Reasoning about Action: A Study in Systems Design

White, Graham

ISSN 1470-5559

Reasoning about Action
A Study in Systems Design

Graham White

Queen Mary
University of London

RR-06-04

April 2006

Department of Computer Science

Reasoning about Action
A Study in Systems Design[*]

Graham White
Computer Science
Queen Mary, University of London
London E1 4NS
`graham@dcs.qmul.ac.uk`
`http://www.dcs.qmul.ac.uk/~graham`

April 29, 2006

Contents

[*] Submitted to *Fundamenta Informaticae*

List of Tables

1 Introduction

The appropriate use of logic to reason about action has been a persistent theme in AI. In this work, we use tools from category-theoretic semantics to analyse one of the most significant AI approaches, that of Reiter. His work is described as a whole in [9], and in detail in a series of papers by him and his school.

There are several reasons for embarking on work such as this. One of these reasons is merely analytical: there is significant logical structure to be found in Reiter's theory, but this logical structure is somewhat hidden by the details of the logical machinery that he uses. His theory revolves around axiomatising a key group of concepts: situation, fluent, and action. These concepts are axiomatised as individuals and relations, like any others, in a first-order universe. No special logical structure is given to them, but this does not reflect the important role that they play in his conceptual analysis: much of the interest in Reiter's system is to be found in these concepts, the relations between them, and in the high-level idioms which feature them. We could think of these concepts as "design patterns" in his reasoning, and, when these design patterns are made explicit, they lead to a remarkably elegant logical system. This distinction between high and low level logics is a common one. We might make an analogy with modal logic: this, too, is a high level language which can, admittedly, be compiled into first order logic. However, the possibility of this compilation does not make modal logic superfluous: rather, modal logic is still useful in that, firstly, it describes tractable and useful fragments of first order logic, and, secondly, it may – for example, when reasoning about transition systems – enforce an appropriate abstraction.

Secondly, we can, using our approach, prove things that Reiter could not prove. In particular, we have a proof theory with cut elimination, which allows us to resolve the questions that Reiter raises in [9, §9.1.3]: see our Section 4.1.

Thirdly, we can use our analysis to make clear the design decisions which Reiter has implicitly made. The system of this paper is very closely modelled on Reiter's: we do, though, extend his system, on mathematical grounds, by adding a notion of equality between actions (see Section 4.2). Surprisingly, this notion turns out to be closely related to arguments due to Davidson about equality between actions: so this notion of equality seems to be well-motivated both mathematically and philosophically.

We can also define two other systems, based on the same sort of intuition as Reiter's system. One of these alternatives is intuitionistic in character, while the other is modelled on classical bidirectional modal logic (as in [10]). The contrasts between these systems are instructive: Reiter's system and our intuitionistic system can both represent more or less the same phenomena, but they do it in different ways. And our full modal system can be regarded as a nondeterministic extension of Reiter. We will present these other two systems, without proofs, in an appendix.

Fourthly, our system overcomes a problem which affects many of the circumscription based treatments of the frame problem, namely that they are fatally sensitive to the language that they are formulated in: logically equivalent languages can yield different results on circumscription [14]. Our category-theoretic formulation gives a "coordinate-free" framework, within which suitable action descriptions can be evaluated without any artificial biases given by particular choices of a logical language.

Finally, one should note that the logical systems themselves are quite interesting: they have strong affinities to the deep inference modal systems of [12, 13, 3, 4] and to the modal display calculus of [7].

2 Technical Programme

In this section we describe some of Reiter's key ideas, and use them to lead into our fibrational theory, which is described in the following sections.

The fundamental problem for Reiter is what is called *regression*: that is, given a transition $s \xrightarrow{\alpha} t$ between situations s and t, and given a proposition P at t – what Reiter would describe as a *fluent* – the regression problem is to find a proposition P' at s which which will be true iff P is true at t.

One might think that another notion would be more fundamental: after all, the traditional AI scenario is that one knows about the present and wants to predict the future. This concept is known as *progression*: that is, given an action $s \xrightarrow{\alpha} t$, and given a theory Ξ_s describing the state s, find the theory Ξ_t describing t. Reiter does, in fact, define this notion: however, his insight was that regression is technically simpler to work with, and that progression can be defined in terms of it.

In this work, rather than define regression and progression as idioms in the situation calculus, we take them as primitive, and define a high-level logic with them as propositional operators. The regression operator, α^*Q, is read as the weakest precondition of α with postcondition Q: that is, the weakest proposition which, being true before α is run, guarantees afterwards the truth of Q.

Progression, $\amalg_\alpha P$, is read correspondingly as the strongest postcondition. Reiter, in fact, defines regression on propositions and progression on theories, but it is simpler technically to define both regression and progression on propositions: we prove the equivalence of our definition of progression with Reiter's in Section 4.1.

Category-theoretically, this goes as follows. Reiter's situations correspond (roughly speaking) to the objects of a base category, his fluents are objects of a category fibred over that base, and actions correspond to morphisms in the base: weakest preconditions, or Reiter's "regression", are the pullbacks (or reindexing functors) given by the fibration, whereas the strongest postconditions, i.e. Reiter's "progression", are left adjoint to the pullbacks.

Since the system is classical, we also have a right adjoint, given by de Morgan duality; in summary we have a hyperdoctrine [8]. The resulting system is similar, in some ways, to dynamic logic. However, the existence of both adjoints is significant, and gives rise to a very clean proof theory which enjoys cut elimination. We also have a sound and complete semantics.

So, given $s \xrightarrow{\alpha} t$, α^* takes propositions at t to propositions at s: Reiter [9, p 65] constructs α^* in such a way that (supposing that the relevant action preconditions are satisfied) it commutes with $\neg$ and $\wedge$: it thus also commutes with $\vee$, and preserves $\top$ and $\bot$. We will work in a classical framework, as does Reiter, so α^* can be regarded as a homomorphism between boolean algebras: roughly speaking, then, we will have a category fibred in boolean algebras over a transition system.

Well-known results of Jónsson and Tarski [6, 1] will then imply that, if the Boolean algebras are atomic, the operators α^* are *relational*: any operator, such as the α^*, which preserves $\top$ and commutes with $\wedge$, can be expressed as

$$(\alpha^*\psi)(\sigma : \mathcal{S}) \quad = \quad \forall \tau : \mathcal{T}. \sigma R \tau \rightarrow \psi(\tau), \tag{1}$$

for some suitable relation R between $\mathcal{S}$ and $\mathcal{T}$. Furthermore, if α^* commutes with $\vee$, then R must be a partial function from $\mathcal{S}$ to $\mathcal{T}$, and if, in addition, α^* preserves $\bot$, that function must be total. So, our operators α^* correspond to total functions between the sets of atoms of the boolean algebras concerned. Since these atoms are maximal consistent sets of propositions, they can be regarded as possible worlds.

We have talked about the actions: we should now consider the nature of the types s and t. Reiter's own formalism has the following properties:

1. actions are deterministic (he does define a probabilistic calculus [9, p. 335] to handle nondeterminism, but we will not discuss that);

2. what he calls "situations" are actually sequences of actions: [9, pp. 49f.] but

3. we are assumed not to have perfect knowledge of the initial (or subsequent) situations. Furthermore,

4. actions have preconditions, and are performable when and only when the preconditions are satisfied [9, p. 20].

Our initial thought is that the types – i.e. the objects of our base category – would be something like the states of a transition system, or what Reiter calls situations. However, the combination of 3 and 4 means that states, or situations, are not fine-grained enough to serve as the objects of a category, since, in the absence of perfect knowledge, we cannot tell whether a given transition can originate from a given state or not. So, in order to accommodate preconditions, the objects of the base should be combinations of a state and a proposition: the proposition would give preconditions for the actions which originate from that object. Category-theoretically, this corresponds to adding subset types to the original base category [5, p. 280].

We should, at this point, clarify an irritating linguistic confusion. Reiter uses what he calls *action preconditions*, and by this he means *enabling preconditions*, that is, propositions which must be true at a state for a particular action to be performable at that state. However, what he calls *regression* is what the programming language semantics community refer to as weakest preconditions: these are *truth preconditions*, that is, propositions which must be true at a state for a particular proposition to be true after the performance of a particular action. From here on, we will use the word “precondition” solely in Reiter’s sense, and will describe the weakest preconditions operator as the regression operator.

3 The System

3.1 Syntax

3.1.1 Fibre Products in the Base

An unusual feature of this system is that we will need, for the sequent calculus, fibre products of types in the base (for example, the rules Π_αL and Π_αR cannot be formulated without it). These do not exist, in general, for transition systems, but can be added to our category of types, given fairly modest conditions.

Fibre products can be defined in terms of n-ary Cartesian products and equalisers. Cartesian projects are harmless: they amount to arguing with n-tuples of states instead of single states. Now we already have subset types, so that all we need for equalisers is propositions, in the fibres, which express equality between morphisms [5, p. 282]. So, we can assume that our base category has fibre products. We will give, later, a concrete syntax which starts from a transition system and constructs a base category with fibre products: for the moment, though, it is shorter and clearer simply to write the fibre products.

3.1.2 The Algebra of Contexts

We will need the following definitions and results to define the cut rule, and to prove cut elimination, in the Reiter system.

Definition 1. A *marked context* $\Gamma[]$, is generated by the rules

$$\begin{aligned}\Gamma[] : t \ &:= \ [: t] \\ &\mid \Gamma[] : t, \Gamma : t \ \mid \Gamma : t, \Gamma[] : t \\ &\mid \{\Gamma[] : u\}^{\beta} \quad (\beta : t \to u)\end{aligned}$$

where Γ stands for a Reiter context as defined in Table 1. The category $\mathfrak{MRC}$ of *marked contexts* has, for objects, the objects of the base category, for morphisms, marked contexts, for composition the substitution operation

$$(\Gamma[: t] : s) \circ (\Gamma'[: u] : t) = \Gamma[\Gamma'[: u]] : t$$

and for identities the marked contexts $[: s] : s$ for each object s.

