AMSI SUMMER RESEARCH SCHOLARSHIPS 2023–24

SET YOUR SIGHTS ON RESEARCH THIS SUMMER

A Category Theoretic Approach to Dynamical Systems

James Kelly

Supervised by Associate Prof. Marcy Robertson, Dr. Léo Diaz

The University of Melbourne

Contents

1 Abstract

We study the theory of polynomial functors and its applications to modelling open, compositional dynamical systems. In particular, we introduce the category **Poly** and demonstrate that it has two monoidal structures. Using the theory of **Poly**, we instantiate some key ideas of dynamical systems including composition, time dependence and stochasticity.

2 Introduction

Category theory occupies a central position in the modern development of areas of pure mathematics including algebraic geometry, representation theory, homological algebra and more. In recent years, there is a growing body of interest in understanding the applications of category theory outside of traditional pure mathematics. In particular, we would like to understand how category theory can be used to give a unifying account of open, compositional dynamical systems. One such approach is to use the theory of polynomial functors. Although these objects have a simple description as the coproduct of representable functors $\mathbf{Set} \to \mathbf{Set}$, the associated category **Poly** has a remarkable amount of structure. In this report, we will first introduce **Poly** and several useful results including two ways to realise **Poly** as a monoidal category. Then we will introduce a definition of a dynamical system in the language of **Poly**. Finally, we will see how to instantiate some key ideas of dynamical systems including compositionality, time dependence and stochasticity.

3 Statement of Authorship

All the results presented in this report are already known and reflect the work of other authors. To establish the basic theory of polynomial functors in §4 as well as the content in §5.1, §5.2 I have studied [NS23], [Spi20], [Spi22]. To introduce time dependence and stochasticity in §5.3 and §5.4 I have studied [St 23b] and aspects of [St 23a]. I introduce $\mathbb{D}yn$ (Definition 5.5) as a particular full subcategory of $Arr(\mathbf{Poly})$ to de-emphasise organising dynamical systems by interface as in [Spi22], [SS23]. Though, I believe this construction would already be known. Additionally, I introduce some basic examples including the $(n, \mathcal{C})$-wired system from Example 5.10. Finally, this report was written by me and looked over by Marcy Robertson, Léo Diaz and Kurt Stoeckl.

4 Polynomial Functors

Before introducing the central definition of this report given in §5, it is necessary to introduce some technical foundations. In this section, we will define the category **Poly** and prove several useful results.

4.1 The category Poly

Definition 4.1. *An endofunctor $p : \boldsymbol{Set} \to \boldsymbol{Set}$ is a polynomial functor (or simply a polynomial) if there exists some $I \in \boldsymbol{Set}$ considered as a discrete category, a functor $p[-] : I \to \boldsymbol{Set}$ and a natural isomorphism*

$$p \cong \coprod_{i \in I} \boldsymbol{Set}(p[i], -)$$

In other words, a polynomial is a coproduct of covariant representable functors. For notation, we will write $\mathbf{Set}(A, -) := y^A$, $y^{\{\bullet\}} := y$ and $Sy^A := \coprod_{s \in S} y^A$.

Remark 4.2. *Notice that for any choice of $A \in \boldsymbol{Set}$, $\boldsymbol{Set}(A, \{\bullet\}) \cong \{\bullet\}$. Hence, $\coprod_{i \in I} \boldsymbol{Set}(p[i], \{\bullet\}) \cong I$. This gives us a canonical way of writing any polynomial functor. Setting $p(\{\bullet\}) := p(1)$, we may write*

$$p \cong \coprod_{i \in p(1)} y^{p[i]}$$

In this regard, we follow the style of [Spi20] and may refer to $p(1)$ as the position set of p and $p[i]$ the direction set at position i.

Definition 4.3. *We define $\boldsymbol{Poly}$ to be the category with objects as polynomials and*

$$\boldsymbol{Poly}(p, q) := \{F : p \to q : F \in \boldsymbol{Nat}(p, q)\}$$

In other words, the morphisms of $\boldsymbol{Poly}$ are natural transformations.

This definition obscures a natural combinatorial interpretation of morphisms of polynomials. This following proposition will allow us to work more concretely with **Poly**.

Proposition 4.4. *Let $p, q \in \boldsymbol{Poly}$. Any morphism $\varphi : p \to q$ is uniquely identified by*

1. *A map of sets $\varphi_1 : p(1) \to q(1)$*
2. *A natural transformation $\varphi_2 : q[\varphi_1(-)] \Rightarrow p[-]$. Equivalently, an $i \in p(1)$ indexed set of functions*

$$(\varphi^{(i)} : q[\varphi_1(i)] \to p[i])_{i \in p(1)}$$

Proof. First, we claim that

$$\mathbf{Poly}(\coprod_{i \in p(1)} y^{p[i]}, q) \cong \prod_{i \in p(1)} \mathbf{Poly}(y^{p[i]}, q)$$

We will define a forward map F as follows. Take $\varphi \in \mathbf{Poly}(\coprod_{i \in p(1)} y^{p[i]}, q)$ and let $i \in p(1)$. Fixing $X \in \mathbf{Set}$ and using the canonical inclusion map, we construct a map of sets

$$X^{p[i]} \xrightarrow{\iota_{i,X}} \textstyle\coprod_{j \in p(1)} X^{p[j]} \xrightarrow{\varphi_X} q(X)$$

Therefore, we have a collection of maps $(X^{p[i]} \xrightarrow{\varphi_X \circ \iota_{i,X}} q(X))_{X \in \mathbf{Set}}$. We verify that this collection constitutes a natural transformation. Let $f \in \mathbf{Set}(X, Y)$ be arbitrary. Consider the following diagram

$$\begin{array}{ccccc}
X^{p[i]} & \xrightarrow{\iota_{i,X}} & \coprod_{j \in p(1)} X^{p[j]} & \xrightarrow{\varphi_X} & q(X) \\
{\scriptstyle f \circ -} \big\downarrow & & \big\downarrow & & \big\downarrow {\scriptstyle q(f)} \\
Y^{p[i]} & \xrightarrow[\iota_{i,Y}]{} & \coprod_{j \in p(1)} Y^{p[j]} & \xrightarrow[\varphi_Y]{} & q(Y)
\end{array}$$

By assumption, the rightmost square is commutative and using the universal property of the co-product, we deduce that the leftmost square is also commutative. Therefore, the collection is a natural transformation and we can map φ to the following dependent function (see Definition A.7).

$$\varphi \mapsto \begin{array}{l} F(\varphi) : (i \in p(1)) \to \mathbf{Poly} \\ i \mapsto (X^{p[i]} \xrightarrow{\varphi_X \circ \iota_{i,X}} q(X))_{X \in \mathbf{Set}} \end{array}$$

Now, we will construct an inverse map G. We identify an arbitrary element $\Delta \in \prod_{i \in p(1)} \mathbf{Poly}(y^{p[i]}, q)$ with a dependent function $\Delta : (i \in p(1)) \to \mathbf{Poly}(y^{p[i]}, q)$. Then for any $X \in \mathbf{Set}$, we can simply define

$$\coprod_{i \in p(1)} X^{p[i]} \to q(X)$$

$$(i, f : p[i] \to X) \mapsto \Delta(i)_X(f)$$

If we let $\Delta(-)_X$ refer to the map defined above, then $G(\Delta) := (\Delta(-)_X)_{X \in \mathbf{Set}}$. To verify that $G(\Delta) \in \mathbf{Poly}(\coprod_{i \in p(1)} y^{p[i]}, q)$, let $f \in \mathbf{Set}(X, Y)$. Then it suffices to show that for $(i, \alpha : p[i] \to X) \in \coprod_{j \in p(1)} X^{p[i]}$

$$q(f)(\Delta(i)_X)(\alpha) = \Delta(i)_Y(f \circ \alpha)$$

However, this is immediate since $(\Delta(X)_i)_{X \in \mathbf{Set}}$ is a natural transformation. By construction, F and G are mutual inverses. Now, writing q in canonical form and applying the Yoneda lemma, we may write

$$\mathbf{Poly}(\coprod_{i \in p(1)} y^{p[i]}, \coprod_{j \in q(1)} y^{q[j]}) \cong \prod_{i \in p(1)} \coprod_{j \in q(1)} \mathbf{Set}(q[j], p[i])$$

