Naval Research Laboratory
Washington, DC 20375-5320

NRL/5540/MR—2023/1

Basic Measurement for One Connective of Distributed Logic

DR. GERARD ALLWEIN
CHRISTOPHER BELMONTE

Center for High Assurance Computer Systems Branch
Information Technology Division

August 17, 2023

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
17-08-2023

2. REPORT TYPE
NRL Memorandum Report

3. DATES COVERED *(From - To)*
10-01-2023 – 09-30-2023

4. TITLE AND SUBTITLE
Basic Measurement for One Connective of Distributed Logic

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
062235N

6. AUTHOR(S)
Dr. Gerard Allwein and Christopher Belmonte

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER
6C59

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Research Laboratory
4555 Overlook Avenue, SW
Washington, DC 20375-5320

8. PERFORMING ORGANIZATION REPORT NUMBER
NRL/5540/MR--2023/1

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Naval Research Laboratory
4555 Overlook Avenue, SW
Washington, DC 20375-5320

10. SPONSOR / MONITOR'S ACRONYM(S)
NRL

11. SPONSOR / MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A: Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This is a report on measuring the distributed possibility connective of Distributed Logic for FPGA applications. The measurement apparatus is particularly well suited for measuring how well a spider is tracking an FPGA application. The mathematics works with regular conditional probability distributions also known as Markov kernels; these are measurable relations in one variable and subprobability measures in another. They satisfy a certain equation and are actually arrows in a category of sigma-lattices as objects, SRel. A subprobability measure is one where the requirement is only that the value of the underlying ambient space less than or equal to 1, not precisely equal to 1. Requiring equality to 1 turns out to be too restrictive. All categories have composition of arrows as one of their basic laws. SRel has a specific prescription for how to compose relations and this turns out to be one of the ingredients required to measure simulation conditions.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			Dr. Gerard Allwein
U	U	U	U	37	**19b. TELEPHONE NUMBER** *(include area code)* (202) 404-3748

This page intentionally left blank.

CONTENTS

This page intentionally left blank

EXECUTIVE SUMMARY

This is a report on measuring the distributed possibility connective of Distributed Logic for FPGA applications. The measurement apparatus is particularly well suited for measuring how well a *spider* is tracking an FPGA application. The mental picture is that the body of a spider sits over an application and its legs stick down into the application. Signals travel up and down the legs. The signals to the spider are *recognition* signals and report on conditions in the application. Signals from the spider are *mitigation* signals and direct the application to perform behavior outside of its normal operation. The measurement in this paper is security properties specified using a *simulation condition*. A simulation from A to B says component A is tracking component B if every move component A can make is echoed by a move that component B can make. One can always reverse the direction by specifying a new simulation from B to A.

The mathematics works with *regular conditional probability distributions* also known as *Markov kernels*; these are measurable relations in one variable and subprobability measures in another. They satisfy a certain equation and are actually arrows in a category of σ-lattices as objects, **SRel**. A subprobability measure is one where the requirement is only that the value of the underlying ambient space less than or equal to 1, not precisely equal to 1. Requiring equality to 1 turns out to be too restrictive. All categories have composition of arrows as one of their basic laws. **SRel** has a specific prescription for how to compose relations and this turns out to be one of the ingredients required to measure simulation conditions.

Distributed Logic was developed by the PI expressly for distributed systems such as FPGA applications. An FPGA application is similar to a circuit board shrunk down to fit on a chip. Like a circuit board, an application is a collection of components all operating in parallel and exchanging signals. The usual logics tend to get a bit out of hand in such an environment because they have no facilities for directly representing this distributed structure. Instead, they rely upon layers of encoding, which makes them clumsy and prone to misuse. Distributed Logic incorporates a graph of nodes and arcs directly as part of the logic; no other logic does this. The graph allows statements in Distributed Logic to mirror the structure of the application. Each component gets its own local logic and distributed connectives connect the local logics. Each local logic is a generally a modal logic so that it can state properties of a component's state machine. The distributed connectives are also modal connectives but take formulas in one local logic and construct formulas in another local logic.

This notion of distribution is much wider than FPGA applications. We have shown in other research how to distribute relation algebras and other logics. In general, this distribution appears to be *perpendicular* to many logic constructs and hence it has a wide applicability.

This page intentionally left blank

BASIC MEASUREMENT FOR ONE CONNECTIVE OF DISTRIBUTED LOGIC

1. INTRODUCTION

Simulation relations tie actions of system to actions of another system. The entities of a system and its actions are not prescribed by the definition of a simulation relation. However, with respect to any particular system, the entities and actions are concretely specified. In our case, we are using finite state machines as the systems and the simulation relation relates states in one machine with states in another. Whenever possible, we will state the mathematics in its most general form. Keeping finite state machines in the back of the reader's mind will help to orient the reader as a ready example.

Simulation relations can be arrows in a category, we shall not use them as such. Rather, we treat relations in general as arrows in a **Set**-based category **Rel**. Relations are used to interpret Distributed Logic. A simulation satisfied by a category of **Set**-based relations, in its most general form, involves four relations; a simulation from A to B involves a relation in A and a relation in B. The simulation relation is actually two relations. For many purposes, these two relations can be made identical.

In interpreting Distributed Logic (DL), the truth semantics yields *true* or *false* as values. These semantics are the *qualitative semantics* for DL. The mathematics of measurement works with *regular conditional probability distributions* also known as *Markov kernels*; these are measurable relations in one variable and subprobability measures in another. They satisfy a certain equation and are actually arrows in a category of σ-lattices as objects, **SRel**. A subprobability measure is one where the requirement is only that the value of the underlying ambient space less than or equal to 1, not precisely equal to 1. Requiring equality to 1 turns out to be too restrictive. All categories have composition of arrows as one of their basic laws. **SRel** has a specific prescription for how to compose relations and this turns out to be one of the ingredients required to interpret possibility connectives. In effect, we will be giving a *quantitative semantics* to DL.

The measurement in this paper is of security properties specified using a *simulation condition*. A simulation from A to B says component A is tracking component B if every move component A can make is echoed by a move that component B can make. One can always reverse the direction by specifying a new simulation from B to A.

The mathematics works with *regular conditional probability distributions* also known as *Markov kernels*; these are measurable relations in one variable and subprobability measures in another. They satisfy a certain equation and are actually arrows in a category of σ-lattices as objects, **SRel**. A subprobability measure is one where the requirement is only that the value of the underlying ambient space less than or equal to 1, not precisely equal to 1. Requiring equality to 1 turns out to be too restrictive. All categories have composition of arrows as one of their basic laws. **SRel** has a specific prescription for how to compose relations and this turns out to be one of the ingredients required to measure simulation conditions.

Distributed Logic was developed by the PI expressly for distributed systems such as FPGA applications. An FPGA application is similar to a circuit board shrunk down to fit on a chip. Like a circuit board, an

---

Manuscript approved August 15, 2023.

application is a collection of components all operating in parallel and exchanging signals. The usual logics tend to get a bit out of hand in such an environment because they have no facilities for directly representing this distributed structure. Instead, they rely upon layers of encoding, which makes them clumsy and prone to misuse. Distributed Logic incorporates a graph of nodes and arcs directly as part of the logic; no other logic does this. The graph allows statements in Distributed Logic to mirror the structure of the application. Each component gets its own local logic and distributed connectives connect the local logics. Each local logic is a generally a modal logic so that it can state properties of a component's state machine. The distributed connectives are also modal connectives but take formulas in one local logic and construct formulas in another local logic.

This notion of distribution is much wider than FPGA applications. We have shown in other research how to distribute relation algebras and other logics. In general, this distribution appears to be *perpendicular* to many logic constructs and hence it has a wide applicability.

2. TECHNICAL IDEA

We will be using forward simulations. The main idea in this paper is to measure DL's possibility connectives that arise in simulation axioms, both the local and distributed connectives. Any of the simulation axioms that can be added to DL have iterated modal connectives. The iteration should be seen as composition in a category. In DL, the composition is at both the level of the logic in the form of the connectives and in the semantic level in the form of composing the relations.

We can re-express any of the simulation axioms using iterated possibilities. The reason to to use possibility connectives is that they are easy in which to apply notions of measurement. The composition will take place at the semantic semantic level in a category **SRel**, for *stochastic relations*. Stochastic relations are the type seen in Markov chains. Essentially this means putting probabilities on the individual pairs (arcs) of the relation. The main reason to use a category is that the distributed stochastic relations are binary relations between objects that in our case will be Stone spaces. This means that there will be two measurable structures involved with these relations and consequently two probability distributions.

The work in [1, 2] uses Polish and analytic spaces. For us, we will be using finite Stone spaces which are always Polish and analytic. However, we will develop the mathematics ignoring this fact only requiring measurable spaces.

2.1 Forward Simulation

$$\mathcal{F}xy \text{ and } \mathcal{H}xx' \text{ implies } \exists y'(\mathcal{K}yy' \text{ and } \mathcal{G}x'y')$$

has the diagram

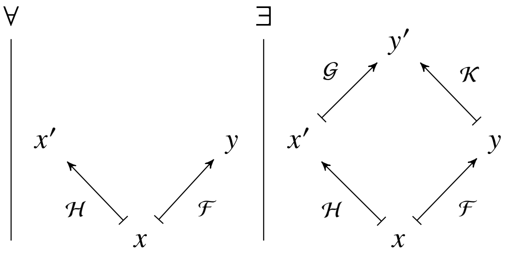

Figure 2.1: Forward Simulation

This is not a category theory diagram but rather the arrows showing pairs of their respective relations.

This condition validates the forward simulation axiom.

$$[f^{\diamond}\rangle\,[k^{\square}\rangle Q \supset [h^{\square}\rangle\,[g^{\diamond}\rangle Q$$

Lemma 2.1.1 *Let* $f : \mathsf{h} \rightharpoonup \mathsf{k}$ *in the logic's distribution graph, then the forward simulation axiom*

$$[f^{\diamond}\rangle\,[k^{\square}\rangle Q \overset{\mathsf{h}}{\supset} [h^{\square}\rangle\,[g^{\diamond}\rangle Q$$

is equivalent to the following two alternate forms:

$$[h^{\diamond}\cdot\rangle\,[f^{\diamond}\rangle\,S \overset{\mathsf{h}}{\supset} [g^{\diamond}\rangle\,[k^{\diamond}\cdot\rangle S \qquad [f^{\diamond}\cdot\rangle\,[h^{\diamond}\rangle\,P \overset{\mathsf{k}}{\supset} [k^{\diamond}\rangle\,[g^{\diamond}\cdot\rangle P$$

All the forms are interderivable.

Note that in the second equivalent formula, the forward simulation axiom and this equivalent are in different localities. Recall for $f : \mathsf{h} \rightharpoonup \mathsf{k}$, that $[f^{\diamond}\rangle$ and $[g^{\diamond}\rangle$ pull formulas back along f and g respectively while $[f^{\diamond}\cdot\rangle$ and $[g^{\diamond}\cdot\rangle$ push formulas forward along f and g respectively.

The equivalent validating conditions (respectively) are

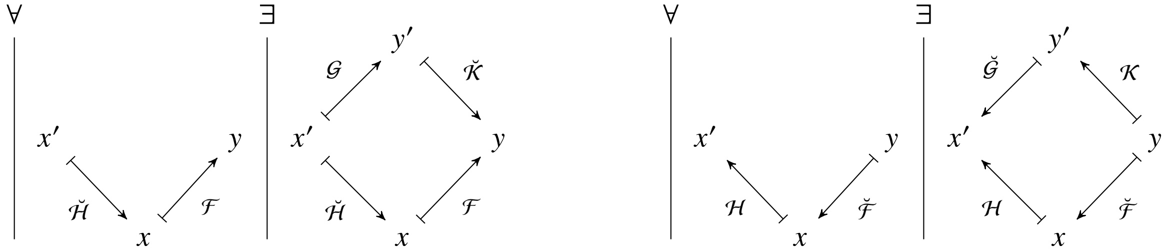

Figure 2.2: Equivalent Forward Simulation Conditions

where $\breve{\mathcal{H}}$ and $\breve{\mathcal{K}}$ the converse of $\mathcal{H}$ and $\mathcal{K}$.

