



NRL/MR/5540--14-9565

Distributed Logics

GERARD ALLWEIN

*Center for High Assurance Computer Systems
Information Technology Division*

WILLIAM L. HARRISON

*University of Missouri
Columbia, Missouri*

October 3, 2014

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE				Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.					
1. REPORT DATE (DD-MM-YYYY) 03-10-2014		2. REPORT TYPE Memorandum Report		3. DATES COVERED (From - To)	
4. TITLE AND SUBTITLE Distributed Logics				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) Gerard Allwein and William L. Harrison*				5d. PROJECT NUMBER 55-8089-HG	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER 9888	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Research Laboratory, Code 5540 4555 Overlook Avenue, SW Washington, DC 20375-5320				8. PERFORMING ORGANIZATION REPORT NUMBER NRL/MR/5540--14-9565	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Assistant Secretary of Defense for Research and Engineering Research/Information Systems and Cyber Security 4800 Mark Center Drive Alexandria, VA 22350-3600				10. SPONSOR / MONITOR'S ACRONYM(S) ASDR&E	
				11. SPONSOR / MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.					
13. SUPPLEMENTARY NOTES *Dept. of Computer Science, University of Missouri, Columbia, Missouri					
14. ABSTRACT Modal logics typically have only one domain of discourse---i.e., the collection of worlds or states. For distributed computing systems, however, it makes sense to have several collections of worlds for each component and to relate one component's local worlds to another using either relations or special maps. To this end, we introduce distributed logics. Distributed logics lift the distribution structure of a distributed system directly into the logic, thereby parameterizing the logic by the distribution structure itself. Each domain supports a "local modal logic." The connections between domains are realized as "distributed modal connectives" where these connectives take propositions in one logic to propositions in another. More generally, weak distributed systems require neighborhood semantics and hence the connection between domains becomes a neighborhood map linking each world in one domain to a collection of worlds in another domain. In sufficiently strong distributed systems, the maps may be Kripke relations linking worlds from two different domains. We illustrate distributed logics with the outline of a security verification for a hardware distributed system (i.e., a system-on-a-chip) with components that must be woven into proofs of security statements.					
15. SUBJECT TERMS Logic Distributed systems System-on-a-Chip					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Gerard Allwein
Unclassified	Unclassified	Unclassified	Unclassified	14	19b. TELEPHONE NUMBER (include area code) (202) 404-3748
Unlimited	Unlimited	Unlimited	Unlimited		

Distributed Logics

March 15, 2014

1 Introduction

Distributed systems are ubiquitous in computing and engineering, yet they have been somewhat obscured in the philosophical world. A *distributed logic* is a collection of *local modal logics* linked together by *distributed modal connectives* each of which takes formulas in one logic and returns formulas in a different logic. Semantically, each local logic is interpreted over a collection of worlds. Let this collection be called the *local collection* for this local logic. A *local neighborhood (nbd) map* takes each world to a set of worlds taken from the local collection and is used to interpret the modal connectives of the local logic. The distributed modal connectives are also interpreted using nbd maps; here, the nbd maps take worlds from a local collection of worlds to nbds of worlds from a different local collection.

Extra properties, via logical axioms and rules, can be imposed on the interpreting nbd maps. This is precisely analogous to traditional modal logic and imposing conditions on Kripke relations or nbd maps. Many of the usual conditions such as normality or functionality can be generalized from their traditional counterparts. The selection of axioms reflects the model theory one needs for an application. If one adds enough axioms to force the distributed modal connectives to be normal modal connectives (even though they map from one logic to another), the interpreting nbd maps can be defined to be Kripke relations that, here, span local collections. There are other approaches to locality in logic: channel theory [6, 2], institutions [13], Chu spaces [7], etc. There are also multi-agent logic systems [12]. What distinguishes distributed logics from these are that the morphisms, i.e., the nbd maps, have been lifted into the logic and hence are given properties via logical axioms and rules.

The obvious practical question is “What are distributed logics good for?”. Consider Fig. 1. This is a simplified view of an actual system. The *cpu* issues a request to the *bus master* to read from the bus. The *mux* either connects line *u* to the bus or leaves it undefined as a “tri-state value”, $\perp$, which will be used as a predicate in the security specification below. The control line tells the *mux* when to make the connection. The formulas are distributed logic statements that hold of the *bus master*:

$$(control = 0) \supset [c](\perp(u)), \quad (control = 1) \supset [c](bus = u)$$

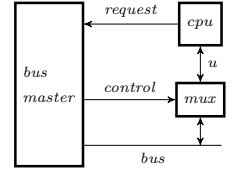


Figure 1

The *bus master* does not have access to the line *u* and, hence, *u* cannot be part of the *bus master*’s state. The two statements hold of any state in the *bus master* since the *control* line is either 0 or 1. Every state in the *bus master* is related to at least one state of the *cpu-mux* via the *control* line; this co-occurrence relation, which will be called *C*, is used in interpreting the (necessity) distributed modal connective $[c]$.

Let σ be a state in the *bus master*’s worlds where *control* = 0. The evaluation of the first statement is then

$$\begin{aligned}
 & \sigma \models_{bus\ master} (control = 0) \supset [c](\perp(u)) \\
 & \therefore \sigma \models_{bus\ master} [c](\perp(u)) \\
 & \therefore \text{for all } \tau \in cpu\text{-mux} (C\sigma\tau \text{ implies } \tau \models_{cpu\text{-mux}} \perp(u))
 \end{aligned}$$

Note how the appellation of the semantic turnstile changes from *bus master* to *cpu-mux* as the formula is evaluated.