Therefore, an element $\varphi \in \mathbf{Poly}(p, q)$ is identified with a dependent function

$$\varphi : (i \in p(1) \to \coprod_{j \in q(1)} \mathbf{Set}(q[j], p[i]))$$

$$i \mapsto (j, f : q[j] \to p[i])$$

Then by taking the first and second projections of the image of φ, we precisely recover a function

$$f_1 : p(1) \to q(1)$$

and an $i \in p(1)$ indexed collection of functions

$$(q[f_1(i)] \to p[i])_{i \in p(1)}$$

as required. $\square$

In view of Proposition 4.4, we will often refer to the data of $\varphi \in \mathbf{Poly}(p, q)$ as a pair $(\varphi_1, (\varphi^{(s)})_{s \in p(1)})$. We will call these maps forward and backwards maps respectively. This result naturally leads to an alternate, though more abstract, description of **Poly** which will become important in §5.

Proposition 4.5. *There is an equivalence of categories* $\boldsymbol{Poly} \xrightarrow{\sim} \int (\boldsymbol{Set}/-)^p$ *where* $(\boldsymbol{Set}/-)^p$ *denotes the contravariant slice functor* $\boldsymbol{Set}^{op} \to \boldsymbol{Cat}$ *taking* $A \in \boldsymbol{Set} \mapsto (\boldsymbol{Set}/A)^{op}$ *(see Definitions A.1, A.2, A.3).*

Proof. First, we will describe a functor $F : \mathbf{Poly} \to \int(Set/-)^p$. Extending Remark 4.2, the data of any polynomial functor $p \in \mathbf{Poly}$ can be written as a function $\pi : \coprod_{i \in p(1)} p[i] \to p(1)$ where π is the canonical projection map. In this view, the fibres (preimages) of π determine the data of $p[-]$. Then

$$F(p) := (p(1), \pi : \coprod_{i \in p(1)} p[i] \to p(1))$$

Let $\varphi : p \to q$ be a morphism of polynomials. A morphism

$$(p(1), \coprod_{i \in p(1)} p[i] \xrightarrow{\pi_p} p(1)) \to (q(1), \coprod_{j \in q(1)} q[j] \xrightarrow{\pi_q} q(1))$$

in $\int(\mathbf{Set}/-)^p$ consists precisely of a function $f_1 : p(1) \to q(1)$ and a choice of function such that the following diagram commutes.

$$\begin{array}{ccc} \coprod_{j \in q(1)} q[j] \times_{q(1)} p(1) & \xrightarrow{f_2} & \coprod_{i \in p(1)} p[i] \\ & \searrow \quad \swarrow & \\ & p(1) & \end{array}$$

We may rewrite $\coprod_{j \in q(1)} q[j] \times_{q(1)} p(1) \cong \coprod_{i \in p(1)} q[f_1(i)]$. Then the data of the above commutative diagram is equivalent to demanding that the induced function $\coprod_{i \in p(1)} q[f_1(i)] \xrightarrow{f_2} \coprod_{i \in p(1)} p[i]$ preserves indexing so that $f_2(i, a) = (i, b)$. This is precisely the data of a functor $q[f_1(-)] \to p[-]$ and hence, using Proposition 4.4, we can prescribe $F(p) \xrightarrow{F(\varphi)} F(q)$. It is clear then that $id \in \mathbf{Poly}(p, p)$ is mapped to the $F(p) \xrightarrow{id} F(p)$ since id consists of the identity $p(1) \to p(1)$ and the identity functor $p[-] \to p[-]$. Given $\varphi \in \mathbf{Poly}(p, q)$ and $\phi \in \mathbf{Poly}(q, w)$, the composite $\phi \circ \varphi : p \to w$ consists of the composite function $p(1) \xrightarrow{\varphi_1} q(1) \xrightarrow{\phi_1} w(1)$ and composite functor $w[\phi \circ \varphi(-)] \to p[-]$. Referring to the definition of composition in Definition A.3, the induced map $F(\phi) \circ F(\varphi)$ is given by the composite $\phi_1 \circ \varphi_1$ and a commuting diagram

$$\begin{array}{ccccc} \coprod_{i \in w(1)} w[i] \times_{w(1)} p(1) & \longrightarrow & \coprod_{j \in q(1)} q[i] \times_{w(1)} q(1) & \longrightarrow & \coprod_{k \in p(1)} p[k] \\ & \searrow & & \swarrow & \\ & & p(1) & & \end{array}$$

The composite function of the top row is equivalently a function $\coprod_{k \in p(1)} w[\phi_1 \circ \varphi_1(k)] \to \coprod_{k \in p(1)} p[k]$ and the commutativity of the diagram ensures that it gives the data of a functor $w[\phi_1 \circ \varphi_1(-)] \to p[-]$. Hence, $F(\phi \circ \varphi) = F(\phi) \circ F(\varphi)$ and we conclude F is a functor. Instead of constructing an inverse functor, we claim that F is full, faithful and dense. Given $(A, B \xrightarrow{f} A) \in \int(\mathbf{Set}/-)^p$, we can construct a polynomial p by demanding $p(1) := A$ and for $a \in A, p[a] := f^{-1}(a)$. Then $F(p) \cong (A, B \xrightarrow{f} A)$ and so F is dense. From our definition of $F(\varphi) : F(p) \to F(q)$, it follows that $F(\varphi)$ is uniquely determined by the data of φ consequently, F is faithful. Moreover, any morphism $\gamma : F(p) \to F(q)$ consists of precisely the same data of a morphism $p \to q$. Therefore, F is full. We conclude that $\mathbf{Poly} \xrightarrow{\sim} \int(\mathbf{Set}/-)^p$. □

4.2 Monoidal structures on Poly

In the final part of §4, we will establish some additional structure on **Poly**. In particular, we define a tensor product of polynomials which we will call the Dirichlet product. This will induce a monoidal structure. Finally, we will see that polynomials are closed under functor composition yielding an additional monoidal structure.

4.2.1 Dirichlet product

Definition 4.6. *Let* $\coprod_{i\in p(1)} y^{p[i]}, \coprod_{j\in q(1)} y^{q[j]} \in$ ***Poly****. We define the Dirichlet product of two polynomials as*

$$\coprod_{i\in p(1)} y^{p[i]} \otimes \coprod_{j\in q(1)} y^{q[j]} := \coprod_{(i,j)\in p(1)\times q(1)} y^{p[i]\times q[j]}$$

The Dirichlet product of two polynomials is clearly then a polynomial. Indeed, this can be extended to define a monoidal structure on **Poly**.

