THE PENNSYLVANIA STATE UNIVERSITY
SCHREYER HONORS COLLEGE

DEPARTMENT OF MATHEMATICS

A TOPOLOGICAL APPROACH TO SELECTION RULE STOCHASTICITY

JACK PIAZZA
SPRING 2023

A thesis
submitted in partial fulfillment
of the requirements
for a baccalaureate degree
in Mathematics
with honors in Mathematics

Reviewed and approved* by the following:

Jan Reimann
Associate Professor of Mathematics
Thesis Supervisor

Sergei Tabachnikov
Professor of Mathematics
Honors Adviser

*Signatures are on file in the Schreyer Honors College.

Abstract

Many classical notions of algorithmic randomness are defined using computable selection rules, partial functions from 2^ω to itself. While several results are known about these randomness notions, there is no general framework allowing one to study them in a unified way. In this thesis, we introduce a method for using a class of selection rules on a set X to turn X into an Alexandroff topological space. We will first explore point-set topological properties of these spaces, such as connectedness and compactness. We will then investigate homotopy in these spaces and prove a sufficient condition for two of them to be homotopy equivalent. Finally, we will discuss potential applications to classical definitions of stochasticity for infinite binary sequences.

Table of Contents

List of Figures

Acknowledgements

My deepest gratitude goes to Dr. Jan Reimann for his constant support and guidance. In addition to teaching me mathematical logic for the past three years, Dr. Reimann introduced me to the logic group at Penn State and gave me lots of invaluable advice, both related to math and outside of it. His mentorship made my academic experience at Penn State much more meaningful than it would have been otherwise.

I would also like to thank the Goldwater Foundation and Penn State for providing financial support which allowed me to conduct the research in this thesis.

Finally, thank you to my family for their love and support, and for encouraging me to pursue math at the undergraduate level and beyond.

Chapter 1

Introduction and Background

1.1 Introduction

Let X be a set and $\mathcal{C}$ a collection of partial functions from X to itself. For any $x \in X$, we can consider the set

$$V(x) = \{f(x) : f \in \mathcal{C} \text{ and } f(x) \text{ is defined}\}$$

Informally, $V(x)$ can be thought of as the set of points that are "reachable" in one step by an application of a function in $\mathcal{C}$. The following are some nice properties that we might want these sets to have:

- Any point x is reachable from itself.
- If y can be reached from x in finitely many steps, it can be reached from x in one step.

To ensure that both of these hold, it suffices to require that $\mathcal{C}$ is closed under function composition and contains the identity function $\mathrm{id} : X \to X$. When this is the case, we can define a subset $Y \subseteq X$ to be "inescapable" if no point outside Y is reachable from a point inside Y. It turns out that in this scenario, the collection of inescapable sets is closed under unions and intersections. In fact, they form a certain kind of topology on X, known as an *Alexandroff topology*, where open sets are closed under arbitrary intersections rather than just finite ones.

In computability theory, there are plenty of instances of classes of partial functions from 2^ω to itself that satisfy the properties above. Among these are Turing functionals, computable subsequences, and Church selection rules, all of which will be defined later in this chapter. Using these examples, we can obtain many different Alexandroff topologies on 2^ω, all of which are rooted in computability theory but behave in very different ways.

The goal of this thesis is to define and explore these topological spaces and their properties. In this chapter, we will briefly establish some notation and background in computability theory and the topology of Alexandroff spaces. The reader should have some familiarity with basic concepts in computability theory (at the level of Part I of [9]), but nothing about the topology of Cantor space or algorithmic randomness will be assumed. For topology, only point-set topology and some basic algebraic topology (at the level of the first two chapters of [3]) will be assumed.

In Chapter 2, we will define the selection rule topology and provide characterizations of topological concepts such as interiors and closures, continuous functions, and bases in these spaces. We will then analyze their point-set topological properties, including compactness and connectedness. Finally, we will discuss various notions for what it means for a subset of one of these spaces to be "small" or "large."

In Chapter 3, we will prove the existence or non-existence of homotopy equivalences between many of these spaces using a criterion for homotopy equivalence in Alexandroff spaces. The most involved of these proofs will be the construction of a homotopy equivalence between the topological spaces associated with two classical concepts of randomness, Church and MWC stochasticity.

Finally, in Chapter 4 we will discuss implications of these results and potential directions for future work.

1.2 Notation

Let 2^ω and $2^{<\omega}$ be the sets of infinite and finite strings over $\{0,1\}$, and let $2^{\leq\omega}$ be their union. We will write $f :\subseteq X \to Y$ to indicate that f is a partial function from X to Y. Unless otherwise specified, a partial computable function from $\mathbb{N}$ or $2^{<\omega}$ to $\{0,1\}$ will simply be referred to as *computable*. If it is total it will specifically be referred to as *total computable*. We will fix an enumeration $\{\Phi_e\}_{e\in\mathbb{N}}$ of Turing programs and partial computable functions. The notation $\Phi_{e,s}(x)\downarrow$ or $\Phi_{e,s}(x)\uparrow$ means that the e'th Turing program halts or does not halt on input x in s steps. Similarly, $\Phi_e(x)\downarrow$ or $\Phi_e(x)\uparrow$ means that the e'th Turing program halts or does not halt on input x in infinitely many steps.

1.3 Computability Background

1.3.1 Cantor Space

Let $\sigma \in 2^{\leq\omega}$ be a string. We will denote bits of σ using parentheses, so that σ equals $\sigma(0)\ldots\sigma(n)$ if it is finite and $\sigma(0)\sigma(1)\ldots$ if it is infinite. The *length* of σ, denoted $|\sigma|$, is the number of bits it contains if it is finite or ∞ if it is infinite. If σ is finite and τ is any string, their *concatenation* $\sigma\tau$ consists of the bits of σ followed by the bits of τ. σ is an *initial segment* of τ, written $\sigma \preccurlyeq \tau$ if $\tau = \sigma\rho$ for some $\rho \in 2^{\leq\omega}$, which could possibly be empty or infinite. If $|\tau| \geq n$, the initial segment of τ consisting of n bits is denoted $\tau|_n$. A sequence of strings (τ_n) is said to *converge* to an infinite string τ if for all $m \in \mathbb{N}$, there is some $N \in \mathbb{N}$ such that for all $n \geq N$, $\tau_n(m)$ exists and equals $\tau(m)$. On the other hand, (τ_n) converges to a finite string σ if there is some $N \in \mathbb{N}$ such that for all $n \geq N$, $\tau_n = \sigma$. σ and τ are said to be *compatible* if one is an initial segment of the other. Finally, the *join* of two infinite strings σ and τ is given by $\sigma \oplus \tau = \sigma(0)\tau(0)\sigma(1)\tau(1)\ldots$. Now we will define the standard topology on 2^ω.

Definition 1.3.1. For $\sigma \in 2^{<\omega}$, the *cylinder* of σ is the set

$$[\sigma] = \{\tau \in 2^\omega : \sigma \preccurlyeq \tau\}$$

Cantor space is the set 2^ω endowed with the topology $\mathcal{C}$ whose open sets are unions of cylinders.

To ensure that this is a topological space, we need to show that the cylinder sets form a basis. It suffices to verify that cylinders are closed under finite intersections and that 2^ω is a union of cylinders. The former claim follows because $[\sigma] \cap [\tau]$ is either $[\sigma]$ or $[\tau]$ if the two strings are compatible, or $\emptyset$ if they are not. The latter follows because $2^\omega = [\epsilon]$, where ϵ is the empty string.

Many properties of Cantor space are summarized by the following proposition. For a proof, see Theorem 69 of [7].

Proposition 1.3.2. $(2^\omega, \mathcal{C})$ *is the unique non-empty, perfect, compact, totally disconnected, metrizable topological space up to homeomorphism.*

One standard metric on 2^ω which induces the topology of Cantor space is

$$d(\sigma,\tau) = \begin{cases} 0 & \sigma = \tau \\ 2^{-n} & \sigma \neq \tau, \text{ where } n = \min\{i \in \mathbb{N} : \sigma(i) \neq \tau(i)\} \end{cases}$$

1.3.2 Selection Rule Stochasticity

Definition 1.3.3. A computable function $r :\subseteq 2^{<\omega} \to \{0,1\}$ is called a *von Mises-Wald-Church (MWC) selector*. If it is total, it is called a *Church selector*.

We can view an MWC or Church selector r as inducing a partial function $s_r :\subseteq 2^{\leq\omega} \to 2^{\leq\omega}$ in the following sense:

Construction: Let r be an MWC or Church selector and $\sigma \in 2^{\leq\omega}$. We will construct a subsequence $s_r(\sigma)$ of σ in stages. At stage 0, let

$$s_r^0(\sigma) = \begin{cases} \sigma(0) & r(\epsilon) = 1 \\ \epsilon & r(\epsilon) = 0 \end{cases}$$

if $r(\epsilon)\downarrow$, and we will say the construction diverges if $r(\epsilon)\uparrow$. More generally, at stage $n+1$ suppose we have constructed a subsequence $s_r^n(\sigma)$ of σ. Then we let

$$s_r^{n+1}(\sigma) = \begin{cases} s_r^n(\sigma)\sigma(n) & r(\sigma|_n) = 1 \\ s_r^{n+1}(\sigma) & r(\sigma|_n) = 0 \text{ or } |\sigma| < n \end{cases}$$

and we will say the construction diverges if $r(\sigma|_n)\uparrow$. If the construction never diverges at any stage, we will obtain a sequence $s_r^n(\sigma)$ with $s_r^n(\sigma) \preccurlyeq s_r^{n+1}(\sigma)$ for each n. This ensures that the sequence $(s_r^n(\sigma))$ converges to either a finite or infinite string, so we let

$$s_r(\sigma) = \lim_{n\to\infty} s_r^n(\sigma)$$

□

We will also call the map s_r an *MWC/Church selection rule*. From this perspective, we can view r as a function which indicates whether or not we should select the bit $\sigma(n)$ from σ, given that we have read the previous n bits. Note that if r is a Church selector then every computation in the above construction halts, so s_r is total. Moreover, if r is computable then s_r is as well.

Let MWC and Church denote the classes of MWC and Church selection rules, respectively. In the next proposition we will prove some basic properties of MWC and Church selection rules.