Definition 2. *Nesting*, $\boldsymbol{\nu}$, is a functor from $\mathfrak{MRC}^{\mathsf{op}} \to \mathcal{C}$ defined on morphisms as follows:

$$\begin{aligned}\boldsymbol{\nu}([: t]) &= \mathrm{Id}_t \\ \boldsymbol{\nu}(\Gamma, \Gamma'[], \Gamma'') &= \boldsymbol{\nu}(\Gamma'[]) \\ \boldsymbol{\nu}(\{\Gamma[]\}^{\alpha}) &= \boldsymbol{\nu}(\Gamma[]) \circ \alpha\end{aligned}$$

We define the following normal form for both contexts and marked contexts:

Definition 3. A context $\Gamma : s$ is in *split normal form* iff it is of the form

$$\Gamma_1, \ldots, \Gamma_k, \qquad (2)$$

with $\Gamma_i = \{P_i\}^{\alpha_i : s \to t_i}$, $i = 1, \ldots, k$. For a marked context, we require that exactly one of the Γ_i – namely, the one with the hole in it – should be of the form $\{[: t_i]\}^{\alpha_i}$.

Lemma 1. *Every context Γ is equivalent to one in split normal form, and similarly for marked contexts.*

Proof. Use the equivalence rules repeatedly. □

Lemma 2. *Split normal forms are unique, up to equality of the α_i and reordering of the Γ_i.*

Proof. The multiset of formulae and their nestings are invariants of the equivalence class of a context: and, given these, we can recover the split normal form, up to equality of the α_i and reordering of the Γ_i. □

Now we have

Proposition 1. *$\boldsymbol{\nu} : \mathfrak{MRC}^{op} \to \mathcal{C}$ is fibred in monoids over $\mathcal{C}$.*

Proof. Let the Cartesian morphisms of $\mathfrak{MRC}^{\mathsf{op}}$ be those of the form $\{[: t]\}^{\alpha : s \to t}$. To show that these satisfy the Cartesian property, we must show that every marked context $\Gamma[]$, with nesting $\alpha : s \to t$, is equivalent to one of the form $\Gamma'[\{[]\}^{\alpha}$, for a context $\Gamma'[]$ with nesting Id_s, and that, up to equivalence, such a $\Gamma'[]$ is unique. Split normal form establishes this. □

The following is a standard property of categories fibred over a base with pullbacks.

Lemma 3. *If $\mathcal{C}$ has pullbacks, then, for a marked context $\Gamma[: t] : u$, and any $\alpha : s \to t$, there is a unique* pullback context $(s \times_t \Gamma)[: s] : s \times_t u$ *with the property that*

$$\{\Gamma[: t] : u\}^{\alpha \times_t u} \approx (s \times_t \Gamma)[\{[: t]\}^\alpha : s]$$

Proof. We apply the universal property of the Cartesian lifting $\{[: t]\}^\alpha : s$ of α to the morphism $\{\Gamma[: t] : u\}^{\alpha \times_t u}$. □

The following is a useful corollary:

Lemma 4. *If $\mathcal{C}$ has pullbacks, then, for a marked context $\Gamma[: t] : u$ and $\alpha : s \to t$, we have*

$$s \times_t \Gamma[\varepsilon] \approx \{\Gamma[\varepsilon]\}^{\alpha \times_t u}.$$

Proof. We use the universal property of pullback contexts, together with the equivalence

$$\{\varepsilon : t\}^\alpha \approx \varepsilon : s.$$

□

Note that, here as elsewhere, expressions of the form $s \times_t \Gamma[\Gamma']$, with Γ' an ordinary context, are unambiguous: since pullback is an operation that is only defined on marked contexts, the substitution must be performed *after* the pullback.

3.1.3 The Calculus

We can, after these preliminaries, present the calculus: it is given in Table 4. The types will be objects of the base category: correspondingly, we assume that the morphisms in the base category satisfy the category-theoretic rules of Table 3. We have two context-forming operators: the usual comma, which is, semantically, $\wedge$ on the left and $\vee$ on the right, and the operator $\{\cdot\}^\alpha$ (for base morphisms α), which is, semantically, α^*. The rules in Table 2 guarantee functoriality of the context-forming operators, as well as the usual equivalences for the comma: the last rule in Table 2 expresses the fact that $\{\cdot\}^\alpha$ commutes with the comma, which follows from the fact that α^* is a boolean algebra morphism.

The sentential operators, besides the usual boolean ones, are the regression operator α^* and its left adjoint, the strongest postconditions $\amalg_\alpha$ operator, which we have already discussed: in the Reiter calculus, however, α^* is de Morgan self-dual (i.e. $\alpha^* \cong \neg\alpha^*\neg$, and so we also have, by de Morgan duality, a *right* adjoint Π_α to α^*.

The rules $\amalg_\alpha$L, Π_αR, and the cut rule use the pullback operations on marked contexts which we have defined in Section 3.1.2.

propositions	$P : t \;\; (P \text{ an atom}, t \in \mathrm{Ob}(\mathcal{C}))$
	$\top : t \quad \bot : t$
	$\dfrac{P : t \quad Q : t}{P \wedge Q : t} \quad \dfrac{P : t}{\neg P : t}$
	$\dfrac{P : t \quad \alpha : s \to t}{\alpha^* P : s}$
	$\dfrac{P : s \quad \alpha : s \to t}{\Pi_\alpha P : t}$
	$\dfrac{P : s \quad \alpha : s \to t}{\Pi_\alpha P : t}$
contexts	$\Gamma : t \;\; := \;\; P : t \mid \Gamma : t, \Gamma : t$ $\mid \{\Gamma : u\}^\alpha \;\; (\alpha : t \to u)$
entailments	$\dfrac{\Gamma : t \quad \Delta : t}{\Gamma \vdash \Delta}$

Table 1: Types and Syntax

$$\frac{}{\Gamma \approx \Gamma} \qquad \frac{\Gamma \approx \Gamma' \quad \Gamma' \approx \Gamma''}{\Gamma \approx \Gamma''}$$

$$\frac{\Gamma \approx \Gamma' \quad \Gamma_1 \approx \Gamma_1'}{\Gamma, \Gamma_1 \approx \Gamma', \Gamma_1'} \qquad \frac{}{\Gamma, \varepsilon \approx \Gamma}$$

$$\frac{}{\Gamma, \Gamma' \approx \Gamma', \Gamma} \qquad \frac{\Gamma : t \approx \Gamma'' : t \quad \alpha : s \to t}{\{\Gamma\}^{\alpha} \approx \{\Gamma'\}^{\alpha}}$$

$$\frac{\alpha = \beta}{\{\Gamma\}^{\alpha} \approx \{\Gamma\}^{\beta}} \qquad \frac{}{\{\Gamma\}^{\alpha \circ \beta} \approx \{\{\Gamma\}^{\beta}\}^{\alpha}}$$

$$\frac{}{\{\Gamma\}^{\mathrm{Id}} \approx \Gamma} \qquad \frac{}{\{\Gamma\}^{\alpha : s \to t}, \{\Gamma'\}^{\alpha : s \to t} \approx \{\Gamma, \Gamma'\}^{\alpha : s \to t}}$$

$$\frac{}{\{\varepsilon\}^{\alpha} \approx \varepsilon}$$

Table 2: Equivalence of Contexts

$$\frac{}{\mathrm{Id} \circ \alpha = \alpha} \qquad \frac{}{\alpha = \alpha \circ \mathrm{Id}}$$

$$\frac{}{(\alpha \circ \beta) \circ \gamma = \alpha \circ (\beta \circ \gamma)}$$

$$\frac{\alpha = \beta}{\gamma \circ \alpha = \gamma \circ \beta} \qquad \frac{\alpha = \beta}{\alpha \circ \gamma = \beta \circ \gamma}$$

Table 3: Rules for Category Theory

$$\frac{\Gamma \vdash \Delta \quad \Gamma \approx \Gamma' \quad \Delta \approx \Delta'}{\Gamma' \vdash \Delta'}\approx$$

$$\frac{}{\Gamma[P] \vdash \Delta[P]}\text{Axiom} \qquad \frac{\Gamma : t \vdash \Delta : t}{\{\Gamma\}^{\alpha:s\to t} \vdash \{\Delta\}^{\alpha:s\to t}}\{\}^{\alpha}$$

$$\frac{\Gamma[] \vdash \Delta}{\Gamma[\Gamma'] \vdash \Delta}\text{LW} \qquad \frac{\Gamma \vdash \Delta[]}{\Gamma \vdash \Delta[\Delta']}\text{RW}$$

$$\frac{\Gamma[P, P] \vdash \Delta}{\Gamma[P] \vdash \Delta}\text{LC} \qquad \frac{\Gamma \vdash \Delta[P, P]}{\Gamma \vdash \Delta[P]}\text{RC}$$

$$\frac{}{\Gamma[\bot] \vdash \Delta}\bot\,\text{L} \qquad \frac{}{\Gamma \vdash \Delta[\top]}\top\text{R}$$

$$\frac{\Gamma[P_1] \vdash Q \quad \Gamma[P_2] \vdash Q}{\Gamma[P_1 \vee P_2] \vdash Q}\vee\text{L} \qquad \frac{\Gamma \vdash \Delta[Q1, Q2]}{\Gamma \vdash \Delta[Q_1 \vee Q_2]}\vee\text{R}$$

$$\frac{\Gamma[P_1, P_2] \vdash \Delta}{\Gamma[P_1 \wedge P_2] \vdash \Delta}\wedge\text{L} \qquad \frac{\Gamma \vdash \Delta[Q_1] \quad \Gamma \vdash \Delta[Q_2]}{\Gamma \vdash \Delta[Q_1 \wedge Q_2]}\wedge\text{R}$$

$$\frac{\Gamma[] \vdash \Delta[P]}{\Gamma[\neg P] \vdash \Delta[]}\neg\text{L} \qquad \frac{\Gamma, \{Q\}^{\alpha:s\to t} \vdash \Delta}{\Gamma \vdash \{\neg Q\}^{\alpha:s\to t}, \Delta}\neg\text{R}$$

$$\frac{\Gamma[\{P\}^{\alpha}] \vdash \Delta}{\Gamma[\alpha^* P] \vdash \Delta}\alpha^*\text{L} \qquad \frac{\Gamma \vdash \Delta[\{Q\}^{\alpha}]}{\Gamma \vdash \Delta[\alpha^* Q]}\alpha^*\text{R}$$

