Partially-ordered Modalities

Gerard Allwein

US Naval Research Laboratory, Code 5540, Washington, DC, USA

William L. Harrison

Department of CS, University of Missouri, Columbia, Missouri, USA

Abstract

Modal logic is extended by partially ordering the modalities. The modalities are normal, i.e., commute with either conjunctions or disjunctions and preserve either Truth or Falsity (respectively). The partial order does not conflict with type of modality (**K**, **S4**, etc.) although this paper will concentrate on **S4** since partially ordered **S4** systems appear to be numerous. The partially-ordered normal modal systems considered are both sound and complete. Hilbert and Gentzen systems are given. A cut-elimination theorem holds (for partially ordered **S4**), and the Hilbert and Gentzen systems present the same logic. The partial order induces a 2-category structure on a coalgebraic formulation of descriptive frames. Channel theory is used to 'move' modal logics when the source and target languages may be different. A particular partially ordered modal system is shown to be applicable to security properties.

Keywords: Modal logic, partial order, Hilbert, Gentzen, channel theory

1 Introduction

This paper presents modal logics with several modalities where the modalities are partially ordered. The partial order can be added to any normal modal logic, however individual partial orders derive from some particular (application oriented) domain of discourse. Propositional dynamic logic [9] places a fair amount of algebraic structure on modalities. A weakened form of this is had by replacing the algebraic structure with a partial order. The partial order typically arises from some application area where the modalities express abstract features of the area and the partial order expresses a relationship among the modalities.

Partially ordered modal systems have pleasant properties; the Hilbert-style axiomization is simple and, in the **S4** case (and we suspect others) a convenient Gentzen-style

calculus which admits a cut elimination theorem. Using this theorem, it is easy to show that the Hilbert and Gentzen systems present the same logic. The simplicity of the logic is mirrored in the simplicity of the semantics. The collection of Kripke relations becomes partially ordered under the subset order. Soundness and completeness for partially ordered modal systems are also shown. One could mix modalities for the modal system **K** with **S4**, although a Gentzen system for such a logic might be a bit complicated. We suspect partial ordering modalities can be extended to other modal systems which are non-normal, leaving that extension to a later paper.

The semantics of a partially ordered system of modalities partially orders the relations of the Kripke semantics. The implications are more clearly seen when expressing that semantics in coalgebraic form. The coalgebra maps, which code the relations, are then partially ordered, and as such, form a category themselves. The result is the coalgebra maps are elements of 2-cells for a 2-category. The usual p-morphisms of modal logic are not changed except to enforce an additional requirement upon them that they respect the partial order of the coalgebras. This in effect makes them functors of the coalgebraic maps taken as the category of the underlying partial order. General frame morphisms are not effected by the partial order except for the same additional constraint imposed on the p-morphisms, i.e., that they respect the partial order.

The Vietoris topology usually given on (set) objects as the target of the Vietoris functor is not affected by the partial order on the coalgebras. So this too is independent of the partial order. Consequently, the Vietoris polynomials [10] are similarly unaffected.

A use of modal logic is in computer security. One wishes to 'move' theorems about a coarse grained security model to a fine grained system implementation model. However, the language for the security model and implementation model can be different. This paper shows way around this difficulty through the use of channel theory; a theorem about the security model can be 'moved' to the implementation model. The modalities require that the relation in the channel be a simulation relation.

This paper also presents a use of a partially ordered modal logic in the generalization of security properties which are second-order in nature [11]. Some second-order properties are expressible using modal logic and in this form, the properties are defined via certain functions on trace sets of data sequences in computer systems. These functions can be used to define closure operators; the closure operators have a natural partial order associated with them which is not a lattice order.

Section 2 presents the Hilbert and Gentzen systems. Section 3 presents the models for the partially ordered systems. Section 4 shows how to move modal logics using channel theory. Section 5 analyzes a particular generalization of security properties and shows how to connect them to a logic where the modalities represent those security properties in the logic.

2 Partially-ordered Modal Logics

The concentration will be on partially ordered **S4** modal logics since these are readily generated for computer systems by closing under functions on system behavior. They also have a nice Gentzen system that generalizes easily for the partially ordered modal-

ities.

2.1 Hilbert-style Systems

A normal modal logic [13] is any set of formulae which contains the classical propositional tautologies and is closed under Modus Ponens and Substitution, and also contains the *normality formula* $\vdash \Box(A \to B) \to (\Box A \to \Box B)$ and closed under the rule: $\vdash A$ implies $\vdash \Box A$ ($\vdash$ is a provability turnstile here). These prescriptions can be suitably altered to include the **S4** nature of the logic and the partial order on the modalities.

Definition 2.1 The modal Hilbert system with partial order $(H, \geq)$ has the axioms of classical propositional logic and the axiom $[h](A \to B) \to ([h]A \to [h]B), h \in H$, and, in addition,

A1: $[k]A \to [h]A$ for $k \geq h$ and $k, h \in H$.

which clearly shows that the relationship between the necessity modalities and the partial order. One also has the rules for proofs from assumptions (repetition, modus ponens), and modal generalization:

$$\frac{A \in \Gamma}{\Gamma \vdash A}\ rep \qquad \frac{\Gamma \vdash A \qquad \Gamma \vdash A \to B}{\Gamma \vdash B}\ mp \qquad \frac{\vdash A}{\vdash [h]A}\ gen$$

and allowing that $\langle h \rangle A$ can be defined as $\neg [h] \neg A$. Here, Γ is a set of formulas and $\vdash$ is the provability relation.

To axiomitize **S4**, one adds the usual axioms:

A2: $[h]A \to A$.

A3: $[h]A \to [h][h]A$.

Axioms A1 and A3 may be replaced with:

A3′: $[k]A \to [h][k]A$, $k \geq h$.

The axiom A1 is the axiom that codes the partial order, it may also be expressed using possibility as:

A1′: $\langle k \rangle A \to \langle h \rangle A$ for $k \leq h$.

There are two derived rules for the Hilbert-system when proofs are allowed to have assumptions, the usual deduction theorem and an extension of *gen*.

Theorem 2.2 *The classical deduction theorem continues to hold and an expanded gen rule is a derived rule of the Hilbert-style system:*

$$[k_1]B_1, \ldots, [k_n]B_n \vdash A \text{ implies}$$
$$[k_1]B_1, \ldots, [k_n]B_n \vdash [h]A,\ k_i \geq h.$$

2.2 Gentzen System for Partially Ordered **S4**

The rules for the classical propositional logic substrate of the modal system are Gentzen's original rules except that Permutation has been removed in favor of multisets. The context formulas in sequents are denoted with capital Greek letters.

Let the *active formula* in a premise of a rule be the instance of the formula which is altered and in a conclusion be the instance of the newly introduced formula. Let the *modal class* of a formula be either necessary, possible, or neutral depending upon whether the modal operator prefixing the formula is a necessity, possibility, or neither.

Definition 2.3 [Modal Condition (MC)] The Modal Condition is that all formulae on the same side of the $\vdash$ as the active formula must have the opposite modal class as the active formula, and all formulae on the opposite side of the $\vdash$ as the active formula must have the same modal class as the active formula.

The following rules define the modal partial order.

Definition 2.4 [Partially Ordered Modal Condition (NC)] Let NC be the condition

$$MC \text{ and } \forall C \in \Gamma \cup \Delta . c(C) \geq h$$

where $c(C)$ is the "closure" value of a formula using the modal partial order.

$$\frac{\Gamma, A \vdash \Delta \qquad NC}{\Gamma, \langle h \rangle A \vdash \Delta} \langle h \rangle \vdash \qquad \frac{\Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, \langle h \rangle B} \vdash \langle h \rangle$$

$$\frac{\Gamma \vdash \Delta, A \qquad NC}{\Gamma \vdash \Delta, [h] A} \vdash [h] \qquad \frac{\Gamma, A \vdash \Delta}{\Gamma, [h] A \vdash \Delta} [h] \vdash$$

The cut rule can be eliminated as in the classical and the **S4** modal systems (see the appendix for the proof). The proof that the Hilbert system is translatable to the Gentzen system requires cut elimination. The proof that the Gentzen system is translatable to the Hilbert system uses the two derived Hilbert rules.