Proposition 1.3.4. *With the operation of function composition,* MWC *and* Church *have a monoid structure. In other words,*

(i) The identity function $\mathrm{id} : 2^{\leq\omega} \to 2^{\leq\omega}$ *is in* MWC

(ii) If $s_1, s_2 \in \mathrm{MWC}$*, then* $s_2 \circ s_1 \in \mathrm{MWC}$

and similarly for Church.

Proof. To see that the identity function is in both classes, consider the map $r : 2^{<\omega} \to \{0,1\}$ given by $r(\sigma) = 1$ for all σ. r is clearly total computable, so it is an MWC and a Church selector. Looking at our construction, the selection rule it generates always appends the next bit of its argument to the subsequence, regardless of the previous bits. Therefore, $s_r(\sigma) = \sigma$ for all $\sigma \in 2^{\leq\omega}$, so $s_r = \mathrm{id}$ is an MWC and Church selection rule.

For closure under composition, suppose first that s_1 and s_2 are MWC selection rules generated by selectors r_1 and r_2. Define a selector r by

$$r(\sigma) = r_2(s_1(\sigma)) \wedge r_1(\sigma)$$

for $\sigma \in 2^{<\omega}$, where $\wedge$ is the standard "AND" operation on the Boolean algebra $\{0, 1\}$. To see that $s_r = s_2 \circ s_1$, consider what it means for a bit $\sigma(n)$ from a string σ to be selected by $s_2 \circ s_1$. It must first be selected by s_1, which means $r_1(\sigma|_n)$ must be 1. It then must be selected by s_2, which means r_2 must output 1 on the set of bits that have been selected so far by s_1. This set is precisely $s_1(\sigma|_n)$, so we have

$$s_2 \circ s_1(\sigma) \text{ selects } \sigma(n) \iff r_1(\sigma|_n) = 1 \wedge r_2(s_1(\sigma|_n)) = 1 \iff r(\sigma|_n) = 1$$

as desired. Therefore, MWC selection rules are closed under composition. If r_1 and r_2 are total computable then so is r, which shows that Church selection rules are closed under composition as well. □

The main historical applications of these selection rules are MWC and Church stochasticity, classical notions of randomness first described by von Mises in [11].

Definition 1.3.5. Let the subset of *balanced strings* in 2^ω be given by

$$H = \left\{ \sigma \in 2^\omega : \lim_{N\to\infty} \sum_{i=0}^{N} \frac{\sigma(i)}{N} = \frac{1}{2} \right\}$$

A string $\sigma \in 2^\omega$ is said to be *MWC (resp. Church) stochastic* if for all MWC (resp. Church) selection rules s, $s(\sigma) \in H$.

Intuitively, a string is stochastic with respect to one of these definitions if any partial or total algorithm which tries to predict bits of σ having seen the preceding bits performs no better than random chance. It is immediate from the definition that any string which is MWC stochastic is also Church stochastic. However, both of these are still relatively weak forms of randomness; for example, Shen showed in [8] that they are strictly weaker than Martin-Lof randomness.

1.3.3 Generalized Selection Rules

We now want to generalize classes of functions with the properties observed in Proposition 1.3.4.

Definition 1.3.6. A *selection rule* on a set X is a partial function $s :\subseteq X \to X$. If s and s' are selection rules, their *composition* is the partial function $s' \circ s$ such that $s' \circ s(x) = s'(s(x))$ whenever x is in the domain of s and $s(x)$ is in the domain of s', and $s' \circ s(x)$ is undefined otherwise. A class S of selection rules is *monoidal* if

(i) The identity function $\mathrm{id} : X \to X$ is in S.

(ii) S is closed under composition.

We will almost always take X to be $2^{\leq\omega}$ or 2^ω in our examples, but many results about monoidal classes of selection rules will be proved more generally for an arbitrary set X.

We showed above that MWC and Church are monoidal classes of selection rules. We will now present several more examples of monoidal classes, categorized into four groups.

Trivial classes of selection rules

First, we will consider three classes of maps which can immediately be seen to be monoidal.

Definition 1.3.7. (i) Let Id be the class containing only the identity function id.

(ii) Let Fun be the class of all partial functions $f :\subseteq 2^\omega \to 2^\omega$.

(iii) Let Shift be the class of all the *shift maps*,

$$\mathrm{Shift} = \{L_i : i \geq 0\}$$

where $L_i : 2^\omega \to 2^\omega$ is given by $L_i(\sigma) = \sigma(i)\sigma(i+1)\dots$

The first two of these are clearly monoidal. For Shift, we use the fact that $L_i \circ L_j = L_{i+j}$ and $L_0 = \mathrm{id}$.

Subsequential classes of selection rules

Church and MWC selection rules can be thought of as "smart" or "adaptive" ways to select subsequences from a sequence, since they can use information about a prefix of a string before deciding whether to include another bit. In contrast, one could also consider oblivious selection rules, which must decide whether or not to select the bit at a certain position independent of the bits in previous positions.

Definition 1.3.8. (i) For every infinite subset

$$X = \{x_0 < x_1 < \dots\}$$

of $\mathbb{N}$, we can define a unique function f_X from 2^ω to itself given by

$$f_X(\sigma) = \sigma(x_0)\sigma(x_1)\dots$$

Denote the class of functions f_X, where X ranges over all infinite subsets of $\mathbb{N}$, by Sub.

(ii) In the previous definition, if we restrict X to be a computable infinite set, we obtain the class of selection rules $\mathrm{Sub[C]}$.

(iii) We can also restrict the set X by its density in $\mathbb{N}$. Let the *upper density* $\rho^+(X)$ and the *lower density* $\rho^-(X)$ of X be

$$\rho^+(X) = \limsup_{n\to\infty} \frac{1}{n}\#\{0 \leq i \leq n-1 : i \in X\}$$
$$\rho^-(X) = \liminf_{n\to\infty} \frac{1}{n}\#\{0 \leq i \leq n-1 : i \in X\}$$

If these two are equal, we define the *density* $\rho(X)$ of X to be their common value. If we restrict the sets in (i) to those which have positive density or positive lower density, we obtain the classes $\mathrm{Sub[D]}$ and $\mathrm{Sub[LD]}$ of selection rules.

(iv) If we require that our sets X be both computable and have positive density or lower density, we obtain the classes $\mathrm{Sub[C][D]}$ and $\mathrm{Sub[C][LD]}$.

All of these classes contain id since $\mathrm{id} = f_{\mathbb{N}}$, and $\mathbb{N}$ is infinite, computable, and has positive density and lower density. Thus, to see that they are monoidal, we must study the composition of two functions f_X and f_Y. The next definition was given by Miller in [6].

Definition 1.3.9. Let

$$A = \{a_0 < a_1 < \dots\}$$
$$B = \{b_0 < b_1 < \dots\}$$

be two infinite subsets of $\mathbb{N}$. Let the set $B \triangleright A$, or B into A, be

$$B \triangleright A = \{a_{b_0} < a_{b_1} < \dots\}$$

From the definition, we can see that $f_B \circ f_A = f_{B \triangleright A}$. Therefore, to show that the above classes are monoidal, it suffices to show that A and B being infinite, computable, etc. implies that $B \triangleright A$ is as well.

Proposition 1.3.10. *Suppose A and B are infinite sets which both satisfy one of the following properties. Then $B \triangleright A$ is infinite and satisfies this property as well.*

(i) Computable

(ii) Positive lower density

(iii) Positive density

Proof. That $B \triangleright A$ is infinite is immediate from the definition.

Suppose that both A and B are computable. To compute the n'th element of $B \triangleright A$, a Turing machine can simply compute the n'th element b_n of B, then compute the b_n'th element of A, so $B \triangleright A$ is computable.

Next suppose that $\rho^-(A) = a$ and $\rho^-(B) = b$ are both positive. Then for all $\varepsilon > 0$, there exists an N such that for all $n \geq N$,

$$\frac{1}{n}\#\{0 \leq i \leq n-1 : i \in A\} \geq a - \varepsilon$$

and

$$\frac{1}{n}\#\{0 \leq i \leq n-1 : i \in B\} \geq b - \varepsilon$$

Choose some $K \geq a_{b_{N-1}}$. Then for all $k \geq K$, we have

$$\frac{1}{k}\#\{0 \leq i \leq k-1 : i \in B \triangleright A\} = \frac{1}{k}\#\{0 \leq i \leq k-1 : a_{b_i} \leq k\} \geq (a-\varepsilon)(b-\varepsilon)$$

For ε sufficiently small the right-hand side is positive, so $\rho^-(B \triangleright A) > 0$ as desired. If we are considering density instead of lower density, we can repeat a similar argument with the lower and upper densities to conclude that $\rho(B \triangleright A) = \rho(B)\rho(A) > 0$.

□

Reduction-based classes of selection rules

The final class of selection rules we will consider come from standard definitions of computability theory. Recall that the computable reductions $\leq_T$, $\leq_1$, $\leq_m$, $\leq_{wtt}$, and $\leq_{tt}$ can be defined by letting $x \leq y$ if there is a computable function $f : 2^\omega \to 2^\omega$ satisfying some additional property such that $f(y) = x$. This motivates the following definition.

Definition 1.3.11. (i) Recall that we can view an oracle Turing machine as a partial function from 2^ω to itself. We can also identify each Φ_e with the corresponding oracle machine $\Phi_e^\emptyset$ with empty tape; more details about this viewpoint can be found in Section 2.4.1 of [2]. These functionals are used to define Turing reducibility, and they form a class Turing of selection rules.

(ii) We can similarly define the class WTT by requiring that each functional has computably bounded use.

(iii) Many-one, one-one, and truth table reductions are defined using a computable function which queries some fixed finite set of bits of one string to determine a single bit of another string. If we fix the computable function and apply it to all elements of 2^ω, we obtain a computable map from 2^ω to itself. Thus, these reductions can be used to define classes Many, One, and TT of selection rules.

The fact that each of these classes of selection rules is monoidal essentially follows from the proofs that the partial orders induced by these reductions are transitive and reflexive.

1.4 Topology Background

Our eventual goal will be to use a monoidal class of selection rules on a set X to construct a topology on X. This topology will satisfy a strong additional property called the *Alexandroff* property. In this section, we will define Alexandroff spaces and collect several results we will need about them. For a more complete introduction to these spaces, see [4].