Note the reversal of some of the arrows from the original in Figure 2.1. An easy rule of thumb is that the connectives pullback propositions along relations. For example $[f^{\Diamond}\rangle\, S$ is pulling S back along $\mathcal{F}$ and $[k^{\Box}\rangle\, S$ is pulling S back along $\mathcal{K}$. The connectives $[-^{\Diamond}\rangle$ and $[-^{\Box}\rangle$ pullback in different ways.

Proof: The proofs requires the use of residuation rules and these are valid in the logic, for any h, k, $f : \mathsf{h} \rightharpoonup \mathsf{k}$, $P \in \mathsf{h}$ and $Q \in \mathsf{k}$:

$$\frac{P \overset{\mathsf{h}}{\supset} [f^{\Box}\rangle\, Q}{\overline{[f^{\Diamond}\cdot\rangle\, P \overset{\mathsf{k}}{\supset} Q}} \qquad \frac{Q \overset{\mathsf{k}}{\supset} [f^{\Box}\cdot\rangle\, P}{\overline{[f^{\Diamond}\rangle\, Q \overset{\mathsf{h}}{\supset} P}}$$

where the double lines means the rules are bidirectional. The rules say that if the premise is provable, then the conclusion is provable (in both directions). ■

The axioms can be put in 2-diagrammatic form where the single arrows are relations, composition of arrows is relational composition, and the double arrow in the middle is interpreted as $\subseteq$. Binary relations and Stone spaces constitute a category. Categorical composition uses the symbol $\circ$ and is defined in terms of relational composition; the latter uses the symbol $\cdot$. Composition is defined for $\mathcal{R} : X \rightharpoonup Y$ and $\mathcal{S} : Y \rightharpoonup Z$ with

$$\begin{aligned} (\mathcal{S} \circ \mathcal{R})xz \text{ iff } & (\mathcal{R} \cdot \mathcal{S})xz \\ \text{iff } & x(\mathcal{R} \cdot \mathcal{S})z \\ \text{iff } & \exists y \in Y(\mathcal{R}xy \text{ and } \mathcal{S}yz). \end{aligned}$$

where in this instance we use the alternate depiction of a relational pair in infix for the second iff. As an example, the diagrammatic form of one form of forward simulation is then

$$\begin{tikzcd}
 & K \arrow[dr, "\breve{\mathcal{K}}"] & \\
H \arrow[ur, "\mathcal{G}"] \arrow[dr, "\breve{\mathcal{H}}"'] & \Uparrow & K \\
 & H \arrow[ur, "\mathcal{F}"'] &
\end{tikzcd}$$

Figure 2.3: Forward Simulation

which says that

$$\mathcal{F} \circ \breve{\mathcal{H}} = \breve{\mathcal{K}} \circ \mathcal{G}.$$

In relational composition, this matches with the forward axiom, i.e.,

$$\breve{\mathcal{H}} \cdot \mathcal{F} = \mathcal{G} \cdot \breve{\mathcal{K}} \qquad [h^{\Diamond}\cdot\rangle\, [f^{\Diamond}\rangle\, Q \subseteq [g^{\Diamond}\rangle\, [k^{\Diamond}\cdot\rangle Q.$$

3. PROBABILITIES

In [4, 5] (the latter seems to include the former and the former appears unpublished), there some useful formulas. He references Giry [6]. He assumes (X, Σ_X) and (Y, Σ_Y) to be σ-fields of sets, and that X has a probability measure P with $\mathcal{F} : X \twoheadrightarrow Y$ a measurable function. We will think of this as a measurable relation, and use the lower case f. Also, if f is a measurable relation, f^{-1} still makes sense.

3.1 Probability Spaces

The sets in Σ are the UCLA propositions from which we derive a Kripke frame and these propositions comprise a Boolean set algebra. In $(H, S, \mathcal{H}, BA(\mathsf{h}))$, $BA(\mathsf{h})$ is promoted from being a Boolean algebra of sets to being a σ-lattice of sets. Every Boolean algebra can be extended to a σ-algebra.

We assume all our measures are probability measures. This does beg the question of whether there are other useful measures for FPGA applications but we will not explore that topic here.

Lambda notation is used in Panangaden, the formula

$$f(x) = x^2$$

written in lambda notation is

$$f = \lambda x \; . \; x^2.$$

This frees the name of the function from its argument. $f(x)$ is a bit odd; is this f applied to a known value x or is it x merely a variable? The lambda notation allows us to specify functions without necessarily naming them, e.g.,

$$\lambda x \; . \; x^2.$$

Applying a function to an argument is displayed as concatenation. To apply this function so some value a is

$$(\lambda x \; . \; x^2)a = a^2.$$

At various places, the operator sup is used. This is $\bigvee$ (large join) in the real numbers under $\leq$ as the order as a linearly ordered, dense set. The sup operators are over values in the real numbers $\mathbb{R}$ and in all places and are analogous to limits. They are generally parameterized in that sup will be some sup of a function of elements of T, i.e., $f : T \twoheadrightarrow \mathbb{R}$ and

$$\sup_{x \in T}\{f(x) \mid \phi(x)\}$$

where elements T are the indices, ϕ is some condition that we use to filter the elements of T and only chose those that satisfy ϕ. Notice that sup of the values come from $f(x) \in \mathbb{R}$.

Definition 3.1.1 A *probability space* $(X, \Sigma_X, \mathsf{P})$ such that

- X is a non-empty set,
- the σ-algebra $\Sigma_X \subseteq \mathcal{P}X$ is a set of subsets of X (typically called events) such that:
  - $X, \emptyset \in \Sigma_X$,
  - Σ_X is closed under complements, countable unions, and countable intersections.
- the probability measure $\mathsf{P} : \Sigma_X \to [0,1]$ satisfies
  - P is countably additive: if $\{A_i\}_{i=1}^{\infty} \subseteq \Sigma_X$ is a countable collection of pairwise disjoint sets, then

    $$\mathsf{P}(\bigcup_{i=1}^{\infty} A_i) = \sum_{i=1}^{\infty} \mathsf{P}(A_i),$$

  - $\mathsf{P}(X) = 1$ and $\mathsf{P}(\emptyset) = 0$.

☐

For us, we will choose generic Stone spaces outfitted with a σ-algebra. Every Stone space generates a σ-algebra, just close up under countable unions, countable intersections, and complements. However, this is usually more than we can expect. Rather, we choose σ-algebras as sublattices of Boolean lattices. The elements of the σ-algebra are the measurable sets. These sets we can think of as those UCLA propositions for which we have total information. They can be used to approximate all the UCLA propositions.

Definition 3.1.2 Let Σ_X be the σ-algebra of a measurable space (X, Σ_X). A function $f : (X, \Sigma_X) \to (Y, \Sigma_Y)$ is measurable when $f^{-1}\Sigma_Y \subseteq \Sigma_X$. ☐

In what follows, we only need be concerned with non-negative functions. Signed functions require a bit more work.

3.2 Simple Functions

Characteristic functions from set theory are known as indicator functions in measure theory; they are used in one way to define Lebesgue integration. Let S be a UCLA proposition, i.e., a set of points, then

$$1_S(x) = \begin{cases} 1 & x \in S \\ 0 & \text{otherwise} \end{cases}$$

We can think of 1 and 0 as truth values or we can think of them as actual real numbers. Probability theory uses the latter interpretation. Assume we have a measure μ on some σ-algebra of propositions, then the following definition is used:

Definition 3.2.1

$$\int_x 1_S \, d\mu(x) \overset{def}{=} \mu(S).$$

☐

The function $\int$ in Lebesgue integration is interpreted differently than in Riemann integration. The value of 1_S is not being used as a numerical value summed in the integration but rather as function to determine a set to be measured. This is one of the differences between Lebesgue integration and Riemann integration, where in the latter, the numerical value of the function being integrated is used as a value in the sum in the sense that

$$\int 1_S \, dx \overset{def}{=} \sup_{x \in [x_0, x_1]} \{1_S(x) * (x_1 - x_0) \mid x_0 \leq x \leq x_1\} = (x_1 - x_0)$$

where $S = \{x \mid x_0 \leq x \leq x_1\}$ and $x_0 < x_1$. There is no metric space underlying Lebesgue integration. The Lebesgue integration could be written more pedantically as

$$\int_x 1_S \, d\mu(x) = \mu\{x \mid 1_S(x) = 1\} = \mu(S).$$

Let $\mathbb{I}$ be the collection of indicator functions and $\mathcal{M}$ be the collection of measures (not necessarily probability measures), then it is best to view $\int$ as a function of type

$$\int : \mathbb{I} \times \mathcal{M} \to [0, \infty).$$

That is, $\int$ needs both an indicator function and a measure before it returns a value.

Definition 3.2.2 (Simple Functions) A *simple* non-negative function $s : \Sigma_X \to [0, \infty)$ is defined as finite linear combination of indicator functions of pairwise disjoint sets $\{S_k \mid 0 \leq k < n\}$ for some n with non-negative coefficients a_k such that

$$s(x) = \sum_k a_k 1_{S_k}(x).$$

□

We will always pair the lower case function let s with the upper case set S. So in effect we have

$$s(x) = \begin{cases} a_k & x \in S_k \\ 0 & \text{otherwise} \end{cases}$$

Let $S = S_0 \cup S_1$, and picture the function s (representing S) with

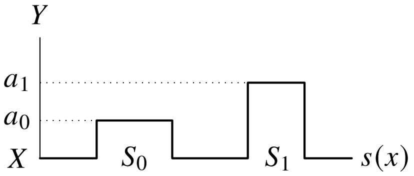

Figure 3.1: Simple Function s

3.3 Primer on Lebesgue Integration for Probability Measures

In the sequel we will need to evaluate the conditions of the form

$$0 \leq s \leq f.$$

for s a simple function and f a function. This stands for

$$\forall z \in X(0 \leq s(z) \leq f(z)),$$

where s is a simple function. We redefine our simple functions from Section 3. A way to picture the relationship we want is

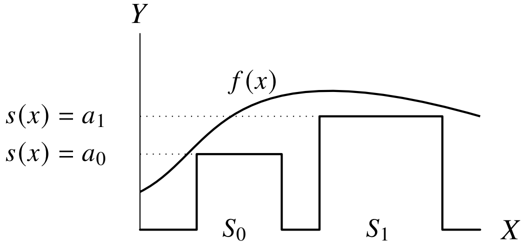

Figure 3.2: Lebesgue Approximation: $0 \leq s \leq f$

where $S_0, S_1 \subseteq X$.