Lemma 4.7. *The Dirichlet product is a bifunctor* $-\otimes- :$ ***Poly*** $\times$ ***Poly*** $\to$ ***Poly***

Proof. From Definition 4.6 we have defined a map on objects. Given $(\varphi,\phi)\in \mathbf{Poly}(p,p')\times\mathbf{Poly}(q,q')$, we must first define

$$\varphi\otimes\phi : \coprod_{(i,j)\in p(1)\times q(1)} y^{p[i]\times q[j]} \to \coprod_{(i,j)\in p'(1)\times q'(1)} y^{p'[i]\times q'[j]}$$

Using the notation of Proposition 4.4, we define $(\varphi\otimes\phi)_1$ as the product map for $\varphi_1\times\phi_1$ and for $(a,b)\in p(1)\times q(1)$, $(\varphi\otimes\phi)^{(a,b)}$ as the product map $\varphi^{(a)}\times\phi^{(b)}$. It then follows that (id_p, id_q) is mapped to the identity morphism on $p\otimes q$. Let $(\varphi_1,\phi_1)\in \mathbf{Poly}\times\mathbf{Poly}((p,q),(v,w))$, $(\varphi_2,\phi_2)\in\mathbf{Poly}\times\mathbf{Poly}((v,w),(r,t))$. Observing the fact that $\times$ is a bifunctor on **Set** and the definition of composition in **Poly** $\times$ **Poly**, we conclude that the following two maps are equal

$$(\varphi_2\circ\varphi_1)\otimes(\phi_2\circ\phi_1) : p\otimes q\to r\otimes t$$

$$(\varphi_2\otimes\phi_2)\circ(\varphi_1\otimes\phi_1) : p\otimes q\to r\otimes t$$

and hence $-\otimes-$ is a bifunctor.

□

Proposition 4.8. $($***Poly***$,\otimes,y)$ *is a symmetric monoidal category.*

Let $p,q,r,s\in\mathbf{Poly}$. Given $p\otimes(q\otimes r)$ and $(p\otimes q)\otimes r$, we compute the position set to be $p(1)\times(q(1)\times r(1))$ and $(p(1)\times q(1))\times r(1)$ respectively. Similarly, fixing $(i,(j,k))\in p(1)\times(q(1)\times r(1))$ and $((i,j),k)\in(p(1)\times q(1))\times r(1)$, the respective direction sets are $p[i]\times(q[j]\times r[k])$ and $(p[i]\times q[j])\times r[k]$. Therefore, there is an obvious associator $\alpha_{p,q,r} : p\otimes(q\otimes r)\xrightarrow{\sim}(p\otimes q)\otimes r$ induced by the associator of $(\mathbf{Set},\times,\{\bullet\})$. Similarly,

$$p\otimes y = \coprod_{(i,j)\in p(1)\times\{\bullet\}} y^{p[i]\times\{\bullet\}} \cong p$$

$$y\otimes p = \coprod_{(i,j)\in \{\bullet\}\times p(1)} y^{\{\bullet\}\times p[j]} \cong p$$

$$p\otimes q = \coprod_{(i,j)\in p(1)\times q(1)} y^{p[i]\times q[j]} \cong \coprod_{(j,i)\in q(1)\times p(1)} y^{q[j]\times p[i]} = q\otimes p$$

induced by the left, right unitors and swap map of **Set**. One can more precisely check the coherence diagrams commute by relying on the fact that $(\mathbf{Set},\times,\{\bullet\})$ is a symmetric monoidal category. Though, we omit this explicitly here.

4.2.2 Composition product

Since the underlying objects of **Poly** are functors **Set** $\to$ **Set**, we have an associated notion of composition of functors inherited from the functor category [**Set**, **Set**]. We can ask whether the composition of a polynomial is again a polynomial. Surprisingly, the answer is yes.

Proposition 4.9. *Let $p, q \in$ **Poly**. Then $p \circ q \in$ **Poly**.*

Proof. Let $X \in \mathbf{Set}, p, q \in \mathbf{Poly}$ and $i \in p(1)$. First, we claim there exists a bijection

$$\prod_{a\in p[i]} \coprod_{j\in q(1)} \mathbf{Set}(q[j], X) \xrightarrow{\sim} \coprod_{f\in \mathbf{Set}(p[i],q(1))} \mathbf{Set}(\coprod_{a\in p[i]} q[f(a)], X)$$

An element of $\prod_{a\in p[i]} \coprod_{j\in q(1)} \mathbf{Set}(q[j], X)$ is identified with a dependent function $F : (a \in p[i]) \to \coprod_{j\in q(1)} \mathbf{Set}(q[j], X)$ sending $a \in p[1] \mapsto (j, f : q[j] \to X)$. Let π_1, π_2 denote the first and second projection maps. Then

$$\pi_1 \circ F : p[1] \to q(1)$$

The second projection map yields a function

$$q[\pi_1 \circ F(a)] \to X$$

Hence, $(\pi_1 \circ F, \coprod_{a\in[i]} \pi_2 \circ F(a)) \in \coprod_{f\in \mathbf{Set}(p[i],q(1))} \mathbf{Set}(\coprod_{a\in p[i]} q[f(a)], X)$. Going in the reverse direction, an element of $\coprod_{f\in \mathbf{Set}(p[i],q(1))} \mathbf{Set}(\coprod_{a\in p[i]} q[f(a)], X)$ is a pair

$$(f : p[i] \to q(1), g : \sum_{a\in p[i]} q[f(a)] \to X)$$

To construct a dependent function $F : (a \in p[i]) \to (j, q[j] \to X)$ we can simply take

$$a \mapsto (f(a), q[f(a)] \xrightarrow{\iota} \sum_{a\in p[i]} q[f(a)] \to X)$$

These maps are inverse to each other and moreover, natural in choice of $X \in \mathbf{Set}$. Next, we use the fact that for any $A \in \mathbf{Set}$, we can write $A \cong \coprod_{a\in A}\{\bullet\}$ and adapting some of the ideas of the proof of Proposition 4.4, to write

$$\begin{aligned} p \circ q &\cong \coprod_{i\in p(1)} \prod_{a\in p[i]} \coprod_{j\in q(1)} y^{q[j]} \\ &\cong \coprod_{i\in p(1)} \coprod_{f\in \mathbf{Set}(p[i],q(1))} \mathbf{Set}(\coprod_{a\in p[i]} q[f(a)], -) \\ &\cong \coprod_{(i,f)\in p(1)\times \mathbf{Set}(p[i],q(1))} y^{\coprod_{a\in p[i]} q[f(a)]} \end{aligned}$$

where in the second line we have used the initial claim. □

Corollary 4.10. *(**Poly**, $\circ, y$) is a (non-symmetric) monoidal category.*

Proof. We already know that the functor category [**Set**, **Set**] is monoidal with respect to composition. From our work above we have established that for $p, q \in \mathbf{Poly}$ then $p \circ q \in \mathbf{Poly}$. Hence, **Poly** inherits this monoidal structure. ☐

Remark 4.11. *Let $\alpha : p \to p' \in \boldsymbol{Poly}(p, p')$ and $\beta : q \to q' \in \boldsymbol{Poly}(q, q')$. It will be useful to describe how to concretely construct the induced morphism $p \circ q \to p' \circ q'$. In particular, we want to define*

$$\alpha \circ \beta : \coprod_{i \in p(1)} \coprod_{f: p[i] \to q(1)} y^{\coprod_{a \in p[i]} q[f(a)]} \to \coprod_{j \in p'(1)} \coprod_{g: p'[j] \to q'(1)} y^{\coprod_{b \in p'[j]} q[g(a)]}$$

using knowledge of α, β. First, we define the forward map. Let $(i \in p(1), f : p[i] \to q(1)) \in (p \circ q)(1)$. The corresponding element $(p' \circ q')(1)$ is the pair $(\alpha_1(i), g : p'[\alpha_1(i)] \xrightarrow{\alpha_2} p[i] \xrightarrow{f} q(1) \xrightarrow{\beta_1} q'(1))$. The corresponding map on directions is given by the following function

$$\sum_{x \in p'[\alpha_1(i)]} q'[g(x)] \to \sum_{y \in p[i]} q[f(y)]$$

$$(x, a) \mapsto (\alpha^{(i)}(x), \beta^{(f(y))}(a))$$

5 Dynamical Systems

Now that we have established the relevant foundations in the theory of polynomial functors, we introduce the central definition of this report.