$$\frac{(s \times_t \Gamma)[P] \vdash \{\Delta\}^{\alpha \times_t u}}{\Gamma[(\mathrm{II}_\alpha P) : t] : u \vdash \Delta : u}\mathrm{II}_\alpha\text{L} \qquad \frac{\Gamma \vdash \Delta[Q]}{\Gamma \vdash \Delta[\{\mathrm{II}_\alpha Q\}^{\alpha}]}\mathrm{II}_\alpha\text{R}$$

$$\frac{\Gamma[P] \vdash \Delta}{\Gamma[\{\Pi_\alpha P\}^{\alpha}] \vdash \Delta}\Pi_\alpha\text{L} \qquad \frac{\{\Gamma\}^{\alpha \times_t u} \vdash (s \times_t \Delta)[Q]}{\Gamma : u \vdash \Delta[(\Pi_\alpha Q) : t] : u}\Pi_\alpha\text{R}$$

$$\frac{\Gamma : s \vdash \Delta[P : t] : s \quad \Gamma'[P : t] : u, \vdash \Delta' : u}{\{\Gamma\}^{s \times_t \beta}, (s \times_t \Gamma')[] \vdash (\Delta \times_t u)[], \{\Delta'\}^{\alpha \times_t u}}\text{cut}$$

side conditions:

$$\begin{array}{rl}
\text{Axiom}, \neg\text{L}, \neg\text{R} & \boldsymbol{\nu}(\Gamma[]) = \boldsymbol{\nu}(\Delta[]) \\
\mathrm{II}_\alpha\text{L}, \mathrm{II}_\alpha\text{R}, & \alpha : s \to t \\
\Pi_\alpha\text{L}\Pi_\alpha\text{R} & \\
\text{cut} & \boldsymbol{\nu}(\Delta[]) = \alpha : s \to t \\
 & \boldsymbol{\nu}(\Gamma'[]) = \beta : u \to t
\end{array}$$

Table 4: The Reiter Sequent Calculus

3.2 Semantics

The semantics of this logic should be as follows:

Definition 4. Let $\mathcal{C}$ be a category with fibre products. A *Reiter category* over $\mathcal{C}$ is a category $\mathcal{E}$ fibred by Boolean algebras over $\mathcal{C}$: the reindexing functors α^* should be Boolean algebra homomorphisms, and should have both left and right adjoints, $\amalg_\alpha$ and Π_α, and the adjoints should satisfy the following Beck-Chevalley conditions [5, p. 97]:

$$\begin{aligned}\alpha^*\amalg_\beta P &\dashv\vdash \amalg_{s\times\beta}(\alpha \times_t u)^* P\\ \alpha^*\Pi_\beta P &\dashv\vdash \Pi_{s\times\beta}(\alpha \times_t u)^* P\end{aligned}$$

for morphisms $\alpha : s \to t, \beta : u \to t$ in the base and for $P : u$.

By applying Stone duality, we can derive the dual notion:

Definition 5. The *Stone space model* of a Reiter category is given by the covariant functor from $\mathcal{C}$ to the category of Stone spaces and continuous maps given by applying the Stone space functor to the indexed category version of the Reiter category.

Definition 6. An *assignment* is an choice, for every $t \in \mathrm{Ob}(\mathcal{C})$ and every atomic $P \in \mathfrak{L}$, of an element $[\![P]\!]_t \in \mathrm{Ob}(\mathcal{E}_t)$.

Given an assignment, we can define, for each $P : t$, its semantic value $[\![P]\!]_t$ by induction on its syntactic complexity: the sentential operators are interpreted in the usual way, α^* is interpreted as the reindexing functor (also written α^*), $\amalg_\alpha$ is interpreted as the left adjoint, and Π_α the right adjoint, to reindexing.

We also define $[\![\Gamma]\!]_t$ for contexts: here we have to make a distinction between left and right contexts, since the comma is interpreted differently on the left and on the right.

Definition 7. The semantic value of a left context is given by the clauses

- $[\![\Gamma]\!]_t = [\![P]\!]_t$ if $\Gamma = P$
- $[\![\Gamma, \Gamma']\!]_t = [\![\Gamma]\!]_t \wedge [\![\Gamma']\!]_t$
- $[\![\{\Gamma : u\}^\alpha]\!]_t = \alpha^*([\![\Gamma]\!]_u)$ (for $\alpha : t \to u$).

The semantic value of a right context is given by the clauses

- $[\![\Delta]\!]_t = [\![P]\!]_t$ if $\Delta = P$
- $[\![\Delta, \Delta']\!]_t = [\![\Delta]\!]_t \vee [\![\Delta']\!]_t$
- $[\![\{\Delta : u\}^\beta]\!]_t = \beta^*([\![\Delta]\!]_u)$ (for $\beta : t \to u$).

And, finally, a definition of semantic entailment:

Definition 8.

$$\Gamma : t \Vdash \Delta : t$$

iff

$$[\![\Gamma]\!]_t \leq [\![\Delta]\!]_t$$

for every assignment.

3.3 Soundness

First a pair of lemmas:

Lemma 5 (Semantic Monotonicity). *If* $(\Gamma[\Delta : u]) : t$ *is a left or right context, and if* $[\![\Delta]\!]_u \leq [\![\Delta']\!]_u$ *for some* $\Delta' : u$*, then* $[\![\Gamma[\Delta]]\!]_t \leq [\![\Gamma[\Delta']]\!]_t$*.*

Proof. The obvious induction, using the functoriality of α^*. □

Lemma 6 (Distributivity of Contexts). *If* $\Gamma[P \vee Q] : t$ *is a left context, then* $[\![\Gamma[P \vee Q]]\!]_t = [\![\Gamma[P]]\!]_t \vee [\![\Gamma[Q]]\!]_t$*; if* $\Gamma[P \wedge Q] : t$ *is a right context, then* $[\![\Gamma[P \wedge Q]]\!]_t = [\![\Gamma[P]]\!]_t \wedge [\![\Gamma[Q]]\!]_t$*.*

Proof. This follows from the distributivity of $\vee$ (or $\wedge$) over the comma on the left (or right), together with the fact that α^* is a Boolean algebra homomorphism. □

Proposition 2. *The rules for the equivalence of contexts in Table 2 are sound with respect to* $\Vdash$*: that is, if* $\Gamma : t \approx \Gamma' : t$*, then* $[\![\Gamma]\!]_t = [\![\Gamma']\!]_t$ *(and so, in particular, if* $\Gamma : t \approx \Gamma' : t$ *and* $\Delta : t \approx \Delta' : t$*, we have* $\Gamma \Vdash \Delta$ *iff* $\Gamma' \Vdash \Delta'$*).*

Proof. This follows from the functoriality of α^*, the fact that α^* is a Boolean algebra homomorphism, and the definitions of the semantic value of contexts. □

Proposition 3. *The rules LW, RW, LC, RC,* $\vee R$ *and* $\wedge L$ *are sound for* $\Vdash$*.*

Proof. Standard, using semantic monotonicity as necessary. □

Proposition 4. *The rules* $\vee L$ *and* $\wedge R$ *are sound for* $\Vdash$*.*

Proof. This follows from Lemma 6. □

Proposition 5. *The rules* $\neg L$ *and* $\neg R$ *are sound for* $\Vdash$*.*

Proof. This follows from the fact that α^*, being a Boolean algebra homomorphism, commutes with $\neg$, together with standard Boolean algebra. □

Proposition 6. *The rules* $\alpha^* L$ *and* $\alpha^* R$ *are sound for* $\Vdash$*.*

Proof. By definition, $[\![\alpha^* P]\!]_s = \alpha^*[\![P]\!]_t = [\![\{P\}^\alpha]\!]_s$, and the result follows. □

Proposition 7. *The rules $\Pi_\alpha L$ and $\coprod_\alpha R$ are sound for $\Vdash$.*

Proof. Consider $\coprod_\alpha$R: by semantic monotonicity, it suffices to prove that, for any $P : s$ and for $\alpha : s \to t$,

$$\begin{aligned} \llbracket P \rrbracket_s &\leq \llbracket \{\coprod_\alpha P\}^\alpha \rrbracket_s \\ &= \alpha^* \llbracket \coprod_\alpha P \rrbracket_t \\ &= \alpha^* \coprod_\alpha \llbracket P \rrbracket_s , \end{aligned}$$

but this is simply the unit of the adjunction $\coprod_\alpha \dashv \alpha^*$.

The proof for Π_αL is dual. □

For the next proposition, we will need a generalised Beck-Chevalley condition:

Lemma 7. *Let $\alpha : s \to t$. For a marked left context $\Gamma[: t] : u$ we have $\llbracket \Gamma[\coprod_\alpha P] \rrbracket_u = \coprod_{s \times_t \alpha} \llbracket s \times_t \Gamma[P] \rrbracket_{s \times_t u}$, and, for a marked right context $\Gamma[: t] : u$ we have $\llbracket \Gamma[\Pi_\alpha P] \rrbracket_u = \Pi_{s \times_t \alpha} \llbracket s \times_t \Gamma[P] \rrbracket_{s \times_t u}$.*

Proof. Induction on the syntactic complexity of $\Gamma[]$. □

Proposition 8. *The rules $\coprod_\alpha L$ and $\Pi_\alpha R$ are sound for $\Vdash$.*

Proof. Consider $\coprod_\alpha$L. Suppose the premise:

$$\begin{aligned} \llbracket s \times_t \Gamma[P] \rrbracket_{s \times_t u} &\leq \llbracket \{\Delta\}^{\alpha \times_s t} \rrbracket_{s \times_t u} \\ &= (\alpha \times_s t)^* \llbracket \Delta \rrbracket_u \end{aligned}$$

so, since $\coprod_{\alpha \times_s t} \dashv \alpha \times_s t^*$,

$$\coprod_{\alpha \times_s t} \llbracket s \times_t \Gamma[P] \rrbracket_{s \times_t u} \leq \llbracket \Delta \rrbracket_u$$

and so, by Lemma 7,

$$\llbracket \Gamma[\coprod_\alpha P] \rrbracket_u \leq \llbracket \Delta \rrbracket_u .$$

The proof for Π_αR is dual. □

Finally

Proposition 9. *The cut rule is sound for $\Vdash$.*

Proof. This follows from standard Boolean algebra, together with the fact that the operators $(\alpha \times_t u)^*$, etc., commute with the logical operators. □

So, putting all these results together, we have

Theorem 1. *$\Vdash$ is sound for our sequent calculus.*

3.4 Completeness

Theorem 2. *The semantics is complete: that is, if, for a given base category $\mathcal{C}$, and for an object t of $\mathcal{C}$,*

$$[\![\Gamma]\!]_t \leq [\![\Delta]\!]_t$$

for two contexts $\Gamma : t$ and $\Delta : t$, then

$$\Gamma \vdash \Delta.$$

This theorem will be proved by constructing a term, or generic, model, which we define as follows.