Definition 3.3.1 (Lebesgue Integration For Non-negative Simple Functions) Let S_k, $0 \leq k < n$ the S_k are pairwise disjoint, and that $\mu(S_k) < \infty$,

$$\int s \, d\mu = \int \left(\sum_k a_k 1_{S_k} \right) d\mu = \sum_k a_k \int 1_{S_k} \, d\mu = \sum_k a_k \, \mu(S_k).$$

where s is the simple function for the set

$$S = \bigcup_{0 \leq k < n} S_k.$$

If A is a measurable subset of X and $s : \Sigma_X \to [0, \infty)$ is a non-negative, measurable simple function, then

$$\int_A s \, d\mu = \sum_k a_k \, \mu(S_k \cap A).$$

For a domain X, the collection of simple functions is denoted SF_X. In general, we always associate a simple function s_D with the set D. In the case of an unadorned s, we associate it with the set S. $\square$

To be assured of getting probabilities, we restrict the distribution function μ to a probability measure and simple functions:

Definition 3.3.2 Let $\mu : \Sigma_X \rightarrow [0,1]$ satisfy Definition 3.1.1 (with μ for P). A *simple function s* for probability is defined as

$$s(x) \overset{def}{=} \sum_k a_k 1_{S_k}(x), \qquad \forall k(0 \le a_k \le 1).$$

$\square$

The effect is that in evaluating integrals of simple functions, we have

$$\begin{aligned}
\int s\, d\mu &= \int \left(\sum_k a_k 1_{S_k} \right) d\mu && \text{Definition 3.3.1} \\
&\le \sum_k \int 1_{S_k}\, d\mu && \text{let } a_k = 1, \text{ for all } k \\
&= \sum_k \mu(S_k) && \text{Definition 3.2.1} \\
&\le \mu(X) && \textstyle\bigcup_k (S_k) \subseteq X,\ \mu \text{ is monotone} \\
&\le 1 && X \text{ is a probability space.}
\end{aligned}$$

Finally, the stock definition of Lebesgue integration for $f : X \rightarrow [0,1]$ is

$$\int_X f\, d\mu = \sup \left\{ \int_X s\, d\mu \mid 0 \le s \le f,\ s \in SF_X \right\}. \tag{1}$$

To define the integral over a measurable subset $A \subseteq X$ is just to adjust the domain of the integration:

$$\int_A f\, d\mu = \sup \left\{ \int_A s\, d\mu \mid 0 \le s \le f,\ s \in SF_X \right\}. \tag{2}$$

The equations (1) and (2) are a bit odd since the variable of the integration is not mentioned. The sup itself is over the values of the integral, i.e., a sup in $[0,1]$:

$$\int_{x \in X} f(x)\, d\mu = \sup \left\{ \int_{x \in X} s(x)\, d\mu \mid \forall z \in X \big(0 \le s(z) \le f(z) \big),\ s \in SF_X \right\}. \tag{3}$$

and

$$\int_{x \in A} f(x)\, d\mu = \sup \left\{ \int_{x \in A} s(x)\, d\mu \mid \forall z \in A \big(0 \le s(z) \le f(z) \big),\ s \in SF_X \right\}. \tag{4}$$

The domain of f and the simple functions approximating f are restricted to A under the integral. More to the point, from Lebesgue integration, we have

In restricting integration to a measurable set A, i.e.,

$$\int_A s\, d\mu = \sum_{0\leq k<n} a_k\, \mu(S_k \cap A),$$

notice that the measure μ does not change. μ continues to have the type $\Sigma_X \to [0,1]$ because any for any measurable set D and since A is measurable, $D \cap A$ is also a measurable set. The point is that μ continues to measure sets with respect to X, not with respect to A in the sense that if we were working with a finite X, then

$$\mu Q = |Q|/|X|, \quad \mu(Q \cap A) = |Q \cap A|/|X|.$$

Effectively this means that

$$SF_A \overset{def}{=} \{s_{D\cap A} \mid D \in \Sigma_X,\ s_{D\cap A} \text{ simple}\}$$

From now on, we will always assume there is some ambient collection of simple functions SF_X for domain X appropriate for any sup operator.

The term $s(x)$ is annoying and should equate to

$$s(x) = \sum_k a_{0\leq k<n} 1_{S_k}(x).$$

for some pairwise disjoint collection $\{S_k \mid 0 \leq k < n\}$ for some n. Further

$$\int_{x\in X} s(x) d\mu = \int_{x\in X} \left(\sum_k a_k 1_{S_k}(x) \right) d\mu = \sum_k a_k \int_{x\in X} 1_{S_k}(x)\, d\mu = \sum_k a_k\, \mu(S_k).$$

where on the last identity, x disappears, i.e., it is thrown away. The last identity

$$\sum_k a_k \int_{x\in X} 1_{S_k}(x)\, d\mu = \sum_k a_k \mu(S_k).$$

can be replaced with

$$\begin{aligned}\sum_k a_k \int_{x\in X} 1_{S_k}(x)\, d\mu &= \sum_k a_k * \int_{x\in X} \mu\{x \mid 1_{S_k}(x) = 1\} \\ &= \sum_k a_k \mu(S_k).\end{aligned}$$

where the term after the first = is missing in the usual explication of Lebesgue integration.

The connection between $s(x)$ and collection S_k is obscure. Just to reiterate, $s(x)$ is always associated with an S such that $S = \bigcup_{0 \leq k < n} S_k$.

4. DISTRIBUTING LEBESGUE INTEGRATION

These sections show modifications necessary to use Lebesgue integration in a distributed setting.

4.1 Distributing μ

To handle distributed relations, the measurement function μ must be parameterized. μ itself has a very simple parameterization. It is a measure of the identity relation $\mathcal{I}_X : X \times X \twoheadrightarrow \{0, 1\}$, i.e.,

$$\mathcal{I}_X x y = \begin{cases} 1 & \text{if } x = y \\ 0 & \text{otherwise} \end{cases}$$

As a function, $\mathcal{I}_X : X \twoheadrightarrow \mathcal{P}(X)$ is defined by

$$\mathcal{I}_X x = \{x\}.$$

As a step towards measurement, we can view $\mathcal{I}_X : X \times \mathcal{P}(X) \twoheadrightarrow \{0, 1\}$ as

$$\mathcal{I}_X \, x \, C = \begin{cases} 1 & \text{if } x \in C \\ 0 & \text{otherwise} \end{cases}$$

The term $x \in C$ can be replaced by $\{x\} \cap C \neq \emptyset$. This means that the prescription above is a version of the semantics for $[{}^{i^\diamond}\rangle C$, i.e.,

$$x \in [{}^{i^\diamond}\rangle C \text{ iff } \exists y (\mathcal{I} x y \text{ and } y \in C) \text{ iff } x \in C,$$

which states that $[{}^{i^\diamond}\rangle C = C$.

The probabilistic version restricted to measurable sets is $\iota_X : X \times \Sigma_X \twoheadrightarrow [0, 1]$ is

$$\iota_X \, x \, C = \mathsf{P}_X(C)$$

for P_X the probability of any x actually being in the set C.

The probability distributions we will need in the sequel are functions of the form $g : X \times \Sigma_Y \twoheadrightarrow [0, 1]$. The meaning of $g(x, S)$ is the same as for $g(x, B)$ with B being a general set, i.e., starting at point x, $g(x, S)$ yields the probability of landing in the measurable set S. These distributions will come in the guise of $g(x, dy)$ where dy can be thought of a variable over Σ_Y.

4.2 Changing the Type of dx

We will make a change for Lebesgue integration notation to make sense of Doberkat's notation [1, 2]. From Billingsley [7] on page 211,

> In this section, f, g, and so on will denote real measurable functions, the values $\pm\infty$ allowed, on a measure space $(\Omega, \mathscr{F}, \mu)$. The object is to define and study the definite integral
>
> $$\int f\, d\mu = \int_\Omega f(\omega)\, d\mu(\omega) = \int_\Omega f(\omega)\, \mu(d\omega). \tag{5}$$

Here, Billingsley is using the general measure symbol μ in place of the probabilistic measure symbol P. The equivalent formula to (5) in our notation replaces Ω with X, $\mathscr{F}$ with Σ_X

$$\int_{x\in X} f(x)\, d\mu = \int_X f(x)\, d\mu(x) = \int_X f(x)\, \mu(dx).$$

The Equations (3) and (5) do not explicitly say that $d\mu$ only makes sense with respect to an unstated (at that point) collection of simple functions. This can been seen in use of the sup in Eqation (3) where the set of simple functions, SF appears out of nowhere. The functional dx is missing an argument. We wish to turn d into a function that returns a measurable set so that $\mu(dx)$ makes sense. Assume s is a simple function taking non-zero values on elements of S_k, then for $1_S(x) = 1$ iff $x \in S$ and $S = \bigcup_{0\leq k<n} S_k$ where the S_k are simple pairwise disjoint sets,

$$d_{s_k}(x) \overset{def}{=} S_k \qquad \text{such that } x \in S_k.$$

Consequently,

$$\int s_k(x)\mu(d_{s_k}x) = a_k\, \mu(S_k).$$

We then have

$$\begin{aligned}\int s(x)\mu(d_s x) &= \sum_k \int s_k(x)\mu(d_{s_k}x)\\ &= \sum_k a_k\, \mu S_k\end{aligned}$$

like we had before except now the differentials d_s have been properly parameterized. We make d an explicit functional.

Definition 4.2.1 The collection of simple functions over X be denoted SF_X. □

The type of d is now

$$d : X \times SF_X \to \Sigma_X.$$

We define

$$d_s x \overset{def}{=} (\lambda x \; . \; \lambda s \; . \; .d\,(x)\,(s))$$

That is, the environment in which d finds itself will supply the x and s. We can now expand how Equation (3) works:

Lemma 4.2.2

$$\int_{x \in X} f(x)\, d\mu = \sup \left\{ \sum_k a_k\, \mu(S_k) \mid \forall z \in X \Big(0 \le s(z) \le f(z)\Big),\ s \in SF_X \right\}.$$

This just repeats Equation (3) but the proof is different.

Proof:

$$\begin{aligned}
\int_{x \in X} f(x)\, d\mu &= \int_{x \in X} f(x)\, \mu(d(x)) \\
&= \sup \left\{ \int_{x \in X} s(x)\; \mu(d\,x\,s) \mid \forall z \in X \Big(0 \le s(z) \le f(z)\Big),\ s \in SF_X \right\} \\
&= \sup \left\{ \int_{x \in X} s(x)\; \mu(d_s(x)) \mid \forall z \in X \Big(0 \le s(z) \le f(z)\Big),\ s \in SF_X \right\} \\
&= \sup \left\{ \int_{x \in X} 1_S(x)\; \mu(d_s x) \mid \forall z \in X \Big(0 \le s(z) \le f(z)\Big),\ s \in SF_X \right\} \\
&= \sup \left\{ \sum_k \int_{x \in X} 1_{S_k(x)}\; \mu(d_{s_k} x) \mid \forall z \in X \Big(0 \le s(z) \le f(z)\Big),\ s \in SF_X \right\} \\
&= \sup \left\{ \sum_k a_k\, \mu(S_k) \mid \forall z \in X \Big(0 \le s(z) \le f(z)\Big),\ s \in SF_X \right\}.
\end{aligned}$$

■

With this change in notation, we can now make sense of the following from [2]. A subprobability measure is a probability measure that does not necessarily sum to 1 for a collection of disjoint sets that entirely cover the domain. Let $\mathfrak{G}(X, \Sigma_X)$ be the collection of all subprobability measures on (X, σ_X). For any X-Y measurable map $f : X \to Y$, $\mathfrak{G}(f) : \mathfrak{G}(X, \Sigma_X) \to \mathfrak{G}(Y, \Sigma_Y)$ is defined, for $B \in \Sigma_Y$,

$$\mathfrak{G}(f)(\mu)(B) \overset{def}{=} \mu(f^{-1}(B)) \tag{6}$$

where it is left to the reader to determine that $\mu : X \twoheadrightarrow [0,1]$ is a probability measure or probability distribution on X. With this understanding, $\mathfrak{G}(f)$ is a probability measure on X.

The left hand side is used in the context of an integration for $f : X \twoheadrightarrow Y$ and $g : Y \twoheadrightarrow Z$ stochastic relations yielding

$$\int_Y g(y)\mathfrak{G}(f)(\mu)(dy)$$

where the integration context will supply the s required to produce $d_s(y)$. The substitution of dy for B for the left side of Equation (6) makes no sense without properly typing the functional d.