Definition 5.1. *A dynamical system is a choice of $S \in \boldsymbol{Set}, p \in \boldsymbol{Poly}$ and $\varphi \in \boldsymbol{Poly}(Sy^S, p)$*

For intuition, we consider polynomials $p, q \in \mathbf{Poly}$ to be systems with an associated set of states. A morphism $p \to q$ then gives an *interaction protocol.* Using this interpretation a dynamical system from 5.1 gives the data of an internal state S, an interface p and then an associated interaction between the internal state and its interface.

Remark 5.2. *In [Spi20], Spivak considers a more general definition of Definition 5.1 where Sy^S is replaced with a comonoid C (where composition $- \circ -$ is taken as the monoidal structure on **Poly**). We do not pursue this here. However, we will revisit the usefulness of comonoids later.*

Considering the interface polynomial consisting of $(p(1) \in \mathbf{Set}, p[-] : p(1) \to \mathbf{Set})$, the simplest example of a dynamical system occurs when $p[-]$ is the constant functor. In this instance, we may write $p \cong p(1)y^{p[1]}$.

Definition 5.3. *Let $A, B, S \in \boldsymbol{Set}$. An (A, B)-Moore machine consists of a pair of functions $r : S \to B$ and $u : A \times S \to S$.*

In this definition we interpret r as the *readout* function which determines the semantics of the internal state. The function u is interpreted as an *update* function encoding the dynamics of a Moore machine.

Proposition 5.4. *An (A, B)-Moore machine corresponds to a dynamical system $\varphi : Sy^S \to By^A$*

Proof. First, observe that $S\{\bullet\}^S \cong S, B\{\bullet\}^A \cong B$. Using Proposition 4.4, the data of φ consists precisely of a function $\varphi_1 : S \to B$ and for each $s \in S$, a function $\varphi_2^s : A \to S$. Set $r := \varphi_1$ and using the adjunction $\mathbf{Set}(X \times Y, Z) \cong \mathbf{Set}(X, \mathbf{Set}(Y, Z))$ we can construct u from the backwards component of φ. □

The Moore machine gives a concrete way to think of dynamical systems as in Definition 5.1. Consider now when p is arbitrary. That is, we do not necessarily demand that $p[-]$ is a constant functor. In this instance, we may still interpret the forward map $\varphi_1 : S \to p(1)$ as the readout function determining the semantics of the internal state. However, the data of our update function (in the context of Proposition 5.4) instead gives a dependent function

$$\varphi^{(-)} : (s \in S) \to \mathbf{Set}(p[\varphi_1(s)], S)$$

We may interpret this as instantiating *mode dependent* dynamics. To see how, consider two polynomials p, q. We can combine them into a new polynomial, which we call $p + q$, as $(p(1) \coprod q(1), (p+q)[-])$ where

$$(p+q)[(i,a)] := \begin{cases} p[a] & i = 1 \\ q[a] & i = 2 \end{cases}$$

Consider two interfaces $By^A, Dy^C \in \mathbf{Poly}$. A dynamical system $Sy^S \xrightarrow{\varphi} By^A + Dy^C$ then gives the data of a function $\varphi_1 : S \to B \coprod D$ and for $s \in S$ a function

$$\varphi^{(s)} = \begin{cases} f_1 : A \to S & \pi_1(\varphi_1(s)) = \pi_1((i,x)) = 1 \\ f_2 : C \to S & \pi_1(\varphi_1(s)) = \pi_1((i,x)) = 2 \end{cases}$$

Hence, we can think of $Sy^S \xrightarrow{\varphi} By^A + Dy^C$, and more generally $Sy^S \to p$ as a *generalised* Moore machine.

5.1 Composing dynamical systems

Consider two dynamical systems

1. $\varphi_1 : S_1 y^{S_1} \to \{\color{blue}{\bullet}, \color{red}{\bullet}, \color{green}{\bullet}\}y^{\{\color{red}{\bullet}\}}$
2. $\varphi_2 : S_2 y^{S_2} \to \{\color{red}{\bullet}\}y^{\{\color{orange}{\bullet}\}}$

Since the Dirichlet product is a bifunctor on **Poly**, we can induce a morphism

$$S_1 y^{S_1} \otimes S_2 y^{S_2} \cong S_1 \times S_2 y^{S_1 \times S_2} \xrightarrow{\varphi_1 \otimes \varphi_2} \{\color{blue}{\bullet}, \color{red}{\bullet}, \color{green}{\bullet}\}y^{\{\color{red}{\bullet}\}} \otimes \{\color{red}{\bullet}\}y^{\{\color{orange}{\bullet}\}} \cong \{\color{blue}{\bullet}, \color{red}{\bullet}, \color{green}{\bullet}\} \times \{\color{red}{\bullet}\}y^{\{\color{orange}{\bullet}\} \times \{\color{red}{\bullet}\}}$$

Observing the proof of Proposition 4.7, the induced morphism $\varphi_1 \otimes \varphi_2$ acts to appose each dynamical system. That is, it acts to put the respective dynamical systems in parallel. Now, for a morphism

$$\rho : \{\color{blue}{\bullet}, \color{red}{\bullet}, \color{green}{\bullet}\} \times \{\color{red}{\bullet}\}y^{\{\color{orange}{\bullet}\} \times \{\color{red}{\bullet}\}} \to \{\color{purple}{\bullet}\}y^{\color{purple}{\bullet}}$$

consider the dynamical system

$$S_1 \times S_2 y^{S_1 \times S_2} \xrightarrow{\rho \circ \varphi_1 \otimes \varphi_2} \{\color{purple}{\bullet}\}y^{\color{purple}{\bullet}}$$

This composite morphism acts to first appose the dynamical systems and then wrap them in a choice of common interface. We would like to collect and formalise this operadic process as a particular morphism in some category.