Theorem 2.5 *Cut is an admissible rule in the Gentzen system without cut.*

Theorem 2.6 *The Partial Order* **S4** *Gentzen system and the Partial Order* **S4** *Hilbert system are equivalent.*

3 Models

A *Kripke Frame* $(X, (\mathcal{R}, \geq))$ is a collection of points (worlds, states, etc.) and a partial order of binary relations $(\mathcal{R}, \geq)$. The relations of $\mathcal{R}$ will be indexed by the variables h and k in the presentation below. Hence $R_h \subseteq R_k$ is presented as $k \geq h$. The Kripke relations satisfy the following:

K1: Monotonicity: $R_h xy$ and $k \geq h$ implies $R_k xy$.

In addition, for **S4**, the following axioms are added

K2: Reflexivity: $R_h xx$

K3: Transitivity: $R_h zx$ and $R_h xy$ implies $R_h zy$.

One can also take, in place of K1 and K3, the following:

K3′: Transitivity + Monotonicity: for $k \geq h$, $R_k yz$ and $R_h xy$ implies $R_k xz$.

The modalities are evaluated using the usual prescription from modal logic using the following definition.

Definition 3.1

$$x \models \langle h \rangle P \text{ iff } \exists y. R_h xy \text{ and } y \models P$$
$$x \models [h] P \text{ iff } \forall y. R_h xy \text{ implies } y \models P.$$

It follows easily that: $[h] \neg P = \neg \langle h \rangle P$.

When the modal frame arises from a modal algebra (which is a Boolean lattice with modal operators), the modalities have the following canonical definition:

Definition 3.2 For A a set of maximal filters of the modal algebra,

$$[h] A = \{x \mid \forall y. R_h xy \text{ implies } y \in A\}, \quad \langle h \rangle A = \{x \mid \exists y. R_h xy \text{ and } y \in A\}$$

It is widely known that not all normal modal logics are complete with respect to Kripke frames. To obtain completeness, valuations must be added so that all frames and all valuations are considered. This is similar to regaining completeness for second-order logic by including an algebra of sets in a frame where the algebra is not the entire power set of elements in the domain.

Following [4] (originally [8]) but using [10], a general frame $(X, (\mathcal{R}, \geq), X_*)$ is Kripke frame $(X, (\mathcal{R}, \geq))$ and an Boolean algebra of sets X_* closed under derived modal operators using the prescriptions for $[h] A$ and $\langle h \rangle A$ in Definition 3.2. A frame is *differentiated* if for all $x, y \in X$ with $x \neq y$, there is a 'witness' $a \in X_*$ such that $x \in a$ and $y \notin a$; *tight* if whenever y is not an R_h-successor (for $R_h \in \mathcal{R}$) of x, there a 'witness' a such that $y \in a$ and $x \notin \langle R \rangle a$; and *compact* if for every $C \subseteq X_*$, if C has the finite intersection property, then $\bigcap C \neq \emptyset$.

Typically, X_* is thought of as the clopen basis for the Stone topology on the Kripke frame. The question arises as to the relationship between that topology and the "closed sets" of **S4** possibility operators. The clopen basis is a Boolean algebra and that algebra is closed under induced modal operators given by Definition 3.2.

It is possible to describe the clopen sets of the Boolean algebra as arising from the identity relation. The identity modal operator $[1_X]$ corresponds to the identity relation on X, and $[1_X] C = \langle 1_X \rangle C$ for all elements of X_* (or propositions) C. All partial orders of relations can be extended with this relation with little effect on the dual algebras.

Lemma 3.3 *For all C, $[1_X] C = C = \langle 1_X \rangle C$.*

The identity relation has the effect of making the lattice of sets of a general frame a modal algebra. Put another way, every Stone space has a modal dual, albeit a trivial one. Hence, every normal modal logic can be extended to a partially ordered modal logic by including the identity relation. If the modal logic is at least **T** meaning it satisfies

at least the reflexivity axiom Rxx for all x, then the modal order is $1_X \subseteq R$ for R the modal relation for **T**.

Theorem 3.4 *The Gentzen rules for the system with the NC conditions are sound with respect to descriptive frames.*

The completeness argument is the usual algebraic argument using contraposition and a representation theorem. The modal representation theorem represents a modal algebra as an algebra of sets using the Kripke frame (Stone space) of the algebra. One defines the 1-1 homomorphism $\beta : A \longrightarrow \mathcal{P}(\mathcal{P}A)$ (where $\mathcal{P}$ is the powerset) from the modal algebra A to the double power set of A by:

$$\beta a = \{x \mid a \in x \text{ and } x \text{ is a maximal filter}\}.$$

It is not hard to show that $\beta\,[h]\,a = [h]\,\beta a$ and $\beta\,\langle h\rangle\,a = \langle h\rangle\,\beta a$. Set union, intersection, and complement interpret the classical logic logic connectives $\vee$, $\wedge$, and $\neg$. The Lindenbaum-Tarski modal algebra is generated via the logic by dividing out the word algebra of the logic by bi-implication and defining the operators via elements of the equivalence classes, i.e., $[P] \wedge [Q] \stackrel{def}{=} [P \wedge Q]$ where $[\ ]$ indicates bi-implication equivalence classes. To get a Kripke model requires that one take the (dual) Stone space containing all the maximal filters of the algebra and define the Kripke relations with:

$$R_h xy \text{ iff } [h]\,a \in x \text{ implies } a \in y.$$

Since $[h]$ and $\langle h\rangle$ are DeMorgan duals of each other, R_h admits an equivalent definition:

$$R_h xy \text{ iff } a \in y \text{ implies } \langle h\rangle\,a \in x.$$

The *canonical model* is the Kripke model generated by the Lindenbaum-Tarski algebra.

Lemma 3.5 *Monotonicity K1 holds in the canonical model. K2, and K3 hold if the frame is an* **S4** *frame.*

The following theorem holds via the usual contraposition argument.

Theorem 3.6 *The partially ordered, normal modal logics are complete with respect to descriptive frames.*

4 Moving Modal Logics

A tried and tested way to relate Kripke frames is via p-morphisms. These are also known as bounded morphisms, zig-zap maps, system maps, etc., and are the morphisms for the category of descriptive Kripke frames. The conditions guarantee that the power set algebras on the frames map properly when the inverse morphisms as inverse set maps are used.

An extrapolation of p-morphisms are bisimulation relations and their kin, simulation relations. These are inadequate when the modal formulas to be related come from

different languages. This occurs when one is relating properties of a high-level model or specification to a low-level implementation. One way around this difficulty is to use channel theory.

4.1 p-morphisms for Partially Ordered Descriptive Frames

Kripke frames can be expressed in terms of coalgebras for the covariant power set functor $\mathcal{P}$. $\mathcal{P}$ takes X to the set of subsets of X and $f : X \to Y$ to the forward image of f, i.e., $\mathcal{P}(f)(A) = \{f(x) \mid x \in A\}$. The coalgebra for Kripke relation R in $\mathbb{X} = (X, (\mathcal{R}, \geq))$ is defined with:

$$R_h x = \{y \mid R_h xy\}$$

(where the symbol R_h is overloaded).