More abstractly, some security properties of distributed systems can be expressed using these forms of logic statements. Distribution prevents taking large cross products of states which tend to degrade the performance of model checking algorithms beyond reasonable levels. Intuitively, although space prevents us from explicating it here, distributed logic statements can be paired with a process algebra where the terms yield something like a tensor product of states of the components.

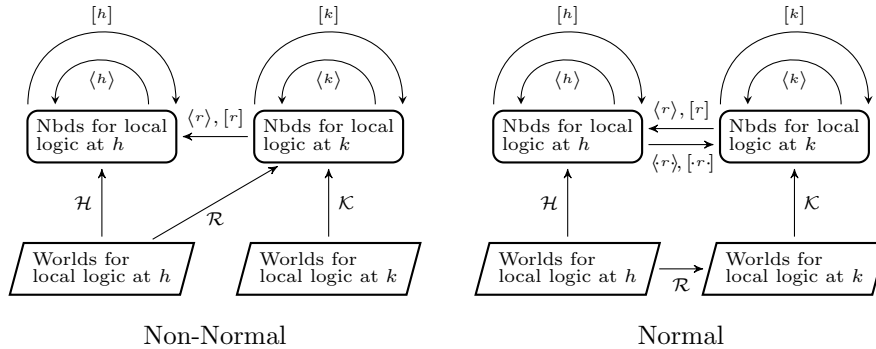
There is another use for distributed logics in testing systems. The situation frequently arises where one is tasked with producing a distributed system for a system-on-a-chip where what is known as “foreign IP (intellectual property)” must be used. While in one state of a known component, tests are made to a foreign IP component. The tests generate neighborhoods about a state in which the test was made. The situation is similar to the non-normal diagram in the next section. The worlds are the states and the $\mathcal{R}$ neighborhood map indicates tests for each state (world).

2 The Logic

Distributed logics refer to all the logics with a distribution structure as we will specify it for non-normal and normal modal logics. A distributed logic starts with a directed graph where every node constitutes a *local logic*. Each node is a classical propositional logic with a set of modal connectives, and any axioms and rules to govern behavior. The graph makes apparent the structure of the collection of the local logics. Using an arc for every modal connective can get a bit “noisy” due to classical negation and defining possibility from necessity or vice versa. Instead, arcs specify semantic maps that must exist in any interpretation. Each arc is then a bit of abstract syntax which, in an interpretation, will be turned in for a nbd map.

2.1 Conventions

The semantic picture for models of two local logics h and k semantically connected by either a nbd map $\mathcal{R}$ or a relation $\mathcal{R}$ is the following diagram:



The $\langle r \rangle$ and $[r]$ are *forward looking* modal connectives in that their interpretation by the neighborhood map $\mathcal{R}$ looks forward along $\mathcal{R}$ from head to tail. The $\langle r \rangle$ and $[r]$ are backwards looking modal connectives. Let x be world for h and y be a world for k , then in the first diagram, $\mathcal{H}x$, $\mathcal{R}x$, and $\mathcal{K}y$ are each a collection of neighborhoods. One can add axioms for the distributed modal connectives to force the nbd maps to be simulation relations in the normal case and to respect a simulation condition for neighborhoods in the non-normal case.

Other axioms can require that the relations be functions. Using both simulation and function axioms requires that the relations be p-morphisms, and the resulting logic is simulation logic [4]. We simplify a bit and allow the indices h and k to refer to a local logic as well as indexing the local logic’s modal connectives, and we also assume there are only the modal connectives $[k]$, $\langle k \rangle$ in the logic for k and similarly for h . There are no problems adding more modal connectives and axioms and rules to govern their behavior. In particular, one can add conditions expressing the interaction between local modal connectives and distributed modal

connectives. We use the simulation axiom (see Axiom F1 below) to illustrate this. There are a wealth of choices that are driven by the particular distributed system under consideration.

In sufficiently weak modal systems, it is not necessary that a point be a member of its neighborhoods. Here, it is almost a requirement or the notion of distribution is not present. Model theoretically, $\mathcal{R}$ relates two different neighborhood systems. These neighborhood maps, as morphisms, compose and there is an identity for each domain of worlds. In the normal case, the morphisms can be represented as relations with suitable modifications of the definitions.

The notation $\text{dom}(r)$ refers to the domain or source of the arc r in a graph and $\text{cod}(r)$ refers to the codomain or target of the arc, $r : \text{dom}(r) \curvearrowright \text{cod}(r)$. We use the locution $\langle h \rangle \in \text{dom}(r)$ to refer to a modal connective in the logic associated with the node which is the source for the arc $r : h \curvearrowright k$. The symbol $\equiv$ is used for *bi-implication*, i.e., $P \equiv Q$ stands for $(P \supset Q) \wedge (Q \supset P)$. We use the following letter conventions:

entity	description
h, k, l	nodes and endo-arcs in a graph $\mathfrak{G}$
$\langle h \rangle, [h], \langle k \rangle, [k]$	local modal connectives at nodes h and k
r, s	arcs in a graph $\mathfrak{G}$
$\langle r \rangle, [r], \langle s \rangle, [s]$	forward-looking modal connectives for arcs r and s
$\langle \cdot r \rangle, [\cdot r]$	backward-looking modal connectives for arc r
H, K	sets of worlds in interpretations for logics at h, k
$\mathcal{H}, \mathcal{K}$	interpret modal connectives for endo-arcs at h, k
$(H, \mathcal{H}, \mathbb{H}), (K, \mathcal{K}, \mathbb{K})$	neighborhood frames for the logics at h and k
$\mathcal{R}, \mathcal{S}$	interpret modal connectives for arcs r and s

We will assume, without loss of generality, that each local logic can be interpreted with a single neighborhood map. Hence, the node and its endo-arc can share the same label with use disambiguating meaning. This allows us to equate a node usually labeled h or k with the modal logic at that node.