5. MEASURING DISTRIBUTED SIMULATIONS

5.1 The Basic Machinery

Definition 5.1.1 Assume that X has a probability measure P defined on it and that $f : X \twoheadrightarrow Y$ is a measurable function. A *regular conditional probability distribution* is a function $h : X \times \Sigma_Y \twoheadrightarrow [0,1]$ such that

(i) for each fixed $B \in \Sigma_Y$ the function $h(\cdot, B)$ is measurable

(ii) for each $x \in X$, $h(x, \cdot)$ defines a probability measure

(iii) and the following equation is satisfied:

$$\forall A \in \Sigma_X, \forall B \in \Sigma_Y \; . \; \int_A h(x,B)d\mathsf{P}(x) = \mathsf{P}(f^{-1}(B) \cap A). \tag{7}$$

□

Panangaden remarks

> Thus we can think of h as the conditional probability which gives that $f(x)$ is in B given that $x \in A$.

To put the left side of equation (7) into more familiar form and recalling that B is fixed and x is a variable and letting $\mu = d\mathsf{P}(x)$, we have

$$\int_A h(x,B)d\mathsf{P}(x) = \sup\left\{ \int_A s\, d\mu : 0 \le s \le \lambda x \; . \; h(x,B),\ s \in SF_X \right\}.$$

From [5],

Thus we can think of h as the conditional probability which gives the probability that $\mathcal{F}(x)$ is in B given that $x \in A$. Conditional probability has the equation, for $\mathsf{P}(A) \neq 0$,

$$\mathsf{P}(B|A) = \frac{\mathsf{P}(B \cap A)}{\mathsf{P}(A)}.$$

The equation with the integral above replaces the discrete equation

$$\mathsf{P}(B|A) * \mathsf{P}(A) = \mathsf{P}(B \cap A). \tag{8}$$

[8], pp. 146 also talks about Regular Probability Distributions.

It is important to remember that the equation is not defining conditional probability. Rather, given a conditional probability h, it must satisfy the equation. Another feature of the notation is that the integral has no mention of f in any of its syntax, this is left implicit. Also, $d\mathsf{P}(x)$ is referring to what exactly? It presumably picks out measurable sets to approximate $f^{-1}(B)$. Another notation, but not really better, is

$$\forall A \in \Sigma_X, \forall B \in \Sigma_Y \, . \int_A h_f(x, B) d\mathsf{P}_f(x) = \mathsf{P}(f^{-1}(B) \cap A).$$

We can overload the notation a bit by agreeing to use f in place of h_f with $f : X \twoheadrightarrow Y$ being the measurable map, and $f : X \times \Sigma_Y$ being the function it spawns:

$$\forall A \in \Sigma_X, \forall B \in \Sigma_Y \, . \int_A f(x, B) d\mathsf{P}_f(x) = \mathsf{P}(f^{-1}(B) \cap A). \tag{9}$$

The restriction bar of $B|A$ must become a logical connective; we will denote $A \mathrel{\Vert\!\!\twoheadrightarrow} B$. The stock understanding of conditional probability is to restrict the domain and hence in this qualitative connective, we must restrict the domain. See the Chapter 7 for more on this.

There is no reasonable computation of probability for classical logic's implication, $\supset$. We will use the restriction operator answering to conditional probability:

$$\mathsf{P}(A \mathrel{\Vert\!\!\twoheadrightarrow} B) = \mathsf{P}(B|A) = \frac{\mathsf{P}(B \wedge A)}{\mathsf{P}(A)}.$$

We can look at $|$ as a logical connective or as a set operator on UCLA propositions in

$$\mathsf{P}(B|A) = \frac{\mathsf{P}(B \cap A)}{\mathsf{P}(A)}$$

using UCLA propositions A and B, yet that is precisely what is going on, i.e., $\subseteq$ holds with a probability. Also,

$$\mathsf{P}(C) = \mathsf{P}(C|X) = \frac{\mathsf{P}(C \cap X)}{\mathsf{P}(X)} = \frac{\mathsf{P}(C)}{1} = \mathsf{P}(C).$$

This is still mysterious. We cannot use $\mathsf{P}(B \cap A)$ since B and A are in different spaces. To get them into the same space, we use f^{-1} and then weight the elements x in A with $d\mathsf{P}(x)$, and cycling through all the x in A.

We will assume that we always have some Polish or Analytic space of states, $(H, \mathcal{H})$ and $(K, \mathcal{K})$.

A further bit of notation from DL. The notation $f : \mathsf{h} \rightharpoonup \mathsf{k}$ indicates that relation $\mathcal{F} : \mathsf{h} \rightharpoonup \mathsf{k}$. The modal connective we will use, $[f^\circ\rangle : \mathsf{k} \rightharpoonup \mathsf{h}$, takes sets in the set lattice at k into sets in the set lattice at h. The reason $[f^\circ\rangle$ is used rather than $[f^\square\rangle$ is that the existential quantifier used in evaluating $[f^\circ\rangle$ forces $[f^\circ\rangle$ to act like $\mathcal{F}^{-1}$. Taking the inverse image of $\mathcal{F}$ is similar to taking the inverse image of any function, i.e.,

$$[f^\circ\rangle\, S = \{x \mid \exists y(\mathcal{F}xy \text{ and } y \in_{\mathsf{k}} S)\} = \mathcal{F}^{-1}S.$$

Assume that we wish to compute the probability of $[f^\circ\rangle\, S$ for $f : \mathsf{h} \rightharpoonup \mathsf{k}$. Underlying this is the relation $\mathcal{F}$ with the valuation condition for the formula (where we use the usual abuse of notation for h to refer to H and k to refer to K and the relation $\mathcal{F}$ underlies the specification $f : \mathsf{h} \rightharpoonup \mathsf{k}$)

$$x \in_{\mathsf{h}} [f^\circ\rangle\, S \text{ iff } \exists y \in_{\mathsf{k}} K(\mathcal{F}xy \text{ and } y \in_{\mathsf{k}} S).$$

Hidden a bit in this notation is that x is ranging over all of H which is collection of points of the space at h. Probabilities give a weight to sets so we wish

$$\mathsf{P}([f^\circ\rangle\, S) = \mathsf{P}(\{x \mid \exists y \in_{\mathsf{k}} K(\mathcal{F}xy \text{ and } y \in_{\mathsf{k}} S)\}).$$

We can turn $\mathcal{F}$ into a two-place (bivariate) function f using the set theoretic notation for relations,

$$f(x, S) \text{ iff } \exists y \in_{\mathsf{k}} K(\mathcal{F}xy \text{ and } y \in S),$$

and we write the associated measurable function

$$\lambda x \,.\, f(x, S).$$

That is,

$$(\lambda x \,.\, f(x, S))a = f(a, S).$$

λ is the usual function abstraction notation. We do not usually use this notation in DL because it implies the existence of a powerset functor or at least a functor to jump the set-theoretic type level. Note that

$$\mathsf{P}(H) = \int_{x \in H} d\mathsf{P}(x) = 1,$$

the probability used here is a probability function, not a subprobability function. The subprobability functions are used for the arrows of the category. Since $[f^\circ\rangle\, S \equiv [f^\circ\rangle\, S \mid H$, we retrieve the probability by scanning through all the $x \in H$ and adding the little bits of weight they contribute to $[f^\circ\rangle\, S$:

$$\begin{aligned}
\mathsf{P}([f^\circ\rangle\, S) &= \mathsf{P}(([f^\circ\rangle\, S) \cap H) \\
&= \mathsf{P}([f^\circ\rangle\, S \mid H) P(H) \\
&= \int_{x \in H} ((\lambda x \,.\, f(x, S))x) d\mathsf{P}(x) \\
&= \int_{x \in H} f(x, S) d\mathsf{P}(x).
\end{aligned}$$

Panangaden (see below) goes on to define the category **SRel** which uses subprobability measures instead of probability measures. The reason for this is that we work with measurable relations and not measurable functions. Relations need not be defined over the entire domain.

We take h (in place of $\mathcal{H}$) to be measurable relation in that it may only be partially defined, i.e., not entire. Regular conditional probability distributions are also known as Markov kernels. These kinds of distributions are useful because they cannot deliver undefined values unlike Kolmogorov conditional probabilities. The composition rule is precisely as in Giry [6] except for the use of subprobability measure in place of the probability measure.

In the following definition, Panangaden uses h where we have used f.

Definition 5.1.2 ([5]) The precategory **SRel** has as objects (X, Σ_X), sets equipped with a σ-field. The morphisms are conditional probability densities or Markov kernels. More precisely, a morphism (X, Σ_X) to (Y, Σ_Y) is a function $f : X \times \Sigma_Y \to [0, 1]$ such that

(i) $\forall x \in X, \lambda B \in \Sigma_Y \,.\, f(x, B)$ is a subprobability measure on Σ_Y,

(ii) $\forall B \in \Sigma_Y, \lambda x \in X \,.\, f(x, B)$ is a measurable function.

The composition rule is as follows. Suppose $f : (X, \Sigma_X) \to (Y, \Sigma_Y)$, $g : (Y, \Sigma_Y) \to (Z, \Sigma_Z)$, $x \in X$, and $C \in \Sigma_Z$:

$$(g \circ f)(x, C) = \int_Y g(y, C) f(x, dy). \tag{10}$$

□

Typical Billingsley notation is

$$\int f \, d\mu = \int_\Omega f(\omega) \, d\mu(\omega) = \int_\Omega f(\omega) \, \mu(d\omega).$$

Replacing μ with P, Ω with Y, ω with y, and f with $g(y, C)$ (C is constant and $g(\cdot, C)$ is a measurable function), we have

$$\int_Y g(y, C)\ d\mathsf{P}(y) = \int_Y g(y, C)\mathsf{P}(dy).$$

Next, replace $\mathsf{P}(dy)$ with $r(x, dy)$ where x is constant and $f(x, \cdot)$ is a subprobability measure, we have

$$\int_Y g(y, C)\ df(x, y) = \int_Y g(y, C) f(x, dy).$$

The reason this works as an integral is that $f(x, \cdot)$ is a subprobability measure, just like P. Hence where one would poke in $d\,\mathsf{P}(y)$ one can poke in $d\mu(x, y)$, or as we have it $df_x(y)$. However, $df(x, y)$ does not help in picking out the differential variable. These tend to get written as $d\mathsf{P}(y)$ and $f(x, dy)$. In the composition formula, x is a free variable and hence held constant; so we could have displayed $f(x, dy)$ as $f_x(dy)$.

Note that $(g \circ f)(x, C)$ is a conditional probability, i.e., a Markov kernel. It is helpful here to point out that integration is being used for two very different purposes. The integration in the formula satisfied by a Markov kernel is used to define a Markov kernel; that is, a function satisfying certain constraints, one of them involving integration. The other use is in composition. Markov kernels are composed using integration to make the middle domain of the composition disappear by using the analysis notion of integration in place of existential quantification.

For bog standard relations $\mathcal{F} : X \rightharpoonup Y$ and $\mathcal{G} : Y \rightharpoonup Z$, composition is defined as

$$(\mathcal{G} \circ \mathcal{F})xz \text{ iff } \exists y(\mathcal{F}xy \text{ and } \mathcal{G}yz).$$

So composition of relations $\mathcal{F}$ and $\mathcal{G}$ uses an existential to make the domain Y disappear; composition of Markov kernels f and g also makes Y disappear.