A p-morphism $p : \mathbb{X} \to \mathbb{Y}$ is then a system map which means the square commutes for all $R_h \in \mathcal{R}$ where pR_h is the relation in $\mathbb{Y}$ which is the target of the p-morphism for R_h. $\mathbb{Y}$ could well have many other Kripke relations. The commutation means that, as relations, (1) $R_h xy$ implies $(pR_h)(px)(py)$ and (2) $(pR_h)(px)y$ implies there is some z such that $R_h xz$ and $pz = y$. To form the category of all coalgebras on $\mathbb{X}$, partially order the relations. This partially orders the relations as coalgebra morphisms. Let $\mathsf{Coalg}(\mathbb{X})$ be the collection of coalgebra morphisms on $\mathbb{X}$. As a set, $\mathsf{Coalg}(\mathbb{X})$ then forms a simple category. A morphism of frames $p : \mathbb{X} \to \mathbb{Y}$ then can be expected to be p-morphism for all the relations of $\mathbb{X}$ with the additional constraint that it also be a functor $p : \mathsf{Coalg}(\mathbb{X}) \to \mathsf{Coalg}(\mathbb{Y})$.

$$\begin{array}{ccc} X & \xrightarrow{\ p\ } & Y \\ {\scriptstyle R_h}\downarrow & & \downarrow{\scriptstyle pR_h} \\ \mathcal{P}(X) & \xrightarrow[\mathcal{P}(f)]{} & \mathcal{P}(Y) \end{array}$$

A morphism $p : \mathbb{X} = (X, (\mathcal{R}, \geq), X_*) \to \mathbb{Y} = (Y, (\mathcal{S}, \geq), Y_*)$ is a general frame morphism if it is a morphism for partially ordered frames and $p^{-1} : Y_* \to X_*$ is a modal homomorphism. General frame morphisms are also descriptive frame morphisms.

4.2 Channel Theory

The basic structures of channel theory [3,1] are deceptively simple. "Channel theory" is the colloquial term for Barwise and Seligman's term "information flow". Channel theory is both a qualitative information theory and a logic for distributed systems. The elements that are distributed are contexts called *classifications*. The classifications are connected by *infomorphisms*. Classifications and their morphisms appear in mathematics in various guises. What is unique about channel theory is that it uses a specific type of cocone called a "channel" as an organizing principle.

A classification contains two distinct collections of objects, tokens and types, connected with a binary relation. They could be anything that makes sense in using a classification as a model. However, most of modern language theory tends to use the term *types* in a different sense. The tokens will be model-theoretic entities such as states, theories, traces through a state space, etc. Bold-faced, slanted typefont always denote classifications.

Formally, the objects and morphisms are the same as Chu spaces [2]. However, in Chu spaces, no mention is made of the theory of a classification (see below) and most

the work appears to be directed at their categorical structure. Here, the categorical structure, while used, is not of primary importance. Also, Scott in [14] uses similar structures but particularizes the formalism to talk about computation. There is also an extensive literature on institutions [6]; this reference integrates institutions with the work of Barwise and Seligman.

Definition 4.1 A *classification*, $\boldsymbol{X}$, is a pair of sets, $Tok(\boldsymbol{X})$, and $Typ(\boldsymbol{X})$, and a relation, $\models_{\boldsymbol{X}} \subseteq Tok(\boldsymbol{X}) \times Typ(\boldsymbol{X})$ written in infix, e.g., $x \models_{\boldsymbol{X}} A$.

Information in a classification is of the form "x being A"; x need not be a model for a logic. In Section 5, x would stand for an arbitrary trace in a security model. x is a carrier of information with A being some of the information x carries. We express this by saying "$x \models_{\boldsymbol{X}} A$" is an example of the basic unit of information in channel theory.

Channel theory has its own notion of morphism, called an *infomorphism*. It is similar to a pair of adjoint functors in that it is a pair of opposing arrows with a condition similar to the adjoint's bijection.

Definition 4.2 A morphism $f : \boldsymbol{X} \rightarrow \boldsymbol{Y}$ of classifications, sometimes called an **infomorphism**, is a pair of opposing maps, $\overrightarrow{f}$ and $\overleftarrow{f}$ such that $\overrightarrow{f} : Typ(\boldsymbol{X}) \rightarrow Typ(\boldsymbol{Y})$ and $\overleftarrow{f} : Tok(\boldsymbol{Y}) \rightarrow Tok(\boldsymbol{X})$, and for all x and A, the following condition is satisfied: $x^f \models_{\boldsymbol{X}} A$ iff $x \models_{\boldsymbol{Y}} A^f$. For ease of presentation, $\overleftarrow{f}(x)$ is displayed as x^f and $\overrightarrow{f}(A)$ as A^f.

General frame morphisms are instances of infomorphisms with $Tok(\boldsymbol{X})$ being the points X of a Kripke frame, $Typ(\boldsymbol{X})$ being the set algebra X_* and the $\models_{\boldsymbol{X}}$ relation being $\in$ relation.

A commuting *cocone* consists of a graph homomorphism G from a graph to the category of classifications, a vertex classification $\boldsymbol{C}$ called the channel's *core*, and a collection of arrows $g_i : G(i) \rightarrow \boldsymbol{C}$. It is required that for all $f : i \rightarrow j$, $g_i = g_j \circ G(f)$. The base of the cocone is the objects and arrows identified by G.

Definition 4.3 An *information channel* is a co-cone in the category of classifications and infomorphisms.

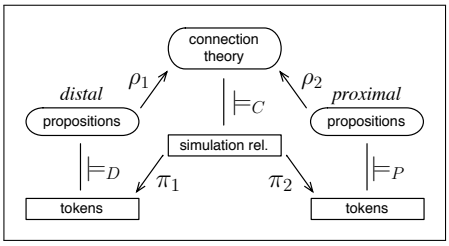
The smallest channel over a base is a colimit. Frequently, the smallest channel is not the most useful because a channel is used as a model. The smallest channel would simply connect the base with no additional modeling apparatus. A colimit in the category of classifications is a colimit on types and a limit on tokens.

Assuming a fixed classification $\boldsymbol{C}$, a *sequent* $\Gamma \vdash_{\boldsymbol{C}} \Delta$, is two sets of types connected by a relation $\vdash$. A *valid* sequent has the force of a meta-level implication of the form: for all tokens x, if $x \models_{\boldsymbol{C}} A$ for all of the types A in Γ, then $x \models_{\boldsymbol{C}} B$ for at least one type B in Δ. A classification's valid sequents are the classification's *theory*, also called the classification's *constraints*. A *channel's theory* refers to the theory in the core.

A channel $\boldsymbol{C}$ may connect a proximal classification $\boldsymbol{P}$, say a high level specification, with the distal classification $\boldsymbol{D}$, say a low level implementation [1]. Possibly

[1] The terms *proximal* and *distal* are merely convenient terms we use to refer to the two classifications

there are several design layers with every two adjacent layers connected by a channel, but the simple picture inset at the right will do for making the main argument apparent. In the diagram, the π_i are projections and ρ_i are injections into a disjoint sum. The rule for the morphisms $\langle \rho_i, \pi_i \rangle$ is: $\pi_i\langle x, y\rangle \models A$ iff $\langle x, y\rangle \models \rho_i A$. The proximal and distal languages are the type sets and are allowed to be different. The connection (channel) theory contains the rules for translation in the form of sequents.