2.2 Axioms and Rules

A local logic is “local” in that it is associated with one node in the graph. In this paper, the accompanying notion of a global logic does not entail formulas “spanning” two local logics in the sense of P in one logic implying Q in another where implying is reified as an implication connective (and similarly with other two place connectives). Each formula lives entirely within a single local logic although it may contain subformulas from others.

The distributed logic graphs we use have *endo-diagrams*, each of which is a labeled node and a single endo-arc (self-arc). Each endo-arc will be translated into an endo-morphism. Each node is required to have at least one endo-diagram whose arc will be translated into an identity morphism. This is necessary since the models for the logic will be a category. The graph axioms specify which local logics there are to be, which morphisms are to appear in any model, and force identity morphisms to exist. Each local logic may have its own propositional atoms and local modal connectives. The **S** specification and **A** and **B** axioms are not optional.

Graph Specification **S**:

- | | |
|---|--|
| S1. A graph $\mathfrak{G}$ of nodes and arcs
A set $\mathfrak{D}$ of endo-diagrams | S2. An endo-diagram with an
arc i for each node in $\mathfrak{G}$ |
|---|--|

Axiom Schemes **A**: For each node in $\mathfrak{G}$,

- | | |
|---|--|
| A1. all truth functional theorems
of a propositional logic | A2. Modal axioms for a logic
at this node |
|---|--|

Each node h must contain an endo-diagram for each class of modal operators in its local logic. A class is the collection $\{[h], \langle h \rangle\}$ if the local logic is non-normal and $\{[h], \langle h \rangle, [\cdot h], \langle \cdot h \rangle\}$ if the local logic is normal.

Axiom Schemes **B**: These axioms force arcs to be interpreted as morphisms in a category. For arcs $r : h \curvearrowright k$ and $s : k \curvearrowright l$,

$$\text{B1. } P \equiv [i] P$$

$$\text{B2. } [r] [s] P \equiv [s \circ r] P$$

Axiom Schemes C: Taken all together these axioms would force the distributed modal connectives to be normal. Each may be optionally added.

$$\text{C1. } [r] P \wedge [r] Q \supset [r](P \wedge Q)$$

$$\text{C2. } [r](P \wedge Q) \supset [r] P \wedge [r] Q$$

$$\text{C3. } \top \supset [r] \top$$

The Axiom Schemes **C** should be present to specify simulation logic [4]; they also allow the specification of backward looking connectives residuated (see [11]) with their forward looking counterparts. Simulation logic could also be built on a non-normal basis using the same main simulation axiom. However, the semantic conditions then involve neighborhoods, not relations.

Definition of Possibility: $\langle m \rangle P \stackrel{\text{def}}{=} \neg [m] \neg P, \quad m \in \{k, r\}$

Rules A: For each local logic k ,

$$\frac{\vdash_k P \quad \vdash_k P \supset Q}{\vdash_k Q}$$

$$\frac{\vdash_k (P_1 \wedge \dots \wedge P_n) \equiv P}{\vdash_k ([k] P_1 \wedge \dots \wedge [k] P_n) \equiv [k] P}$$

Rule B: For each $r : h \curvearrowright k$ arc in $\mathfrak{G}$,

$$\frac{\vdash_k (P_1 \wedge \dots \wedge P_n) \equiv P}{\vdash_h ([r] P_1 \wedge \dots \wedge [r] P_n) \equiv [r] P}$$

where the subscripted $\vdash$ indicates the local logic to which the proof sign attaches.

We will only be concerned with the forward versions of necessity and possibility connectives since the backwards versions are so similar. The backward versions are only present for normal systems.

2.3 Options

Axiom Schemes D: The **D** axioms are examples of extra properties to be enforced on the interpreting morphisms. Other axioms can be added as well, we use these as paradigm examples:

$$\text{D1. } [r] P \supset \langle r \rangle P$$

$$\text{D2. } \langle r \rangle P \supset [r] P$$

In non-normal systems, the axiom D1 specifies consistency and the axiom D2 specifies completeness, both with respect to the collection of neighborhoods about any world when the world is in the source of the nbd map used in interpreting $[r]$ and $\langle r \rangle$. In normal systems, the first specifies the interpreting relation be total on its domain and the second that it act functionally (see Section 3.1).

Axiom Schemes E: The axiom E1 is only necessary if you wish the classical proposition logic at $\text{dom}(r)$ to be included in the logic at $\text{cod}(r)$. This condition is part of the definition of simulation [9] although it is not strictly necessary in that it can be removed without damaging the logic.

For all propositional letters p ,

$$\text{E1. } p \supset [r] p$$

From now on, a distributed logic contains at least the specification **S** and axiom schemes **A** and **B**, and the Definition of Possibility, and the rules **A** and **B**. Normal distributed logics include the non-normal axioms and rules and the Axioms Schemes **C**. Axiom Schemes **C** can also be added individually rather than en masse if only a subset of the properties of normality are desired. The Axiom Schemes **D** are of interest and we have modeling conditions for them. The Axiom Scheme **E** must be handled quite separately in the semantics. Other axioms can be added, we stop with the list chosen for the purposes of this presentation.