Incidentally,

$$\int_Y g(y, C) f(x, dy) = \sup\left\{ \int_Y s\, f_x(dy) : 0 \leq s \leq g(\cdot, C),\ s \in SF_Y \right\}$$

This looks a bit less intimidating in the form

$$\int_Y g_C(y) f_x(dy) = \sup\left\{ \int_Y s\, f_x(dy) : 0 \leq s \leq g_C,\ s \in SF_Y \right\}$$

Let $\breve{h} : \mathsf{h} \rightharpoonup \mathsf{h}$, $\breve{k} : \mathsf{k} \rightharpoonup \mathsf{k}$ and $f, g : \mathsf{h} \rightharpoonup \mathsf{k}$ (not the f of [7]), and let H be the space at h, K the space at k, and $Q \in \mathsf{k}$. The h and k are next-state relations on their respective Stone spaces at h and k respectively. Hence $\breve{h}$ and $\breve{k}$ are the previous-state relations. We have the following equivalent diagrams:

$$\begin{tikzcd}[column sep=small]
\forall & & \\
x' & & y \\
& x \arrow[ul, mapsto, "\mathcal{H}"] \arrow[ur, mapsto, "\mathcal{F}"'] &
\end{tikzcd}
\quad\Bigg|\quad
\begin{tikzcd}[column sep=small]
\exists & y' & \\
x' \arrow[ur, mapsto, "\mathcal{G}"] & & y \arrow[ul, mapsto, "\mathcal{K}"'] \\
& x \arrow[ul, mapsto, "\mathcal{H}"] \arrow[ur, mapsto, "\mathcal{F}"'] &
\end{tikzcd}
\qquad
\begin{tikzcd}[column sep=small]
\forall & & \\
x' \arrow[dr, mapsto, "\check{\mathcal{H}}"'] & & y \\
& x \arrow[ur, mapsto, "\mathcal{F}"'] &
\end{tikzcd}
\quad\Bigg|\quad
\begin{tikzcd}[column sep=small]
\exists & y' \arrow[dr, mapsto, "\check{\mathcal{K}}"] & \\
x' \arrow[ur, mapsto, "\mathcal{G}"] \arrow[dr, mapsto, "\check{\mathcal{H}}"'] & & y \\
& x \arrow[ur, mapsto, "\mathcal{F}"'] &
\end{tikzcd}$$

Figure 5.1: Equivalent Forward Simulation Conditions

Now we change the variables by swapping the instances of x and x', and y and y'. This is so that the x' and y' are the variables quantified away in the integral formulas below:

$$\begin{tikzcd}[column sep=small]
\forall & & \\
x \arrow[dr, mapsto, "\check{\mathcal{H}}"'] & & y' \\
& x' \arrow[ur, mapsto, "\mathcal{F}"'] &
\end{tikzcd}
\quad\Bigg|\quad
\begin{tikzcd}[column sep=small]
\exists & y \arrow[dr, mapsto, "\check{\mathcal{K}}"] & \\
x \arrow[ur, mapsto, "\mathcal{G}"] \arrow[dr, mapsto, "\check{\mathcal{H}}"'] & & y' \\
& x' \arrow[ur, mapsto, "\mathcal{F}"'] &
\end{tikzcd}$$

Figure 5.2: Forward Simulation with Change of Variables

We can measure

$$[f^\diamond\rangle\,[k^\square\!\cdot\rangle Q \overset{\mathsf{h}}{\supset} [h^\square\!\cdot\rangle\,[g^\diamond\rangle\, Q$$

by measuring

$$[h^\diamond\!\cdot\rangle\,[f^\diamond\rangle\, Q \mathrel{\Vdash\!\!\!\blacktriangleright} [g^\diamond\rangle\,[k^\diamond\!\cdot\rangle Q$$

and using the interpretation of $\supset$ as $\Vdash\!\!\!\blacktriangleright$, i.e., in terms of restriction.

The valuation conditions are

$$x \in [h^\diamond\!\cdot\rangle\,[f^\diamond\rangle\, Q \text{ iff } \exists x' \in H,\ \exists y \in K(\check{\mathcal{H}}xx' \text{ and } \mathcal{F}x'y \text{ and } y \in Q)$$

and

$$x \in [g^\diamond\rangle\,[k^\diamond\!\cdot\rangle Q \text{ iff } \exists y \in K,\ \exists y' \in K(\mathcal{G}xy \text{ and } \check{\mathcal{K}}yy' \text{ and } y' \in Q).$$

Note that

$$\mathcal{F}x'Q \text{ iff } \mathcal{F}x' \cap Q \neq \emptyset \text{ iff } \exists y(\mathcal{F}x'y \text{ and } y \in Q).$$

A similar statement holds for $\breve{\mathcal{K}}$, so we can rewrite these conditions (and reordering the conjuncts) as

$$x \in [h^\diamond\cdot\rangle\,[f^\diamond\rangle\, Q \text{ iff } \exists x' \in H(\mathcal{F}x'Q \text{ and } \breve{\mathcal{H}}xx') \qquad x \in [g^\diamond\rangle\,[k^\diamond\cdot\rangle Q \text{ iff } \exists y \in K(\breve{\mathcal{K}}yQ \text{ and } \mathcal{G}xy).$$

We will use the lower case counterparts of the relations in the composition statements:

$$(f \circ \breve{h})(x, Q) \overset{def}{=} \int_{x' \in H} f(x', Q)\breve{h}(x, dx') \qquad (\breve{k} \circ g)(x, Q) \overset{def}{=} \int_{y' \in K} \breve{k}(y', Q)g(x, dy')$$

These latter are subprobability measures since they are expressions of composition in **SRel**. Note that the position of h and f in $[h^\diamond\cdot\rangle\,[f^\diamond\rangle\, Q$ is opposite the position of $\breve{h}$ and f in $(f \circ \breve{h})(x, Q)$. A similar statement holds for g and k for the second formula.

We wish to compute a conditional probability from

$$[h^\diamond\cdot\rangle\,[f^\diamond\rangle\, Q \mathrel{\Vdash\!\!\blacktriangleright} [g^\diamond\rangle\,[k^\diamond\cdot\rangle Q.$$

Let H be the subprobability measure at h. For the formula of conditional probability,

$$\mathsf{H}(A \supset B) = \mathsf{H}(B|A) = \frac{\mathsf{H}(B \cap A)}{\mathsf{H}(A)},$$

we define:

$$A \overset{def}{=} [h^\diamond\cdot\rangle\,[f^\diamond\rangle\, Q, \qquad B \overset{def}{=} [g^\diamond\rangle\,[k^\diamond\cdot\rangle Q.$$

Blindly plugging in the required sets and measurable relations into the format provided by [5], then, using the replacement formula for the left side of equation (8) (twice), we get

$$\mathsf{H}(A) = \mathsf{H}(A \cap H) = \int_{x \in H} (f \circ \breve{h})(x, Q)d\,\mathsf{H}(x) \qquad \mathsf{H}(B \cap A) = \int_{x \in A} (\breve{k} \circ g)(x, Q)d\,\mathsf{H}(x).$$

In more drawn out form, we have

$$\mathsf{H}(A) = \mathsf{H}(A \cap H) = \int_{x \in H} \int_{x' \in H} f(x', Q)\breve{h}(x, dx')d\,\mathsf{H}(x)$$

and

$$\mathsf{H}(B \cap A) = \int_{x \in A} \int_{y' \in K} \breve{k}(y', Q)g(x, dy')d\,\mathsf{H}(x).$$

Hence, we can now measure

$$\mathsf{H}\left([h^\diamond\cdot\rangle\,[f^\diamond\rangle\, Q \overset{\mathsf{h}}{\supset} [g^\diamond\rangle\,[k^\diamond\cdot\rangle Q\right).$$

We will work with the first formula by holding x constant on the inner integral:

$$\begin{aligned}
\int_{x\in H}\int_{x'\in H} f(x',Q)\breve{h}(x,dx')d\,\mathsf{H}(x) &= \int_{x\in H}\left(\int_{x'\in H} f(x',Q)\breve{h}(x,dx')\right) d\,\mathsf{H}(x)\\
&= \int_{x\in H}\left(\int_{x'\in H} f_Q(x')\breve{h}_x(dx')\right) d\,\mathsf{H}(x)\\
&= \int_{x\in H}\left(\overbrace{\sup\left\{\int_{x'\in H} s\,\breve{h}_x(dx') : 0\le s\le f_Q,\ s\in SF_H\right\}}^{r(x,Q)}\right) d\,\mathsf{H}(x)\\
&= \sup\left\{\int_{x\in H} s\ \ d\,\mathsf{H}(x) : 0\le s\le r(\cdot,Q),\ s\in SF_H\right\}
\end{aligned}$$

How for the second formula:

$$\begin{aligned}
\int_{x\in A}\int_{y'\in K} \breve{k}(y',Q)g(x,dy')d\,\mathsf{H}(x) &= \int_{x\in A}\left(\int_{y'\in K} \breve{k}(y',Q)g(x,dy')\right) d\,\mathsf{H}(x)\\
&= \int_{x\in A}\left(\int_{y'\in K} \breve{k}_Q(y')g_x(dy')\right) d\,\mathsf{H}(x)\\
&= \int_{x\in A}\left(\overbrace{\sup\left\{\int_{y'\in K} s\,g_x(dx') : 0\le s\le \breve{k}_Q,\ s\in SF_K\right\}}^{t(x,Q)}\right) d\,\mathsf{H}(x)\\
&= \sup\left\{\int_{x\in A} s\ \ d\,\mathsf{H}(x) : 0\le s\le t(\cdot,Q),\ s\in SF_A\right\}
\end{aligned}$$

5.2 Change of Variables

This section is not needed in the succeeding section but does allow us to connect up with Doberkat's [1, 2] work.

From [2] Lemma 1.6.20 and [1] Proposition 1.95 is

Lemma 5.2.1 (Change of Variables) *Let (X,Σ_X) and (Y,Σ_Y) be analytic spaces and assume $f : X \twoheadrightarrow Y$ is X-Y measurable. If $g : Y \twoheadrightarrow \mathbb{R}$ is Y measurable and bounded, and $C \subseteq Y$ a measurable set,*

$$\int_Y g(y)\mathfrak{G}(f)(\mu)(dy) = \int_X (g\circ f)(x)\mu(dx) \tag{11}$$

In particular

$$\int_C g(y)\mathfrak{G}(f)(\mu)(dy) = \int_{f^{-1}(C)} (g\circ f)(x)\mu(dx)$$

From (9), we can substitute X for A since $X \subseteq X$ so that $f^{-1}(B) \cap X = f^{-1}(B)$,

$$\forall B \in \Sigma_Y \,.\, \int_X f(x, B) d\mathsf{P}(x) = \mathsf{P}(f^{-1}(B)).$$

Now swapping changing notation from P to μ and holding B constant, we get Panangaden's (7).

$$\int_X f(x, B) d\mu(x) = \mu(f^{-1}(B)).$$

From [2], let $\mathfrak{G}(X, \Sigma_X)$ be the collection of all subprobability measures on (X, Σ_X). For any X-Y measurable map $f : X \to Y$, $\mathfrak{G}(f) : \mathfrak{G}(X, \Sigma_X) \to \mathfrak{G}(Y, \Sigma_Y)$ is defined, for $B \in \Sigma_Y$,

$$\mathfrak{G}(f)(\mu)(B) \overset{def}{=} \mu(f^{-1}(B)) \tag{12}$$

where it is left to the reader to determine that $\mu : X \to [0, 1]$ is a probability measure (others call it a "distribution") on X. With this understanding, $\mathfrak{G}(f)$ is a probability measure on X.