Definition 9. Let $\mathcal{C}$ be a category with fibre products. The term model, $\mathcal{E}_{\mathcal{C}}$, over $\mathcal{C}$ is given by the following data:

Objects these are given by pairs $P : s$, where s is an object of $\mathcal{C}$ and P is a proposition of type s

Morphisms a morphism between $P : s$ and $Q : t$ is given by a proof

$$P \vdash \{Q\}^{\alpha}$$

for some morphism $\alpha : s \to t$ of $\mathcal{C}$. Two such morphisms are equal iff their source and target are the same, and the corresponding morphisms of $\mathcal{C}$ are equal.

Composition suppose we have two morphisms corresponding to proofs

$$\begin{matrix}\Pi\\ \vdots\\ P : s \vdash \{Q : t\}^{\alpha}\end{matrix} \qquad \text{and} \qquad \begin{matrix}\Pi'\\ \vdots\\ Q : t \vdash \{R : u\}^{\beta}\end{matrix}$$

Their composition is given by the proof

$$\dfrac{\begin{matrix}\Pi\\ \vdots\\ P : s \vdash \{Q : t\}^{\alpha}\end{matrix} \qquad \dfrac{\dfrac{\begin{matrix}\Pi'\\ \vdots\\ Q : t \vdash \{R : u\}^{\beta}\end{matrix}}{\{Q : t\}^{\alpha} \vdash \{\{R : u\}^{\beta}\}^{\alpha}}\,\{\}^{\alpha}}{\{Q : t\}^{\alpha} \vdash \{R : u\}^{\alpha\circ\beta}}\,\approx}{P : s \vdash \{R : u\}^{\alpha\circ\beta}}\,\text{cut}$$

Identity morphisms these are given by the proofs

$$\dfrac{\dfrac{}{P : t \vdash P : t}\,\text{Axiom}}{P : t \vdash \{P : t\}^{\mathrm{Id}}}\,\approx$$

The display functor this is the map p which sends a typed proposition $P : t$ to the object t, and a proof of $P : s \vdash \{Q : t\}^\alpha$ to the morphism $\alpha : s \to t$.

Liftings we lift base morphisms as follows. Let $\alpha : s \to t$ be a morphism in the base, and let $P : t$ be an object of the fibre $\mathcal{E}_t$ over t: let the lifting of α be the following proof:

$$\cfrac{\cfrac{}{\{P : t\}^\alpha \vdash \{P : t\}^\alpha}\ \text{Axiom}}{(\alpha^* P : t) : s \vdash \{P : t\}^\alpha}\ \alpha^*\text{L}$$

In the indexed category viewpoint, this corresponds to using α^* as substitution functors.

Adjoints Left and right adjoints to the substitution functors α^* are given by Π_α and Π_α.

We now prove

Proposition 10. *$\mathcal{E}_\mathcal{C}$ is a Reiter category.*

Proof. It is clear than $\mathcal{E}_\mathcal{C}$ is a category (equality of morphisms is so strong that laws like associativity are immediate). It is likewise clear that our "display functor", p, is actually a functor. We have to check that the liftings are Cartesian: so, consider composable morphisms $\alpha : s \to t$ and $\beta : t \to u$ in the base, together with a proof Π of $P : s \vdash \{R : u\}^{\beta\circ\alpha}$ lying over $\alpha \circ \beta$. We need to produce a proof of $P : s \vdash \{(\alpha^* R) : t\}^\alpha$ (commutativity of the resulting diagram is trivial). But this is immediate:

$$\cfrac{\cfrac{\begin{array}{c}\Pi\\ \vdots\\ P : s \vdash \{R : u\}^{\beta\circ\alpha}\end{array}}{P : s \vdash \{\{R : u\}^\beta\}^\alpha}\ \approx}{P : s \vdash \{(\beta^* R) : t\}^\alpha}\ \beta^*\text{R}$$

This establishes the functoriality of the α^*. Consequently, $\mathcal{E}_\mathcal{C}$ is fibred over $\mathcal{C}$: the fibres are Boolean algebras, because the inference rules are a superset of the normal classical inference rules.

We need to show that Π_α and Π_α are left and right adjoint functors to the α^*. Functoriality is easy. For example, the following construction, which produces a proof of $\Pi_\alpha P \vdash \Pi_\alpha Q$ from a proof of $P \vdash Q$, establishes functoriality for Π_α:

$$\cfrac{\cfrac{\begin{array}{c}\Pi\\ \vdots\\ P : s \vdash Q : s\end{array}}{\{(\Pi_\alpha P) : t\}^\alpha \vdash Q : s}\ \Pi_\alpha\text{L}}{(\Pi_\alpha P) : t \vdash (\Pi_\alpha Q) : t}\ \Pi_\alpha\text{R}$$

Given functoriality, we only need to establish the unit and counit for each adjunction. The unit for $\amalg_\alpha \dashv \alpha^*$ is proven like this:

$$\dfrac{\dfrac{\dfrac{}{P : s \vdash P : s}\;\text{Axiom}}{P : s \vdash \{\amalg_\alpha P : s\}^\alpha}\;\amalg_\alpha\text{R}}{P : s \vdash \alpha^*\amalg_\alpha(P : s)}\;\alpha^*\text{R}$$

and the counit like this:

$$\dfrac{\dfrac{\dfrac{}{\{Q : t\}^\alpha \vdash \{Q : t\}^\alpha}\;\text{Axiom}}{\alpha^*(Q : t) \vdash \{Q : t\}^\alpha}\;\alpha^*\text{L}}{\amalg_\alpha\alpha^*(Q : t) \vdash Q : t}\;\amalg_\alpha\text{L}$$

The proof of the adjunction for $\amalg_\alpha$ is dual.

Finally, we must verify the Beck-Chevalley conditions: for $\amalg_\alpha$, we prove these as follows. Suppose that $\alpha : s \to t$, $\beta : u \to t$, and that $P : u$. Then we have

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\{P\}^{\alpha\times_t u} \vdash \{P\}^{\alpha\times_t u}}\;\text{Axiom}}{\{P\}^{\alpha\times_t u} \vdash (\alpha\times_t u)^*P}\;(\alpha\times_t u)^*\text{R}}{\{P\}^{\alpha\times_t u} \vdash \{\amalg_{(s\times_t\beta)}(\alpha\times_t u)^*P\}^{s\times_t\beta}}\;\amalg_{(s\times_t\beta)}\text{R}}{\{\amalg_\beta P\}^\alpha \vdash \amalg_{(s\times_t\beta)}(\alpha\times_t u)^*P}\;\amalg_\beta\text{L}}{\alpha^*\amalg_\beta P \vdash \amalg_{(s\times_t\beta)}(\alpha\times_t u)^*P}\;\alpha^*\text{L}$$

and

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\{P\}^{\alpha\times_t u} \vdash \{P\}^{\alpha\times_t u}}\;\text{Axiom}}{(\alpha\times_t u)^*P \vdash \{P\}^{\alpha\times_t u}}\;(\alpha\times_t u)^*\text{L}}{(\alpha\times_t u)^*P \vdash \{\{\amalg_\beta P\}^\beta\}^{\alpha\times_t u}}\;\amalg_\beta\text{R}}{(\alpha\times_t u)^*P \vdash \{\amalg_\beta P\}^{(\alpha\times_t u)\circ\beta}}\;\approx}{(\alpha\times_t u)^*P \vdash \{\amalg_\beta P\}^{(s\times_t\beta)\circ\alpha}}\;\approx}{(\alpha\times_t u)^*P \vdash \{\{\amalg_\beta P\}^\alpha\}^{s\times_t\beta}}\;\approx}{(\alpha\times_t u)^*P \vdash \{\alpha^*\amalg_\beta P\}^{s\times_t\beta}}\;\alpha^*\text{R}}{\amalg_{(s\times_t\beta)}(\alpha\times_t u)^*P \vdash \alpha^*\amalg_\beta P}\;\amalg_{(s\times_t\beta)}\text{L}$$

(where we have expanded the definitions of the pullback contexts in β^*L and $s\times_t\beta^*$L).

The proofs of the Beck-Chevalley conditions for $\amalg_\alpha$ are dual. This concludes the proof that $\mathcal{E}_\mathcal{C}$ is a Reiter category. □

Definition 10. Let Γ be a left context: let the *propositionalisation* of Γ, $\overline{\Gamma}$, be defined by

$$\begin{aligned} \overline{P} &= P \\ \overline{\Gamma', \Gamma''} &= \overline{\Gamma'} \wedge \overline{\Gamma''} \\ \overline{\{\Gamma\}^{\alpha}} &= \alpha^{*}\overline{\Gamma} \end{aligned}$$

If Δ is a left context, the *propositionalisation* of Δ, $\overline{\Delta}$, is defined dually.