Definition 5.5. *We define* $\mathbb{D}yn$ *to be the full subcategory of the arrow category* $Arr(\textbf{Poly})$ *where the domain of any arrow* $p \xrightarrow{\varphi} q$ *is isomorphic to* Sy^S *for some* $S \in \textbf{Set}$ *(see Definition A.8). In particular, the objects of* $\mathbb{D}yn$ *are dynamical systems and a morphism of* $\mathbb{D}yn(Sy^S \to p, Ty^T \to q)$ *is a pair* $(\alpha : Sy^S \to Ty^T, \beta : p \to q)$ *such that the following diagram is commutative.*

$$\begin{array}{ccc} Sy^S & \longrightarrow & p \\ {\scriptstyle\alpha}\downarrow & & \downarrow{\scriptstyle\beta} \\ Ty^T & \longrightarrow & q \end{array}$$

Remark 5.6. *The definition of morphisms in Definition 5.5 has a natural interpretation. Consider a morphism of dynamical systems* $(\alpha, \beta) : (Sy^S \xrightarrow{\varphi} By^A) \to (Ty^T \xrightarrow{\phi} Dy^C)$*. In particular,* $\phi \circ \alpha = \beta \circ \varphi$ *which tells us the following: First, the respective induced forward maps* $S \xrightarrow{\alpha_1} T \xrightarrow{\phi_1} D$ *and* $S \xrightarrow{\varphi_1} B \xrightarrow{\beta_1} D$ *are equal. In the language of Proposition 5.4, this tells us that translating states from* S *to* T *and then interpreting their image in* T *is equivalent to interpreting a state in* S *and mapping it to the semantics of* T*. In a similar fashion, if we fix* $s \in S$*, the induced backwards maps* $C \to T \to S$ *and* $C \to A \to S$ *are equal, telling us that the translation of states* $S \leftrightarrow T$ *coheres with the respective dynamics of each dynamical system.*

Proposition 5.7. $\mathbb{D}yn$ *is a symmetric monoidal category with unit* $y \xrightarrow{id} y$

Proof. From Proposition 4.8, we know $(\textbf{Poly}, \otimes, y)$ is a symmetric monoidal category. Observing that $\otimes$ is a bifunctor, this induces a monoidal structure on $Arr(\textbf{Poly})$. It suffices to confirm that $\mathbb{D}yn$ is closed under $\otimes$. Indeed, this is immediate from the definition of $\otimes$ since $Sy^S \otimes Ty^T = S \times Ty^{S \times T}$. ☐

Consider a tuple of dynamical systems

$$(S_i y^{S_i} \xrightarrow{\varphi_i} p_i)_{i=1}^n$$

As before, we find that the Dirichlet product acts to put the dynamical systems in parallel. A morphism of polynomials $\gamma : p_1 \otimes ... \otimes p_n \to q$ then defines a new interaction protocol from the collection of interfaces of the dynamical systems to a new interface, q. This process is precisely a morphism in $\mathbb{D}yn$ of the following form.

$$\begin{array}{ccc} \otimes_{i=1}^n S_i y^{S_i} & \xrightarrow{\otimes_{i=1}^n \varphi_i} & \otimes_{i=1}^n p_i \\ \Vert & & \downarrow{\scriptstyle\gamma} \\ \otimes_{i=1}^n S_i y^{S_i} & \xrightarrow[\gamma \circ \otimes_{i=1}^n \varphi_i]{} & q \end{array}$$

5.2 Dynamic wiring

In Proposition 4.8, we established that **Poly** had a monoidal structure where the product was the Dirichlet product. However, it is possible to say something even stronger. In particular, $\otimes$ is closed with internal hom $[-, -]$.

Proposition 5.8. *There exists a functor* $[-, -] : \textbf{Poly}^{op} \times \textbf{Poly} \to \textbf{Poly}$ *such that there is a natural isomorphism*

$$\textbf{Poly}(p \otimes q, r) \cong \textbf{Poly}(p, [q, r])$$

and an isomorphism $[p,q] \cong \coprod_{\varphi \in \textbf{Poly}(p,q)} y^{\coprod_{i\in p(1)} q[\varphi_1(i)]}$.

For brevity, we will not give a proof here. We refer the interested reader to [NS23, Ch 4 §5]. We introduce this concept because it allows us to instantiate dynamical systems with dynamic organisation patterns as observed in [Spi22], [SS23]. In particular, consider a dynamical system of the form

$$Sy^S \xrightarrow{\varphi} [p,q]$$

Using Proposition 4.4, φ consists of

1. A function $\Delta : S \to \textbf{Poly}(p,q)$ which gives for $s \in S$, a choice of morphism $p \to q$.
2. For each $s \in S$, a function $\coprod_{i\in p(1)} q[\Delta(s)_1(i)] \to S$

Importantly, this particular type of dynamical system associates to each each state, a change of interface. To illustrate these ideas in an example consider the following illustration.

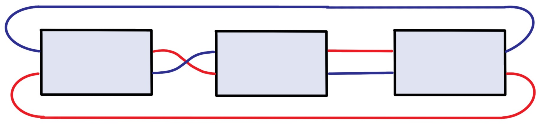


Figure 1: $(3, \{\textcolor{red}{\bullet}, \textcolor{blue}{\bullet}\})$-wired system.

To each box, we consider a system roughly in the context of Definition 5.1. In particular, to each box we associate:

1. An unobserved internal state given by $S_i \in \textbf{Set}$
2. An interface given by an element of $\{\textcolor{red}{\bullet}, \textcolor{blue}{\bullet}\}^2$
3. A mapping $S_i \times \{\textcolor{red}{\bullet}, \textcolor{blue}{\bullet}\}^2 \to B$ for some $B \in \textbf{Set}$ giving the semantics.
4. An update map $\{\textcolor{red}{\bullet}, \textcolor{blue}{\bullet}\}^2 \times \{\textcolor{red}{\bullet}, \textcolor{blue}{\bullet}\}^2 \times S_i \to S_i$ giving the dynamics.

By demanding that the update map depends on a box's interface as well as the interfaces of two other boxes, we recover the wiring pattern depicted in the illustration above. We would like to describe a system which can update its wiring pattern and describe this in the language of **Poly**.

Definition 5.9. *Let $n \in \mathbb{N}$, $B, \mathcal{C} \in$ **Set**. We define an $(n,\mathcal{C})$-wired system as a dynamical system of the form*

$$\otimes_{i=1}^n S_i y^{S_i} \to \otimes_{i=1}^n [\mathcal{C}^2 y^B, \mathcal{C}^2 y^{S_i \times \mathcal{C}^2}]$$

Example 5.10. *For our purposes, we will define $\mathcal{C} := \{\textcolor{red}{\bullet}, \textcolor{blue}{\bullet}, \textcolor{green}{\bullet}\}$. First, consider a $(1,\mathcal{C})$-wired system. Applying the formula from Proposition 5.8, we recover the data of two maps. First, map $S \to \textbf{Poly}(\mathcal{C}^2 y^B, \mathcal{C}^2 y^{S_1 \times \mathcal{C}^2})$ and secondly a map $\mathcal{C}^2 \times \mathcal{C}^2 \times S_1 \to S_1$. We can further break the data of the first map into two respective maps $S_1 \times \mathcal{C}^2 \to \mathcal{C}^2$ and $S_1^2 \times \mathcal{C}^2 \to B$. Together, they give the data of the four dot points described above.*

However, the interface is described by a map $S_1 \times \mathcal{C}^2 \to \mathcal{C}^2$ Significantly, the interface dynamically changes depending on the current state and interface. More generally for $n > 2$, we demand that each box's dynamics $\mathcal{C}^2 \times \mathcal{C}^2 \times S_i \to S_i$ depends on the interface of two other distinguished boxes. For example, a $(5, \mathcal{C})$-wired system is drawn as follows for a fixed wiring pattern.

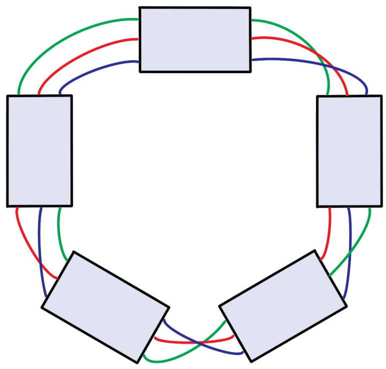


Figure 2: $(5, \{\bullet, \bullet, \bullet\})$*-wired system.*

The wiring pattern then changes dynamically depending on each box's own internal dynamics and the connection a box has with the adjacent boxes.