Combining Equations (7) and (12), we get

$$\mathfrak{G}(f)(\mu)(B) = \int_X f(x, B) d\mu(x),$$

where in the context of the integration below (see Sections 4.1 and 4.2),

$$\mathfrak{G}(f)(\mu)(dy) = \int_X f(x, dy) d\mu(x). \tag{13}$$

Doberkat's aim is to turn $\mathfrak{G}(f)(\mu)(\text{-})$ into a probability measure. The right hand side is the probability measure

$$\int_X f(x, \text{-}) d\mu(x).$$

We note that Equation (10) can be put into different form due to the fact that C may be considered a constant measurable set:

$$\begin{aligned}(g \circ f)(x, C) &= \int_Y g(y, C) f(x, dy) \\ &= \int_Y g_C(y) f(x, dy) \\ &= (g_C \circ f)(x).\end{aligned}$$

We will need the Fubini Theorem:

Theorem 5.2.2 ([9]) *Let X and Y be σ-finite measure spaces, i.e., their measures are such that $\mu_X(C), \mu_Y(C) \leq \infty$. Also let $X \times Y$ have the product measure. The product measure is unique since X and Y are σ-finite. If f is $X \times Y$-integrable, i.e., f is a measurable function and*

$$\int_{X\times Y} |f(x,y)|\, d(x,y) < \infty,$$

then

$$\int_X \left(\int_Y f(x,y)\, dy\right) dx = \int_Y \left(\int_X f(x,y)\, dx\right) dy = \int_{X\times Y} f(x,y)\, d(x,y).$$

Tonelli extended Fubini's theorem by relaxing the requirement that f has a finite integral to f being a non-negative measurable function.

The key feature of Fubini's theorem we will need is that the integrals over X and Y commute.

Lemma 5.2.3 *Let (X, Σ_X) and (Y, Σ_Y) be analytic spaces and assume $f : X \twoheadrightarrow Y$ is X-Y measurable. If $g : Y \twoheadrightarrow \mathbb{R}$ is Y measurable and bounded, and $C \subseteq Y$ a measurable set. The Change of Variables Lemma 5.2.1 is the definition for composition in **SRel**.*

Proof: Let C be fixed and $g(\text{-}, C) : Y \twoheadrightarrow \mathbb{R}$ is measurable

$$\begin{aligned}
\int_X \Big(g \circ f\Big)(x, C)(\mu(dx)) &= \int_X \Big(g_C \circ f\Big)(x)(\mu(dx)) && C \text{ is constant}\\
&= \int_Y g_C(y)\, \mathfrak{G}(f)(\mu)(dy) && \text{Change of Variables, Eq. (11)}\\
&= \int_Y g_C(y) \int_X f(x, dy)\, d(\mu x) && \text{Eq. (13)}\\
&= \int_Y g_C(y) \int_X f(x, dy)\, \mu(dx) && \text{Billingsley, Eq. (5)}\\
&= \int_Y \int_X g_C(y) f(x, dy)\, \mu(dx) && g_C(y) \text{ constant for } f(x, dy)\\
&= \int_X \Big(\int_Y (g_C(y)) f(x, dy)\, \mu(dx)\Big) && \text{Theorem 5.2.2}\\
&= \int_X \Big(g \circ f\Big)(x, C)\, \mu(dx) && \text{Comp. in SREL, Equation (10)}
\end{aligned}$$

■

The point is that composition in **SRel** is intimately tided to the Change of Variables theorem. Note that the Change of Variables or composition in **SRel** is not used for $P(B|A)$ for the formula $A \supset B$ because the A and B are not connected via composition. They are separate formulas of the form $A = [h^{\diamond}\cdot\rangle\,[f^{\diamond}\rangle\, Q$ and $B = [g^{\diamond}\rangle\,[k^{\diamond}\cdot\rangle Q$. Composition in **SRel** is used to evaluate A and B internally, not an external relationship between them.

5.3 Identities in the Category of SRel

Every category must have identity arrows. Panangaden [5] identifies them as simple Lebesgue integrations.

According to [5], we have

$$h(x, B) = \int_X h(x', B)\, \delta(x, dx')$$

where δ is a characteristic function. In other words, the right hand side functions as an identity arrow in the category **SRel**. In the words of [5], this is "a simple computation of a Lebesgue integral". This simple computation will occupy us for a bit.

What he is really saying is that $\delta(x, dx')$ is a parameterized form of $\mu(dx')$. In our parlance, that becomes $\mu_x(d_s x')$ in the integration context. Translating back to Panangaden, $\delta_x(d_s x')$, i.e.,

$$h(x, B) = \int_X h(x', B)\, \delta_x(d_s x').$$

Since here the integral is "pinned" to the free variable x, we can treat x as a constant and recast δ as

$$\delta_x A = \begin{cases} 1 & x \in A \\ 0 & \text{otherwise} \end{cases}$$

That $\delta(x, \cdot)$ is the identity stems from treating the above integral as composition in the category **SRel** using the prescription for compositions in that category from Definition (ii). δ_x has type $\delta_x : (X, \Sigma_X) \twoheadrightarrow (X, \Sigma_X)$. Let $h : (X, \Sigma_X) \twoheadrightarrow (Y, \Sigma_Y)$ be a measurable function, then

$$(h \circ \delta)(x, B) = \int_X h(x', B) \delta_x(dx').$$

Expanding this, we have

$$\begin{aligned}
\int_X h(x', B)\, \delta(x, d_s x') &= \sup \left\{ \int_{x' \in X} s(x')\, \delta_x(d_s x') \mid 0 \le s \le h(\cdot, B),\ s \in SF_X \right\} \\
&= \sup \left\{ \int_{x' \in X} 1_S(x')\, \delta_x(d_s x') \mid 0 \le s \le h(\cdot, B),\ s \in SF_X \right\} \\
&= \sup \{ a * \delta_x(S) \mid 0 \le s \le h(\cdot, B),\ s \in SF_X \} \\
&= \sup \{ a * \delta_x(h^{-1}B) \mid 0 \le s_{h^{-1}B} \le h(\cdot, B) \} \\
&= h(x, B) * \delta_x(h^{-1}B) \\
&= h(x, B)
\end{aligned}$$

Since h is a measurable function and B is measurable, then $h^{-1}B$ is measurable and is the largest measurable set S such that for all $z \in h^{-1}B$, we have $0 \le s(z) \le h(z, B)$. That is, $s = s_{h^{-1}B}$. Comparing s with δ_x we have

$$s_{h^{-1}B} = \begin{cases} a * 1 & x \in h^{-1}B \\ 0 & \text{otherwise} \end{cases} \qquad a * \delta_x(h^{-1}B) = \begin{cases} a * 1 & x \in h^{-1}B \\ 0 & \text{otherwise} \end{cases}$$

It is clear that $a = h(x, B)$.

6. FINITE STATE MACHINE EXAMPLES

In control systems, the system being controlled is called the *plant*. We adopt this term here for the system being watched by the spider. We assume there is spider watching over a plant. The measurement is intended to explain how close the spider can determine the plant's behavior if the spider has incomplete information about the plant. The incompleteness is where measurement comes in.

6.1 Measuring Properties of Finite State Machines

Let the plant be at locality h and the spider at locality k with $\mathcal{F}$ the simulation or connection relation that relates states in the plant with states in the spider. We will use the forward simulation condition and assume $\mathcal{G} = \mathcal{F}$ although we will continue to use $\mathcal{G}$ and subsequently g in the mathematics to make it easier to distinguish from where the computations come.

The simulation condition, altered to use the converse of $\mathcal{H}$ so that we may use DL formulas only involving possibility, i.e. $[\text{-}^{\circ}\cdot\rangle$, is

$$\begin{tikzcd}
\forall \arrow[ddd, dash] & & & & \exists \arrow[ddd, dash] & & y' \arrow[dr, |->, "\breve{\mathcal{K}}"] & \\
& & & & & & & \\
& x' \arrow[dr, |->, "\breve{\mathcal{H}}"'] & & y & & x' \arrow[uur, |->, "\mathcal{G}"] \arrow[dr, |->, "\breve{\mathcal{H}}"'] & & y \\
& & x \arrow[ur, |->, "\mathcal{F}"'] & & & & x \arrow[ur, |->, "\mathcal{F}"'] &
\end{tikzcd}$$

Figure 6.1: Forward Simulation

says

> if the plant starts at state x which we have determined to be related to the spider at state y under the simulation relation $\mathcal{F}$, and the plant makes one move to x' under $\mathcal{H}$, then the spider takes one step to y' under $\mathcal{K}$ and y' is related to x' under the simulation.

Note that we can read the simulation relation forward or backward. However, when evaluating probabilities, it very much matters in which direction we are reading the simulation relation. Give the diagram, we must be reading the simulation relation in its forward direction. That direction is from the plant to the spider.

We will be measuring the formula

$$[f^{\circ}\rangle\,[k^{\square}\cdot\rangle Q \overset{\mathsf{h}}{\supset} [h^{\square}\cdot\rangle\,[g^{\circ}\rangle\, Q$$

by measuring the equivalent

$$[h^{\diamond}\cdot\rangle\,[f^{\diamond}\rangle\, Q \Vdash\!\!\blacktriangleright [g^{\diamond}\rangle\,[k^{\diamond}\cdot\rangle Q$$

and using the interpretation of $\supset$ as $\subseteq$, which we now think of as a conditional probability.

We wish to evaluate

$$\mathsf{H}(A \supset B) = \mathsf{H}(B|A) = \frac{\mathsf{H}(B \cap A)}{\mathsf{H}(A)},$$

by letting

$$A \overset{def}{=} [h^{\diamond}\cdot\rangle\,[f^{\diamond}\rangle\, Q, \qquad B \overset{def}{=} [g^{\diamond}\rangle\,[k^{\diamond}\cdot\rangle Q.$$

Hence the stochastic formulas are

$$\mathsf{H}(A) = \mathsf{H}(A \cap H) = \int_{x \in H} (f \circ \check{h})(x, Q)\, d\,\mathsf{H}(x) \qquad \mathsf{H}(B \cap A) = \int_{x \in A} (\check{k} \circ g)(x, Q)\, d\,\mathsf{H}(x).$$

in their expanded forms

$$\mathsf{H}(A) = \mathsf{H}(A \cap H) = \int_{x \in H} \int_{x' \in H} f(x', Q)\check{h}(x, dx')\, d\,\mathsf{H}(x) \quad \mathsf{H}(B \cap A) = \int_{x \in A} \int_{y' \in K} \check{k}(y', Q) g(x, dy')\, d\,\mathsf{H}(x).$$

For UCLA propositions P, define probability distribution H at h as

$$\mathsf{H}(P) \overset{def}{=} |P|/|H|$$

where $|-|$ is the cardinality of its argument and H is the ambient state space, i.e., $P \subseteq H$. Hence H is a probability measure over H. The probability distribution K at k is similar.