A channel's sequents may be used to underwrite information flow through a channel where the pieces of information are tokens and the information they carry are properties. Using the channel in the diagram, let x be a token of $\boldsymbol{D}$, y a token of $\boldsymbol{P}$ and $\langle x, y\rangle$ a token of the channel $\boldsymbol{C}$. Further, let $\Gamma \subseteq Typ(\boldsymbol{D})$ and $\Delta \subseteq Typ(\boldsymbol{P})$ and Γ^{ρ_1} and Δ^{ρ_2} refer to the forward images of these sets under ρ_1 and ρ_2 respectively. If the sequent $\Gamma^{\rho_1} \vdash_{\boldsymbol{C}} \Delta^{\rho_2}$ as a constraint in the channel, it will relate tokens from $\boldsymbol{D}$ to tokens from $\boldsymbol{P}$ using the following form of reasoning:

$$
\begin{array}{lll}
x \models_{\boldsymbol{D}} \Gamma \text{ iff } \pi_1\langle x, y\rangle \models_{\boldsymbol{D}} \Gamma & & \text{assumption} \\
\quad \text{iff } \langle x, y\rangle \models_{\boldsymbol{C}} \Gamma^{\rho_1} & & \text{infomorphism condition} \\
\quad \text{implies } \langle x, y\rangle \models_{\boldsymbol{C}} \Delta^{\rho_2} & & \text{channel constraint} \\
\quad \text{iff } \pi_2\langle x, y\rangle \models_{\boldsymbol{P}} \Delta & & \text{infomophism condition} \\
\quad \text{iff } y \models_{\boldsymbol{P}} \Delta & & \text{assumption}
\end{array}
$$

Our goal will be to transfer a theorem of the form $A' \vdash [h]\, B'$ at the proximal level to a theorem $A \vdash [h]\, B$ in the distal level. A system $\boldsymbol{P}$ simulates a system $\boldsymbol{D}$ with respect to $[h]$ just when there is channel $\boldsymbol{C}$ such that "if $\langle x, x'\rangle$ are in the simulation relation $Tok(\boldsymbol{C})$ and $\boldsymbol{D}$ transitions under the relation R_h from x to y, then $\boldsymbol{P}$ transitions under the relation $R_{h'}$ from x' to y' and $\langle x', y'\rangle \in Tok(\boldsymbol{C})$." For sequents of this form to transfer from proximal to distal, the following conditions must be met:

C1: The connection theory in $\boldsymbol{C}$ relates non-modal proximal and distal types.

C2: The projection π_1 is surjective, i.e., must cover $Tok(\boldsymbol{D})$.

C3: $\boldsymbol{P}$ simulates $\boldsymbol{D}$ via the channel tokens $Tok(\boldsymbol{C})$.

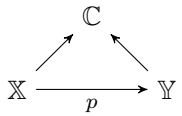
The proof of Theorem 4.4 is in the appendix.

Theorem 4.4 *For channel $\boldsymbol{C}$, if $\boldsymbol{P}$ simulates $\boldsymbol{D}$, $\rho_1 A \vdash_{\boldsymbol{C}} \rho_2 A'$, and $\rho_2 B' \vdash_{\boldsymbol{C}} \rho_1 B$:*

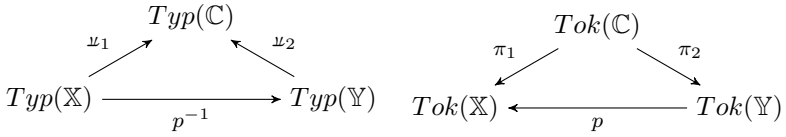
$$\Big(A' \vdash_{\boldsymbol{P}} [h']\, B'\Big) \text{ implies } \Big(A \vdash_{\boldsymbol{D}} [h]\, B\Big).$$

Bisimulations are extrapolations from p-morphisms and bisimulations in channel theory are extrapolations of general frame morphisms. Let $p : \mathbb{X} \to \mathbb{Y}$ be a general

frame morphism. Treat the general frames $\mathbb{X}$ and $\mathbb{Y}$ as classifications with the tokens being the Kripke points (worlds), the types being the set algebras and $\models$ as the $\in$ relational between points and sets. A colimit with vertex $\mathbb{C}$ over the morphism $p : \mathbb{X} \to \mathbb{Y}$ is then a bisimulation the category of classifications.



A simulation from distal to proximal uses the condition Rxy implies $(pR)(px)(py)$ whereas a simulation from proximal to distal uses the condition $(pR)(px)y$ implies there is some z such that $pz = y$ and Rxy. The (bi)simulation relation is the limit of the general frame morphism on the points of the domains and codomain of the general frame morphism.



The colimit $p : \mathbb{X} \to \mathbb{Y}$ on the types identifies types in the source and target of p. The connection theory is empty since that work has now been taken over by the identification via p on the types.

Now if one drops the requirement that there be a morphism linking the two relations and simply use the (bi)simulation relation as the link, one has the definition of (bi)simulation. In place of the identification $pA' = A$ and $pB' = B$, one takes $ɯ_1 A \vdash_{\mathbb{C}} ɯ_2 A'$ and $ɯ_2 B' \vdash_{\mathbb{C}} ɯ_1 B$ for transferring $A' \models_{\mathbb{Y}} [h]\, B'$ in the classification $\mathbb{Y}$ to $A \models_{\mathbb{X}} [h]\, B$ in the classification $\mathbb{X}$.

4.3 High-Level Simulation

Partially ordering the modalities suggest that a "higher level" notion of simulation. Let $\rightsquigarrow$ be a relation on Kripke relations for both $\boldsymbol{P}$ and $\boldsymbol{D}$. Define

$$x \models \boxed{\mathrm{R}} A \text{ iff } \forall S, \forall y. R \rightsquigarrow S \text{ and } Sxy \text{ implies } y \models A.$$

Definition 4.5 $\boldsymbol{P}$ simulates $\boldsymbol{D}$ with respect to R and R' when there is a channel $\boldsymbol{C}$ such that (where $C_{RR'}xy$ stands for a element of the $Tok(\boldsymbol{C})$)

$$R \rightsquigarrow S, Sxy, \text{and } C_{RR'}xx' \text{ implies } \exists S', \exists y'. R' \rightsquigarrow S', S'x'y' \text{ and } C_{RR'}yy'.$$

Theorem 4.6 *For channel* $\boldsymbol{C}$*, if* $\boldsymbol{P}$ *simulates* $\boldsymbol{D}$ *with respect to* R *and* R'*,* $\rho_1 A \vdash_{\boldsymbol{C}} \rho_2 A'$*, and* $\rho_2 B' \vdash_{\boldsymbol{C}} \rho_1 B$*:*

$$\Big(A' \vdash_{\boldsymbol{P}} \boxed{R'} B'\Big) \text{ implies } \Big(A \vdash_{\boldsymbol{D}} \boxed{R} B\Big).$$

Quantifying over all R *properly paired with an* R' *yields a global, higher order necessity.*

The proof is much like the proof for Theorem 4.4.

5 The Logic of Possibilistic Security

Separability, Generalized Noninterference, Noninference, and Generalized Noninference are the four possibilistic security properties handled by McLean [11]. Separability means that given a particular trace of high's behavior, any trace of low's behavior is possible, and vice versa; this relation is called *co-possibility*. Generalized Noninterference abstracts Goguen and Meseguer Noninterference [7]. The co-possibility relation becomes non-symmetric: any high-level trace is co-possible with any low-level trace, and *when only high-level input is considered* any low-level trace is co-possible with any high-level trace. Noninference "purges" high information from the input and output traces by overwriting that information with a constant value. This is weakened in Generalized Noninference, where only high input is purged. The relative strengths of these notions was captured by McLean a partial order; this order is reversed for the purposes of this paper in Fig. 1; the order indicates increasing restrictiveness from top to bottom. We have augmented by an additional element, Nothing, at the bottom for reasons that will become apparent later.

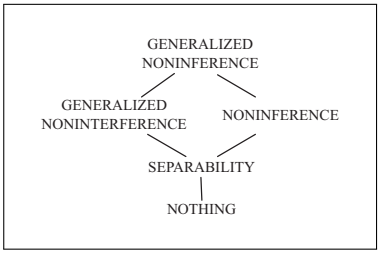


Figure 1

Relative Strengths of Possibilistic Models.

The diagram is somewhat misleading, because when viewed as a partial order of modalities, it turns out not to be a lattice. There is a Kripke relation for each element in the partial order. The partial order is the subset relation on the Kripke relations (as sets). However, Separability is not the set theoretic intersection of Generalized Noninterference and Noninference, and Generalized Noninference is not the set theoretic union. There are security properties which are the set theoretic intersection and union, but they are either unnamed or so far not put to any use. Also, since the relations involved here are at least transitive, the union of two transitive relations is not necessarily transitive.