1.4.1 Alexandroff Spaces

An Alexandroff space is a special kind of topological space which has a lot in common with a topology defined on a finite set.

Definition 1.4.1. A topological space $(X, \mathcal{T})$ is

(i) *Alexandroff* if $\mathcal{T}$ is closed under arbitrary intersections, not just finite ones.

(ii) an *A-space* if it is Alexandroff and T_0.

It follows immediately from the definition that in an Alexandroff space, arbitrary unions of closed sets are closed. This means that open and closed sets can be regarded as equivalent in an Alexandroff space. That is, any theorem about open and closed sets will also hold if we replace the word "open" by "closed" and the word "closed" by "open". We will now explore some of the consequences of this.

Let $(X, \mathcal{T})$ be an Alexandroff space. Each element $x \in X$ has a smallest open neighborhood $V(x)$ obtained by taking the intersection of all of its open neighborhoods. Moreover, each subset Y of X has a smallest open set containing it given by

$$V(Y) = \bigcup_{y \in Y} V(y)$$

The sets $V(x)$ form a basis for the topology since $U = V(U)$ when U is open.

One of the most important properties of an Alexandroff space X is that its topology can be captured by viewing X as a set with a preorder. Define this preorder by letting $y \leq x$ if and only if $V(y) \subseteq V(x)$. This is equivalent to the *specialization preorder* that one can put on any topological space by letting $x \leq y$ if and only if $y \in \overline{\{x\}}$. In the case of an Alexandroff space, the set of elements below an element $x \in X$ is precisely $V(x)$, so a set is open if and only if it is closed downward in the preorder.

We say that two elements x and y in X are *similar* if $x \leq y$ and $y \leq x$ or, equivalently, if $V(x) = V(y)$. By taking the quotient by the equivalence relation $\sim$ which relates similar elements, we obtain a space $K(X) = X/\sim$ called the *Kolmogorov quotient* of X. $K(X)$ is an A-space, as all topologically indistinguishable elements in X are identified in $K(X)$. Moreover, $K(X)$ is still an Alexandroff space and it can be shown that $K(X)$ is homotopy equivalent to X.

The specialization preorder gives a new characterization of continuous maps and homeomorphisms. A map $f : X \to Y$, where X and Y are Alexandroff, is continuous if and only if it is monotone, i.e. if $x_1 \leq x_2 \implies f(x_1) \leq f(x_2)$. It is a homeomorphism if and only if it is an order isomorphism, i.e. it is a bijection and $x_1 \leq x_2 \iff f(x_1) \leq f(x_2)$.

1.4.2 Homotopy Equivalence for Alexandroff Spaces

Next we will describe criteria for when two Alexandroff spaces are homotopy equivalent. First, we will discuss paths in an Alexandroff space.

Lemma 1.4.2. *Let X be an Alexandroff space. If $x, y \in X$ with $x \leq y$, then there is a path $f : [0, 1] \to X$ with $f(0) = x$ and $f(1) = y$.*

Proof. The path defined by $f(t) = x$ for $t < 1$ and $f(1) = y$ is continuous. □

Proposition 1.4.3. *Suppose x_0, x_n are elements of an Alexandroff space X such that there are elements $x_1, \ldots, x_{n-1} \in X$ satisfying $x_i \leq x_{i+1}$ or $x_i \geq x_{i+1}$ for $i \in \{0, \ldots, n-1\}$. Then there is a path $f : [0, 1] \to X$ with $f(0) = x_0$ and $f(1) = x_n$. We will call a path of this form* simple.

Proof. We can use the previous lemma to create paths from x_i to x_{i+1} for each $i \in \{0, \ldots, n-1\}$, then concatenate these paths. The gluing lemma ensures that the concatenated path is continuous. □

The next result, whose proof can be found in [1], shows that simple paths are particularly important in Alexandroff spaces.

Proposition 1.4.4. *(i) Let X be an Alexandroff space and suppose $f : [0, 1] \to X$ is a path with $f(0) = x$ and $f(1) = y$. Then there exists a simple path $f' : [0, 1] \to X$ which is path homotopic to f.*

(ii) An Alexandroff space is connected if and only if it is path connected.

Recall that a *weak homotopy equivalence* between spaces X and Y is a map $f : X \to Y$ which induces an isomorphism on all homotopy groups. The following theorem shows that Alexandroff spaces have a close relationship with abstract simplicial complexes.

Theorem 1.4.5. *For an Alexandroff space X, let K be an abstract simplicial complex whose vertices are points of X and whose simplices are finite chains in the specialization preorder of X. Then there is a weak homotopy equivalence between X and a geometric realization $|K|$ of K.*

From now on, we will refer to chains in the specialization preorder of an Alexandroff space as "simplices."

One significant benefit of considering simplicial complexes instead of general topological spaces is that there are simple criteria for maps between simplicial complexes to be homotopic. Many of these criteria transfer over to the case of Alexandroff spaces with little modification to the proofs. We will now use these to find a simple criterion for two Alexandroff spaces to be homotopy equivalent.

Definition 1.4.6. Let X_1 and X_2 be Alexandroff spaces. Two maps $f, g : X_1 \to X_2$ are *simplicially close* if for all $x \in X_1$, $f(x)$ and $g(x)$ lie in a simplex σ of X_2. Equivalently, for each $x \in X$, either $f(x) \leq g(x)$ or $g(x) \leq f(x)$.

Proposition 1.4.7. *Simplicially close maps are homotopic.*

Proof. See Proposition 5.5.4 of [4]. □

For our purposes, we will usually be interested in the case where X_1 and X_2 are Alexandroff spaces with the same underlying set, but where one topoplogy is finer than the other.

Theorem 1.4.8. *Let (X, τ) and (X, τ') be Alexandroff spaces with $\tau \subseteq \tau'$. Let $\leq$ and $\leq'$ denote the preorders induced by τ and τ', respectively. Suppose there is a map $f : X \to X$ satisfying the following two criteria:*

(i) For all $x, y \in X$ with $x \leq y$, $f(x) \leq' f(y)$

(ii) For all $x \in X$, either $x \leq' f(x)$ or $f(x) \leq' x$.

Then (X, τ) and (X, τ') are homotopy equivalent.

Proof. The homotopy equivalence will be obtained by considering the maps $f : (X, \tau) \to (X, \tau')$ and $\mathrm{id} : (X, \tau') \to (X, \tau)$. id is automatically continuous since $\tau \subseteq \tau'$, so we must first check that f is continuous. This is equivalent to f being monotone, which is guaranteed by condition (i).

Next we must show that $f \circ \mathrm{id}$ and $\mathrm{id} \circ f$ are homotopic to the identity maps on (X, τ') and (X, τ), respectively. It suffices to show that these two pairs of maps are simplicially close to each other by Proposition 1.4.7. In other words, we must check that (a) $f(\mathrm{id}(x)) \leq' x$ or $f(\mathrm{id}(x)) \geq' x$ for $x \in (X, \tau')$ and (b) $\mathrm{id}(f(x)) \leq x$ or $\mathrm{id}(f(x)) \geq x$ for $x \in (X, \tau)$. Since $\mathrm{id} \circ f$, $f \circ \mathrm{id}$, and f are all the same when viewed as maps from the set X to itself and the preorder on τ is larger than the preorder on τ', (b) is implied by (a). Therefore, we only need that $f(x) \leq' x$ or $f(x) \geq' x$ for each $x \in X$, which is precisely condition (ii). □

Chapter 2

Selection Rule Topologies

2.1 Defining the Selection Rule Topology

Let S be a monoidal class of selection rules on a set X. We will show that such a class induces an Alexandroff topology on X. Moreover, this topology will be induced in a contravariant way; that is, if $S \subseteq S'$ are two monoidal classes of selection rules on X, the topology generated by S will be finer than the topology generated by S'.

For each $x \in X$, define

$$V(x) = \{s(x) : s \in S\}$$

to be the set of *derivatives* of x. For now we will ignore the overlap in notation between the set of derivatives of x and the smallest open set containing x, since we will later show that the two coincide. For a subset $Y \subseteq X$, we will also define the two sets

$$V(Y) = \bigcup_{x \in Y} V(x)$$
$$N(Y) = \{x \in X : V(x) \subseteq Y\}$$

which we will call the *derivatives* and *interior* of Y, respectively.

Proposition 2.1.1. *Let* $Y, Y' \subseteq X$ *with* $Y \subseteq Y'$.

(i) $N(Y) \subseteq Y$

(ii) $N(Y) \subseteq N(Y')$

(iii) $N(N(Y)) = N(Y)$

Proof. First suppose $x \in N(Y)$. Then $V(x) \subseteq Y$. Since S is monoidal is contains the identity function, so $x \in V(x)$. Therefore, $x \in Y$ which proves (i).

For (ii), suppose $x \in N(Y)$. Then $V(x) \subseteq Y$ and so $V(x) \subseteq Y'$, which implies (ii) because then $x \in N(Y')$.

From part (i), we know that $N(N(Y)) \subseteq N(Y)$, so to prove (iii) we only need to show the opposite inclusion. Suppose $x \notin N(N(Y))$. Then $V(x)$ is not a subset of $N(Y)$, so there is some $s \in S$ such that $s(x) \in X \setminus N(Y)$. By the same logic applied to $s(x)$, there is some $s' \in S$ such that $s'(s(x)) \in X \setminus Y$. Since S is monoidal, there is $s'' \in S$ such that $s''(x) = s'(s(x))$. Then we have that $s''(x) \in V(x) \setminus Y$, which implies $x \notin N(X)$. □

This proposition suggests that the (suggestively named) function N behaves like a function which takes a set to its interior in a topological space. In fact, any function satisfying Proposition 2.1.1 can be used to define an Alexandroff space.

Proposition 2.1.2. *Let* $\mathcal{T}$ *be the set of subsets* $Y \subseteq X$ *for which* $N(Y) = Y$. *Then* $(X, \mathcal{T})$ *is an Alexandroff space. We will denote this space by* X_S.

Proof. We will check the conditions for a topological space in turn.

First, observe that $N(\emptyset) \subseteq \emptyset$, so it must equal $\emptyset$. On the other hand, $V(x) \subseteq X$ for all x, so $N(X) = X$.