Lemma 8. *For any* Γ *and* Δ*,*

$$\Gamma \vdash \Delta$$

iff

$$\overline{\Gamma} \vdash \overline{\Delta}$$

Proof. The obvious induction. □

Proof of Theorem 2. Suppose that $\Gamma : t \Vdash \Delta : t$. Define an interpretation of the language in $\mathcal{E}_{\mathcal{C}}$ by sending $P : t$ to $P : t$ as an object of $\mathcal{E}_t$. We establish, by induction, that, with respect to this interpretation, $[\![\Gamma]\!]_t = \overline{\Gamma}$, and $[\![\Delta]\!]_t = \overline{\Delta}$. Since $\Gamma : t \Vdash \Delta : t$, we must have $[\![\Gamma]\!]_t \leq [\![\Delta]\!]_t$, and, consequently, $\overline{\Gamma} \leq \overline{\Delta}$: by the definition of $\mathcal{E}_{\mathcal{C}}$, this means that $\overline{\Gamma} \vdash \overline{\Delta}$. By the lemma, we have $\Gamma \vdash \Delta$. □

3.5 Cut Elimination

We can prove cut elimination for this calculus, making the usual induction over cut rank and cut depth. We first need a definition of the rank of a formula:

Definition 11. Let A be a formula in the classical language. We define the rank of A, $|A|$, by the clauses

$$\begin{aligned} |P| &= 1 && \text{for } P \text{ atomic} \\ |\neg P| &= |P| + 1 \\ |\alpha^{*} P| &= |P| + 1 \\ |\amalg_{\alpha} P| &= |P| + 1 \\ |\Pi_{\alpha} P| &= |P| + 1 \\ |P \wedge Q| &= 1 + \max(|P|, |Q|) \\ |P \vee Q| &= 1 + \max(|P|, |Q|) \\ |P \rightarrow Q| &= 1 + \max(|P|, |Q|) \end{aligned}$$

There are, as usual, three cases to consider in the cut elimination process: where either of the premises is an axiom, $\perp$ L, or $\top$R, where the cutformula is non-principal in either of the premises, and were the cutformula is principal in both of the premises.

3.5.1 The Base Cases

The base cases are $\perp$ L, $\top$R, and Axiom. The case of $\perp$ L looks like this:

$$\dfrac{\Gamma[\perp] : s \vdash \Delta[P : t] : s \quad \Gamma'[P : t] : u, \vdash \Delta' : u}{\{\Gamma[\perp]\}^{s \times_t \beta}, (\alpha \times_t \Gamma')[] \vdash (\Delta \times_t \beta)[], \{\Delta'\}^{\alpha \times_t u}}\,\text{cut}$$

and we replace it with

$$\dfrac{}{\{\Gamma[\perp]\}^{s \times_t \beta}, (\alpha \times_t \Gamma')[] \vdash (\Delta \times_t \beta)[], \{\Delta'\}^{\alpha \times_t u}}\perp \text{L}$$

The other base cases are similar.

3.5.2 Moving the Cut Upwards

Most of the cases where we move a cut upwards are handled in exactly the same way as in classical logic, and are omitted: however, we do have to take care when the rule that we cut against changes the typing of the entailment. The cases to consider are as follows.

$\{\}^\alpha$ Here the cut application looks like

$$\dfrac{\dfrac{\begin{matrix}\Pi \\ \vdots \\ \Gamma \vdash \Delta[P]\end{matrix}}{\{\Gamma\}^\alpha \vdash \{\Delta[P]\}^\alpha}\{\}^\alpha \qquad \begin{matrix}\Pi' \\ \vdots \\ \Gamma'[P] \vdash \Delta'\end{matrix}}{\{\Gamma\}^{\alpha \times_u \gamma}, ((\beta \circ \alpha) \times_t \Gamma')[\varepsilon] \vdash (\alpha \times_t (\Delta \times_u \gamma))[\varepsilon], \{\Delta'\}^{(\beta \circ \alpha) \times_t u}}\,\text{cut}$$

where we have $P : u$, $\Gamma, \Delta : t$, $\Gamma', \Delta' : v$, $\alpha : s \to t$, $\boldsymbol{\nu}(\Delta[]) = \beta : u \to t$, and $\boldsymbol{\nu}(\Gamma'[]) = \gamma : v \to t$. Diagrammatically, the situation looks like this.

$$\begin{tikzcd}
 & s \times_u v \arrow[dl, "s \times_u \gamma"'] \arrow[dd, "\alpha \times_u \gamma"] \arrow[dr, "\alpha \times_u v"] \arrow[ddrr, bend left, "(\beta \circ \alpha) \times_u v"] & & & \\
s \arrow[dr, "\alpha"'] & & t \times_u v \arrow[dl, "t \times_u \gamma"] \arrow[dr, "\beta \times_u v"] & & \\
 & \Gamma, \Delta : t \arrow[dr, "\beta"'] & & \Gamma', \Delta' : v \arrow[dl, "\gamma"] & \\
 & & P : u & &
\end{tikzcd}$$

We replace the above application of cut with

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\begin{array}{cc}\Pi & \Pi' \\ \vdots & \vdots \\ \Gamma \vdash \Delta[P] & \Gamma'[P] \vdash \Delta'\end{array}}{\{\Gamma\}^{t \times_u \gamma}, (\beta \times_u \Gamma')[\varepsilon] \vdash (\Delta \times_u \gamma)[\varepsilon], \{\Delta'\}^{\beta \times_u v}}\,\text{cut}}{\{\{\Gamma\}^{t \times_u \gamma}, (\beta \times_u \Gamma')[\varepsilon]\}^{\alpha \times_u v} \vdash \{(\Delta \times_u \gamma)[\varepsilon], \{\Delta'\}^{\beta \times_u v}\}^{\alpha \times_u v}}\,\{\}^{\alpha \times_u v}}{\{\{\Gamma\}^{t \times_u \gamma}\}^{\alpha \times_u v}, \{(\beta \times_u \Gamma')[\varepsilon]\}^{\alpha \times_u v} \vdash \{(\alpha \times_t (\Delta \times_u \gamma))[\varepsilon]\}^{\alpha \times_u v}, \{\{\Delta'\}^{\beta \times_u v}\}^{\alpha \times_u v}}\,\approx}{\{\Gamma\}^{\alpha \times_u \gamma}, (\alpha \times_t \beta \times_u \Gamma')[\{\varepsilon\}^{\alpha}] \vdash \alpha \times_t (\Delta \times_u \gamma)[\{\varepsilon\}^{\alpha}], \{\Delta'\}^{(\beta \circ \alpha) \times_u v}}\,\approx, \text{Lemma 3}}{\{\Gamma\}^{\alpha \times_u \gamma}, ((\beta \circ \alpha) \times_t \Gamma')[\varepsilon] \vdash (\alpha \times_t (\Delta \times_u \gamma))[\varepsilon], \{\Delta'\}^{(\beta \circ \alpha) \times_t u}}\,\approx
$$

Result Cut rank unchanged: cut depth reduced by 1.

α^***L,** α^***R,** $\amalg_\alpha$**R,** $\amalg_\alpha$**L** These rules change the nesting of the contexts, but only around the principal formula: since we are considering rule applications in which the cutformula is not principal, this does not affect the cuts that we are considering, and so the cuts can be dealt with in the same way as with classical logic.

$\amalg_\alpha$**L** The cut application looks like this:

$$
\dfrac{\begin{array}{c}\Pi \\ \vdots \\ \Gamma \vdash \Delta[Q : v] : w\end{array} \qquad \dfrac{\begin{array}{c}\Pi' \\ \vdots \\ s \times_t \Gamma'[P][Q] \vdash \{\Delta'\}^{\alpha \times_t u}\end{array}}{\Gamma'[(\amalg_\alpha P) : t][Q : v] \vdash \Delta' : u}\,\amalg_\alpha\text{L}}{\{\Gamma\}^{\gamma \times_v w}, \{\Gamma'[\amalg_\alpha P][\varepsilon]\}^{u \times_v \delta} \vdash \{\Delta[\varepsilon]\}^{\gamma \times_v w}, \{\Delta'\}^{u \times_v \delta}}\,\text{cut}
$$

where the category theory, and typing, looks like this:

$$
\begin{tikzcd}
 & & & (s \times_t u) \times_u (u \times_v w) \arrow[dll, "(s \times_t u) \times_v \delta"'] \arrow[drr, "\alpha \times_t (u \times_v w)"] & & & \\
 & s \times_t \Gamma' : s \times_t u \arrow[dl] \arrow[drr, "\alpha \times_t u"] & & & & u \times_v w \arrow[dll, "u \times_v \delta"'] \arrow[dr, "\gamma \times_v w"] & \\
P : s \arrow[drr, "\alpha"'] & & & \Gamma', \Delta' : u \arrow[dl, "\beta"] \arrow[drr, "\gamma"'] & & & \Gamma, \Delta : w \arrow[dl, "\delta"] \\
 & & \amalg_\alpha P : t & & & Q : v &
\end{tikzcd}
$$

We replace the above cut with the following:

$$\dfrac{\begin{array}{c}\Pi\\ \vdots\\ \Gamma \vdash \Delta[Q]\end{array} \qquad \begin{array}{c}\Pi'\\ \vdots\\ s \times_t \Gamma'[P : s][Q : v] \vdash \{\Delta'\}^{\alpha \times_t u}\end{array}}{\begin{array}{c}\{\Gamma\}^{(\gamma \times_v w) \circ (\alpha \times_t (u \times_v w))}, \{s \times_t \Gamma'[P : s][\varepsilon : v]\}^{(s \times_t u) \times_v \delta} \vdash \\ \{\Delta[\varepsilon]\}^{(\gamma \times_v w) \circ ((\alpha \times_t (u \times_t w))}, \{\{\Delta'\}^{\alpha \times_t u}\}^{(s \times_t u) \times_v \delta}\end{array}}\,\text{cut}$$

Now, by the naturality of the pullback operation,

$$\{s \times_t \Gamma'[P : s][\varepsilon : v]\}^{(s \times_t u) \times_v \delta} \approx s \times_t \left(\{\Gamma'[P][\varepsilon]\}^{u \times_v \delta}\right);$$

Furthermore, we can work on the last expression on the right, using the context equivalence rule $\{\{\cdot\}^{\phi}\}^{\psi} \approx \{\cdot\}^{\psi \circ \phi}$ and the fact that

$$(\alpha \times_t u) \circ ((s \times_t u) \times_v \delta = (u \times_v s) \circ (\alpha \times_t (u \times_v w)) :$$

so, we can continue the proof as follows:

$$\dfrac{\dfrac{\begin{array}{c}\{\Gamma\}^{(\gamma \times_v w) \circ (\alpha \times_t (u \times_v w)}, s \times_t \{\Gamma'[P][\varepsilon]\}^{u \times_v \delta} \vdash \\ \{\Delta[\varepsilon]\}^{(\gamma \times_v w) \circ ((\alpha \times_t (u \times_t w))}, \{\Delta'\}^{(u \times_v s) \circ (\alpha \times_t (u \times_v w))}\end{array}}{\begin{array}{c}\{\{\Gamma\}^{\gamma \times_v w}\}^{\alpha \times_t (u \times_v w)}, s \times_t \{\Gamma'[P][\varepsilon]\}^{u \times_v \delta} \vdash \\ \{\{\Delta\}^{\gamma \times_v w}\}^{\alpha \times_t (u \times_v w)}, \{\{\Delta'\}^{u \times_v \delta}\}^{\alpha \times_t (u \times_v w)}\end{array}}\,\approx}{\{\Gamma\}^{\gamma \times_v w}, \{\Gamma'[\coprod_\alpha P][\varepsilon]\}^{u \times_v \delta} \vdash \{\Delta[\varepsilon]\}^{\gamma \times_v w}, \{\Delta'\}^{u \times_v \delta}}\,\coprod_\alpha\text{L}$$

result Cut depth reduced by one, cut rank unchanged.