McLean [11] uses state spaces (i.e., collections of system traces of input-output behavior) to show that the security properties do not attach themselves to traces but rather to sets of traces. He does this via an example showing that a particular property is not preserved through a reduction of system traces.

To formulate each possibilistic security property, McLean defines types of interleaving functions on traces of system behavior, where an *interleaving function* takes two traces and manufactures a third trace using some elements of the two traces. Each security property is then associated with a particular type of interleaving function. We summarize McLean's framework here for completeness.

Definition 5.1 [State Space] For non-negative integers m and n, let the sequences $\langle in_1, \ldots, in_m \rangle$ and $\langle out_1, \ldots, out_n \rangle$ be (respectively) tuples of distinct input and output variables such that the i-th input variable ranges over some alphabet I_i and the j-th

output variable ranges over some alphabet O_j. A **state space** Σ is the set

$$\{\langle\langle in_1, \ldots, in_m\rangle, \langle out_1, \ldots, out_n\rangle\rangle \mid in_i \in I_i, out_j \in O_j\}.$$

An element of the state space is called a *system state.*

Definition 5.2 Assume that for some $1 \leq p < m$, $in_1, \ldots, in_p$ are inputs of high-level users, the rest are inputs of low-level users. Similarly, for some $1 \leq q < n$ $out_1, \ldots, out_q$ are the outputs of the respective high-level users and the rest the respective outputs of low-level users. The state notation is condensed to $\langle highin : lowin, highout : lowout\rangle$:

$$\langle\overbrace{\langle in_1, \ldots, in_p\rangle}^{highin} : \overbrace{\langle in_{p+1}, \ldots, in_m\rangle}^{lowin}, \overbrace{\langle out_1, \ldots, out_q\rangle}^{highout} : \overbrace{\langle out_{q+1}, \ldots, out_n\rangle}^{lowout}\rangle$$

A *trace* is a (possibly finite) sequence of states. Input $highin_i$ ($lowin_i$) refers to the high (low) inputs of the i-th state in the trace with outputs $highout_i$ and $lowout_i$ defined analogously. The concatenation $\langle highin_i : lowin_i\rangle$ refers to the sequence $\langle in_1, \ldots, in_p, in_{p+1}, \ldots, in_m\rangle$ in the i-th state. The set of all traces is denoted $\hat{\Sigma}$. A state space, Σ, partitioned into high and low, is called a *two level security state space.*

McLean's Example.

One form of confidentiality property is that one kind of behavior is unaffected by another kind of behavior. An example is that any legal low-level behavior, i.e., a trace of states restricted to low-level input and output, must be co-possible with any legal high-level behavior. However, McLean showed that expressing the co-possibility relation with traces is problematic for reasons we now summarize.

Let $P(t)$ be true for trace t just when every possible high-level input-output pair can be paired with t's input-output and the result still be an allowable sequence. Consider the property $Q(t) \equiv in(t) = out(t)$ (i.e., for all i, $in(t)_i = out(t)_i$ where the equality is over sequences of input elements and output elements). Hence $Q(t)$ is true of just those sequences where the high input is mapped directly to low output for all positions (no permutations). Let P and Q stand for their extensions. The maximal trace set $\hat{\Sigma}$ consisting of all possible traces has property P. This implies that all properties, as sets of traces, satisfy P. It follows that $Q \subseteq \hat{\Sigma} \subseteq P$. This is a contradiction since Q does not satisfy P. The intuitive appeal of the language does not match the extensional, first-order semantics of the language.

Definition 5.3 Let state space $\Sigma = \{\langle\langle in_1, \ldots, in_m\rangle, \langle out_1, \ldots, out_n\rangle\rangle \mid in_i \in I_i \wedge out_i \in O_i\}$, let $\mu \in \{0,1,2\}^m$ and let $\nu \in \{0,1,2\}^n$. A function $f : \hat{\Sigma} \times \hat{\Sigma} \to \hat{\Sigma}$ is a *selective interleaving function of type* $F_{\mu,\nu}$ if and only if $f(t_1, t_2) = t$ implies that for all i, j such that $1 \leq i \leq m$ and $1 \leq j \leq n$,

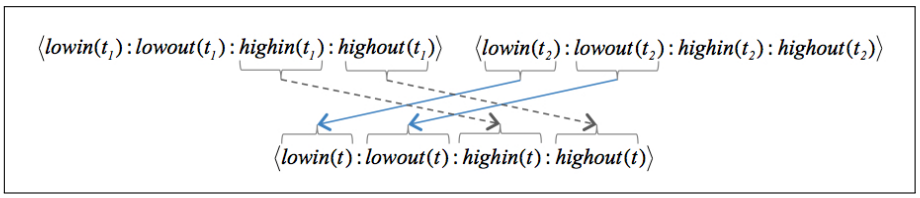
$$in[i](t) = in[i](t_1), \text{ if } \mu[i] = 1 \quad out[j](t) = out[j](t_1), \text{ if } \nu[j] = 1$$
$$in[i](t) = in[i](t_2), \text{ if } \mu[i] = 2 \quad out[j](t) = out[j](t_2), \text{ if } \nu[j] = 2$$

The interleaving class of type $F_{\langle 1^H:2^L\rangle,\langle 0^H:2^L\rangle}$ indicates $H+L$ inputs (from the notation $\langle 1^H : 2^L\rangle$) and $H+L$ outputs (from the notation $\langle 0^H : 2^L\rangle$). An *interleaving class* is a type of interleaving function. In this example, each function $f(t_1, t_2)$ in $F_{\langle 1^H:2^L\rangle,\langle 0^H:2^L\rangle}$ maps the high input of t_1 to the high input of the resulting trace, maps, the low input of t_2 to the low input of the resulting trace, does not care about high output, and maps the low output of t_2 to the low output of the resulting trace. The individual functions of $F_{\langle 1^H:2^L\rangle,\langle 0^H:2^L\rangle}$ may differ on how they set the high output of the resulting trace.

A *security class* is an interleaving class that corresponds to one of the security classes in the partial order. The following table summarizes the security classes with their interleaving class types:

Nothing	No functions
Separation	$F_{\langle 1^H:2^L\rangle,\langle 1^H:2^L\rangle}$
Generalized Noninterference	$F_{\langle 1^H:2^L\rangle,\langle 0^H:2^L\rangle}$
Noninference	$F_{\langle \lambda^H:2^L\rangle,\langle \lambda^H:2^L\rangle}$
Generalized Noninference	$F_{\langle \lambda^H:2^L\rangle,\langle 0^H:2^L\rangle}$

where λ is some fixed value to which the referenced input and output are set. Separability's lone interleaving function can be pictured as:



5.1 Channel Theory and Possibilistic Security

Example 5.4 [Trace Classification] Given a state space Σ, the trace classification $\boldsymbol{T}$ is

- $Tok(\boldsymbol{T}) = \{\langle s_1, s_2, \dots\rangle \mid s_i \in \Sigma\} \cup \{\langle s_1, \dots, s_n\rangle \mid s_i \in \Sigma\}$ $(= \hat{\Sigma})$;
- $Typ(\boldsymbol{T})$ are properties, i.e., open formulas of first-order logic with one free variable ranging over the tokens;
- $t \models_{\boldsymbol{T}} A$ is the satisfaction relation (trace t satisfies property A).

A subset of the token set is called a *trace set*. A trace set U is a **reduction** of a trace set V if and only if $U \subseteq V$.

5.2 Security Properties and Reductions.

McLean and others use the term *refinement* for the term *reduction*. A reduction of a property P is a system S such that $S \subseteq P$ and so S is said to "refine" or "reduce" P. Consider a possible reduction infomorphism $r : \boldsymbol{T} \to \boldsymbol{T}'$ for $\boldsymbol{T}, \boldsymbol{T}'$ trace classifications. The channel types are first-order logic descriptions of traces.

