vdash x \triangleright_i z
\end{array}
\rightsquigarrow
\begin{array}{c}
\begin{array}{c}
\vdots \pi_1 \\
y \vdash a \triangleright_i b \\
\hline
a \blacktriangleleft_i y \vdash b \\
\hline
x \vdash a
\end{array}
\quad
\begin{array}{c}
\vdots \pi_0 \\
x \vdash a \quad b \vdash z \\
\hline
x \blacktriangleleft_i y \vdash b
\end{array}
\quad
\begin{array}{c}
\vdots \pi_2 \\
b \vdash z \\
\hline
b \vdash z
\end{array} \\
\hline
x \blacktriangleleft_i y \vdash z \\
\hline
y \vdash x \triangleright_i z
\end{array}$$
  

$$\begin{array}{c}
\begin{array}{c}
\vdots \pi_0 \\
x \vdash a \quad y \vdash b \\
\hline
x \blacktriangleleft_i y \vdash a \blacktriangleleft_i b
\end{array}
\quad
\begin{array}{c}
\vdots \pi_1 \\
y \vdash b \\
\hline
a \blacktriangleleft_i b \vdash z \\
\hline
a \blacktriangleleft_i b \vdash z
\end{array}
\quad
\begin{array}{c}
\vdots \pi_2 \\
a \blacktriangleleft_i b \vdash z \\
\hline
a \blacktriangleleft_i b \vdash z
\end{array} \\
\hline
x \blacktriangleleft_i y \vdash z
\end{array}
\rightsquigarrow
\begin{array}{c}
\begin{array}{c}
\vdots \pi_1 \\
y \vdash b \\
\hline
x \blacktriangleleft_i b \vdash z \\
\hline
y \vdash x \triangleright_i z
\end{array}
\quad
\begin{array}{c}
\vdots \pi_2 \\
a \blacktriangleleft_i b \vdash z \\
\hline
a \vdash z \triangleleft_i b \\
\hline
x \vdash z \triangleleft_i b \\
\hline
b \vdash x \triangleright_i z
\end{array} \\
\hline
x \blacktriangleleft_i y \vdash z
\end{array}$$
  

$$\begin{array}{c}
\begin{array}{c}
\vdots \pi_1 \\
y \vdash a \blacktriangleright_i b \\
\hline
y \vdash a \rightarrow_i b
\end{array}
\quad
\begin{array}{c}
\vdots \pi_0 \\
x \vdash a \quad b \vdash z \\
\hline
a \rightarrow_i b \vdash x \blacktriangleright_i z
\end{array}
\quad
\begin{array}{c}
\vdots \pi_2 \\
b \vdash z \\
\hline
b \vdash z
\end{array} \\
\hline
y \vdash x \blacktriangleright_i z
\end{array}
\rightsquigarrow
\begin{array}{c}
\begin{array}{c}
\vdots \pi_1 \\
y \vdash a \blacktriangleright_i b \\
\hline
a \triangleleft_i y \vdash b \\
\hline
x \vdash a
\end{array}
\quad
\begin{array}{c}
\vdots \pi_0 \\
x \vdash a \quad b \vdash z \\
\hline
x \triangleleft_i y \vdash b
\end{array}
\quad
\begin{array}{c}
\vdots \pi_2 \\
b \vdash z \\
\hline
b \vdash z
\end{array} \\
\hline
x \triangleleft_i y \vdash z \\
\hline
y \vdash x \blacktriangleright_i z
\end{array}$$

In each case above, the cut in the original derivation is strongly uniform by assumption, and after the transformation, cuts of lower complexity are introduced which can be easily verified to be strongly uniform for each  $0 \leq i \leq 3$ .

## References

- [1] Alexandru Baltag, Bob Coecke, and Mehrnoosh Sadrzadeh. Epistemic actions as resources. *J. Log. Comput.*, 17(3):555–585, 2007.
- [2] Alexandru Baltag, Lawrence S. Moss, and Slawomir Solecki. The logic of public announcements, common knowledge and private suspicious. Technical Report SEN-R9922, CWI, Amsterdam, 1999.