$\coprod_\alpha$**L** Dual to the above.

3.5.3 Principal against Principal

The final case is where the cutformula is principal in both sides. The classical connectives give few surprises: we will, then, only consider the functorial connectives.

α^* The cut here will look like

$$\dfrac{\dfrac{\begin{array}{c}\Pi\\ \vdots\\ \Gamma \vdash \Delta[\{P\}^{\alpha}]\end{array}}{\Gamma \vdash \Delta[\alpha^* P]}\,\alpha^*\text{R} \qquad \dfrac{\begin{array}{c}\Pi'\\ \vdots\\ \Gamma'[\{P\}^{\alpha}] \vdash \Delta'\end{array}}{\Gamma'[\alpha^* P] \vdash \Delta'}\,\alpha^*\text{L}}{\{\Gamma\}^{\beta \times_s v}, \{\Gamma'[\varepsilon]\}^{u \times_s \gamma} \vdash \{\Delta[\varepsilon]\}^{\delta \times_s v}, \{\Delta'\}^{u \times_s \gamma}}\,\text{cut}$$

where the category theory, and typing, looks like

$$\begin{tikzcd}
 & u \times_s v \arrow[rr, "\kappa"] \arrow[dl, "u\times_s\gamma"'] \arrow[drr, "\beta\times_s v"] & & u \times_t v \arrow[dlll, "u\times_t(\alpha\circ\gamma)"] \arrow[dl, "(\alpha\circ\beta)\times_t v"] \\
\Gamma', \Delta' : u \arrow[dr, "\beta"] \arrow[ddr, "\alpha\circ\beta"'] & & \Gamma, \Delta : v \arrow[dl, "\gamma"'] \arrow[ddl, "\alpha\circ\gamma"] & \\
 & \alpha^* P : s \arrow[d, "\alpha"] & & \\
 & P : t & &
\end{tikzcd}$$

Here κ is the canonical morphism $u \times_s v \to u \times_t v$. Note that, because cut is applicable, the two occurrences of $\alpha^* P$ must be typed with the same object, and determinacy means that there can only be one α-typed morphism from that object: consequently, the two occurrences of P must be typed with the same object.

We replace the cut with

$$\dfrac{\dfrac{\dfrac{\begin{array}{cc}\Pi & \Pi' \\ \vdots & \vdots \\ \Gamma \vdash \Delta[\{P\}^\alpha] & \Gamma'[\{P\}^\alpha] \vdash \Delta'\end{array}}{\{\Gamma\}^{(\beta\circ\alpha)\times_t v}, \{\Gamma'[\varepsilon]\}^{u\times_t(\gamma\circ\alpha)} \vdash \{\Delta[\varepsilon]\}^{(\beta\circ\alpha)\times_t v}, \{\Delta'\}^{u\times_t(\gamma\circ\alpha)}}\,\text{cut}}{\begin{array}{c}\{\{\Gamma\}^{(\beta\circ\alpha)\times_t v}, \{\Gamma'[\varepsilon]\}^{u\times_t(\gamma\circ\alpha)}\}^\kappa \vdash \\ \{\{\Delta[\varepsilon]\}^{(\beta\circ\alpha)\times_t v}, \{\Delta'\}^{u\times_t(\gamma\circ\alpha)}\}^\kappa\end{array}}\,\{\}^\kappa}{\begin{array}{c}\{\Gamma\}^{\beta\times_s v}, \{\Gamma'[\varepsilon]\}^{u\times_s\gamma} \vdash \\ \{\Delta[\varepsilon]\}^{\beta\times_s v}, \{\Delta'\}^{u\times_s\gamma}\end{array}}\,\approx$$

Here the final application of $\approx$ comes from the equalities

$$(u \times_t (\alpha \circ \gamma)) \circ \kappa = u \times_s \gamma$$

and

$$((\alpha \circ \beta) \times_t) \circ \kappa = \beta \times_s v.$$

Result Cut rank reduced by one, cut depth reduced by one.

II_α The cut looks like

$$\dfrac{\dfrac{\begin{array}{c}\Pi\\ \vdots\\ \Gamma\vdash\Delta[P]\end{array}}{\Gamma\vdash\Delta[\{\mathrm{II}_\alpha P\}^\alpha]}\,\mathrm{II}_\alpha\mathrm{R}\qquad\dfrac{\begin{array}{c}\Pi'\\ \vdots\\ s\times_t\Gamma'[P]\vdash\{\Delta'\}^{\alpha\times_t v}\end{array}}{\Gamma'[\mathrm{II}_\alpha P]\vdash\Delta'}}{\{\Gamma\}^{u\times_t\gamma},\{\Gamma'[\varepsilon]\}^{(\alpha\circ\beta)\times_t v}\vdash\{\Delta[\varepsilon]\}^{u\times_t\gamma},\{\Delta'\}^{(\alpha\circ\beta)\times_t v}}\,\mathrm{cut}$$

The category theory looks like

$$\begin{tikzcd}
 & u\times_s(s\times_t v) \arrow[dl, "u\times_s(s\times_t\gamma)"'] \arrow[drrr, "\beta\times_s(s\times_t v)"] & & u\times_t v \arrow[ll, "\eta"'] \arrow[dlll, "u\times_t\gamma"] \arrow[dr, "\beta\times_s v"] \arrow[ddr, "(\alpha\circ\beta)\times_t v"'] & \\
\Gamma,\Delta:u \arrow[dr, "\beta"] \arrow[ddr, "\alpha\circ\beta"'] & & & & s\times_t\Gamma':s\times_t v \arrow[dlll, "s\times_t\gamma"] \arrow[d, "\alpha\times_t v"] \\
 & P:s \arrow[d, "\alpha"] & & & \Gamma',\Delta':v \arrow[dlll, "\gamma"] \\
 & \mathrm{II}_\alpha P:t & & &
\end{tikzcd}$$

Here η is the map $\langle\pi_1,\langle\beta\circ\pi_1,\pi_2\rangle\rangle$.

We replace the cut with the following:

$$\dfrac{\Gamma\vdash\Delta[P]\quad s\times_t\Gamma'[P]\vdash\{\Delta'\}^{\alpha\times_t v}}{\{\Gamma\}^{u\times_s(s\times_t\gamma)},\{s\times_t\Gamma'[\varepsilon]\}^{\beta\times_s(s\times_t v)}\vdash\{\Delta[\varepsilon]\}^{u\times_s(s\times_t\gamma)},\{\{\Delta'\}^{\alpha\times_t v}\}^{\beta\times_s(s\times_t v)}}\,\mathrm{cut}$$

By the naturality of pullback contexts, together with Lemma 4, we have

$$\{s\times_t\Gamma'[\varepsilon]\}^{\beta\times_s(s\times_t v)}\approx u\times_t\Gamma'[\varepsilon]\approx\{\Gamma'[\varepsilon]\}^{(\alpha\circ\beta)\times_t v},$$

so we can continue the proof as follows:

$$\dfrac{\dfrac{\begin{array}{l}\{\Gamma\}^{u\times_s(s\times_t\gamma)},\{\Gamma'[\varepsilon]\}^{(\alpha\circ\beta)\times_t v}\\ \qquad\vdash\{\Delta[\varepsilon]\}^{u\times_s(s\times_t\gamma)},\{\Delta'\}^{(\alpha\circ\beta)\times_t v}\end{array}}{\begin{array}{l}\{\{\Gamma\}^{u\times_s(s\times_t\gamma)},\{\Gamma'[\varepsilon]\}^{(\alpha\circ\beta)\times_t v}\}^\eta\\ \qquad\vdash\{\{\Delta[]\}^{u\times_s(s\times_t\gamma)},\{\Delta'\}^{(\alpha\circ\beta)\times_t v}\}^\eta\end{array}}\,\{\}^\eta}{\begin{array}{l}\{\Gamma\}^{u\times_t\gamma},\{\Gamma'[\varepsilon]\}^{(\alpha\circ\beta)\times_t v}\\ \qquad\vdash\{\Delta[\varepsilon]\}^{u\times_t\gamma},\{\Delta'\}^{(\alpha\circ\beta)\times_t v}\end{array}}\,\approx$$

Result Cut rank decreased by one: cut depth decreased by one.

$\amalg_\alpha$ Dual to the above.

So, finally, we have

Theorem 3. *The Reiter system satisfies cut elimination.*

Proof. We make an induction on cut rank, with an inner induction on cut depth. Consider an uppermost cut in a proof Π: by induction, we can decrease the depth of the cut first, until we have either a cut against an axiom or a cut where both formulae are principal: in that case, we can decrease the cut rank. We can thus eliminate all of the topmost cuts, and so, by induction, we eliminate all cuts. □

4 Applications

4.1 The Definition of Progression

Reiter has, in [9, §9.1], a discussion of the correct definition of the progression operator: he leaves some issues open, and we can resolve them.

His official definition of progression, in [9, §9.1.1], is formulated in terms of theories, rather than propositions, and model-theoretically rather than proof-theoretically. However, if we formulate it propositionally, and cash out the model theory using our term model, we have the following

Definition 12 (Reiter's Definition of Progression)**.** Given $\alpha : s \to t$, and a proposition $P : s$, a proposition $Q : t$ is a *progression* of P iff, for any $\beta : t \to u$, and any $R : u$, we have

$$P \vdash \{R\}^{\beta\circ\alpha} \quad \text{iff} \quad Q \vdash \{R\}^{\beta}$$

Proposition 11. *Reiter's progression agrees with our* $\amalg_\alpha$.