Axiom Scheme F: Simulation logic [4] requires for an arc $r : h \curvearrowright k$ in $\mathfrak{G}$, and modal connectives $[h] \in \text{dom}(r), [k] \in \text{cod}(r)$,

$$\text{F1. } \langle r \rangle [k] P \supset [h] \langle r \rangle P$$

In normal distributed logics, the axiom F1 forces the arcs in the graph to be interpreted as simulation relations and B2 forces composition of relations to hold, where a simulation relation is one “half” of a bisimulation [16]. One common use of the simulation relation is when the interpretation of $\langle r \rangle$ via a relation $\mathcal{R}$ is a p-morphism. To force this, add the Axiom Schemes **C** and **D** to the simulation axiom.

3 Frames and Algebras

In keeping with our simplifications, assume there is only one local modality per frame, including both a $\Box$ and $\Diamond$ since they are inter-definable. More modal connectives can be added if needed if needed by the particular distributed system under consideration.

3.1 Frames

Definition 3.1.1 A neighborhood frame is a structure $\mathcal{H} = (H, \mathcal{H}, \mathbb{H})$ such that H is a collection of worlds, $\mathbb{H}$ is a collection of neighborhoods which are subsets of H and the entire collection is closed under the Boolean operations and under the operations $[h], \langle h \rangle : \mathbb{H} \rightarrow \mathbb{H}$ given by:

$$[h]C \stackrel{\text{def}}{=} \{x \in H \mid C \in \mathcal{H}x\}, \quad \langle h \rangle C \stackrel{\text{def}}{=} \{x \in H \mid -C \notin \mathcal{H}x\},$$

with where $-C$ is the set complement of C in H . $\mathcal{H} : H \rightarrow \mathcal{P}\mathbb{H}$ is a nbd map taking every world of H into a collection of neighborhoods. We use the same symbol for the frame and its nbd map, and let use disambiguate what is meant.

Each node in a distributed logic’s graph has a local logic associated with it. That local logic, in turn, must have a neighborhood frame associated with it.

Definition 3.1.2 Let $\mathcal{H}$ and $\mathcal{K}$ be neighborhood frames. A nbd map $\mathcal{R} : \mathcal{H} \rightarrow \mathcal{K}$ is a map (also using the symbol $\mathcal{R}$) $\mathcal{R} : H \rightarrow \mathcal{P}\mathbb{K}$ such that for any $C \in \mathbb{K}$,

$$[r]C \stackrel{\text{def}}{=} \{x \in H \mid C \in \mathcal{R}x\} \in \mathbb{H}, \quad \langle r \rangle C \stackrel{\text{def}}{=} \{x \in H \mid -C \notin \mathcal{R}x\} \in \mathbb{H}.$$

Let $\mathcal{R} : \mathcal{H} \rightarrow \mathcal{K}$ and $\mathcal{S} : \mathcal{K} \rightarrow \mathcal{L}$ be morphisms. The identity morphism $I : \mathcal{H} \rightarrow \mathcal{H}$ and the composition $\mathcal{S} \circ \mathcal{R} : \mathcal{H} \rightarrow \mathcal{L}$ are defined with $(x \in H)$

$$Ix \stackrel{\text{def}}{=} \{C \in \mathbb{H} \mid x \in C\}, \quad (\mathcal{S} \circ \mathcal{R})_x \stackrel{\text{def}}{=} \{C \in \mathbb{L} \mid \{y : C \in \mathcal{S}y\} \in \mathcal{R}_x\}.$$

Each arc $r : h \curvearrowright k$ of the graph must be associated with a *semantic morphism* in the interpretation. The semantic morphisms are *neighborhood maps* $\mathcal{R} : H \rightarrow \mathcal{P}\mathbb{K}$ where $\mathbb{K}$ is the collection of neighborhoods, i.e., the $\mathbb{K}$ in $(K, \mathcal{K}, \mathbb{K})$. In the normal case, the neighborhood maps can be replaced with relations. These relations are derivable in the usual way [10], i.e., $\mathcal{R}xy$ iff $y \in \bigcap \mathcal{R}x$; that is, take intersection of all the neighborhoods at x under $\mathcal{R}$.

Note that the definition for composition can be rewritten as

$$(\mathcal{S} \circ \mathcal{R})_x \stackrel{\text{def}}{=} \{C \in \mathbb{L} \mid [s]C \in \mathcal{R}_x\}$$

using the Definition 3.1.2 for $[s]C$. The definition is found in Manes [14] for the Kleisli category of the double power set monad. Our models are always in the category of neighborhood frames.

Each node representing a distinct local logic must be mapped to a distinct frame object in any interpretation. This informal way of restricting interpretations is the result of treating the graph as not defining everything in a distributed logic, but the alternative would make the logic impenetrable.