The measurable sets for finite H or K can be viewed as a complete lattice. Atomic measurable sets are singletons of either H or K. Using those, we can complete the lattice of sets by taking unions and including $\emptyset$. Σ_H or Σ_K is $(\Sigma_X, \cup, \cap)$ for $X \in \{H, K\}$. Let $\mathcal{H} = \{x_0, x_1, x_2, x_3\}$ and $\mathcal{K} = \{y_0, y_1, y_2, y_3\}$, then either Σ_H or Σ_K can be pictured as

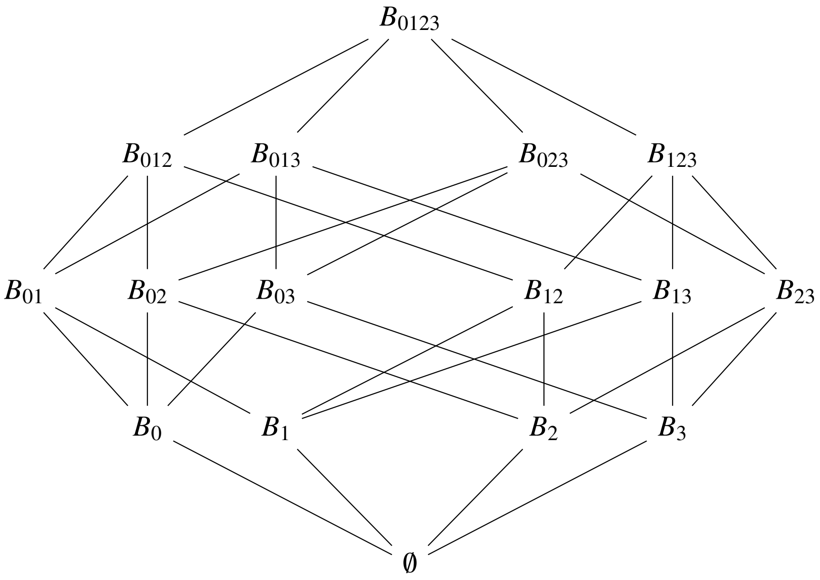

Figure 6.2: measurable sets in $\mathcal{S}$

where there is a line from B to B' iff $B \subseteq B'$ for B, B' elements of the lattice and, for example, $B_{ij} = B_i \cup B_j$, so that $B_{013} = B_0 \cup B_1 \cup B_3$. All the possible propositions are measurable sets here. However, in complicated situations, even though dealing with finite state machines, not all the measurable sets will be simple. Some may only be approximated if we have incomplete information about what is in them, say, when foreign IP is used. In this instance, when we are not dealing with the foreign IP complication, we will let the above lattice stand for measurable sets.

The probability of any of these sets is defined via

$$\mathsf{P}(B) = |B|/4, \quad \text{the cardinality of } B \text{ divided by the cardinality of the space.}$$

The unions will be important even in this finite state machine example because we must evaluate a supremum over the lattice order. It will make the sequel a bit easier if we note that

$$\varphi : \; s'(x) \leq s(x) \text{ iff } S_{s'} \subseteq S_s. \tag{14}$$

If we associate each measurable set of the lattice with a simple function, i.e., for B in the lattice there is a simple function s_B such that $S_{s_B} = B$, then the lattice is isomorphic to a lattice of simple functions.

We define the lattice of simple function for finite X ($X \in \{H, K\}$) defined via the bijection $\varphi : \Sigma_X \twoheadrightarrow SF_X$ (14) and the operation identities

$$\begin{aligned} s_B \wedge s_{B'} = \varphi(B) \wedge \varphi(B') &\overset{def}{=} \varphi(B \cap B') = s_{B \cap B'} \\ s_B \vee s_{B'} = \varphi(B) \vee \varphi(B') &\overset{def}{=} \varphi(B \cup B') = s_{B \cup B'} \\ \neg s_B = \neg\varphi(B) &\overset{def}{=} \varphi(\neg B) = s_{\neg B}. \end{aligned}$$

This allows us to move freely back and forth between the lattices, and allows us to make sense of $\sup_{s \in SF}$.

For state machines, measurable sets are of the form

$$S = \sum_{0 \leq k < n} S_k$$

and the S_k are pairwise disjoint and in

$$\int s d\mu = \sum_{0 \leq k < n} a_k * \mu(S_k).$$

We will need to consider simple functions that contribute to the sup for $\mathcal{F} : \mathsf{h} \rightharpoonup \mathsf{k}$ (h and k might be the same locality, and choose X accordingly)

$$\sup \left\{ \int_X s(x)\, d\mu(x) \mid \forall z \in X (0 \leq s(z) \leq f(z, B)),\ s \in SF_X \right\}. \tag{15}$$

where $f(z, B)$ is the probability of landing in B under the relation $\mathcal{F}$, i.e., $\mathcal{F}zy$ and $y \in B$. Note that the sup here is not in the SF_X lattice; the sup is rather in $[0, 1]$. The $s \in SF_X$ is merely an index for simple functions satisfying the constraint $\forall z \in X(\ldots)$.

Consider the set

$$[f^\circ\rangle B = \{z \mid \exists y (\mathcal{F}zy \text{ and } y \in B)\}.$$

When considering a simple function s_C, in computing the sup we need only concern ourselves with those C such that $C \subseteq [f^\circ\rangle B$. To see this, let $C \nsubseteq [f^\circ\rangle B = \mathcal{F}^{-1}B$ and note that for each $z \in [f^\circ\rangle B$, we have $f(z, B) > 0$ and for each $z \notin B$, we have $f(z, B) = 0$. Since $C \nsubseteq [f^\circ\rangle B$, there is some $z \in C$ and $z \notin [f^\circ\rangle B$, then $f(z, B) = 0$. Let $D = \{z\}$. Since C is finite, then

$$C = \bigcup_{k<n} D_k$$

for some collection of singleton sets D_k. However, $D = D_k$ for some k. Hence

$$s_D(z) = \mathsf{X}(D) = a_D * \frac{|D|}{|X|} > 0 = f(z, B).$$

Since $s_D \leq s_C$, then $s_C(z) = s_D(z)$ and so $s(C)(z) > f(z, B)$. What does hold is that if

$$s_{C'} \leq s_C$$

and s_C does meet the criteria, then we need not consider $s_{C'}$ because the sup will include s_C whose values swamp those of $s_{C'}$.

The upshot is that we need only consider sets in the principle ideal of Σ_X formed from $[f^\circ\rangle B$. Any set C not in that ideal must contain at least one z for which $s_C(z) > 0 = f(z, B)$.

Another upshot is that if $0 \le s_{[f^\circ\rangle B} \le f(\cdot, B)$, then we need only consider $s_{[f^\circ\rangle B}$ since sets underneath it in the lattice will not contribute any values to the sup that are not subsumed by $s_{[f^\circ\rangle B}$ and simple functions s_D for $s_D \not\le s_{[f^\circ\rangle B}$ will be such that D contains elements z such that $s_D(z) > s_{[f^\circ\rangle B}$.

We slightly redefine simple functions to label them with the set to which they are associated. This allows for the succeeding theorem to be used when integrating simple functions.

Definition 6.1.1 For the simple function s_D where D is some measurable set,

$$s_D(x) = s(x, D) \overset{def}{=} \begin{cases} 1 & x \in D \\ 0 & \text{otherwise} \end{cases}$$

Let $\mathcal{D}$ be the measurable relation

$$\mathcal{D}xy \text{ iff } x = y \text{ and } y \in D,$$

so that $\mathcal{D}^{-1}D = D$.

Theorem 6.1.2 *For finite state machines at localities* x *and* y, $\mathcal{R} : \mathsf{x} \rightharpoonup \mathsf{y}$, *and* $\mu = \mathsf{X}$, *we can always choose step functions s such that*

$$\forall z_k \in r^{-1}Q(s(z_k) = r(z_k, Q)).$$

Thus, for A measurable, $A \subseteq X$, and $D_k = \{z_k\}$

$$\int_{x \in A} r(x, Q)\, d\mu = \sum_k r(x_k, Q) * \mu(D_k),\ x_k \in r^{-1}Q \cap A,$$

and $\mu(D_k) = |D_k|/|X|$. In particular, for measurable set D,

$$\int_{x \in A} s(x, D)\, d\mu = \mu(D \cap A).$$

Proof: Let $Q = \{y_0, \ldots, y_{m-1}\}$ and $r(x, y_i)$ be the probability on the arc from x to y_i under the relation $\mathcal{R} : \mathsf{x} \rightharpoonup \mathsf{y}$ (again where it is possible $\mathsf{x} = \mathsf{y}$, choose X accordingly):

$$r(x, Q) \overset{def}{=} \sum_{0 \le i < m} r(x, y_i).$$

That is, to get the probability of landing in Q from x under $\mathcal{R}$, we sum up the probabilities of hitting any one of the elements in Q from x under $\mathcal{R}$. Now consider a simple function s satisfying

$$0 \le s \le r(\cdot, Q) \text{ iff } \forall z \in X(0 \le s(z) \le r(z, Q)).$$

Since we are considering finite state machines, the domain X is finite. Let $X = \{x_0, \ldots, x_{m-1}\}$. We only need to consider sets in the principle ideal whose top element is $r^{-1}Q$, so $r^{-1}Q \subseteq X$. Let

$$r^{-1}Q = \{z_0, \ldots, z_{n-1}\}$$

We can always choose s such that

$$\forall z_k \in r^{-1}Q(s(z_k) = r(z_k, Q)).$$

That is, $r(z_k, Q)$ is itself a step function taking discrete values for all z_k. Since each $z_k \in \{x_k\}$ and letting $D_k = \{z_k\}$, we define

$$s_{D_k}(x) = \begin{cases} r(x, Q) & x = z_k \\ 0 & \text{otherwise} \end{cases}$$

This causes

$$\forall z \in X\Big(0 \le s_{r^{-1}Q}(z) \le r(z, Q)\Big) \text{ iff } \forall z \in r^{-1}Q\Big(0 \le s_{r^{-1}Q}(z) \le r(z, Q)\Big).$$

So now we can always choose s such that

$$\forall z_k \in r^{-1}Q(s(z_k) = s_{r^{-1}Q}(z_k) = r(z_k, Q)).$$

The simple function $s_{r^{-1}Q}$ dominates all simple functions in the principle ideal generated by $s_{r^{-1}Q}$. Stretched out, for all $C \in \mathcal{P}(r^{-1}Q)$, $s_C \le s_{r^{-1}Q}$.

$$\begin{aligned}
\int_{x\in A} r(x, Q)\, d\mu &= \sup\left\{ \int_{x\in A} s(x)\, d\mu \mid \forall z \in A\Big(0 \le s(z) \le r(z, Q)\Big) \right\} \\
&= \sup\left\{ \int_{x\in A} s(x)\, d\mu \mid \forall z \in A\Big(0 \le s(z) \le r(z, Q),\ s \in \{s_{r^{-1}Q}\}\Big) \right\} \\
&= \int_A s_{r^{-1}Q}(x)\, d\mu \\
&= \int_A \left(\sum_k a_k * 1_{D_k}(x) \mid D_k \subseteq A \cap r^{-1}Q \text{ and } D_k = \{x_k\} \right) d\mu \\
&= \int_A \left(\sum_k r(x_k, Q) * 1_{D_k}(x) \right) d\mu, \quad x_k \in A \cap \mathcal{R}^{-1}Q \text{ and } D_k = \{x_k\} \\
&= \sum_k r(x_k, Q) \int_A 1_{D_k}\, d\mu, \quad x_k \in A \cap \mathcal{R}^{-1}Q \text{ and } D_k = \{x_k\} \\
&= \sum_k r(x_k, Q) * \mu(D_k \cap A), \quad x_k \in A \cap \mathcal{R}^{-1}Q \text{ and } D_k = \{x_k\} \\
&= \sum_k r(x_k, Q) * \mu(D_k), \quad x_k \in A \cap \mathcal{R}^{-1}Q \text{ and } D_k = \{x_k\}
\end{aligned}$$

Since we are working with finite state machines, we can always compute $\mu(D_k) = |D_k|/|X|$.