Now suppose $(Y_i)_{i \in I}$ is a family of sets such that $N(Y_i) = Y_i$ for each $i \in I$. Denote the union $\bigcup_{i \in I} Y_i$ by Y. We want to show that $N(Y) = Y$. By part (i) of 2.1.1 we know that $N(Y) \subseteq Y$, so we only need to show that $Y \subseteq N(Y)$. Suppose $x \in Y$. Then $x \in Y_i$ for some $i \in I$. It follows that $x \in N(Y_i)$ since $Y_i = N(Y_i)$. Then $x \in N(Y)$ since $Y_i \subseteq Y$ by part (ii) of 2.1.1.

Finally, consider the same family of sets, but this time denote their intersection $\bigcap_{i \in I} Y_i$ by Y'. By a similar argument, we know that $N(Y') \subseteq Y'$ so we only need to show that $Y' \subseteq N(Y')$. Suppose $y \in Y'$. Then $y \in Y_i$, and hence $y \in N(Y_i)$, for all $i \in I$. This means that $V(y) \subseteq Y_i$ for all $i \in I$, so $V(y)$ is also a subset of their intersection, Y'. This is exactly the requirement for $y \in N(Y')$, as desired. □

Observe that we did not use part (iii) of Proposition 2.1.1 to show that this is an Alexandroff space. Instead, this part can be used to show that V and N indeed give the smallest open set containing a given set and the interior of a set, respectively.

Proposition 2.1.3. *(i) For all $x \in X_S$, $V(x)$ is the smallest open neighborhood of x.*

(ii) For all $Y \subseteq X_S$, $V(Y)$ is the smallest open set containing Y.

(iii) For all $Y \subseteq X_S$, $N(Y)$ is the largest open set contained in Y.

Proof. First we will prove (i). It is clear that $x \in V(x)$ for each x since the identity function is a selection rule, but it is not immediate that $V(x)$ is open. It suffices to show that $V(x) \subseteq N(V(x))$. Suppose $y \in V(x)$. Then $y = s(x)$ for some $s \in S$. We wish to show that $V(y) \subseteq V(x)$. Indeed, any element of $V(y)$ is of the form $s'(y) = s'(s(x))$, which is in $V(x)$ by the compositionality property of S.

Now let U be any open neighborhood of x. Since $U \subseteq N(U)$ and $x \in U$, we have $x \in N(U)$ so $V(x) \subseteq U$. Therefore, $V(x)$ is the smallest such open neighborhood.

By part (i), any open set containing Y must contain $V(y)$ for each $y \in Y$. $V(Y)$ is open since it is a union of the open sets $V(y)$. Therefore, $V(Y)$ is the smallest open set containing Y, which proves (ii).

Finally, we will prove (iii). From part (iii) of Proposition 2.1.1, we know that $N(Y)$ is open. Suppose that $U \subseteq Y$ is open but not a subset of $N(Y)$. Then there is some $y \in U \setminus N(Y)$, so $V(y)$ is not a subset of Y. However, $V(y) \subseteq U$ since $y \in U$ by the openness of U, which contradicts $U \subseteq Y$. Thus, we must have $U \subseteq N(Y)$ and so $N(Y)$ is maximal among open sets contained in Y. □

Dual to these notions, we can characterize closed subsets of X_S as well as the largest closed set contained in a given set.

Proposition 2.1.4. *(i) A set $Y \subseteq X_S$ is closed if and only if whenever $s(x) \in Y$ for some $s \in S$, $x \in Y$ as well.*

(ii) The closure of $Y \subseteq X_S$ is $\overline{Y} = \{x \in X_S : \exists s \in S\,(s(x) \in Y)\}$.

(iii) The largest closed set contained in $Y \subseteq X_S$ is

$$C(Y) = \{x \in X_S : (\forall y \in X_S)(\forall s \in S)(s(y) = x \implies y \in Y)\}$$

Proof. For (i) observe that Y is closed if and only if $N(X_S \setminus Y) = X_S \setminus Y$ by definition. Thus, closure means for every $x \notin Y$, $s(x) \notin Y$ for all $s \in S$. This is equivalent to saying whenever $s(x)$ is in Y for some $s \in S$, x is in Y as desired.

To prove (ii), recall that $\overline{Y} = X_S \setminus N(X_S \setminus Y)$. Since

$$N(X_S \setminus Y) = \{x \in X_S : s(x) \in X_S \setminus Y \text{ for all } s \in S\}$$

its complement is

$$\overline{Y} = \{x \in X_S : s(x) \in Y \text{ for some } s \in S\}$$

which is what we wanted.

Finally, for (iii), notice that since open and closed sets are interchangeable in Alexandroff spaces, we have $C(Y) = X_S \setminus V(X_S \setminus Y)$. Since $V(X_S \setminus Y)$ is the set of all $s(x)$ where $x \in X_S \setminus Y$, its complement is the set of all z which are not equal to $s(x)$ for any $x \in X_S \setminus Y$ and $s \in S$. In other words, if such a z is equal to $s(y)$ for some y, then we must have $y \in Y$, which is what is written in the proposition. □

2.2 Basic Topological Properties

Now that we have defined an abundance of Alexandroff topological spaces (one for each monoidal collection of selection rules), we want to explore their topological properties. Throughout this section, fix a set X and a monoidal class S of selection rules on X.

Proposition 2.2.1. *If S is countable and X is uncountable, then X_S is not compact or Lindelöf. However, regardless of the cardinality of S or X, $V(x)$ is compact for each x so X_S is locally compact.*

Proof. Consider the cover of X_S by the open sets $V(x)$. Each $V(x)$ is countable since S is countable. Since a countable union of countable sets is countable and X is uncountable, any subcover of this cover must be uncountable. Therefore, X_S is not compact or Lindelöf.

On the other hand, suppose $(U_i)_{i \in I}$ is an open cover of $V(x)$. Then some U_0 must contain x. Since $V(x)$ is the smallest open set containing x, U_0 must contain $V(x)$ and so $\{U_0\}$ is a finite subcover of $V(x)$. □

This proposition applies to the majority of the examples we considered in the previous section, since $2^{\leq\omega}$ and $2^{<\omega}$ are uncountable but any class of computable selection rules (such as Church, MWC, or Turing) is countable.

By Proposition 1.4.4, X_S is path connected (or, equivalently, connected) if and only if every pair of points $x, y \in X_S$ is connected by a simple path. We can use this criterion to classify the connectedness of each of our examples from Chapter 2.

Proposition 2.2.2. *The topologies generated by the classes* Id *and* Shift *are not connected or path connected. Every other class of selection rules discussed in Chapter 2 generates a path connected space.*

Proof. It is clear that Id generates the discrete topology on 2^ω since $V(x) = \{x\}$ for each x, so it fails to be connected.

For Shift, we will show that there is no simple path connecting $x = 000...$ and $y = 111....$ Recall that L_i is the function which shifts a string left by i bits. Observe that if a string z ends with an infinite sequence of 0's or 1's, then any z' with $L_i(z') = z$ or $L_i(z) = z'$ does as well. Therefore, if there is a list $x_0, \ldots, x_n$ with $x = x_0$ and $x_i \leq x_{i+1}$ or $x_i \geq x_{i+1}$ for each i, it follows by induction that x_n ends with infinitely many 0's. Thus, x and y cannot be connected by a simple path.

For every other example in Chapter 2, we can use the same construction to show that X_S is path connected. Indeed, it can be verified that for any two strings $x, y \in 2^\omega$, both x and y can be selected from $x \oplus y$ in all of these examples. Therefore, we have $x \leq x \oplus y \geq y$, which shows that x and y are connected by a simple path. □

Finally, we will look at which separation axioms our selection rule topologies satisfy.

Proposition 2.2.3. *X_S is T_0 if and only if $x \leq y$ and $y \leq x$ implies $x = y$ for all $x, y \in X_S$.*

Proof. First suppose X_S is not T_0. Then there are two elements $x \neq y$ in X which share the same open sets. In particular this implies that $y \in V(x)$ and $x \in V(y)$, so $x \leq y$ and $y \leq x$.

Conversely, suppose there are $x \neq y$ with $x \leq y$ and $y \leq x$. Then $x \in V(y)$ and $y \in V(x)$. If U is any open set containing x, then $V(x) \subseteq U$, so $y \in U$. Similarly, if V is any open set containing y, then $x \in V$, so x and y are topologically indistinguishable and X_S is not T_0.

□

In all of our examples except Id, there are selection rules which can select the string $x = 01010...$ from $y = 10101...$ and vice versa. Therefore, none of these spaces are T_0.

Let us compare this to Cantor space, the usual topological space on the set 2^ω. In Proposition 1.3.2, we saw that Cantor space is totally disconnected, compact, and metrizable (hence Hausdorff). On the other hand, the previous set of propositions show that most natural examples of selection rule topologies are path connected and are not compact or T_0.

2.3 Largeness Properties

2.3.1 Density

Continuing with the theme in Alexandroff spaces that open and closed sets can be regarded as equal, we can formulate alternate notions of properties like density by switching the roles of the two.

Definition 2.3.1. Let X be an Alexandroff space. A set $D \subseteq X$ is *A-dense* if $V(D) = X$.

A similar concept to A-density could be defined in any topological space by declaring a set to be A-dense if the only open set containing it is the entire space X. Unfortunately, in most non-Alexandroff cases this concept serves little purpose. For example, suppose Y is a proper subset of $\mathbb{R}$ (or any Hausdorff space). Then there is some $x \in \mathbb{R} \setminus Y$, and $\mathbb{R} \setminus \{x\}$ is an open set containing Y so Y is not A-dense. Therefore, the only A-dense set in $\mathbb{R}$ is $\mathbb{R}$ itself.

On the other hand, in Alexandroff spaces there are many nontrivial A-dense sets. For example, the set H of half-density strings is A-dense in 2^ω_{Church}. To see this, let $\overline{\sigma}$ denote the string whose bits differ from the bits of σ at every position. Then note that $\sigma \oplus \overline{\sigma}$ is in H for all σ, and $\sigma \leq \sigma \oplus \overline{\sigma}$. Therefore, $V(H) = 2^\omega$.

There is a loose connection between A-dense sets and subsets of $\mathbb{R}$ with full Lebesgue outer measure. Recall that the Lebesgue outer measure of a subset Y of $\mathbb{R}$ is the infimum of the measures of open sets containing Y. Therefore, a set has full Lebesgue outer measure if and only if every open set containing it has full measure. On the other hand, a set is A-dense if the only open set containing it is the space itself.