The corresponding Kripke frame conditions for the logical axioms are

Frame Conditions S:

FS1. A category of *local neighborhood frames* and neighborhood maps

FS2. An identity morphism for the i arc in $D \in \mathfrak{D}$

Frame Conditions A: For each node in $\mathfrak{G}$,

FA1. A set of classical worlds

FA2. Frame conditions for a local logic at this node

Frame Conditions B: For $I : H \rightarrow \mathbb{H}$, $\mathcal{R} : H \rightarrow \mathbb{K}$ and $\mathcal{S} : K \rightarrow \mathbb{L}$ in $\mathfrak{G}$

FB1. $I_x = \{C \in \mathbb{H} \mid x \in C\}$

FB2. $(\mathcal{S} \circ \mathcal{R})_x = \{C \in \mathbb{L} \mid [s] C \in \mathcal{R}_x\}$

Frame Conditions C:

FC1. $B, C \in \mathcal{R}_x$ implies $B \cap C \in \mathcal{R}_x$

FC2. $B \in \mathcal{R}_x$ and $B \subseteq C$ implies $C \in \mathcal{R}_x$

FC3. $\top \in \mathcal{R}_x$

Frame Conditions D:

FD1. $C \in \mathcal{R}_x$ implies $\neg C \notin \mathcal{R}_x$

FD2. $C \notin \mathcal{R}_x$ implies $\neg C \in \mathcal{R}_x$

Frame Condition F:

FF1. $-\{y \mid C \in \mathcal{K}y\} \notin \mathcal{R}x$ implies $\{z \mid \neg C \notin \mathcal{R}z\} \in \mathcal{H}x$

with the convention that the nbd maps that use upper case script relation letters will interpret modal connectives that use the corresponding lower case Roman letters. Each distributed frame category interpreting a distributed logic will have the conditions matching the axioms. The frame conditions **S**, **A**, and **B** are always assumed, the others are required if the corresponding axioms are present in the modeled local logic.

Slightly different frames are used for the axiom E1; the local frames will contain functions to interpret constants, one for every atomic proposition of the local logic for which the local frame provides a model.

The following proposition allows for the use of one neighborhood frame per local logic.

Proposition 3.1.3 *There are no provable instances of formulas of the form $P \bullet Q$ for $\bullet \in \{\supset, \wedge, \vee\}$ with P in one local logic and Q in different local logic.*

The proof is an easy induction on the axiom schemes and rules. The consequence is that no formula in the logic has a binary connective between formulas in two different local logics.

Note that we stated the above proposition in terms of formula “instances” rather than formulas because it is possible to attach a local logic to more than one node in the graph. In effect, this would give more than one instance of the logic in the entire distributed logic.

Using the semantics conditions, it is easy to show that

$$x \models_{\mathcal{H}} \neg[r] \neg P \text{ iff } x \models_{\mathcal{H}} \langle r \rangle P,$$

hence the definition of $\langle r \rangle$ in terms of $[r]$ makes sense. A distributed category model has neighborhood frames for every node with a valuation for each node. The morphisms are neighborhood maps.

Definition 3.1.4 *A distributed category model is a neighborhood frame category with a valuation and a local frame for each local logic. The local frame and its valuation are called a local model. A valuation specifies a collection of points in the local frame where the atomic propositions are true.*

3.2 Algebras

We rely on heterogeneous (multisorted) algebras [8] for the free algebra construction. The categorical version is most easily accessible in [1] who attribute the multisorted (non-categorical) case to [8].

Definition 3.2.1 (Birkhoff and Lipson [8]) A heterogeneous algebra is a system $A = [\mathcal{L}, F]$ in which

1. $\mathcal{L} = \{S_i\}$ is a family of non-void sets S_i of different types of elements, each called a phylum of the algebra A . The phyla S_i are indexed by some set I ; i.e., $S_i \in \mathcal{L}$ for $i \in I$ (or are called by appropriate names).
2. $F = \{f_\alpha\}$ is a set of finitary operations operations, where each f_α is a mapping

$$f_\alpha : S_{i(1,\alpha)} \times S_{i(2,\alpha)} \times \cdots \times S_{i(n(\alpha),\alpha)} \rightarrow S_{p(\alpha)}$$

for some non-negative integer $n(\alpha)$, function $i_\alpha : j \rightarrow i(j, \alpha)$ from $n(\alpha) = \{1, 2, \dots, n(\alpha)\}$ to I , and $p(\alpha) \in I$. The operations f_α are indexed by some set Ω ; i.e., $f_\alpha \in F$ for $\alpha \in \Omega$ (or are called by appropriate names).

Definition 3.2.2 A distributed algebra appropriate for a distributed logic is a heterogeneous algebra with a modal algebra, called a local modal algebra, for each node of a graph, identity modal operators for each node, and distributed operators $\langle r \rangle$ and $[r]$ for every arc r of the graph. For $r : h \curvearrowright k$ in the graph,

- $[r][s]a = [s \circ r]a$;
- $[i]a = a$, for the i arc in an endo-diagram;
- if the Axiom Schemes **C** are used
 - $[r]a \wedge [r]b \leq [r](a \wedge b)$;
 - $[r](a \wedge b) \leq [r]a \wedge [r]b$;
 - $\top_{\mathbb{H}} = [r]\top_{\mathbb{K}}$, for $\top$ the top of a Boolean lattice;;
- if Axiom Schemes **D** are used
 - $[r]a \leq \langle r \rangle a$;
 - $\langle r \rangle a \leq [r]a$;
- $\langle r \rangle[k]a \leq [h]\langle r \rangle a$, if Axiom Scheme **F** is used.

The axiom E1 will be handled in the next subsection where we must add constant operations and functions to help interpret the propositional atoms.

Appropriate distributed algebras give a “localization” view of heterogeneous algebras which is isomorphic to the definition given above. Each phylum S_i with operators defined only upon S_i is a local modal algebra. The operations associated with $r : h \curvearrowright k$ of the graph map from a local modal algebra to a local modal algebra. This stratifies the heterogeneous distributed algebra and treats every local modal algebra as an object in the surrounding distributed algebra.

Algebraic versions of soundness and completeness depend on the Lindenbaum-Tarski (LT) algebra. We must first show that the operators all respect the congruence of bi-implication induced on the local word algebras by the local logics. The only operators not already covered in previous modal algebraic work are the distributed operators.