Proof.

$$\begin{aligned} P \vdash \{R\}^{\beta\circ\alpha} \quad & \text{iff} \quad P \vdash \{\beta^* P\}^{\alpha} \\ & \text{iff} \quad P \vdash \alpha^*(\beta^* P) \\ & \text{iff} \quad \amalg_\alpha P \vdash \beta^* P \\ & \text{iff} \quad \amalg_\alpha P \vdash \{P\}^{\beta} \end{aligned}$$

so $\amalg_\alpha P$ is a progression of P, according to Reiter's definition. □

Reiter then says:

> "Why not simply let the progression be, *by definition*, the set $\mathcal{F}_{S_a}$ of sentences uniform in S_a entailed by [the initial theory]? . . . currently, the problem remains open. [9, §9.1.3]

In our terms, this would amount to the following: that, provided we had arbitrary conjunctions, we would have

$$\amalg_\alpha P \quad \cong \quad \bigwedge_{P\vdash\{Q:t\}^\alpha} Q \tag{3}$$

But this holds in our system:

Proposition 12. *Whenever the required conjunctions exist,* (3) *holds.*

Proof. This follows trivially from the adjunction $\amalg_\alpha \vdash \alpha^*$. □

4.2 Concrete Syntax

We have claimed, in Section 3.1.1, that, given subobjects and equality between morphisms, we can construct fibre products in the base. We need to make this claim more precise by exhibiting a concrete syntax for the system we obtain. The constructions here are purely category-theoretic: however, it is an aid to intuition if we think of the corresponding Stone space model.

Suppose that we have a base category with Cartesian products (written $s\times t$ for objects s and t) and subset types (which we write $\{t|P\}$, for P a proposition over t): then, using [5, pp. 190ff], we can construct a proposition $\alpha =_t \beta$, with type s. In terms of Stone spaces, $\alpha =_t \beta$ will be true, at a point x of s, iff $\alpha(x) = \beta(x)$ in the Stone space over t (we are here simply identifying α and β with the corresponding continuous maps of Stone spaces). Since we have subset types, we can now construct an equaliser $\{s|\alpha =_t \beta\}$ of α and β [5, p. 282]: it is a subobject of s. Finally, if we have morphisms $\alpha : s \to t$ and $\beta : u \to t$, we can construct a fibred product

$$s \times_t u \;=\; \{s \times t|\alpha \circ p_1 =_t \beta \circ p_2\},$$

where $p_1 : s \times t \to s$ and $p_2 : s\times_t \to t$ are the canonical projections.

Note that the notions of equality that we have here are, in the usual jargon, *internal*: they only measure what the morphisms α and β do to the propositions in the fibres, and, if there are not enough propositions in the fibres, they are not guaranteed to yield the "true", *external* equality between morphisms in the base. Internal equality will, in general, be coarser grained than external equality.

Given, then, this internal equality, we can rewrite our sequent calculus to make use of it. We will write the equalities in a separate area to the left of each sequent: in general, then, our sequents will accumulate a set of such equality stipulations, which we will write Θ.

The rules in which fibre products are actually used are cut, $\amalg_\alpha$L and $\amalg_\alpha$R.

Cut will look like this:

$$\frac{\Theta|\Gamma : s \vdash \Delta[P : t] : s \quad \Theta'|\Gamma'[P : t] : u, \vdash \Delta' : u}{\Theta, \Theta', \alpha =_t \beta|\Gamma : s \times u, \Gamma'[\varepsilon] : s \times u \vdash \Delta[\varepsilon] : s \times u, \Delta' : s \times u}\,\text{cut}$$

and the other two rules will be similar but more complex.

4.2.1 Discussion

This version of the calculus strictly extends what Reiter does: the equality predicates with which we construct fibre products are (when the Cartesian products are expanded) two-place primitive predicates where each argument place is occupied by a different situation. Reiter's *primitive* fluents, on the other hand, can have only one situation argument place. It seems, however, to be a mathematically well-motivated extension.

It is also philosophically well-motivated. Davidson [2, p. 109] has an extensive series of examples of common-sense reasoning which are, basically, equational: the notion of equality between actions here pays a crucial role. It is quite gratifying to find that these equality predicates, introduced into our system for the purposes of cut elimination, should turn out to express notions which have, on quite other grounds, played a crucial role in the philosophy of action.

A Appendix: Alternative Systems

So far we have been adhering to Reiter's assumptions: that the logic is classical and that the transition system is deterministic. We here give two other systems, which each negate one of these assumptions: an intuitionistic, deterministic system, and a classical, nondeterministic system. We present the systems without proofs, for the sake of comparison with Reiter's system. In both cases the crucial choice is the semantics of the context forming operators: Reiter's system had an operator $\{\cdot\}^\alpha$ with the semantics of regression. Determinacy, and an involutory negation, then made this operator de Morgan self dual. Our alternative systems do not have an operator with those properties, so the context forming operators are less constrained.

A.1 The Intuitionistic System

Here we need contexts on the left, and single propositions on the right. An appropriate context forming operator turns out to be $\{\cdot\}_\alpha$, with the semantics of $\amalg_\alpha$: we still have $\amalg_\alpha \vdash \alpha^*$, and the system is given by the rules in Table 6, with the rules for the equivalence of contexts given in Table 5.

Note that, because $\amalg_\alpha$ only commutes with colimits, we do not have any equivalences of the form $\{\Gamma, \Gamma'\}_\alpha \approx \{\Gamma\}_\alpha, \{\Gamma'\}_\alpha$ or of the form $\{\varepsilon\}_\alpha \approx \varepsilon$: we do have the join rule, but it is not an equivalence and must be represented explicitly in the system.

Semantics for this system are categories fibred in complete Heyting algebras over a transition system: we do not need fibre products in the base. We can prove soundness, completeness, and cut elimination.

The operator α^*, having a left adjoint, commutes with $\wedge$ and $\top$, but not necessarily $\vee$ or $\bot$. Consider a proposition $P : s$ such that, for $\alpha : s \to t$,

$$P \vdash \alpha^* \bot;$$

$\frac{}{\Gamma \approx \Gamma}$	$\frac{\Gamma \approx \Gamma' \quad \Gamma' \approx \Gamma''}{\Gamma \approx \Gamma''}$
$\frac{\Gamma \approx \Gamma' \quad \Gamma_1 \approx \Gamma_1'}{\Gamma, \Gamma_1 \approx \Gamma', \Gamma_1'}$	$\frac{}{\Gamma, \varepsilon \approx \Gamma}$
$\frac{}{\Gamma, \Gamma' \approx \Gamma', \Gamma}$	$\frac{\Gamma : s \approx \Gamma' : s \quad \alpha : s \to t}{\{\Gamma\}_\alpha \approx \{\Gamma'\}_\alpha}$
$\frac{\alpha = \beta}{\{\Gamma\}_\alpha \approx \{\Gamma\}_\beta}$	$\frac{}{\{\Gamma\}_{\beta \circ \alpha} \approx \{\{\Gamma\}_\alpha\}_\beta}$
$\frac{}{\{\Gamma\}_{\mathrm{Id}} \approx \Gamma}$	

Table 5: Equivalence of Intuitionistic Contexts

if such a proposition were true at s, then performing α at s would result in a contradiction. So, $\alpha^* \bot$ gives a non-executability condition for α: it is a proposition such that, if true, α cannot be executed. There are no corresponding positive guarantees for the executability of α: the proposition $\neg\alpha^*(\bot)$ only gives something like the unprovability of the non-executability of α.

A.2 The Classical Nondeterministic System

Reiter's system had a chain of adjunctions $\coprod_\alpha \vdash \alpha^* \vdash \Pi_\alpha$: the existence of the right adjoint entails that the regression operator commutes with $\vee$ and $\bot$, which, in turn, means that our transition system is deterministic. If we retain the classical logic, but drop determinacy, we end up with two regression operators, which we can write $\Box^\alpha$ and $\Diamond^\alpha$: there are two adjoint operators, $\Box^\alpha$ and $\Diamond^\alpha$. We have $\Diamond^\alpha \vdash \Box_\alpha$ and $\Diamond_\alpha \vdash \Box_\alpha$, so the operators can be seen as adjoint modalities as in Ryan and Schobbens [10].

The context forming operators are $\{\cdot\}_\alpha$ and $\{\cdot\}^\alpha$, with the semantics of $\Box_\alpha$ and $\Box^\alpha$, respectively; equivalence of contexts is given by the same rules as for the intuitionistic system, enlarged with corresponding rules for $\{\cdot\}^\alpha$.

The system itself is given in Table 7: it is a one-sided system with propositions in negation normal form. It has strong affinities to the deep inference modal systems of [12, 13, 3, 4].