5.3 Time dependent dynamical systems

Many dynamical systems evolve over time. In particular, we would expect that evolving a dynamical system some time t_1 and then evolving the system again for t_2 units of time should be equivalent to evolving the system for $t_1 + t_2$ units of time. Our starting point to incorporating this into our framework is the following. Fix $S, T \in$ **Set** and consider the dynamical system $Sy^S \xrightarrow{\varphi} y^T$. Using Proposition 4.4, we find that φ is equivalently the data of a function $T \times S \to S$. If T is a monoid, when is the resulting map a monoid action?

Proposition 5.11. *Let $S \in$ **Set**, $(T, +, 0)$ be a commutative monoid and consider **Poly** as a monoidal category with respect to composition $- \circ -$. The set of monoid actions $T \times S \to S$ is in one-to-one correspondence with comonoid homomorphisms (See Definitions A.5,A.6) of the form $Sy^S \to y^T$.*

Before proving this, we need to understand how to equip Sy^S and y^T with comonoid structures.

Lemma 5.12. *Let $S \in$ **Set**. Then Sy^S can be equipped with a comonoid structure.*

Proof. We must first define the erasure and duplicator maps. Any arbitrary morphism $Sy^S \to y$ is equivalent to a function $S \times \{\cdot\} \cong S \to S$. In this view, we define $\epsilon : Sy^S \to y$ corresponding to $id_S : S \to S$. Now, we calculate

$$\begin{aligned} Sy^S \circ Sy^S &\cong \coprod_{s_1 \in S} \coprod_{f : S \to S} y^{\coprod_{s_2 \in S} S} \\ &\cong S \times \mathbf{Set}(S, S) y^{S \times S} \end{aligned}$$

Consequently, any morphism $Sy^S \to Sy^S \circ Sy^S$ is determined by a choice of $f_1 : S \to S \times \mathbf{Set}(S,S)$ and $f_2 : S \times S \times S \to S$. We define the duplicator to be the morphism δ corresponding to $f_1(s) = (s, id_S)$ and $f_2(s_1, s_2, s_3) = s_3$. Next, we check that ϵ and δ satisfy the diagrams in Definition A.5. We claim that the following composite map is an isomorphism.

$$Sy^S \xrightarrow{\delta} S \times \mathbf{Set}(S,S)y^{S\times S} \cong Sy^S \circ Sy^S \xrightarrow{id\circ\epsilon} Sy^S \circ y \cong Sy^S$$

To prove this claim we rely on the computation in Remark 4.11. On positions the map $id \circ \epsilon$ acts to send $(s, f : S \to S) \mapsto (s, S \to \{\cdot\})$. Since $s \mapsto (s, id_S)$ under δ, we see that the composite gives an isomorphism on positions. Next, we consider the backwards maps. Fix $(s^* \in S, f : S \to S)$. Then the backwards map of $id \circ \epsilon$ acts to send $s \mapsto (s, s^*)$. Instead, if we fix $s' \in S$ then the backwards map of δ acts to send $(s_1, s_2) \mapsto s_2$. Therefore, the composition yields $id : S \to S$ and the composite map is an isomorphism. Proving the right co-unit law proceeds similarly so we will omit it. Next, we demonstrate the co-associativity law. Consider $Sy^S \circ Sy^S \xrightarrow{\delta\circ id} (Sy^S \circ Sy^S) \circ Sy^S$. On positions, we compute that $(s^*, f : S \to S) \in (Sy^S \circ Sy^S)(1) \mapsto ((s^*, id_S), (s_1, s_2) \in S \times S \mapsto f(s_2))$. If we fix $(s^*, f : S \to S) \in (Sy^S \circ Sy^S)(1)$, then the corresponding backwards map has form

$$\coprod_{(s_1,s_2)\in S\times S} S \to \coprod_{s\in S} S$$
$$(s_1, s_2, s_3) \mapsto (s_2, s_3)$$

Now, consider $Sy^S \circ Sy^S \xrightarrow{id\circ\delta} Sy^S \circ (Sy^S \circ Sy^S)$. Similarly, on positions we compute $(s^*, f : S \to S) \in (Sy^S \circ Sy^S)(1) \mapsto (s^*, (f(s^*), id_S))$. The backwards map is then given by

$$\coprod_{(s_1,s_2)\in S\times S} S \to \coprod_{s\in S} S$$
$$(s_1, s_2, s_3) \mapsto (s_1, s_3)$$

It is then clear the composite of the respective backwards maps yields the identity. We conclude that (Sy^S, ϵ, δ) is a comonoid. □

Lemma 5.13. *Let $(T, +, 0)$ be a monoid. Then y^T has a canonical comonoid structure.*

Proof. Any morphism $y^T \to y$ is defined by a single choice of $t \in T$. Define the erasure map ϵ to be the morphism corresponding to the choice of $t = 0$. We compute that $y^T \circ y^T \cong y^{T\times T}$. Therefore, any map $y^T \to y^T \circ y^T$ gives the data of a map $T \times T \to T$. We choose this map to correspond to the monoidal operation $+$. Then the co-unit laws follow from the fact that 0 is a two sided unit and co-associativity follows directly from the fact that $+$ is associative. □

Now, we may prove Proposition 5.11.

Proof. Throughout we fix $S \in \mathbf{Set}$ and $(T, 0, +)$ a commutative monoid. We have established that a morphism $Sy^S \to y^T$ is equivalently the data of a function $f : T \times S \to S$. Using the observation that the position set of

y^T and y are singletons, we see that the following square is commutative

$$\begin{array}{ccc} Sy^S & \xrightarrow{\varphi} & y^T \\ {\scriptstyle \epsilon_S}\downarrow & & \downarrow{\scriptstyle \epsilon_T} \\ y & \xrightarrow[id]{} & y \end{array}$$

if and only if for all $s \in S$, $\epsilon_T^{(s_2(0))} = \epsilon_S^{(s)}(s) = s$. Equivalently, if and only if $f(0, s) = s$ for all $s \in S$. Similarly, we see that the following square is commutative

$$\begin{array}{ccc} Sy^S & \xrightarrow{\varphi} & y^T \\ {\scriptstyle \delta_S}\downarrow & & \downarrow{\scriptstyle \delta_T} \\ Sy^S \circ Sy^S & \xrightarrow{\varphi \circ \varphi} & y^T \circ y^T \end{array}$$

if and only if for all $s \in S, t_1, t_2 \in S$, $\varphi_S^s \circ \varphi_T(t_1, t_2) = \delta_S^s(\delta())$. Equivalently, if $f(t_1 + t_2, s) = f(t_2, f(s, t_1))$. We conclude that any monoid action $T \times S \to S$ is uniquely identified with a comonoid morphism $Sy^S \to y^T$. $\square$

To generalise this notion to arbitrary dynamical systems, we introduce the following definition.

Definition 5.14. *A time dependent dynamical system is a choice of monoid $(T, +, 0)$, $p \in$ **Poly** and morphism of polynomials*

$$Sy^S \otimes Ty \to p$$

such that for every morphism $p \to y$, the induced morphism $Sy^S \to y^T$ is a comonoid homomorphism.