Lemma 3.2.3 The distributed operators respect bi-equivalence.

The connective $[r]$ respects bi-equivalence because of the Rule **B**. Using Boolean negation, it is easy to show that $\langle r \rangle$ does as well.

Next, we must show that the LT algebra is actually a distributed algebra. The only operators that are at issue are the distributed operators.

Lemma 3.2.4 *The LT distributed operators satisfy the required properties for a distributed algebra.*

The equivalence classes for the LT algebras are defined (as usual) with $\llbracket P \rrbracket = \{Q \mid \vdash_{\mathcal{H}} P \equiv Q\}$. The operators are defined inductively, i.e., $\llbracket P \rrbracket \wedge \llbracket Q \rrbracket = \llbracket P \wedge Q \rrbracket$, $[r]\llbracket P \rrbracket = \llbracket [r]P \rrbracket$.

Corollary 3.2.5 *The LT heterogeneous algebra is a distributed algebra.*

Proof: (Proof Outline) The free heterogeneous algebra is the usual algebra of equivalence classes of terms in the variables as generators. One runs the induction procedure to get the word algebras over all the local logics simultaneously [8], then divide out by the equalities in each algebra. Proposition 3.1.3 shows that no additional sorts over and above the local modal algebra carrier sets are necessary. Lemma 3.2.3 shows that the replacement property for the bi-implication congruence holds for each operator. Finally, Lemma 3.2.4 shows each of LT operators satisfy the distributed algebra axioms. ■

Theorem 3.2.6 *Distributed Logic is sound with respect to the algebraic and distributed frame category models.*

outline: Soundness over the algebraic models is an induction starting with a valuation into a distributed algebra and then using the fact that the LT algebra is a free algebra for the heterogeneous class of distributed algebras. From this, it is easy to see that $\supset$ interprets to $\leq$ in the algebra. The axioms of the LT algebra clearly interpret to the axioms of the logic, and the rules of the logic preserve truth in the algebra. The free heterogeneous algebras are then used to generate the universal morphism for any interpretation into a heterogeneous modal algebra thus validating the axioms and rules.

The Frame Conditions FS1, FS2, FB1, and FB2, given the work in Manes [14] on the double power set monad restricted to neighborhoods, show that the neighborhood maps are the Kleisli morphisms and hence form a category, so the identity and associative laws of categories are met. In the presence of the normal axioms, the previous prescription for manufacturing relations from neighborhood maps shows these frame conditions ensure the maps act like Kleisli morphisms for the power set monad restricted to neighborhoods.

The rest of the axioms and rules are easily checked. ■

The *canonical frame* is generated by the LT algebra; the frame's neighborhoods are the output of representation function for the LT algebra. The representation function β is defined by

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

Let $\text{MA}(h), \text{MA}(k)$ stand for the local modal algebras and $\text{CF}(h), \text{CF}(k)$ stand for the canonical frames at h and k respectively. To get a frame category from the LT modal algebra requires that one take the (dual) Stone space containing all the maximal filters of each local algebra and define the local neighborhood maps with:

$$\beta a \in \mathcal{H}x \text{ iff } [h]a \in x.$$

Since $[h]$ and $\langle h \rangle$ are DeMorgan duals of each other and β is a homomorphism,

$$-\beta a \notin \mathcal{H}x \text{ iff } \beta -a \notin \mathcal{H}x \text{ iff } [h]-a \notin x \text{ iff } \neg[h]-a \in x \text{ iff } \langle h \rangle \in x.$$

These same definitions work for the canonical relation $\mathcal{R}$ for $r : h \curvearrowright k$ where now $a \in \text{MA}(k)$, $[r]a, \langle r \rangle a \in \text{MA}(h)$, $x \in \text{CF}(h)$, and $\mathcal{R}x \subseteq \mathbb{K}$ for $\mathbb{K}$ the neighborhoods of $\text{CF}(k)$.

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 set complement interpret the classical logic connectives $\vee$, $\wedge$, and $\neg$. The only question is the status of $\langle r \rangle, [r]$ for $r : h \curvearrowright k$.

Lemma 3.2.7 For $a \in \text{MA}(k)$ and $\langle r \rangle a \in \text{MA}(h)$,

$$\beta[r]a = [r]\beta a \text{ and } \beta\langle r \rangle a = \langle r \rangle \beta a.$$

Proof: $x \in \beta[r]a$ iff $[r]a \in x$ iff $\beta a \in \mathcal{R}x$ iff $x \in [r]\beta a$. The proof for $\langle r \rangle$ is similar. ■

The modal completeness argument is the usual algebraic argument [11] using contraposition and the frame argument uses the canonical frame derived from a representation theorem [3, 11]. The modal representation theorem represents a modal algebra as an algebra of sets using the canonical frame (Stone space) of the algebra. One defines the 1-1 homomorphism β on the distributed algebra for each carrier set and the operations using the above prescriptions.

Theorem 3.2.8 *Distributed Logic is complete with respect to the distributed algebras and the distributed category models.*

Proof: From Proposition 3.1.3, we need only concern ourselves with formula (instances) which sit entirely within a single local logic. So one presents the formula instance at issue and then picks the local logic for which it must be determined whether it is a theorem. The argument is a contraposition argument using the LT heterogeneous algebra and its canonical frame category.