Taking inspiration from the definition of Church and MWC stochastics, we can define stochastics with respect to any subset D of a selection rule space X_S. We will think of D as the set of elements which satisfy a certain "generic" property (such as, in the case of Church and MWC stochasticity, having asymptotically equally many zeroes and ones).

Definition 2.3.2. Let $D \subseteq X_S$. The D-*stochastic* elements of X_S are those in $N(D)$.

2.3.2 Largeness and Smallness Hierarchies

We will now generalize the above discussion to a family of related largeness and smallness concepts. Fix a selection rule topology X_S. The *largeness hierarchy* for sets in X_S is defined as follows.

Definition 2.3.3. A set $D \subseteq X_S$ is

- Σ^0_1 if $V(D) = X_S$.
- Π^0_1 if $\overline{D} = X_S$.
- Σ^0_{n+1} for $n \geq 1$ if $V(D)$ is Π^0_n.
- Π^0_{n+1} for $n \geq 1$ if $\overline{D}$ is Σ^0_n.
- Δ^0_n if it is both Σ^0_n and Π^0_n.

We will also say D is Σ^0_0, Π^0_0, and Δ^0_0 if and only if $D = X_S$.

From this definition we see immediately that D is Σ^0_1 if and only if it is A-dense and Π^0_1 if and only if it is dense. More generally, we can define a non-recursive characterization of Σ^0_n and Π^0_n sets using the closure and V operators alternately. For $D \subseteq X_S$, let $L_1(D) = \overline{V(D)}$ and $L_2(D) = V(\overline{D})$.

Proposition 2.3.4. *A set $D \subseteq X_S$ is Σ^0_n if and only if*

- $L_1^{n/2}(D) = X_S$, *if n is even.*
- $V(L_1^{(n-1)/2}(D)) = X_S$, *if n is odd.*

D is Π^0_n if and only if

- $L_2^{n/2}(D) = X_S$, *if n is even.*

- $\overline{L_2^{(n-1)/2}(D)} = X_S$*, if* n *is odd.*

Proof. This follows immediately using induction on the definition of Σ_n^0 and Π_n^0 sets. □

Corollary 2.3.5. *(i) If* D *lies at any level of the largeness hierarchy and* $D \subseteq D'$*, then* D' *lies at the same level.*

(ii) Any Σ_n^0 *or* Π_n^0 *set, for* $n \geq 0$*, is* Δ_{n+1}^0*.*

Proof. (i) follows from the proposition because both V and the closure operator are monotone, so L_1 and L_2 are monotone as well.

For (ii), first suppose D is Σ_n^0. Let cl denote the closure operator. If n is even, then $\mathrm{cl} \circ V \circ \cdots \circ \mathrm{cl} \circ V(D) = X_S$, where there are $n/2$ copies of each of cl and V. It follows by monotonicity of these operators that

$$\mathrm{cl} \circ V \circ \cdots \circ \mathrm{cl} \circ V(\mathrm{cl}(D)) = X_S$$

where we have added one instance of cl as the innermost operator, which shows that $D \in \Pi_{n+1}^0$. Also,

$$\mathrm{cl} \circ V \circ \cdots \circ \mathrm{cl} \circ V(D) = X_S$$

where there are now $n/2 + 1$ of each operator, which shows $D \in \Sigma_{n+1}^0$. In the other hand, if n is odd then $V \circ \mathrm{cl} \circ \cdots \circ V(D) = X_S$, where there are $(n+1)/2$ copies of V and $(n-1)/2$ copies of cl. Then we have

$$V \circ \mathrm{cl} \circ \cdots \circ V(\mathrm{cl}(D)) = X_S$$

where we have added one cl as the innermost operator, which shows that $D \in \Pi_{n+1}^0$, and

$$V \circ \mathrm{cl} \circ \cdots \circ V(D) = X_S$$

where there are $(n+3)/2$ copies of V and $(n+1)/2$ copies of cl, which shows $D \in \Sigma_{n+1}^0$. Thus, $D \in \Delta_{n+1}^0$. The proof that this holds if $D \in \Pi_n^0$ is similar, but we exchange all instances of cl and V. □

The above corollary shows that our set of largeness properties really does form a hierarchy. However, if we put certain assumptions on X_S, this hierarchy only has a few levels which are nontrivial.

Proposition 2.3.6. *Suppose that for every* $x, y \in X_S$*, there is a* z *such that* $x, y \leq z$*. Then every nonempty* $D \subseteq X_S$ *is in* Π_2^0*. Similarly, if for every* x *and* y *there is a* z *such that* $z \leq x, y$*, then every nonempty* $D \subseteq X_S$ *is in* Σ_2^0*.*

Proof. We will prove the first claim, since the proof of the second is nearly identical. Suppose $D \neq \emptyset$ is in Σ_2^0. Then $V(\overline{D}) = X_S$. Take some $x_0 \in X_S$. For any y, we know there is a z with $x_0, y \leq z$. Since $x_0 \leq z$, $z \in \overline{D}$ by Proposition 2.1.4. Then since $y \leq z$, $y \in V(\overline{D})$ by the definition of V. Since y was arbitrary, this shows that $V(\overline{D}) = X_S$. □

Next we will define the smallness hierarchy in X_S.

Definition 2.3.7. A set $D \subseteq X_S$ is

- σ_1^0 if $N(D) = \emptyset$.
- π_1^0 if $M(D) = \emptyset$.
- σ_{n+1}^0 for $n \geq 1$ if $N(D)$ is π_n^0.
- π_{n+1}^0 for $n \geq 1$ if $C(D)$ is σ_n^0.
- δ_n^0 if it is both σ_n^0 and π_n^0.

We will also say D is σ_0^0, π_0^0, and δ_0^0 if and only if $D = \emptyset$.

The following proposition is immediate from the fact that $N(D) = X_S \setminus \overline{X_S \setminus D}$ and $C(D) = X_S \setminus V(X_S \setminus D)$.

Proposition 2.3.8. *Let $n \geq 0$. $D \subseteq X_S$ is*

- σ_n^0 *if and only if $X_S \setminus D$ is Π_n^0.*
- π_n^0 *if and only if $X_S \setminus D$ is Σ_n^0.*
- δ_n^0 *if and only if $X_S \setminus D$ is Δ_n^0.*

From this and Corollary 2.3.5, it follows that a σ_n^0 or π_n^0 set is δ_{n+1}^0 for $n \geq 0$. Thus, we obtain an infinite hierarchy of smallness properties as well.

Chapter 3

Homotopy Equivalence in Selection Rule Topologies

In this section we will consider the selection rule topology for all of the classes of selection rules described in Chapter 2 and deduce which of them are homotopy equivalent to each other. The main tool we will use is the following version of Theorem 1.4.8.

Theorem 3.0.1. *Let X_S and $X_{S'}$ be two selection rule spaces with $S \subseteq S'$. Suppose there is a map $f : X_{S'} \to X_S$ satisfying the following two criteria:*

(i) If $x, y \in X$ and there is some $s' \in S$ such that $s'(x) = y$, then there is some $s \in S$ such that $s(x) = y$.

(ii) For all $x \in X$, there is some $s \in S$ such that either $s(x) = f(x)$ or $s(f(x)) = x$.

Then X_S and $X_{S'}$ are homotopy equivalent.

Proof. Since $S \subseteq S'$, the basis elements of S' are larger than the basis elements of S, so the topology of X_S is finer than the topology of $X_{S'}$. Then this theorem is just a restatement of Theorem 1.4.8 using the fact that $x \leq y$ in X_S if and only if there is some $s \in S$ with $s(y) = x$ and $x \leq y$ in $X_{S'}$ if and only if there is some $s' \in S'$ with $s'(y) = x$. □

We will call a homotopy of the above form a *good* homotopy equivalence. For each of the examples of monoidal classes of selection rules in Chapter 2, we will restrict the underlying set from $2^{\leq\omega}$ to 2^ω if necessary by restricting the domain of each selection rule to 2^ω and letting s be undefined on input σ if $s(\sigma)$ is a finite string. By doing this, we can consider each class as having the same underlying set, so we can check whether a good homotopy exists for any pair of monoidal classes where one contains the other.

Figure 3.1 illustrates the containment relations we will be considering between the classes in Chapter 2. A line between two classes represents containment, with the larger class being higher on the page. Therefore, the topologies become finer as one reads from top to bottom, starting with the trivial topology 2^ω_{Fun} and ending with the discrete topology 2^ω_{Id}.

For each line in the diagram connecting two classes $S \subseteq S'$, we will classify their relationship into one of three categories:

(i) There is a good homotopy equivalence between the two spaces.

(ii) The spaces are not homotopy equivalent.

(iii) No good homotopy equivalence exists between the two spaces, but they may still be homotopy equivalent.

The proofs of these relationships will proceed in three sections. First, we will analyze the homotopy type of the spaces which are very fine or coarse (close to the top or bottom of Figure 3.1). Next, we will explore the equivalences for classes involving computable subsequences. In the last section, we will construct a good homotopy between 2^ω_{MWC} and $2^\omega_{\mathrm{Church}}$, which will be considerably more complicated than the proofs in the other two sections.