First, we note that using Proposition 5.8, any morphism $Sy^S \otimes Ty \xrightarrow{\varphi} p$ is identified with a morphism of the form $Sy^S \to [Ty, p]$. Hence, this definition is a special instance of Definition 5.1. Note that a morphism $p \to y$ is equivalently the choice of $i \in p(1)$ and $x \in p[i]$. Hence, the induced morphism $Sy^S \to y^T$ is given by the map sending

$$(s, t) \in S \times T \mapsto \varphi^{(s,t)}(x) \in S \times \{\bullet\} \cong S$$

Example 5.15. *Let $A, B \in$ **Set** and consider a time dependent dynamical system when $p := By^A$. Comparing with Proposition 5.4, the data consists of*

1. *A readout function $S \times T \to B$*
2. *A map $S \times T \times A \to S$ such that for fixed $a \in A$, the resulting map is a monoid action.*

5.4 Stochastic dynamical systems

In Proposition 4.5, we exhibited a very different means of constructing **Poly**. We would like to slightly modify this construction to instantiate dynamical systems with randomness.

Definition 5.16. *Let M : **Set** $\to$ **Set** be a monad. We define a functor $\mathbf{Set}_M/-$: $\mathbf{Set}^{op} \to$ **Cat** as follows. For any $A \in$ **Set**, we define $\mathbf{Set}_M/A$ to be the category with the same objects as **Set**$/A$. However, a morphism $(X, X \xrightarrow{q} A) \to (Y, Y \xrightarrow{p} A)$ is an $a \in A$ indexed family of functions*

$$\alpha_a : q^{-1}(a) \to M(p^{-1}(a))_{a \in A}$$

The identity morphism is given by the respective component of the unit of the monad $\eta : id \Rightarrow M$ and composition is given in the Kleisli category of M, $\mathcal{K}l(M)$ (see Definition A.4). Given $f \in \textbf{Set}(B, A)$, we induce a functor $\textbf{Set}_M/f : \textbf{Set}_M/A \to \textbf{Set}_M/B$. On objects, $\textbf{Set}_M/f$ acts exactly as $\textbf{Set}/f$ does. That is

$$(Y, Y \to A) \mapsto (Y \times_A B, Y \times_A B \to B)$$

Comparatively, $\textbf{Set}_M/f$ acts on morphisms as

$$(\alpha_a : q^{-1}(a) \to M(p^{-1}(a)))_{a \in A} \mapsto (\alpha_{f(b)} : q^{-1}(f(b)) \to M(p^{-1}(f(b))))_{b \in B}$$

Definition 5.17. *We define $\textbf{Poly}_M := \int (\textbf{Set}_M/-)^p$.*

An object in $\textbf{Poly}_M$ precisely gives a choice $A, B \in \textbf{Set}$, and a function $B \xrightarrow{\alpha} A$. To recover a polynomial $p \in \textbf{Poly}$, we may identify $p(1) := A$ and for any $i \in A$, $p[i] := \alpha^{-1}(i) \subset B$. Next, we consider a morphism $p \to q$. Mirroring the proof of Proposition 4.5, this precisely gives the data of a function $f_1 : p(1) \to q(1)$ and an $i \in p(1)$ indexed collection of functions $(q[f_1(i)] \to M(p[i]))_{i \in p(1)}$. This is formally summarised in the following proposition.

Proposition 5.18. *Let $p, q \in \textbf{Poly}_M$. Then p, q can be identified with polynomial functors and a morphism $\varphi \in \textbf{Poly}_M(p, q)$ can be identified with the following data*

1. *A function $\varphi_1 : p(1) \to q(1)$*
2. *An $i \in p(1)$ indexed family of functions*

$$(\varphi^{(i)} : q[\varphi_1(i)] \to M(p[i]))_{i \in p(1)}$$

If we compare this data to Proposition 4.4 we see that the only difference now is that backwards maps belong to $\mathcal{K}l(M)$ rather than **Set**. Now, we will see this give a specific example.

Proposition 5.19. *There exists a monad $\mathcal{D} : \textbf{Set} \to \textbf{Set}$ such that for any $X \in \textbf{Set}$*

$$\mathcal{D}(X) = \{f \in \textbf{Set}(X, [0, 1]) : \sum_{x \in X} f(x) = 1 \text{ and } f(x) \neq 0 \text{ for finitely many } x \in X\}$$

We call $\mathcal{D}$ the discrete distribution monad.

We will not fully describe the implied monad structure of $\mathcal{D}$ here. The salient feature of $\mathcal{D}$ is that it maps a set X to the set of finitely supported probability distributions on X. Importantly, $\mathcal{D}$ enables us to give an example of a *stochastic* dynamical system.

Definition 5.20. *Let $\mathcal{D}$ be the discrete distribution monad. A stochastic dynamical system is a choice of $S \in \textbf{Set}$, $p \in \textbf{Poly}_\mathcal{D}$ and $\varphi \in \textbf{Poly}_\mathcal{D}(Sy^S, p)$.*

Example 5.21. *Let $t \in \mathbb{N}$ and consider a trader speculating on the price of a commodity. We let $p_t \in \mathbb{R}$ refer to the price of the commodity at time step t. The trader predicts p_{t+1} knowing the historical prince $p_0, p_1, ..., p_t$*

assigning some (finitely supported) distribution $\mathbb{R}^t \to \mathcal{D}(\mathbb{R})$ to the price at the next time step. Realising the trade, the trader can determine the associated profit or loss which is a function $\mathbb{R} \to \mathbb{R}$. This determines the data of a stochastic dynamical system

$$\mathbb{R}y^{\mathbb{R}} \to \mathbb{R}y^{\mathbb{R}^t}$$

6 Conclusion

In this report, we have introduced the category **Poly**. Following this, we presented the central definition of a dynamical system as a particular arrow in **Poly** and explored an assortment of interesting ideas associated to dynamical systems that we could instantiate within this formalism. Initially, we had hoped to better explore Spivak's special double category $\mathbb{O}rg$ and the associated definition of *dynamical categorical structures* outlined in [SS23]. Though given constraints on time and the report, we were not able to adequately realise these goals. In the future, we are also interested in understanding **Poly** for its own sake especially for generalisations to categories other than **Set**. A particularly motivating result is the fact that a polynomial comonoid can be identified with a (small) category.

7 Acknowledgements

I am incredibly grateful to Marcy Robertson, Léo Diaz and Kurt Stoeckl for their time, guidance and support throughout this project. I would also like to give a special thanks to the Australian Mathematical Sciences Institute for this invaluable opportunity.

References

[NS23] Nelson Niu and David I. Spivak. *Polynomial Functors: A Mathematical Theory of Interaction*. 2023. arXiv: `2312.00990 [math.CT]`.

[Spi20] David I. Spivak. *Poly: An abundant categorical setting for mode-dependent dynamics*. 2020. arXiv: `2005.01894 [math.CT]`.

[Spi22] David I. Spivak. "Learners' Languages". In: *Electronic Proceedings in Theoretical Computer Science* 372 (Nov. 2022), pp. 14–28. ISSN: 2075-2180. DOI: `10.4204/eptcs.372.2`. URL: `http://dx.doi.org/10.4204/EPTCS.372.2`.

[SS23] Brandon T. Shapiro and David I. Spivak. "Dynamic Operads, Dynamic Categories: From Deep Learning to Prediction Markets". In: *Electronic Proceedings in Theoretical Computer Science* 380 (Aug. 2023), pp. 183–202. ISSN: 2075-2180. DOI: `10.4204/eptcs.380.11`. URL: `http://dx.doi.org/10.4204/EPTCS.380.11`.

[St 23a] Toby Benedict St Clere Smithe. “Mathematical foundations for a compositional account of the Bayesian brain”. en. PhD thesis. 2023. DOI: 10.5287/ORA-KZJQYOP2D. URL: https://ora.ox.ac.uk/objects/uuid:e5eae609-0306-44fe-97bf-0eabba684983.