Lastly,

$$
\begin{aligned}
\int_{x\in A} s(x,D)\, d\mu &= \sup\left\{ \int_{x\in A} s(x)\, d\mu \mid \forall z \in X\Big(0 \le s(z) \le s(z,D),\ s \in SF_X\Big) \right\} \\
&= \sup\left\{ \int_{x\in A} s(x)\, d\mu \mid \forall z \in X\Big(0 \le s(z) \le s(z,D),\ s \in \{s(\cdot,D)\}\Big) \right\} \\
&= \int_{x\in A} s(x,D)\, d\mu \\
&= \sum_k \mathcal{D}(x_k, D_k) \int_A 1_{D_k}(x)\, d\mu, \quad x_k \in A \cap \mathcal{D}^{-1}D \\
&= \sum_1 \mathcal{D}(x_1, D) * \mu(A \cap D),\ x_1 \in A \cap \mathcal{D}^{-1}D \\
&= 1 * \mu(D \cap A) \\
&= \mu(D \cap A)
\end{aligned}
$$

■

7. DISTRIBUTED LOGIC EXTENSIONS

While the above work is technically correct, there are some holes that will need to be filed before it is well grounded. In particular, Distributed Logic must be extended and the algebraic models of Distributed Logic must be necessarily upgraded. A new topological model must be developed expressly for measurement.

Distributed Logic, in order to be a logic of probability, must include countable disjunctions. This notion must be axiomatized in the logic and the relationship with the other connectives must also be axiomatized.

Let $\mathcal{F} : H \rightharpoonup A$ be the restriction relation. This relation must be axiomatized in Distributed Logic so that it can be used in interpreting the restriction connective, $A \Vdash\!\xrightarrow{f} B$. The reason for this is that just for probability *simpliciter*, it could simply be defined on a single probability space. However, given the work above with Markov kernels, it is clear we must iterate this connective and make it a connective of Distributed Logic.

With the extensions to the logic, the algebraic semantics must be extended as well. In particular, σ-lattices must be used in place of the Boolean lattices as the base algebraic structure at every node in the distribution graph. Also, the distribution structure itself must be altered to accommodate the restriction connective.

The algebraic and topological (Kripke) semantics must be changed for the quantitative case to include measurement structure. The qualitative nature of the current semantics can be used as a guide. In particular, the qualitative semantics is built up by induction from valuations on the atomic propositions. The measurement semantics must also be built up by induction the atomic propositions.

8. APPENDIX

8.0.1 Polish Spaces

From nLab on Polish Spaces,

> A Polish space is a topological space that's homeomorphic to a separable complete metric space. Every second countable locally compact Hausdorff space is a Polish space, among others.

For those us who have been away from topology for awhile, a dense subset of a topological space is one that intersects every open set. A *separable space* is one with a countable dense subset. This appears to say that the collection of opens is countable, just pick one point from the dense subset in each open. Hence this choice gives us an index set for the opens. This is not quite enough to get a σ-algebra from the opens. The open will have countable unions, but still might lack countable intersections.

From nLab on Complete Space

> A space (with space taken in a sense relevant to the field of topology) is complete (or Cauchy-complete) if every sequence, net, or filter that should converge really does converge. We identify the sequences, nets, or filters that should converge as the Cauchy ones.
>
> A space that is not complete has gaps that may be filled to form its completion; it is rather natural to make the space (or equivalently its underlying topological space) Hausdorff at the same time. Forming the completion of a Hausdorff space is an important example of completion in the general abstract sense.

From nLab on Second Countable,

> A space (such as a topological space) is second-countable if, in a certain sense, there is only a countable amount of information globally in its topology. (Change globally to locally to get a first-countable space.)
>
> A topological space is second-countable if it has a base for its topology consisting of a countable set of subsets.
>
> A locale is second-countable if there is a countable set B of open subspaces (elements of the frame of opens) such that every open G is a join of some subset of B. That is, we have
>
> $$G = \bigvee \{U : B \mid U \subseteq G\}.$$

From nLab on Locally Compact Topological Space,

> A topological space is locally compact if every point has a neighborhood base consisting of compact subspaces. This means that for every point $x \in X$ every open neighbourhood $U_x \supset \{x\}$ contains a compact neighbourhood $K_x \subset U_x$.

They go one to note

> Every discrete space is locally compact.
>
> (open subspaces of compact Hausdorff spaces are locally compact)
>
> Every open topological subspace $X \underset{\text{open}}{\subset} K$ of a compact Hausdorff space K is a locally compact topological space.
>
> In particular every compact Hausdorff space itself is locally compact.
>
> Conversely, every locally compact Hausdorff space X arises in this way, since it can be considered an open subspace in its one-point compactification $X \sqcup \{\infty\}$.

The following is apropos from nLab

> Spaces of structures and models (in the model theory sense), and spaces of n-types? (again in the model theory sense), quite often provide examples of Polish spaces. For example, if L is a countable language (a countable signature), then the collection of possible L-structures M on the countable universe $\mathbb{N}$, topologized by taking as basic opens
>
> $$U_\phi = \{M \in \mathrm{Struct}(L) : M \models \phi\}$$
>
> where ϕ is a quantifier-free sentence, is a Polish space homeomorphic to the product space
>
> $$\prod_{relations\ R} \mathbf{2}^{\mathbb{N}^{arity(R)}} \times \prod_{functions\ f} \mathbb{N}^{\mathbb{N}^{arity(f)}}$$
>
> (taking constants to be functions of arity 0 in the signature).

If I understand this correctly, the structures for any countable modal language will form a Polish space. A structure here is a maximal filter, it either satisfies or fails to satisfy any formula. This is not giving any indication of a metric on a space the collection of structures to which it may be homeomorphic.

From the nCategory Cafe (the page no longer seems to exist)

> Why are Polish spaces not very big? In other words, why are there none with cardinality exceeding the continuum? It is because any Polish space has a countable dense subset and you can write any point as a limit of a sequence of points in this subset. So, you only need a sequence of integers to specify any point in a Polish space.

8.0.2 Analytic Spaces

Analytic spaces can be seen as extensions of Polish spaces.

Proposition 8.0.1 ([2]) *Let X be a Polish space. Then the following statements are equivalent for $A \subseteq X$:*

a. A is analytic.

b. There exists a Polish space Y and a Borel subset $B \subseteq X \times Y$ with $A = \pi_X[B]$.

c. There exists a continuous map $f : \mathbb{N}^\infty \to X$ with $f[\mathbb{N}^\infty[= A$.

d. $A = \pi_X[C]$ for a closed subset $C \subseteq X \times \mathbb{N}^\infty$.

8.0.3 Distributed Logic

From [10], the axiom set for the base Simulation Logic is

Graph

S1. A directed graph $\mathfrak{G}$ of nodes and arrows

S2. An endo-arrow i_{h} for each node in $\mathfrak{G}$

Axiom Schemes A: For each node in $\mathsf{h} \in \mathfrak{G}$, and endo-arrow i_{h} at h,

A1. all truth functional theorems of a propositional logic at h

A2. Normal modal axioms for a local logic at h

A3. $P \overset{\mathsf{h}}{\equiv} [i^{\square}_{\mathsf{h}}\rangle P$

Axiom Schemes B: For each arrow $f : \mathsf{h} \rightharpoonup \mathsf{k}$ and $g : \mathsf{k} \rightharpoonup \mathsf{l}$ in $\mathfrak{G}$,

B1. $[f^{\square}\rangle\,[g^{\square}\rangle V \overset{\mathsf{h}}{\equiv} [f \circ g^{\square}\rangle V$

B2. $[g^{\square}\cdot\rangle\,[f^{\square}\cdot\rangle P \overset{\mathsf{h}}{\equiv} [f \circ g^{\square}\cdot\rangle P$

The following Axioms are optional and we do not assume them. We add them here to give a flavor of what is possible.

Axiom Schemes D: To force arrows to be functions, use the following **D** axioms, just as they would in any normal modal logic systems. We do not assume these axioms hold in the sequel:

D1. $[f^{\square}\rangle Q \overset{\mathsf{h}}{\supset} [f^{\diamond}\rangle Q$

D2. $[f^{\diamond}\rangle Q \overset{\mathsf{h}}{\supset} [f^{\square}\rangle Q$

Axiom Schemes E: The axiom E1 is only necessary if you wish the classical proposition logic at $\mathrm{Log}(\mathsf{h})$ to be included in the logic at $\mathrm{Log}(\mathsf{k})$. It is not strictly necessary although it does pick up the clause in the definition of simulation [11] requiring this of a simulation. We do not assume these axioms hold in the sequel:

For all propositional letters p,

E1. $p \overset{\mathsf{h}}{\supset} [f^{\square}\rangle p$

E2. $p \overset{\mathsf{k}}{\supset} [f^{\square}\cdot\rangle p$

We assume each locality h has at least a modal relation $\mathcal{H}$ to interpret formulas at $\mathrm{Log}(\mathsf{h})$.

Definitions and Rules (Normal Systems)

Definition of Possibility: $[m^{\diamond}\rangle P \overset{def}{=} \neg[m^{\Box}\rangle\neg P, \quad m \in \{h, k, f\}$

Local Rules: For a local logic at h,

$$\frac{\overset{\mathsf{h}}{\vdash} P \qquad \overset{\mathsf{h}}{\vdash} P \supset R}{\overset{\mathsf{h}}{\vdash} R} \qquad\qquad \frac{\overset{\mathsf{h}}{\vdash} (P_1 \wedge \cdots \wedge P_n) \supset P}{\overset{\mathsf{h}}{\vdash} ([h^{\Box}\rangle P_1 \wedge \cdots \wedge [k^{\Box}\rangle P_n) \supset [k^{\Box}\rangle P}$$

The second rule is a bit redundant with respect to the normal modal logic axioms A2 since it will force any local logic to be normal.

Distributed Rules For each $f : \mathsf{h} \rightharpoonup \mathsf{k}$ arrow in $\mathfrak{G}$,

$$\frac{\overset{\mathsf{k}}{\vdash} (Q_1 \wedge \cdots \wedge Q_n) \supset Q}{\overset{\mathsf{h}}{\vdash} ([f^{\Box}\rangle Q_1 \wedge \cdots \wedge [f^{\Box}\rangle Q_n) \supset [f^{\Box}\rangle Q} \qquad\qquad \frac{\overset{\mathsf{h}}{\vdash} (P_1 \wedge \cdots \wedge P_n) \supset P}{\overset{\mathsf{k}}{\vdash} ([f^{\Box\cdot}\rangle P_1 \wedge \cdots \wedge [f^{\Box\cdot}\rangle P_n) \supset [f^{\Box\cdot}\rangle P}$$

REFERENCES

1. E. E. Doberkat, *Stochastic Relations: Foundations for Markov Transition Systems* (Chapman and Hall/CRC Studies in Informatics Series, 2007).
2. E. E. Doberkat, *Stochastic Coalgebraic Logic* (Springer-Verlag, 2010).
3. J. Barwise and J. Seligman, *Information Flow: The Logic of Distributed Systems* (CUP, 1997). Cambridge Tracts in Theoretical Computer Science 44.
4. P. Panangaden, “Probabilistic Relations,” from Panangaden’s website (1998).
5. P. Panangaden, “The Category of Markov Kernels,” *Electronic Notes in Theoretical Computer Science* **22**, 171–187 (1999).
6. M. Giry, “A Categorical Approach to Probability Theory,” Proceedings of the LNM 915, Categorical Aspects of Topology and Analysis, Carleton University, Ottawa 1981, 1982, pp. 68–85.
7. P. Billingsley, *Probability and Measure* (Wiley & Sons, 2012).
8. K. R. Parthasarathy, ed., *Probability Distributions on Metric Spaces* (AMS Chellsea Publishing, 1967). 2005 Edition.
9. H. L. Royden, *Real Analysis*, third edition ed. (Macmillan, 1964).
10. G. Allwein, W. L. Harrison, and D. Andrews, “Simulation logic,” *Logic and Logical Philosophy* **23**(3), 277–299 (2014).
11. P. Blackburn, M. de Rijke, and Y. Venema, *Modal Logic* (Cambridge University Press, 2001). Cambridge Tracts in Theoretical Computer Science, No. 53.