If A is a description of a collection of traces, then $Tok(A)$ refers to all traces which satisfy A. Let $\overrightarrow{r} = 1_{Typ(\boldsymbol{T})}$ and $\overleftarrow{r} : Tok(\boldsymbol{T}') \rightarrowtail Tok(\boldsymbol{T})$ be an injection. The reduction r takes the property A into the property A^r and is an instance of the rule in the inset figure where Γ^r, Δ^r are r applied element-wise to the formulas in Γ, Δ. This rule preserves validity even when $\overleftarrow{r}$ is not an injection.

$$\frac{\Gamma \vdash_{\boldsymbol{T}} \Delta}{\Gamma^r \vdash'_{\boldsymbol{T}} \Delta^r}\ r\text{–Intro}$$

The notion of a possibilistic security property being a collection of traces is defective for the mere fact that these properties cannot be stated using it if traces are thrown out in going from $\boldsymbol{T}$ to $\boldsymbol{T}'$.

Continuing McLean's Example.

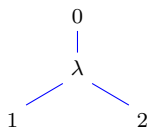
Recall, $P(t)$ is true just when every possible high-level input-output pair can be paired with t's input-output and the result still be an allowable sequence, i.e., still be a token in the trace classification. Let $\boldsymbol{T}$ be a trace set classification whose tokens are an entire state space, $Tok(\boldsymbol{T}) = \hat{\Sigma}$. As earlier, now in channel theoretic language, $Tok(\boldsymbol{T}) = Tok(P)$, and $Q(t) \equiv in(t) = out(t)$ i.e., for all i, $in(t)_i = out(t)_i$ where the equality is over sequences of input elements and output elements. Hence, $Tok(Q) \subseteq Tok(P)$ yet Q does not imply P.

The sequent $Q \vdash_{\boldsymbol{T}} P$ is a constraint of the classification $\boldsymbol{T}$. However, the property Q is being thought of as a classification that describes all of the tokens satisfying Q. Let $\boldsymbol{T}'$ refer to the classification with types Q and P but with only the tokens from $\boldsymbol{T}$ that satisfy Q. The obvious map is $r : \boldsymbol{T} \rightarrow \boldsymbol{T}'$ such that $\overleftarrow{r}$ is the injection induced by $Tok(\boldsymbol{T}') \subseteq Tok(\boldsymbol{T})$ and is the identity on types. The rule inset to the right should have produced a good constraint in $\boldsymbol{T}'$. It does not because r is not an infomorphism. In particular, $t^r \models_{\boldsymbol{T}} P$ iff $t \models_{\boldsymbol{T}'} P$ is false in the forward direction as long as $t \models_{\boldsymbol{T}'} P$ means $P(t)$. Put another way, the quantifiers defining $P(t)$ are not restricted to $Tok(\boldsymbol{T}')$. This is precisely the move (from $t \models_{\boldsymbol{T}'} P$ to $P(t)$) that one uses to conceive of modal logic as being a variant of second-order logic.

$$\frac{Q \vdash_{\boldsymbol{T}} P}{Q^r \vdash_{\boldsymbol{T}'} P^r}\ r\text{–Intro}$$

5.3 Formalizing the Intended Model.

Considering the partial order of security properties, any system closed under Separability is also closed under Generalized Noninterference. This would indicate that if Separability is interpreted using a modal closure operator $\langle s \rangle$ and Generalized Noninterference is interpreted using a modal closure operator $\langle g \rangle$, then $\langle s \rangle\, P \subseteq \langle g \rangle\, P$. However, let Noninference be interpreted by the modal closure operator $\langle n \rangle$. It would be expected that $\langle s \rangle\, P \subseteq \langle n \rangle\, P$. Closing under Noninference's purge function will not include the traces of $\langle s \rangle\, P$ if there is some trace $t \in P$ and $t \neq f(t_1, t_2)$ for f being Separability's interleaving function and $t \neq \langle \langle \lambda^{H+L}, \lambda^{H+L} \rangle, \ldots \rangle$.



Extended Partial Order

In a similar vein, closing under one General Noninterference function will not necessarily imply closure under another. To use the partial order requires that closing be cumulative looking up the partial order and that it be inclusive at any one point in the order.

One way to achieve a cumulative and inclusive order is to use currying to turn the two-place functions into collections of single-place functions. To make the order cumulative, include functions from higher up in the order in a collection lower in the order. To make the order inclusive requires that it be closed under composition. To make the order work to define set closures requires that the identity function be an allowable function. The extended typing introduced for Noninference and Generalized Noninference can be further extended to correspond to the partial order in the Extended Partial Order.

Definition 5.5 Given functions $f_{t_1}t_2 \stackrel{def}{=} f(t_1, t_2)$ and $g_{t_2}t_1 \stackrel{def}{=} g(t_1, t_2)$. $F^{\circ}_{\mu,\nu}$ is the collection of curried interleaving functions generated by the two-place interleaving functions type $F_{\mu,\nu}$ and includes the identity function, $\mathrm{id}_{\hat{\Sigma}}$.

Definition 5.6 The *interleaving type partial order* is $F^{\circ}_{\mu_k,\nu_k} \geq F^{\circ}_{\mu_h,\nu_h}$ iff for all i, j, $\mu_k[i] \geq \mu_h[i]$ and $\nu_k[i] \geq \nu_h[i]$ where $\mu_h, \mu_k, \nu_h, \nu_k$ take values in the Extended Partial Order.

The typing structure is too restrictive for the interleaving type partial order to be cumulative. This can be remedied by taking the union of all the curried interleaving functions above and $\mathrm{id}_{\hat{\sigma}}$.

Definition 5.7 Define $k \geq h$ iff $h \subseteq k$, for $k \stackrel{def}{=} \bigcup\{F^{\circ}_{\mu_h,\nu_h} \mid F^{\circ}_{\mu_k,\nu_k} \geq F^{\circ}_{\mu_h,\nu_h}\}$.

Intuitively, an *interleaving class of operators* is the collection of interleaving operators generated via currying from an interleaving class of functions combined with the interleaving classes of operators further up the partial order. The Nothing interleaving class of operators is the empty set $\emptyset$, and the Separation class contains its lone interleaving operator and the identity operator on traces.

Theorem 5.8 *Let $k \geq h$ and $g \in k, f \in h$ for interleaving classes $F^{\circ}_{\mu_k,\nu_k}, F^{\circ}_{\mu_h,\nu_h}$ respectively, then $g \circ f, f \circ g \in k$ regardless of how they are internally curried, and h and k are semigroups.*

Let h be an interleaving class of operators. If $f, g \in h$, then $f \circ g \in h$. Since function composition is associative, h is a semigroup. If t is a trace and P is closed under a collection of interleaving operators, then t is still in P, hence the identity function, $\mathrm{id}_{\hat{\Sigma}}$ is a valid interleaving operator for all classes. This makes h a monoid. Also, if $k \geq h$ and $f \in h$, then $f \in k$.

To say that P is closed under an interleaving function f is to say that for all $t_1, t_2 \in P$, $f(t_1, t_2) = t$ and $t \in P$. There is a mismatch between the binary relations of modal logic and what appears to be a three place relation, namely $f(t_1, t_2) = t$. It will not work to simply apply g_{t_1} on the curried functions, since the condition is then rendered for all $t_2 \in P$, $g_{t_1}t_2 = t$ and there is no guarantee that $t_1 \in P$. The solution is to use pairs of traces. We define the *intended model* as the model of sequences and

interleaving classes. Let h be an interleaving class, then (regardless of internal currying): $R_h\langle t_1, t_2\rangle\langle t_1, t\rangle$ iff $\exists f.f_{t_1} \in h$ and $f_{t_1}t_2 = t$. To finish the intended model, the R_h relations must be reversed, e.g., $R_hxy \mapsto \breve{R}_hyx$ which yields 'forward looking' relations for the $[h]$ and $\langle h\rangle$ operators.