Note that any theorem without an implication as the main connective can be outfitted with one because $\vdash P$ iff $\vdash T \supset P$ where T is the truth constant in a local logic. Hence we need only check implications. Suppose $\not\vdash P \supset Q$, then $\llbracket P \rrbracket \not\leq \llbracket Q \rrbracket$ in the LT algebra where $[P], [Q]$ are the bi-implicational equivalence classes. This along with Corollary 3.2.5 is enough for algebraic completeness.

For frame completeness, there is maximal separating filter x such that $\llbracket P \rrbracket \in x$ and $\llbracket Q \rrbracket \notin x$, i.e., $x \in \beta\llbracket P \rrbracket$ and $x \notin \beta\llbracket Q \rrbracket$, so $x \models P$ and $x \not\models Q$. Therefore there is a local model falsifying the non-theorem, and hence a distributed category model falsifying the non-theorem.

Taking the contrapositive in the algebraic and frame cases yields the required result. ■

3.3 The Axiom Schemes E

The axiom E1 requires some special treatment. The algebra will now have a collection of constant operators, one for each propositional atom in the language.

Definition 3.3.1 *An E local modal algebra is a local modal algebra with a collection of (local) constant operations. Note that two constant operations, being functions, can point to the same element of the local modal algebra. The Lindenbaum-Tarski E local modal algebra has each constant operation pointing out the equivalence class of the propositional atom to which it attached. In symbols, if p is a propositional atom, then its constant, nullary operation, σ_p , is such that $\sigma_p = p$ in the word algebra of the logic and $\sigma_p = \llbracket p \rrbracket$ in the LT algebra. In addition to any axioms necessary for the local modal logic, we add the axiom*

$$\sigma_p \leq [r]\sigma_p$$

for an arc r in the diagram to another node. This effectively forces $\llbracket p \rrbracket \leq [r]\llbracket p \rrbracket$ for any interpretation $\llbracket - \rrbracket$. We also require the logic at $\text{cod}(r)$ to contain at least the same propositional atoms as those at $\text{dom}(r)$.

Definition 3.3.2 *A E neighborhood frame is a neighborhood frame with a collection of constant functions, f_p , one for each propositional atom. A constant function selects an element of the set algebra, i.e., a neighborhood.*

Fix a distributed algebra with any necessary E local modal algebra. Modal valuations vary over what gets assigned to the propositional atoms. Here, the valuations must be consistent with the nullary operations associated with each atom. We get the variation necessary for valuations by choosing different algebras which agree on everything except the nullary operations. So the variation gets satisfied at a slightly higher level. A similar statement holds for E neighborhood frames. The inductive definition generating interpretations from valuations remains the same and hence the restriction on valuations gets transferred to interpretations.

Definition 3.3.3 An $\mathbf{E}$ local algebra valuation, $\llbracket - \rrbracket$, must take every propositional atom to an element of the carrier set pointed to by the nullary operation for that atom, i.e., if $\sigma_p = a$, then $\llbracket p \rrbracket = a$. Similarly, for a $\mathbf{E}$ local neighborhood frame and valuations $\llbracket - \rrbracket$, we demand $\llbracket p \rrbracket = C$ if $f_p = C$. Also, we demand that for $r : h \curvearrowright k$, the r interpreting relation $\mathcal{R}$ must respect the constant functions in the sense that $x \in f_p$ at the neighborhood frame for h and $f_p \in \mathcal{R}x$ at the neighborhood frame for k .

For the LT algebra, $\sigma_p = p$ in the word algebra forces $\sigma_p = \llbracket p \rrbracket$ in the LT algebra. The result is that we get the same LT algebra as we would have without the nullary constants. The universal property of the free algebra with respect to unique maps to the other $\mathbf{E}$ local modal algebras are unaffected since the restriction on interpretations will force the unique maps to choose the same elements of the algebras to which the nullary operations point for the respective propositional atoms. In the freeness diagram below, p indicates some propositional atom in the language, FA_h is the carrier set of the local modal logic for h inside of the free algebra $\mathcal{A}$. The algebra $\mathcal{B}$ is some other appropriate distributed algebra, and γ is an induced interpretation from the freeness property of $\mathcal{A}$,

$$\begin{array}{ccc} SL(h, k \in \mathfrak{G}) & \xrightarrow{\eta} & \mathcal{A}(FA_h, FA_k \in \{S_i\}, \sigma_p^{FA_h}, \sigma_p^{FA_k} \in Ops_{\mathcal{A}}) \\ & \searrow \gamma & \downarrow g \\ & & \mathcal{B}(B_h, B_k \in \{T_i\}, \sigma_p^{B_h}, \sigma_p^{B_k} \in Ops_{\mathcal{B}}) \end{array}$$

The algebra $\mathcal{B}$ has no notion of propositional atoms. The σ_p , being operations, are preserved by g . Hence, $\eta(p) = \sigma_p^{FA_h}$ and $g(\eta(p)) = g(\sigma_p^{FA_h}) = \sigma_p^{B_h}$. Since the diagram commutes, $\gamma(p) = \sigma_p^{B_h}$.

The extension to distributed algebras and distributed category models are called $\mathbf{E}$ distributed algebras and $\mathbf{E}$ distributed category models.

Theorem 3.3.4 Distributed logics with the $\mathbf{E}$ axioms are sound and complete with respect to $\mathbf{E}$ distributed algebras and $\mathbf{E}$ distributed category models.