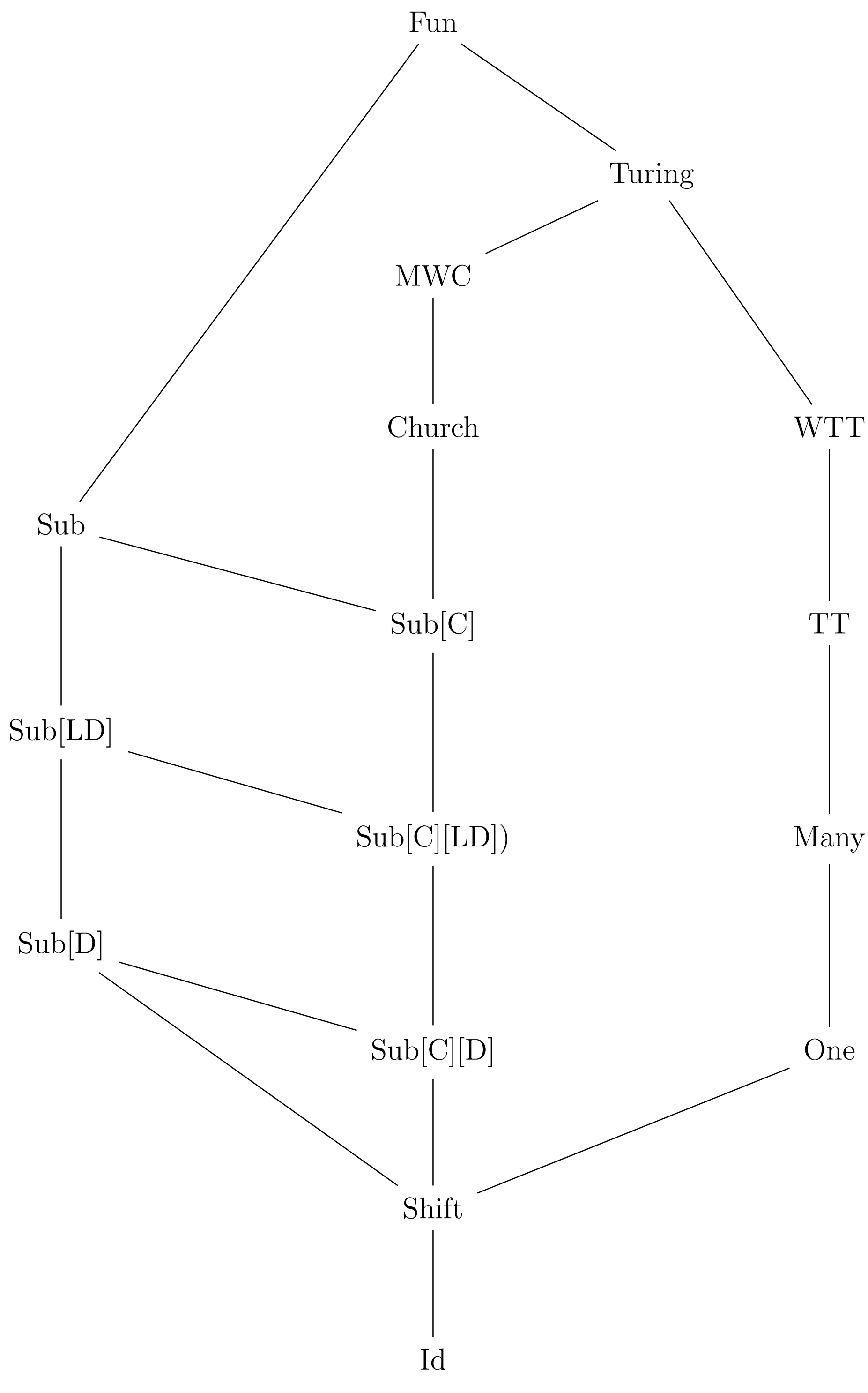

Figure 3.1: Inclusions of classes of selection rules

3.1 Coarse and Fine Topologies

We will later see that homotopy equivalence and related notions can be useful for distinguishing selection rule topologies. However, if the topology is too coarse (which occurs if there are too many selection rules), it becomes less useful. This is encapsulated in the following proposition.

Proposition 3.1.1. *Let S be a monoidal class of selection rules on 2^ω. Suppose there is some x_0 such that for all $y \in 2^\omega$, there is an $s \in S$ such that either $s(x_0) = y$ or $s(y) = x_0$. Then 2^ω_S is homotopy equivalent to 2^ω_{Fun}.*

Proof. We will show that the constant function $f(x) = x_0$ induces a good homotopy equivalence. Continuity is clear since constant functions between topological spaces are always continuous.

Next we must check simplicial closeness. This means that for all $x \in 2^\omega$, there is some $s \in S$ such that either $s(f(x)) = x$ or $s(x) = f(x)$. This is true by assumption since $f(x) = x_0$, so there is a good homotopy equivalence. □

Corollary 3.1.2. *The topologies induced by* Sub, $\mathrm{Sub[LD]}$, $\mathrm{Sub[D]}$, Turing, WTT, TT, *and* Many *are all homotopy equivalent to 2^ω_{Fun}, the trivial topology.*

Proof. For all of the classes of subsequences mentioned in the proposition, the string $x_0 = 1010\ldots$ can be used to select any other string. For Turing, WTT, and TT, the same x_0 can be selected by any string.

For Many, a similar argument works but we must be slightly more careful. First suppose a string x has at least one 0 and one 1. Let e_0 be the position of a 0 and e_1 be the position of a 1. Then the computable function g which sends even numbers to e_1 and odd numbers to e_0 induces a selection rule which sends x to x_0. If x is either $000\ldots$ or $111\ldots$, the constant functions $g(n) = 1$ and $g(n) = 0$ induce reductions from x_0 to x, respectively. Therefore, x_0 still satisfies the condition of the previous proposition. □

Therefore, a finer notion than homotopy equivalence is needed to distinguish the topologies at the top of our diagram. The status of homotopy equivalences between some of the very fine selection rule topologies can also be easily deduced, but for different reasons.

Proposition 3.1.3. *The topologies induced by* Id *and* Shift *are not homotopy equivalent to the topology induced by any other class of selection rules considered in the diagram.*

Proof. Since the zeroth homology group of a space corresponds to the number of path components and homotopy equivalent spaces have the same homology, path connectedness is preserved under homotopy equivalence. The result follows since 2^ω_{Id} and $2^\omega_{\mathrm{Shift}}$ fail to be path connected while the rest of the spaces are, as was shown in Proposition 2.2.2. □

Proposition 3.1.4. *Suppose S is a monoidal class of selection rules containing at least one rule s with a point x_0 such that $s(x_0) \neq x_0$. Then there is no good homotopy equivalence between 2^ω_{Id} and 2^ω_S.*

Proof. Suppose $f : 2^\omega_S \to 2^\omega_{\mathrm{Id}}$ induces a good homotopy equivalence. The simplicial closeness condition means that one of x and $f(x)$ must be able to select the other via the identity selection rule for each x, so $f(x) = x$ for all $x \in 2^\omega$. On the other hand, continuity implies that since $s(x_0)$

can be selected from x_0, we have $\mathrm{id}(f(x_0)) = f(s(x_0))$, or $f(x_0) = f(s(x_0))$. Since f is the identity function, this means $x_0 = s(x_0)$, contradicting our assumption. Thus, no good homotopy equivalence can exist. □

Corollary 3.1.5. *There is no good homotopy equivalence between* 2^ω_{Id} *and* $2^\omega_{\mathrm{Shift}}$.

These results show that while we may or may not expect to find a homotopy equivalence between 2^ω_S and a discrete space for small enough S, we will almost never be able to find a good homotopy equivalence. In other words, the notion of good homotopy equivalence is too strong for spaces which are close to discrete, but not strong enough for spaces which are close to trivial. We can therefore hope that it provides more useful results in the region of Figure 3.1 between these two ends, which is what we will investigate in the following sections.

3.2 Equivalences Between Subsequence Spaces

We will now consider homotopy equivalences involving the classes of selection rules consisting of computable subsequences. We have already seen that when we consider subsequences which are not necessarily computable, we end up with enough selection rules that our space is homotopy equivalent to 2^ω_{Fun}. When we restrict to a countable number of subsequences, however, this no longer holds.

Lemma 3.2.1. *Suppose* $S \subseteq S' \subseteq S''$ *are three monoidal classes of selection rules on a set* X*, and suppose* $f : X_{S''} \to X_S$ *induces a good homotopy equivalence. Then* f *also induces a good homotopy equivalence when viewed as a map from* $X_{S''}$ *to* $X_{S'}$*, or as a map from* $X_{S'}$ *to* X_S*, so all three spaces are homotopy equivalent.*

Proof. Let τ_S, $\tau_{S'}$, and $\tau_{S''}$ be the topologies associated with the three classes. Then $\tau_{S''} \subseteq \tau_{S'} \subseteq \tau_S$. From this it is clear that if f is continuous from $X_{S''}$ to X_S, it is continuous from $X_{S''}$ to $X_{S'}$ and from $X_{S'}$ to X_S as well. For simplicial closeness, we have that for all x, one of $f(x)$ and x can be selected from the other by a rule from S. Such a rule is also in S' and S'', so simplicial closeness holds regardless of which space we take as the domain or codomain of f. □

Proposition 3.2.2. *The topologies induced by the classes* $\mathrm{Sub[C]}$, $\mathrm{Sub[C][LD]}$, *and* $\mathrm{Sub[C][D]}$ *are all homotopy equivalent.*

Proof. We will construct a function $f : 2^\omega_{\mathrm{Sub[C]}} \to 2^\omega_{\mathrm{Sub[C][D]}}$ which induces a good homotopy equivalence. It will follow from Lemma 3.2.1 that all three spaces are homotopy equivalent. Given a string σ, define $f(\sigma)$ as

$$f(\sigma)(n) = \begin{cases} \sigma(m) & \exists m \in \mathbb{N}\, n = m^2 \\ 1 & \text{otherwise} \end{cases}$$

First we need to check that f is continuous. We need to show that if y is a computable subsequence of x, then $f(y)$ is a computable subsequence of $f(x)$ with positive density. Suppose y can be obtained from x by selecting the bits with positions in a computable set A. Consider the set

$$B = \{n^2 : n \in A\} \cup \{n \in \mathbb{N} : n \text{ is not a perfect square}\}$$

Then $f(y)$ can be obtained from $f(x)$ by selecting the bits with positions in B. It is clear that B is computable since A is, and B has density 1 since perfect squares have density 0 in the natural numbers.

Next we will check simplicial closeness. We need to show that for all x, one of x and $f(x)$ can be selected from the other via a computable subsequence. This is true since x can always be selected from $f(x)$ by choosing the bits whose positions are perfect squares. Therefore, f induces a good homotopy equivalence as desired. □

Proposition 3.2.3. *There is no good homotopy equivalence between the topologies induced by*

(i) $\mathrm{Sub[C]}$ *and* Sub.

(ii) $\mathrm{Sub[C][LD]}$ *and* $\mathrm{Sub[LD]}$.

(iii) $\mathrm{Sub[C][D]}$ *and* $\mathrm{Sub[D]}$.