Theorem 5.9 *The intended model satisfies axioms K1, K2, and K3.*

Theorem 5.10 *Separability is not the set theoretic intersection of Generalized Noninterference and Noninference, and Generalized Noninference is not the set theoretic union.*

6 Conclusion

The concentration in this paper was on **S4** since it has a pleasant Gentzen system and many applications can be found where properties are closures under some class of functions. The partial order is on the modalities themselves and represents a higher-order structure of the modalities reminiscent of dynamic logic. The difference between dynamic logic and partially ordered modal logics is the lack of algebraic structure on the modalities. The semantics of partial ordered modal logics reflects the partial order directly as the set theoretic subset relation on the Kripke relations. The partial order can be used to give a higher order notion of simulation for 'moving' logics among classifications in channel theory. This allows for expressing theorems of high level models or specification of system behavior to be transferred to low level implementations when the implementations satisfy several concrete criteria.

The partial order of the logic is reduced in the paper to a single axiom. This axiom can be added conservatively (i.e., preserving completeness) to any normal modal logic. The identity relation makes Boolean propositional logic a modal logic of the most simplest form. Any normal modal logic then becomes a partially ordered modal logic with the addition of the identity relation.

We would like to further explore the utility of partially ordering non-normal modalities as these have a number of Computer Science applications (e.g., they tend to crop up in linear logic). There are other aspects of modal logic we have not yet explored such as interactions with Sahlqvist formulas and correspondence theorems [4]. Substructural logics represent another avenue of research. Positive modal logic [5,12] weakens the Boolean propositional part of conventional modal logic to a logic without negation. It would be interesting to formulate a version of partially ordered modal logic for these conditions.

References

[1] Allwein, G., *A qualitative framework for Shannon information theories*, in: *Proceedings of the New Security Paradigms Workshop, 2004* (2005), pp. 23 – 31.

[2] Barr, M., $*$*-autonomous categories and linear logic*, Mathematical Structures Computer Science **1** (1991), pp. 159–178.

[3] Barwise, J. and J. Seligman, "Information Flow: The Logic of Distributed Systems," Cambridge University Press, 1997, cambridge Tracts in Theoretical Computer Science 44.

[4] Blackburn, P., M. de Rijke and Y. Venema, "Modal Logic," Cambridge University Press, 2001, cambridge Tracts in Theoretical Computer Science, No. 53.

[5] Dunn, J. M., *Positive modal logic*, Studia Logica **55** (1995), pp. 301–317.

[6] Goguen, J., *Information integration in institutions*, in: L. Moss, editor, *Thinking Logically: a Memorial Volume for Jon Barwise*, Indiana University Press, 200x pp. 1–48.

[7] Goguen, J. A. and J. Meseguer, *Security policies and security models*, in: *Proceedings of the 1982 IEEE Symposium on Security and Privacy* (1982), pp. 11–20.

[8] Goldblatt, R., *Metamathematics of modal logic*, Reports on Mathematical Logic **6** (1976), pp. 41–77.

[9] Harel, D., D. Kozen and J. Tiuryn, "Dynamic Logic," MIT Press, 2000.

[10] Kupke, C., A. Kurz and Y. Venema, *Stone coalgebras*, in: H. P. Gumm, editor, *Coalgebraic Methods in Computer Science, Electronic Notes in Theoretical Computer Science*, 1 **82**, 2003, pp. 170–190.

[11] McLean, J., *A general theory of composition for a class of "possibilistic" properties*, IEEE Transactions on Software Engineering **22** (1996), pp. 53–67.

[12] Palmigiano, A., *Coalgebraic semantics for positive modal logic*, in: H. P. Gumm, editor, *Coalgebraic Methods in Computer Science, Electronic Notes in Theoretical Computer Science*, 1 **82**, 2003, pp. 221–236.

[13] Sambin, G. and V. Vaccaro, *Topology and duality in modal logic*, Annals of Pure and Applied Logic **37** (1988), pp. 249–296.

[14] Scott, D. S., *Domains for denotational semantics*, in: *An extended version of the paper prepared for ICALP '82* (1982), pp. 1–47.

7 Appendix

7.1 Cut Elimination Proof

A *conclusion parameter* of an instance of a rule is an instance of a formula which is not newly introduced formula of a logical rule nor the formula introduced by thinning. A *premise parameter* is one that matches in an obvious way a conclusion parameter of an instance of a rule. To say a "formula is generated on the left" means that the bottom rule of the left subtree above the *mix* produces the formula. The formula may already exist in the conclusion of the rule, in that case another copy is generated. Similar remarks hold for generating on the right. Hence a formula is parametric just when it is not generated and generated when it is not parametric.

The proofs below are by examination of the rules using a double induction on (1) the rank, which is the sum of the distances along all the branches of the proof tree from the active cut formula to the leaves of the proof tree, and (2) the degree which is a count of the connective in the cut formula. The notation R_l refers to the rank of the left cut formula and R_r refers to the rank of the right cut formula.

Since cut elimination for the classical base of the logic is well known, only the cases involving the modal rules will be shown here. All uses of double lines in the proof will be to represent multiple uses of a rule. The locution $C \vdash C$ stand for $C \vdash$ and $\vdash C$. First we prove a lemma that will cut the number of cases that have to be independently considered down to a manageable number.

Lemma 7.1 *If the mix formula is parametric in the rule producing the left premise and that premise was not produced with the* $\langle h\rangle \vdash$ *or* $\vdash [h]$ *rules, then we can always reduce the rank of the cut formula in on the left subtree. A similar statement holds for right premises and right subtrees.*

Proof. $\mathrm{R_l} > 1$.

$$\dfrac{\dfrac{\Gamma \vdash \Delta[A][B]}{\Gamma \vdash \Delta[A][\langle h\rangle\, B]} \vdash \langle h\rangle \qquad \Phi[A] \vdash \Psi}{\Gamma, \Phi \vdash \Delta[\langle h\rangle\, B]\Psi}\, cut$$

is transformed into

$$\dfrac{\dfrac{\Gamma \vdash \Delta[A][B] \qquad \Phi[A] \vdash \Psi}{\Gamma, \Phi \vdash \Delta[B], \Psi}\, cut}{\Gamma, \Phi \vdash \Delta[\langle h\rangle\, B], \Psi} \vdash \langle h\rangle$$

The case where the left rule above the *mix* is $[h] \vdash$ is similar. □

Theorem 7.2 *All uses of cut in a proof may be eliminiated.*

Proof.

The proof follows Gentzen's original proof, we only list cases relevant to the modalities.

Case 1: $\mathrm{R_l} + \mathrm{R_r} = 2$

Case 1.1: One premise is an axiom.

Case 1.2: $\vdash [h], [h] \vdash$

$$\dfrac{\dfrac{\Gamma \vdash \Delta[A]}{\Gamma \vdash \Delta[[h]\, A]} \vdash [h] \qquad \dfrac{\Phi[A] \vdash \Psi}{\Phi[[h]\, A] \vdash \Psi}\, [h] \vdash}{\Gamma, \Phi \vdash \Delta, \Psi}\, mix$$

which is transformed into

$$\dfrac{\Gamma \vdash \Delta[A] \qquad \Phi[A] \vdash \Psi}{\Gamma, \Phi \vdash \Delta, \Psi}\, mix$$

Case 1.3: $\vdash \langle h\rangle, \langle h\rangle \vdash$ This case is similar to Case 7.1.

Case 1.4: $\vdash K, \langle h \rangle \vdash$,

$$\dfrac{\dfrac{\Gamma \vdash \Delta[B]}{\Gamma \vdash \Delta[B, \langle h \rangle\, A]} \vdash K \quad \dfrac{\Phi[A] \vdash \Psi}{\Phi[\langle h \rangle\, A] \vdash \Psi} \langle h \rangle \vdash}{\Gamma, \Phi \vdash \Delta[B], \Psi}\, mix$$

which is transformed into

$$\dfrac{\Gamma \vdash \Delta[B]}{\Gamma, \Phi \vdash \Delta[B], \Psi}\, K \vdash, \vdash K$$

This transform is justified since the rule producing the right hand premise assures us that we may thin in all of the elements needed to produce Φ and Ψ.