4 Conclusions and Future Work

Distributed logic is best viewed as a logical toolbox that contains many different logics which are configured by axioms. One selects a graph structure for the local logics and then axioms and rules based upon a particular application. Many of the common modal axioms can be altered to fit distributed modal connectives. The simulation axiom shows this. As a further example, consider the Euclidean axiom (in a normal modal logic) $\langle h \rangle P \supset [h] \langle h \rangle P$ and its validating condition $\mathcal{H}xy$ and $\mathcal{H}xz$ implies $\mathcal{H}yz$. In distributed form for $r : h \curvearrowright k$ in Figure 2, this becomes $\langle r \rangle P \supset [r] \langle k \rangle P$ and the condition becomes $\mathcal{R}xy$ and $\mathcal{R}xz$ implies $\mathcal{K}yz$.

The situation in Figure 2 models a real situation. The relation $\mathcal{R}$ between domain h and k is an artefact of the model and as such, deserves to be represented in a logic over the model. This is the sense in which distributed logic could be considered a model theoretic logic [5]. One must make choices up front before parts of the toolbox come together for a logic; the choices are made because models of a particular kind are needed for an application. A good source of applications which require distributed reasoning are the security guarantees necessary for system-on-a-chip architectures. In on-going and future work, we are expanding the use of distributed logics to provide a programming logic for a hardware specification language called ReWire [15].

More philosophically speaking, modal logics come with a model theory which includes morphisms between models. The logic is abstracted over the model theory giving valid axioms and rules for reasoning about the

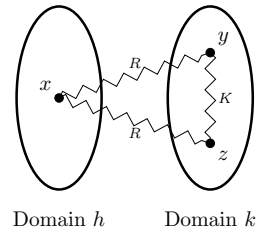


Figure 2

models. Since morphisms are used in the model theory to describe critical aspects of the model, the obvious question is why are these aspects not formalized the logics. The work in this paper (and its predecessor [4]) represents the first steps in this direction.

Part of the problem with including morphisms in a logic is deciding which morphisms should be included and how are they structured. Category theory presents us with the theory of morphisms. Considering modal logic, one could have started with p-morphisms. The approach we have taken is to generalize the notion of what should be considered a model theoretic morphism and then use logical axioms to give the morphisms the properties desired. In effect, we are choosing logical morphisms that preserve only some structure, not all structure unless that is what is desired. The axiom system is then used as an array of control switches to configure distributed logics. In addition, the morphisms can be fine tuned between some local logics but not imposed between all local logics within a distributed logic. This accords well with the notion that distributed logics should be useful for representing reasoning about distributed systems where there is much variation and nuance that must be represented formally.

Space prevents us from also covering two-place intensional connectives such as relevance logic's entailment. That too has a pleasant reconstruction in distributed logic, although the three place relations require an extended notion of categorical morphism. Distributed logic was originally formulated with relations. Consideration of testing for foreign IP in system-on-a-chip designs forced the use of neighborhood systems. The ease of modification of distributed logic forced by two place intensional connectives and weak modal connectives requiring a neighborhood semantics is part of a larger theme for distributed logic: many model theoretic notions are "orthogonal" to distribution in that they do not seem to cause any significant hurdles to their re-expression in a distributed system. Some model theoretic notions, such as morphism, are inherently distributed. Some, such as, Kripke relations, can be re-expressed as distributed notions. The bounds of what is possible seems to be related to the question of what is modality.

References

- [1] Adámek, J. and J. Rosický, "Locally Presentable and Accessible Categories," London Mathematical Society, 1994, Lecture Note Series 189.
- [2] Allwein, G., *A qualitative framework for Shannon information theories*, in: *Proceedings of the New Security Paradigms Workshop, 2004* (2005), pp. 23 – 31.
- [3] Allwein, G. and J. Dunn, *Kripke models for linear logic*, Journal of Symbolic Logic **58** (1993), pp. 514–545.
- [4] Allwein, G., W. Harrison and D. Andrews, *Simulation logic*, Logic and Logical Philosophy **23** (2014), pp. 277–299.
- [5] Barwise, J. and S. Feferman, editors, "Model-Theoretic Logics," Springer-Verlag, 1985.
- [6] Barwise, J. and J. Seligman, "Information Flow: The Logic of Distributed Systems," CUP, 1997, Cambridge Tracts in Theoretical Computer Science 44.
- [7] Benthem, J. V., *Information transfer across Chu spaces*, Logic Journal of the IGPL **8** (2000), pp. 719–731.
- [8] Birkhoff, G. and J. D. Lipson, *Heterogeneous algebras*, Journal of Computational Theory **8** (1968), pp. 115–133.
- [9] Blackburn, P., M. de Rijke and Y. Venema, "Modal Logic," Cambridge University Press, 2001, Cambridge Tracts in Theoretical Computer Science, No. 53.
- [10] Chellas, B. F., "Modal Logic: an introduction," Cambridge University Press, 1980.

- [11] Dunn, J. M. and G. Hardegree, “Algebraic Methods in Philosophical Logic,” Oxford Logic Guides 41, Oxford University Press, 2001.
- [12] Fagin, R., J. Y. Halpern, Y. Moses and M. Y. Vardi, “Reasoning About Knowledge,” MIT Press, 1995.
- [13] Goguen, J. A. and R. M. Burstall, *Institutions: Abstract model theory for specification and programming*, CSLI Research Reports **85-30** (1985), pp. 1–73.
- [14] Manes, E. G., “Algebraic Theories,” Springer-Verlag, 1976.
- [15] Procter, A. M., W. L. Harrison, I. Graves, M. Becchi and G. Allwein, *Semantics-directed machine architecture in ReWire.*, in: *FPT* (2013), pp. 446–449.
- [16] Sangiorgi, D., “Introduction to Bisimulation and Coinduction,” Cambridge University Press, 2012.