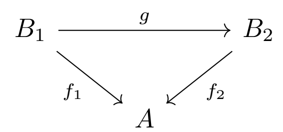
[St 23b] Toby St. Clere Smithe. “Open Dynamical Systems as Coalgebras for Polynomial Functors, with Application to Predictive Processing”. In: *Electronic Proceedings in Theoretical Computer Science* 380 (Aug. 2023), pp. 307–330. ISSN: 2075-2180. DOI: 10.4204/eptcs.380.18. URL: http://dx.doi.org/10.4204/EPTCS.380.18.

A Appendix

Definition A.1. *(Slice category) Let* $\boldsymbol{C}$ *be a small category. Given a choice* $A \in \boldsymbol{C}$*, we may construct the category* $\boldsymbol{C}/A$ *as follows.*

1. *An object of* $\boldsymbol{C}/A$ *is a pair* $(B \in \boldsymbol{C}, f \in \boldsymbol{C}(A, B))$*. We can alternatively identify an object by an arrow* $A \xrightarrow{f} B$*.*

2. *A morphism* $(B_1, f_1) \to (B_2, f_2)$ *is an arrow* $g : B_1 \to B_2$ *such that the follow diagram is commutative.*



Composition $(B_1, f_1) \xrightarrow{g_1} (B_2, f_2) \xrightarrow{g_2} (B_3, f_3)$ *is simply given by* $g_2 \circ g_1$*. The identity arrow for any* (B, f) *corresponds to* $id_B \in \boldsymbol{C}(B, B)$*.*

Definition A.2. *(Opposite slice functor) We denote* $(\boldsymbol{Set}/-)^p$ *as the slice functor* $\boldsymbol{Set}^{op} \to \boldsymbol{Cat}$ *which sends* $A \in \boldsymbol{Set} \mapsto (\boldsymbol{Set}/A)^{op}$*. Given an arrow* $f \in \boldsymbol{Set}(A, B)$*,* $(\boldsymbol{Set}/f)^p$ *is a functor* $\boldsymbol{Set}/B \to \boldsymbol{Set}/A$ *(equivalently a functor between the respective opposite categories) as follows.*

1. *On objects* $(C, C \xrightarrow{h} B) \mapsto (C \times_B A, C \times_B A \xrightarrow{\pi} A)$

2. *On morphisms* $(C_1, C_2 \xrightarrow{h_1} B) \xrightarrow{g} (C_2, C_2 \xrightarrow{h_2} B)$*,* g *is mapped to the induced map*

$$C_1 \times_B A \to C_2 \times_B A$$

Definition A.3. *(Grothendieck construction) Let* $\boldsymbol{Cat}$ *be the category of small categories and let* $\boldsymbol{C} \in \boldsymbol{Cat}$*. Given a functor* $F : \boldsymbol{C} \to \boldsymbol{Cat}$*, the Grothendieck construction written* $\int F$*, is a category consisting of the following data.*

1. *An object in* $\int F$ *is a pair* (c, x) *where* $c \in \boldsymbol{C}$ *and* $x \in F(c)$*.*

2. *A morphism* $(c_1, x_1) \to (c_2, x_2)$ *is an arrow* $f : c_1 \to c_2$ *and a choice of arrow* $g : F(f)(x_1) \to x_2$*. We write this morphism as the pair* (f, g)*.*

3. *The composition of morphisms* $(c_1, x_1) \xrightarrow{(f_1, g_1)} (c_2, x_2) \xrightarrow{(f_2, g_2)} (c_3, x_3)$ *is given by the composite*

$$(f_2 \circ f_1, g_2 \circ F(f)(g_1))$$

Definition A.4. *(Kleisli category) Let* (M, η, μ) *be a monad over a category* $\boldsymbol{C}$*. The Kleisli category, denoted,* $\mathcal{K}l(M)$ *has the same objects as* $\boldsymbol{C}$*. An arrow* $A \to B$ *of* $\mathcal{K}l(M)$ *is an arrow of* $\mathcal{C}(A, M(B))$*. Identity morphisms are given by the components of the unit of* M*,* η*. Finally, given* $f \in \mathcal{C}(X, M(Y))$ *and* $g \in \mathcal{C}(Y, M(Z))$ $g \circ_{\mathcal{K}l(M)} f$ *is given by the composite*

$$X \xrightarrow{f} M(Y) \xrightarrow{M(g)} M^2(Z) \xrightarrow{\mu_Z} M(Z)$$

Definition A.5. *Let* $(C, \otimes, \mathbf{1})$ *be a monoidal category. A comonoid in* C *is an object* M *equipped with a duplicator morphism* $\delta : M \to M \otimes M$ *and an erasure morphism* $M \to \mathbf{1}$*. The following diagrams (known as counit and coassociative laws respectively) are required to commute:*

$$\begin{array}{ccccc} \mathbf{1} \otimes M & \xrightarrow{\sim} & M & \xleftarrow{\sim} & M \otimes \mathbf{1} \\ & {\scriptstyle \epsilon \otimes id} \nwarrow & \downarrow{\scriptstyle \delta} & \nearrow {\scriptstyle id \otimes \epsilon} & \\ & & M \otimes M & & \end{array}$$

$$\begin{array}{ccc} M & \xrightarrow{\delta} & M \otimes M \\ {\scriptstyle \delta}\downarrow & & \downarrow{\scriptstyle id \otimes \delta} \\ M \otimes M & \xrightarrow[\delta \otimes id]{} & M \otimes M \otimes M \end{array}$$

Definition A.6. *Let Let* $(C, \otimes, \mathbf{1})$ *be a monoidal category and* $(M, \epsilon_M, \delta_M), (N, \epsilon_N, \delta_N)$ *be comonoids. A comonoid homomorphism is a morphism* $f : M \to N$ *such that the following diagrams commute:*

$$\begin{array}{ccc} M & \xrightarrow{f} & N \\ {\scriptstyle \epsilon_M}\downarrow & & \downarrow{\scriptstyle \epsilon_n} \\ \mathbf{1} & \xrightarrow[id]{} & \mathbf{1} \end{array}$$

$$\begin{array}{ccc} M & \xrightarrow{f} & N \\ {\scriptstyle \delta_M}\downarrow & & \downarrow{\scriptstyle \delta_n} \\ M \otimes M & \xrightarrow[f \otimes f]{} & N \otimes N \end{array}$$

Definition A.7. *(Dependent function) Let* $I \in \boldsymbol{Set}$ *and consider an* $i \in I$ *indexed collection of sets* $\{X_i\}_{i \in I}$*. A dependent function is an element of* $\prod_{i \in I} X_i$*. We may denote an element as a mapping*

$$f : (i \in I) \to X_i$$

Definition A.8. *(Arrow category) Let* $\boldsymbol{C}$ *be a category. We form the arrow category of* $\boldsymbol{C}$ *denoted at* $Arr(\boldsymbol{C})$ *as follows.*

1. *The objects of* $Arr(\boldsymbol{C})$ *are arrows* $A \to B$ *for* $A, B \in \boldsymbol{C}$

2. *Given two arrows* $A \to B$ *and* $C \to D$. *A morphism is a pair of arrows* $(A \to C, B \to D)$ *such that the following diagram is commutative.*

$$\begin{array}{ccc} A & \longrightarrow & B \\ \downarrow & & \downarrow \\ C & \longrightarrow & D \end{array}$$

3. *Given an arrow* $A \to B$, *the identity morphism is the pair* $(A \xrightarrow{id_A} A, B \xrightarrow{id_B} B)$

4. *Composition of morphism of arrows is given by vertical composition.*