We should not also that the is a display logic, analogous to the systems of [7]: we have

Lemma 9. *For any marked context $\Delta[: s] : t$, there is a marked context $\overline{\Delta}[: t] : s$ and sequences ϑ and $\overline{\vartheta}$ of applications of adj and $\approx$ such that, for any contexts*

$$\frac{\Gamma \vdash Q \quad \Gamma \approx \Gamma'}{\Gamma' \vdash Q}\approx \qquad \frac{\Gamma[\{\Gamma'\}_\alpha,\ \{\Gamma''\}_\alpha] \vdash P}{\Gamma[\{\Gamma', \Gamma''\}_\alpha] \vdash P}\text{join}$$

$$\frac{}{\Gamma, P \vdash P}\text{Axiom} \qquad \frac{}{\Gamma[\bot] \vdash Q}\bot\ \text{L}$$

$$\frac{\Gamma[\varepsilon] \vdash Q}{\Gamma[\Gamma'] \vdash Q}\text{LW} \qquad \frac{\Gamma \vdash}{\Gamma \vdash Q}\text{RW}$$

$$\frac{\Gamma[P, P] \vdash Q}{\Gamma[P] \vdash Q}\text{LC}$$

$$\frac{\Gamma[P_1] \vdash Q \quad \Gamma[P_2] \vdash Q}{\Gamma[P_1 \vee P_2] \vdash Q}\vee\text{L} \qquad \frac{\Gamma \vdash Q_i}{\Gamma \vdash Q_1 \vee Q_2}\vee\text{R}_i$$

$$\frac{\Gamma[P_1, P_2] \vdash Q}{\Gamma[P_1 \wedge P_2] \vdash Q}\wedge\text{L} \qquad \frac{\Gamma \vdash Q_1 \quad \Gamma \vdash Q_2}{\Gamma \vdash Q_1 \wedge Q_2}\wedge\text{R}$$

$$\frac{\Gamma', P \to Q \vdash P : s \quad \Gamma[\Gamma', Q : s] \vdash R}{\Gamma[\Gamma', P \to Q : s] \vdash R}\to\text{L} \qquad \frac{\Gamma, P \vdash Q}{\Gamma \vdash P \to Q}\to\text{R}$$

$$\frac{\Gamma[P : t] \vdash Q \quad \alpha : s \to t}{\Gamma[\{\alpha^* P\}_\alpha] \vdash Q}\alpha^*\text{L} \qquad \frac{\{\Gamma\}_\alpha \vdash Q}{\Gamma \vdash \alpha^* Q}\alpha^*\text{R}$$

$$\frac{\Gamma[\{P\}_\alpha] \vdash Q}{\Gamma[\amalg_\alpha P] \vdash Q}\amalg_\alpha\text{L} \qquad \frac{\Gamma \vdash Q}{\{\Gamma\}_\alpha \vdash \amalg_\alpha Q}\amalg_\alpha\text{R}$$

$$\frac{\Gamma \vdash P : s \quad \Gamma'[P : s] \vdash Q}{\Gamma'[\Gamma] \vdash Q}\text{cut}$$

Table 6: The Intuitionistic Sequent Calculus

$$\frac{}{\vdash \Delta[P, \neg P]}\,\text{axiom} \qquad \frac{\vdash \Delta \quad \Delta \approx \Delta'}{\vdash \Delta'}\,\approx$$

$$\frac{\vdash \Delta[\{\Delta'\}_{\alpha:s\to t}, \{\Delta''\}_{\alpha:s\to t}]}{\vdash \Delta[\{\Delta', \Delta''\}_{\alpha:s\to t}]}\,\text{join}_\alpha \qquad \frac{\vdash \Delta[\{\Delta'\}^{\alpha:s\to t}, \{\Delta''\}^{\alpha:s\to t}]}{\vdash \Delta[\{\Delta', \Delta''\}^{\alpha:s\to t}]}\,\text{join}^\alpha$$

$$\frac{\vdash \{\Delta\}_\alpha, \Delta'}{\vdash \Delta, \{\Delta'\}^\alpha}\,\text{adj}^\alpha \qquad \frac{\vdash \{\Delta\}^\alpha, \Delta'}{\vdash \Delta, \{\Delta'\}_\alpha}\,\text{adj}_\alpha$$

$$\frac{\vdash \Delta[P, P]}{\vdash \Delta[P]}\,\text{C} \qquad \frac{\vdash \Delta[\varepsilon]}{\vdash \Delta[\Delta']}\,\text{W}$$

$$\frac{}{\vdash \Delta[\top]}\,\top$$

$$\frac{\vdash \Delta[P_1] \quad \vdash \Delta[P_2]}{\vdash \Delta[P_1 \wedge P_2]}\,\wedge \qquad \frac{\vdash \Delta[P_1, P_2]}{\vdash \Delta[P_1 \vee P_2]}\,\vee$$

$$\frac{\vdash \Delta[\{P\}_\alpha]}{\vdash \Delta[\Box_\alpha P]}\,\Box_\alpha \qquad \frac{\vdash \Delta[\{P\}^\alpha]}{\vdash \Delta[\Box^\alpha P]}\,\Box^\alpha$$

$$\frac{\vdash \Delta[P]}{\vdash \Delta[\{\Diamond_\alpha P\}^\alpha]}\,\Diamond_\alpha \qquad \frac{\vdash \Delta[P]}{\vdash \Delta[\{\Diamond^\alpha P\}_\alpha]}\,\Diamond^\alpha$$

$$\frac{\vdash P, \Delta \quad \vdash \neg P, \Delta'}{\vdash \Delta, \Delta'}\,\text{cut}$$

Table 7: The Nondeterministic Classical Sequent Calculus

$\Delta' : t$ and $\Delta'' : s$, we have

$$\begin{array}{c} \vdash \Delta[\Delta'], \Delta'' \\ \vdots\, \vartheta \\ \vdash \Delta', \overline{\Delta}[\Delta''] \end{array} \qquad \textit{and} \qquad \begin{array}{c} \vdash \Delta', \overline{\Delta}[\Delta''] \\ \vdots\, \overline{\vartheta} \\ \vdash \Delta[\Delta'], \Delta'' \end{array}$$

Proof. We define $\overline{\Delta}$ as follows:

$$\begin{aligned} \overline{\varepsilon}[] &= \varepsilon[] \\ \overline{\Delta_1, \Delta_2[], \Delta_3} &= \overline{\Delta}_2[\Delta_3, [], \Delta_1] \\ \overline{\{\Delta[]\}^\alpha} &= \{\overline{\Delta}[]\}_\alpha \\ \overline{\{\Delta[]\}_\alpha} &= \{\overline{\Delta}[]\}^\alpha \end{aligned}$$

We then prove the required properties by induction: the base case is trivial, and the inductive cases are proved as follows: first comma,

$$\begin{array}{c} \begin{array}{c} \vdash (\Delta_1, \Delta_2[\Delta'], \Delta_3), \Delta'' \\ \hline\hline \vdash \Delta_2[\Delta'], \Delta_3, \Delta'', \Delta_1 \end{array} \approx \\ \vdots\ \text{inductive step} \\ \begin{array}{c} \vdash \Delta', \overline{\Delta}[\Delta_3, \Delta'', \Delta_1] \\ \hline\hline \vdash \Delta', \overline{(\Delta_1, \Delta_2[\Delta''], \Delta_3)} \end{array} \text{ definition of } \overline{\cdot} \end{array}$$

then $\{\cdot\}^\alpha$

$$\begin{array}{c} \begin{array}{c} \vdash \{\Delta[\Delta']\}^\alpha, \Delta'' \\ \hline\hline \vdash \Delta[\Delta']\,\{\Delta''\}_\alpha \end{array} \text{ adj} \\ \vdots\ \text{inductive step} \\ \begin{array}{c} \vdash \Delta', \overline{\Delta}[\{\Delta''\}_\alpha] \\ \hline\hline \vdash \Delta', \overline{(\{\Delta[\Delta'']\}^\alpha)} \end{array} \text{ definition of } \overline{\cdot} \end{array}$$

($\{\cdot\}_\alpha$ is, of course, exactly similar). This gives us the rule applications ϑ: to produce ϑ', note that the rules in ϑ are all either instances of $\approx$ or of adj, and are thus invertible. □

Because of this display property, the cut rule – although it is only stated for cutformulae at top level – is, in fact, strong enough to yield the results that we want (in particular, we have a categorical semantics of the usual sort).

Note that, if we specialise the base category to have a single object, and morphisms the iterates of a single α, we get a proof theory for **K**: similarly, we can obtain **K4** by specialising to a single object and a morphism α with $\alpha \circ \alpha = \alpha$. We thus have a proof theory for various modal logics in the style of Simpson [11].

References

[1] Robert Bull and Krister Segerberg. Basic modal logic. In D. Gabbay and F. Guenther, editors, *Handbook of Philosophical Logic II: Extensions of Classical Logic*, volume 165 of *Synthese Library*, pages 1–88. Reidel, Dordrecht, first edition, 1984.

[2] Donald Davidson. The logical form of action sentences. In *Essays on Actions and Events*, pages 105–148. Oxford University Press, 1980. Originally published in N. Rescher (ed.), *The Logic of Decision and Action*, University of Pittsburgh Press, 1967.

[3] Robert Hein. Geometric theories and modal logic in the calculus of structures. Master's thesis, Technische Universität Dresden, 2005.

[4] Robert Hein and Charles Stewart. Purity through unravelling. In Paola Bruscoli, François Lamarche, and Charles Stewart, editors, *Structures and Deduction*, pages 126–143. Technische Universität Dresden, 2005. ISSN 1430-211X.

[5] B. Jacobs. *Categorical Logic and Type Theory.* Number 141 in Studies in Logic and the Foundations of Mathematics. North Holland, Amsterdam, 1999.

[6] E. Jónsson and A. Tarski. Boolean algebras with operators: Part 1. *American Journal of Mathematics*, 73:891–939, 1951.

[7] Markus Kracht. Power and weakness of the modal display calculus. In *Proof Theory of Modal Logic*, pages 93–121. Kluwer Academic Publishers, Dordrecht, 1996.

[8] F.W. Lawvere. Adjointness in foundations. *Dialectica*, 23:281–296, 1969.

[9] Raymond Reiter. *Knowledge in Action: Logical Foundations for Specifying and Implementing Dynamical Systems.* MIT Press, Cambridge MA, 2001.

[10] M. Ryan and P.-Y. Schobbens. Counterfactuals and updates as inverse modalities. *Journal of Logic, Language and Information*, 6, 1997.

[11] Alex K. Simpson. *The Proof Theory and Semantics of Intuitionistic Modal Logic.* PhD thesis, University of Edinburgh, 1994.

[12] Charles Stewart and Phiniki Stouppa. A systematic proof theory for several modal logics. In Renate Schmidt, Ian Pratt-Hartmann, Mark Reynolds, and Heinrich Wansing, editors, *Advances in Modal Logic*, volume 5 of *King's College Publications*, pages 309–333, 2005.

[13] Phiniki Stouppa. The design of modal proof theories: The case of S5. Master's thesis, Technische Universität Dresden, 2004.

[14] G. Graham White. Intensionality and circumscription. In Salem Benferhat and Enrico Giunchiglia, editors, *Proceedings of the 9th International Conference on Non-Monotonic Reasoning (NMR2002)*, pages 372–379, 2002. Toulouse, April 2002.