Proof. We will prove the result for $\mathrm{Sub[C]}$ and Sub; the proofs of the other two are very similar.

Suppose $f : 2^\omega_{\mathrm{Sub}} \to 2^\omega_{\mathrm{Sub[C]}}$ induces a good homotopy equivalence. Since f is continuous, whenever y can be selected from x via a subsequence, $f(y)$ can be selected from $f(x)$ via a computable subsequence. In particular, if we take x to be $x_0 = 1010\ldots$, then every $y \in 2^\omega$ can be selected from x_0 via a subsequence. Therefore, every element $f(y)$ in the range of f can be selected from $f(x_0)$ via a computable subsequence. In particular, this means the range of f must be countable.

Denote the range of f by R. By simplicial closeness, every x can either be selected from $f(x)$ or selects $f(x)$ via a computable subsequence. Define a set R' by

$$R' = \{x \in 2^\omega : \exists s \in \mathrm{Sub[C]}\, \exists y \in R\, (s(y) = x \text{ or } s(x) = y)\}$$

Then simplicial closeness of f says that every element of 2^ω is in R'. In other words, every $x \in 2^\omega$ can either (a) be selected from some element of R via a computable subsequence, or (b) contains an element of R as a computable subsequence. By diagonalizing againt the condition that $f_Y(x) = y$ for each computable set Y and string $y \in R$, we can construct uncountably many strings x which fail condition (b). Moreover, since R and $\mathrm{Sub[C]}$ are both countable, only countably many strings satisfy condition (a). Therefore, there is a string which satisfies neither condition, so $R' \neq 2^\omega$, which is a contradiction. □

3.3 The Stochasticity Homotopy Equivalence

In this section we will prove the existence of a good homotopy equivalence between the topologies of Church and MWC.

Theorem 3.3.1. *The topologies induced by* Church *and* MWC *are homotopy equivalent.*

Proof. We will define a function $f : 2^\omega_{\mathrm{MWC}} \to 2^\omega_{\mathrm{Church}}$ which induces a good homotopy equivalence. The proof will be split into two parts: we will first define f, then show it is continuous and satisfies the simplicial closeness condition.

Let $\rho : \mathbb{N}^3 \to \mathbb{N}$ be a computable pairing function. For example, one could define $\rho(x, y, z) = \langle\langle x, y\rangle, z\rangle$ where

$$\langle a, b\rangle = \frac{1}{2}(a + b)(a + b + 1) + b$$

The goal in defining f will be to include all information about how every Turing functional Φ_e acts on X, including the use and halting time of these computations. Explicitly, we will let

$$f(x)(\rho(e, u, k)) = \begin{cases} 1 & k > 0 \text{ and } \Phi_{e,k}(x|_u)\downarrow \\ 0 & k > 0 \text{ and } \Phi_{e,k}(x|_u)\uparrow \\ 1 & k = 0 \text{ and } \Phi_e(x|_u)\downarrow \\ 0 & \text{otherwise} \end{cases}$$

Before we argue that f induces a good homotopy equivalence, we will first prove the following properties about f:

(i) $f(x)$ contains x as a computable subsequence.

(ii) $f(x)$ has $\overline{0} = 000\ldots$ and $\overline{1} = 111\ldots$ as computable subsequences.

(iii) If $s \in \mathrm{MWC}$ with $s(x) = y$, then y is computable from $f(x)$.

To prove property (i), note that there is a Turing machine which halts on input n if and only if the oracle tape has a 1 in the n'th position. Let e_0 be the index of such a machine. Then $f(x)(\rho(e_0, n + 1, 0)) = x(n)$ by the definition of f. The sequence $\rho(e_0, n + 1, 0)$ is computable since ρ is.

For property (ii), let e_1 be the index of a Turing machine which never halts and e_2 the index of a machine which always halts, regardless of what is on the oracle tape. For similar reasons to the previous property, $f(x)(\rho(e_1, n+1, 0)) = 0$ and $f(x)(\rho(e_2, n+1, 0)) = 1$ for all n, so $\overline{0}$ and $\overline{1}$ are contained in $f(x)$ as computable subsequences.

Finally, for property (iii), note that y is computable from x since there is an MWC selection rule selecting y from x. Moreover, x is computable from $f(x)$ by property (i), so y is computable from $f(x)$.

Next we will show that f is continuous. Suppose $s(x) = y$, where $s \in \mathrm{MWC}$. By property (i), there is a computable set A such that the subsequence of bits of $f(x)$ whose positions are in A is x. By property (ii), there are also infinite computable sets B_0 and B_1 such that the bits of $f(x)$ in those positions are always 0 or 1, respectively. Let e_3 be an index for the MWC selector which induces s. We want to show that there is a Church selection rule s' which selects $f(y)$ from $f(x)$. We will describe a general procedure for a MWC selection rule acting on a string σ. Then we will show the selector which induces it is total (so it is actually a Church selection rule) and that it selects $f(y)$ from $f(x)$.

Suppose our procedure has an input string σ and has not yet selected any bits or taken any actions. To select the bit in position $0 = \rho(0, 0, 0)$, it will take the following steps.

- Scan bits of σ until it finds ones whose positions are in A. As it continues scanning bits in all other steps, whenever it finds one in A, store it in a subsequence called σ_0.

- After it finds the first bit $\sigma_0(0)$ in A, scan until it finds the bit in position $\rho(e_3, 1, 0)$. At this point, label this bit of σ_0 as *used*.
- If the bit in position $\rho(e_3, 1, 0)$ is 0, continue reading all future bits of σ, outputting none of them, from now on and ignore all further instructions for the procedure.
- If the bit in position $\rho(e_3, 1, 0)$ is 1, scan for the bits in positions $\rho(e_3, 1, k)$, starting from $k = 0$ and increasing, until it finds one which is 1.
- Once it finds a k_0 such that $\rho(e_3, 1, k_0) = 1$, run $\Phi_{e_3}(\sigma_0|_1)$ for k_0 steps.
- If this does not halt, continue reading σ forever and output no more bits.
- If it halts with output 1, compute $\sigma_0(1)$ if necessary, call its value b, and enumerate it into a sequence τ. If it halts with output 0, repeat the previous steps with the first unused bit of σ_0 and incrementing the middle index of $\rho(e_3, 1, k)$ by one.
- Compute an index q for a Turing machine which, on any input, runs Φ_0 on input b.
- Scan the value of $\rho(q, 0, 0)$. If this value is 0, continue scanning until the algorithm is about to scan a bit with position in B_0, at which point it will select that bit. If it is 1, select a bit with position in B_1 instead.

Before describing what the procedure does at a nonzero stage, we will first show that this procedure has the intended behavior if $\sigma = f(x)$. In this case, we can rewrite the above description in the following way.

- Scan bits of $f(x)$, storing bits which are in x in a subsequence.
- After it has found a bit of x, check the appropriate bit of $f(x)$ to see if $\Phi_{e_3}(x|_1)$ halts. It will, since Φ_{e_3} must halt on every initial segment of x by the existence of s.
- Check the appropriate bits of $f(x)$ to see if $\Phi_{e_3}(x|_1)$ halts in k steps for each k. One of them will, since the computation halts.
- Once it finds one that does, run this computation and record its value - this will tell us whether s selects the first bit from x or not.
- If it does not select the first bit, repeat the above steps to see if it selects the next bit, and so on. Some bit will eventually be selected, since we know s selects y from x.
- Once $y|_0$ has been computed, use an appropriate bit in $f(x)$ to check if Φ_0 halts on input $y|_0$, and output this result using the infinite subsequences of 0s and 1s with positions in B_0 and B_1.

Also notice that regardless of the input σ, the procedure always either outputs a bit or continues reading $f(x)$ forever and outputs no bits. This may make the selection rule procedure output finite strings for some inputs, but the selector remains total (in the sense of Definition 1.3.3) at least for this step.

Now suppose we are at some nonzero stage and our procedure has selected $\rho(e, u, k)$ bits from an input σ, where $k = 0$, and it is trying to select the bit that will go in position $\rho(e, u, k)$ of the output. In this case, it will repeat the exact same steps as above with a few modifications. Instead of just computing one bit of τ, it will compute $u + 1$ bits in succession. Also, the index q it computes will run Φ_e on this initial segment of τ rather than running Φ_0 on b. Otherwise, every step will be the same.

Finally, suppose the procedure has selected $\rho(e, u, k)$ bits from σ where $k > 0$. In this case, most of the procedure will be the same as in the previous case. However, instead of checking if $\rho(e_3, 0, 0)$ is 1 or not, it will check if $\rho(e_3, 0, k)$ is 1 or not. In the case where σ is $f(x)$, this will allow it to check whether $\Phi_e(y|_k)$ halts in some finite number of steps, rather than checking if it halts at all.

The fact that the selector for this procedure is total and selects $f(y)$ from $f(x)$ shows that f is continuous. We have essentially already shown that f satisfies the simplicial closeness property. Indeed, since x is a computable subsequence of $f(x)$ with positions in A, the oblivious Church selection rule which only selects bits whose positions are in A selects x from $f(x)$. Therefore, f induces a good homotopy equivalence as desired.

□

With these homotopy equivalences proven, we can categorize all but three of the inclusions in Figure 3.1. The status of the homotopy equivalences we have classified is shown in Figure 3.2.