- [3] Nuel Belnap. Display logic. *J. Philos. Logic*, 11:375–417, 1982.
- [4] Nuel Belnap. Linear logic displayed. *Notre Dame Journal of Formal Logic*, 31(1):14–25, 1990.
- [5] Ranald Clouston, Jeremy E. Dawson, Rajeev Goré, and Alwen Tiu. Annotation-free sequent calculi for full intuitionistic linear logic. *CSL*, pages 197–214, 2013.
- [6] Ranald Clouston, Jeremy E. Dawson, Rajeev Goré, and Alwen Tiu. Annotation-free sequent calculi for full intuitionistic linear logic - extended version. *CoRR*, 2013.
- [7] Willem Conradie, Silvio Ghilardi, and Alessandra Palmigiano. Unified correspondence. In A. Baltag and S. Smets, editors, *Johan F.A.K. van Benthem on Logical and Informational Dynamics*, Outstanding Contributions to Logic. Springer, in print 2014.
- [8] Willem Conradie and Alessandra Palmigiano. Algorithmic correspondence and canonicity for distributive modal logic. *Annals of Pure and Applied Logic*, 163(3):338 – 376, 2012.
- [9] Willem Conradie, Alessandra Palmigiano, and Sumit Sourabh. Algebraic modal correspondence: Sahlqvist and beyond. *Submitted*.
- [10] Brian A. Davey and Hilary A. Priestley. *Lattices and Order*. Cambridge University Press, 2002.
- [11] Jeremy Dawson, Ranald Clouston, Rajeev Goré, and Alwen Tiu. From display calculi to deep nested sequent calculi: Formalised for full intuitionistic linear logic. In *Proceedings of IFIP Theoretical Computer Science*, Lecture Notes in Computer Science, International Federation for Information Processing, 2014.
- [12] Roy Dyckhoff, Mehrnoosh Sadrzadeh, and Julien Truffaut. Algebra, proof theory and applications for an intuitionistic logic of propositions, actions and adjoint modal operators. *ACM Transactions on Computational Logic*, 14(4), 2013.
- [13] Sabine Frittella, Giuseppe Greco, Alexander Kurz, and Alessandra Palmigiano. A multi-type display calculus for game logic. *In preparation*.
- [14] Sabine Frittella, Giuseppe Greco, Alexander Kurz, and Alessandra Palmigiano. Multi-type display calculus for propositional dynamic logic. *Journal of Logic and Computation, Special Issue on Substructural Logic and Information Dynamics, Forthcoming*, 2014.



- [15] 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, Special Issue on Substructural Logic and Information Dynamics, Forthcoming*, 2014.
- [16] Rajeev Goré. Dual intuitionistic logic revisited. In Roy Dyckhoff, editor, *Automated Reasoning with Analytic Tableaux and Related Methods*, volume 1847 of *Lecture Notes in Computer Science*, pages 252–267. Springer, 2000.
- [17] 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.
- [18] Alexander Kurz and Alessandra Palmigiano. Epistemic updates on algebras. *Logical Methods in Computer Science*, 2013. arXiv:1307.0417.
- [19] Carsten Lutz. Complexity and succinctness of public announcement logic. *AAMAS '06 Proceedings or the fifth international joint conference on Autonomous agents and multiagent system*, pages 137–143, 2006.
- [20] Minghui Ma, Alessandra Palmigiano, and Mehrnoosh Sadrzadeh. Algebraic semantics and model completeness for intuitionistic public announcement logic. *Annals of Pure and Applied Logic*, 2013. <http://dx.doi.org/10.1016/j.apal.2013.11.004>.
- [21] Marc Pauly and Rohit Parikh. Game logic - an overview. *Studia Logica*, 75(2):165–182, 2003.
- [22] Jan Plaza. Logics of public communications. *Synthese*, 158(2):165–179, 2007.
- [23] Francesca Poggiolesi. *Gentzen Calculi for Modal Propositional Logic*. Trends in logic. Springer, 2010.
- [24] Vaughan Pratt. Action logic and pure induction. In *Proceedings JELIA 1990*, volume LNCS 478, pages 97–120. Springer, 1991.
- [25] Greg Restall. *An Introduction to Substructural Logics*. Routledge, London, 2000.
- [26] Gisèle Fischer Servi. Axiomatizations for some intuitionistic modal logics. *Rend. Sem. Mat Polit. di Torino*, 42:179–194, 1984.
- [27] Hans P. van Ditmarsch, Wiebe van der Hoek, and Barteld Kooi. *Dynamic Epistemic Logic*. Springer, 2007.
- [28] Heinrich Wansing. *Displaying Modal Logic*. Kluwer, 1998.

- [29] Heinrich Wansing. Sequent systems for modal logics. *Handbook of Philosophical Logic*, 8:61–45, 2002.