Case 1.5-7: $(\vdash K, [h] \vdash)$, $(\vdash \langle h \rangle, K \vdash)$, $(\vdash [h], K \vdash)$ These are similar to Case 7.1.

Case 2: $\mathrm{R_l} > 1$

Case 2.1: The mix formula is parametric on the left.

Case 2.1.1: The mix formula is generated on the right.

Case 2.1.1.1: $\langle h \rangle \vdash$, any Logical Rule

$$\dfrac{\dfrac{\Gamma[B] \vdash \Delta[A] \quad NC}{\Gamma[\langle h \rangle\, B] \vdash \Delta[A]} \langle h \rangle \vdash \quad \Phi[A] \vdash \Psi}{\Gamma[\langle h \rangle\, B], \Phi \vdash \Delta, \Psi}\, mix$$

is transformed to

$$\dfrac{\dfrac{\Gamma[B] \vdash \Delta[A] \quad \Phi[A] \vdash \Psi}{\Gamma[B], \Phi \vdash \Delta, \Psi}\, mix \quad NC}{\Gamma[\langle h \rangle\, B], \Phi \vdash \Delta, \Psi} \langle h \rangle \vdash$$

From the NC condition of the left cut premise, A is of the form $\langle k \rangle\, C$ and $h \geq k$. Since the mix formula is generated on the right, $\langle k \rangle\, C$ was generated by a use $\langle k \rangle \vdash$. The NC condition on that use means $k \geq c(D)$ for all $D \in \Phi, \Psi$. Hence $h \geq c(D)$ for all $D \in \Gamma, \Delta, \Phi, \Psi$. Also, since the premise for the right hand cut sequent satisfies MC, and since Δ^* only differs from Δ by the elimination of A and similarly for Φ^*, the premise of the cut rule in the conclusion proof fragment also satisfies MC. Hence this premise satisfies NC and the use of the $\langle h \rangle$ is proper.

Case 2.1.1.2: $\vdash [h]$, any Logical Rule. This case is similar to the preceding case.

Case 2.1.1.3-4: ($[h]\vdash$, any non-modal Rule), ($\vdash \langle h\rangle$, any non-modal Rule). These cases are handled by the Lemma or, in the case of the right rule above the mix being $C\vdash$ or $K\vdash$, by Gentzen's original proof.

Case 2.1.2: The *mix* formula is parametric in the right.

Case 2.1.2.1: $\langle h\rangle\vdash, \vdash [k]$

$$\dfrac{\dfrac{\Gamma[A]\vdash\Delta[B] \quad NC}{\Gamma[\langle h\rangle A]\vdash\Delta[B]}\vdash\langle h\rangle \quad \dfrac{\Phi[B]\vdash\Psi[C] \quad NC}{\Phi[B]\vdash\Psi[[k]C]}\vdash[h]}{\Gamma,\Phi^*\vdash\Delta^*,\Psi[[k]C]}\,mix$$

This case cannot happen since the NC condition on the right forces B to be of the form $[h']D$ and this contradicts the NC condition the left.

Case 2.1.2.2-4: ($\langle h\rangle\vdash, \langle k\rangle\vdash$), ($\vdash [h], \langle k\rangle\vdash$), ($\vdash [h], \vdash [h]$). These cases are similar to Case 2.1.2.1.

Case 2.1.2.3-4: ($\vdash [h]$, any Logical Rule), ($\vdash \langle h\rangle$, any Logical Rule). These cases are similar to Cases 2.1.1.3-4 and handled by the Lemma.

Case 2.2: The *mix* formula is generated on the left:

Case 2.2.1: The *mix* formula is generated on the right.

Case 2.2.1.1: $\vdash\langle h\rangle, \langle h\rangle\vdash$

$$\dfrac{\dfrac{\Gamma\vdash\Delta[A][\langle h\rangle A]}{\Gamma\vdash\Delta[\langle h\rangle A]}\vdash\langle h\rangle \quad \dfrac{\Phi[A]\vdash\Psi \quad NC}{\Phi[\langle h\rangle A]\vdash\Psi}\langle h\rangle\vdash}{\Gamma,\Phi^*\vdash\Delta^*,\Psi}\,mix$$

is transformed into

$$\dfrac{\dfrac{\dfrac{\dfrac{\Gamma\vdash\Delta[A][\langle h\rangle A] \quad \Phi[\langle h\rangle A]\vdash\Psi}{\Gamma,\Phi^*\vdash\Delta^*[A],\Psi}\,mix}{\Gamma,\Phi^*\vdash\Delta^*[\langle h\rangle A],\Psi}\vdash\langle h\rangle \quad \Phi[\langle h\rangle A]\vdash\Psi}{\Gamma,\Phi^*,\Phi^*\vdash\Delta^*,\Psi,\Psi}\,mix}{\Gamma,\Phi\vdash\Delta,\Psi}\,C\vdash C$$

The first mix reduces the rank of the mix formula, and the second reduces the rank on the left to 1.

Case 2.2.1.2: $\vdash [h], [h] \vdash$ This case mirrors the previous case. □

7.2 Proof of Simulation Theorem 4.4

Theorem 7.3 *For channel* $\boldsymbol{C}$*, if* $\boldsymbol{P}$ *simulates* $\boldsymbol{D}$*,* $\rho_1 A \vdash_{\boldsymbol{C}} \rho_2 A'$*, and* $\rho_2 B' \vdash_{\boldsymbol{C}} \rho_1 B$*:*

$$\Big(A' \vdash_{\boldsymbol{P}} [h'] B'\Big) \text{ implies } \Big(A \vdash_{\boldsymbol{D}} [h] B\Big).$$

Proof. Assume $x \models_{\boldsymbol{D}} A$ and that $R_h yx$ holds. Using C2, there is some tuple $\langle x, x'\rangle \in Tok(\boldsymbol{C})$ and $\pi_1\langle x, x'\rangle = x$. From the morphism condition on $\langle \pi_1, \rho_1\rangle$, $\langle x, x'\rangle \models_{\boldsymbol{C}} \rho_1 A$. C1 must include $\rho_1 A \vdash_{\boldsymbol{C}} \rho_2 A'$ in which case $\langle x, x'\rangle \models_{\boldsymbol{C}} \rho_2 A'$. The morphism condition on $\langle \pi_2, \rho_2\rangle$ implies $\pi_2\langle x, x'\rangle \models_{\boldsymbol{P}} A'$ and hence $x' \models_{\boldsymbol{P}} A'$. From the antecedent in the theorem, $x' \models_{\boldsymbol{P}} [h] B'$. Using C3, there is some y' such that $\langle y, y'\rangle \in Tok(\boldsymbol{C})$ and that $R'_h y' x'$ holds where R'_h is the modal relation corresponding to $[h]$. From the fact that $x' \models_{\boldsymbol{P}} [h] B'$, it follows that $y' \models B'$. Since $\pi_2\langle y, y'\rangle = y'$, $\pi_2\langle y, y'\rangle \models B'$ and from the morphism condition, $\langle y, y'\rangle \models_{\boldsymbol{C}} \rho_2 B'$. C1 must include $\rho_2 B' \vdash_{\boldsymbol{C}} \rho_1 B$ in which case $\langle y, y'\rangle \models_{\boldsymbol{C}} \rho_1 B$. From the morphism condition, $\pi_1\langle y, y'\rangle \models_{\boldsymbol{D}} B$ and hence $y \models_{\boldsymbol{D}} B$. The resulting conditions show that $x \models_{\boldsymbol{D}} [h] B$ and that x satisfies $A \vdash_{\boldsymbol{D}} [h] B$. □