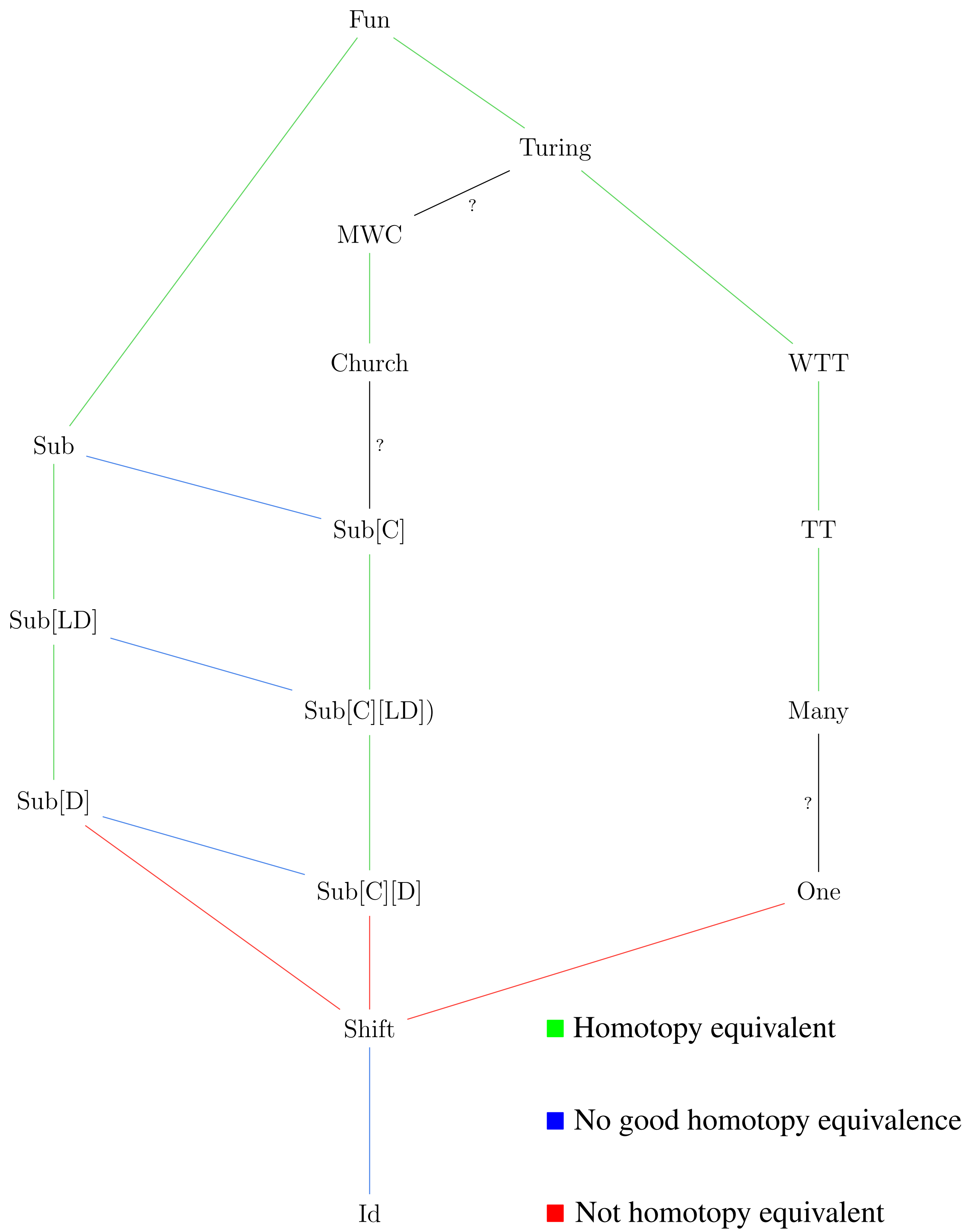

Figure 3.2: Homotopy equivalences between selection rule spaces

Chapter 4

Future Work

In previous investigations of classical selection rules (such as [5], [8], and [10]), the proofs relating stochasticity notions are relatively ad hoc; that is, they typically only work for the properties being studied, and don't easily generalize to the case of general selection rules. The goal of this thesis was to lay the groundwork for a method which could be used to study several classes of selection rules simultaneously.

While homotopy equivalence is one of the easier properties to study for selection rule topologies, it only scratches the surface of what information could be gained about classes of selection rules from analyzing these topological properties. One direction for future work could be to study invariants which are slightly finer than homotopy equivalence. For example, one could define a *computable homotopy equivalence* between topologies on 2^ω to be a homotopy equivalence where the two maps f and g are both computable. The proof of Theorem 3.3.1 gives a good homotopy equivalence, but fails to give a computable good homotopy equivalence between the topologies induced by Church and MWC. Similarly, one could define a $\emptyset'$-computable homotopy equivalence, a positive density homotopy equivalence, etc. By considering these finer invariants, one may be able to distinguish classes like Church and MWC from each other.

Another possible direction is to explore classes of stochastic strings in the spaces discussed in this thesis. Definition 2.3.2 allows us to consider stochasticity with respect to any property D of a string rather than just the classical property H of having a balanced amount of zeroes and ones. One could investigate whether there are any properties of a set D which would guarantee that the set $N(D)$ of D-stochastics share properties with the Church or MWC stochastics. For example, it is immediate from the definitions in Section 2.3.2 that there exists a D-stochastic if and only if D is not σ^0_1. What information about stochastics can be determined if D lies elsewhere in the largeness or smallness hierarchies? Another question to consider is whether there is a topological way of distinguishing stochastics with respect to some property. For example, would the lack of a computable homotopy equivalence between the topologies induced by Church and MWC ensure that the Church and MWC stochastics are not the same set? If not, what other properties of the topologies are needed?

Finally, one could explore classes of selection rules which are not monoidal. One important class of selection rules which were not discussed in this thesis are *Kolmogorov-Loveland selection rules*, a modified version of MWC selection rules which are allowed to read bits of a string in any order, not just monotonically. Merkle showed in [5] that these rules are not closed under composition, so they do not form a monoidal class of selection rules. If we attempt to define the interior operator N in the same way, it will no longer satisfy $N \circ N = N$ and so it will not induce a topology. Is there any way of turning this into a topology? One approach could be to define the interior of a set D to be $\bigcap_{n\in\mathbb{N}} N^n(D)$; does this give us a topology for the Kolmogorov-Loveland selection rules, and if so what properties does it have?

Bibliography

[1] Will Asness, *A brief overview of Alexandrov spaces*, http://math.uchicago.edu/~may/REU2018/REUPapers/Asness.pdf, 2018.

[2] Robert Downey and Denis Hirschfeldt, *Algorithmic randomness and complexity*, Springer, New York, 2010.

[3] Allen Hatcher, *Algebraic topology*, Cambrige University Press, Cambridge, 2002.

[4] J. P. May, *Finite spaces and larger contexts*, https://math.uchicago.edu/~may/FINITE/FINITEBOOK/FINITEBOOKCollatedDraft.pdf.

[5] Wolfgang Merkle, *The Kolmogorov-Loveland stochastic sequences are not closed under selecting subsequences*, Journal of Symbolic Logic **68** (2003), 1362–1376.

[6] Justin Miller, *Intermediate intrinsic density and randomness*, Computability, to appear.

[7] Charles Pugh, *Real mathematical analysis*, Springer, Cham, 2002.

[8] Alexander Shen, *On relations between different algorithmic definitions of randomness*, Soviet Math. Dokl. **38** (1989), 316–319.

[9] Robert Soare, *Turing computability: Theory and applications*, Springer, Berlin, 2016.

[10] Jean Ville, *Étude critique de la notion de collectif*, 1939.

[11] Richard von Mises, *Grundlagen der wahrscheinlichkeitsrechnung*, Mathematische Zeitschrift **5** (1919), 52–99.

ACADEMIC VITA

Jack Piazza

EDUCATION

B.S. in Mathematics, The Pennsylvania State University — Expected 2023
Schreyer Honors College
Graduate coursework in: Topology, Logic, Algebra, Real Analysis, Dynamical Systems

RESEARCH EXPERIENCE

Asymptotic Behavior of the MNS Model — June 2022 - current
Advised by Andrew Ahn — Cornell University

- Used methods from integrable probability and random matrix theory to study the MNS model, a periodic random matrix model.
- Found the limit shape of an asymptotic version of the MNS model; Currently investigating the higher-order moments.

A Topological Approach to Selection-Rule Randomness — January 2022 - current
Advised by Jan Reimann — Penn State University

- Found a new way of characterizing definitions of randomness using ideas from topology.
- Currently investigating how to distinguish randomness notions using topological structure.
- Paper in preparation, will also be submitted as a senior thesis.

Church Stochasticity and Uniform Distribution — January 2020 - December 2021
Advised by Jan Reimann — Penn State University

- Studied algorithmic randomness, complexity theory, and uniform distribution.
- Investigated generalizations of randomness notions using ideas from uniform distribution.

HONORS AND AWARDS

- Barry Goldwater Scholarship — 2022
- Evan Pugh Scholar Award — 2022
  - Awarded to the top 0.5 percent of Penn State juniors and seniors by GPA.
- Erickson Discovery Grant — 2021
  - A Penn State award which funds undergraduate summer research projects.
- Phi Beta Kappa Membership — 2021
- Institutional Membership, Association for Symbolic Logic — 2020

WORK AND FUNDED EXPERIENCES

Cornell Mathematics REU — June 2022 - July 2022
Cornell University

- Studied asymptotic behavior of models in random matrix theory and integrable probability with a team of three other undergraduates under Dr. Andrew Ahn.

Research Assistant May 2021 - August 2021
Penn State University

- Investigated the relationship between computability theory and uniform distribution under Dr. Jan Reimann.
- Funded by an Erickson Discovery Grant from Penn State.

Mathematics Grader August 2020 - current
Penn State University

- Graded for several undergraduate (upper-division and lower-division) math courses, including Introduction to Mathematical Logic, Complex Analysis, Classical Analysis II, Introduction to Analysis I, Integral Vector Calculus, and Matrices.

Teaching Assistant August 2020 - December 2020
Penn State University

- Held daily office hours for the introductory math course The Mathematics of Money, which had an enrollment of 300 students.
- Helped students, most of whom had no math experience beyond high school, with questions about course material.

PRESENTATIONS

Selection Rule Topology and its Applications July 2022
Penn State Logic Seminar Penn State University

Limit Shape of Periodic Random Walks July 2022
Cornell Math REU Symposium Cornell University

Uniform Distribution and Randomness for Infinite Binary Sequences April 2022
Penn State Undergraduate Research Exhibition Penn State University

Church Stochasticity and Uniform Distribution March 2022
Southeastern Logic Symposium University of Florida

Gale-Stewart Games March 2022
Penn State Logic Seminar Penn State University

Structure of Optimal Strategies for Nonlocal Games February 2021
Penn State Logic Seminar Penn State University

Superdense Coding and the Bell Inequality November 2020
Penn State Logic Seminar Penn State University

READING GROUPS AND SEMINARS

Penn State Logic Seminar August 2020 - current

- Gave talks at semester-long reading courses attended primarily by graduate students.

Category Theory Reading Group May 2020 - August 2020

- Helped lead discussions for a weekly summer course.