Towards foundations of categorical cybernetics

Matteo Capucci, Bruno Gavranović,
Jules Hedges, Eigil Fjeldgren Rischel

MSP Group, University of Strathclyde, Écosse Libre

We propose a categorical framework for processes which interact bidirectionally with both an environment and a 'controller'. Examples include open learners, in which the controller is an optimiser such as gradient descent, and an approach to compositional game theory closely related to open games, in which the controller is a composite of game-theoretic agents. We believe that 'cybernetic' is an appropriate name for the processes that can be described in this framework.

1   Introduction

In this paper we propose a categorical framework for processes which interact bidirectionally with both an environment and a 'controller'. Examples include open learners [FST19, CGG$^+$21], in which the controller is an optimiser such as gradient descent, and an approach to compositional game theory closely related to open games [GHWZ18] in which the controller is a composite of game-theoretic agents. We believe that 'cybernetic' is an appropriate name for the processes that can be described in this framework.

A common theme of this paper is that the same categorical ingredients appear repeatedly at multiple 'levels' in the construction of cybernetic systems. The most important of these is the *para construction*, which yields a category **Para**($\mathscr{C}$) of parametrised morphisms given a category $\mathscr{C}$ acted upon by a category of parameters $\mathscr{M}$. We combine this construction with the optics construction [CEG$^+$20], which is known to model bidirectional information flow in dynamical systems [Mye20] and compositional game theory [BHZ19]. We argue that we can think of optics as being built of coupled parametrised and 'co-parametrised' morphisms. This suggests that **Para**($-$) and **Optic**($-$) are related. Putting them together, we obtain the category of parametrised optics, which is the central construction of this paper.

A parametrised optic describes open systems that have bidirectional information flow, with a control on the forward direction and an objective on the backwards direction. This pattern is ubiquitous in cybernetics: whenever an instance of the optic pattern appears, we generally find that they form a parametrised family with an objective function, where the goal of the system is to find a parameter that optimises the objective, all while interacting with its environment bidirectionally. The most straightforward example is a component (for example a layer) of a neural network, which interacts with its neighbours via forwards and backwards passes, depending on its current parameters (weights and biases), and produces a 'local' loss against which the parameters are optimised.

The final part of the construction of a class of cybernetic systems is to describe how the parameters/controls and objectives interact. For many classes of systems, they are coupled to a dynamical system that updates the parameters using an algorithm such as gradient descent. In the case of game theory the situation is more complicated, since game-theoretic agents perform counterfactual reasoning and are classically assumed to be perfect optimisers rather than using some effective approximation of the optimum. This situation can be described using selection functions [HOS$^+$17b]. We find that there is a deep and previously unknown relationship between selection functions and optics; in particular we can build a category whose objects are selection functions and morphisms are optics, with a monoidal product describing Nash equilibria.

Figure 1: (a) A morphism in **Para**$(\mathscr{C})$ (b) A morphism in **CoPara**$(\mathscr{C})$

By combining parametrised optics and selection functions, we obtain a compositional formulation of game theory. This turns out to be more specific than open games: a pair of a parametrised optic and a selection function determines an open game functorially, but not vice versa. However each part of our construction carries a clear game-theoretic meaning, and we also fix a particular technical problem with open games involving games where one agent makes several different decisions. Overall, we claim that this variant of open games is an improvement on the usual one.

For reasons of space we only present two classes of examples: neural networks which can be can be presented entirely using the structure of parametrised optics, and open games which have selection functions as an additional ingredient. However, abstractly the same construction works if selection functions are replaced with any suitable lax monoidal pseudofunctor on a category of optics. Further applications of this framework are given in a companion paper [CGLF21].

2   Para

Throughout this section, let $(\mathscr{M}, J, \odot)$ be a symmetric monoidal category and let $\mathscr{C}$ be a category.

Definition 1. An $\mathscr{M}$-actegory is a category $\mathscr{C}$ together with a functor $\bullet : \mathscr{M} \times \mathscr{C} \to \mathscr{C}$, and natural transformations $\varepsilon : J \bullet X \cong X$ and $\delta : (M \odot N) \bullet X = M \bullet (N \bullet X)$ satisfying certain coherence laws [todo].

For the purposes of this paper, $\mathscr{M}$ is to be thought of as a category of *parameters* and $- \bullet -$ as a *parametrisation operation*. From the structure of an $\mathscr{M}$-actegory then, we can produce a category of $\mathscr{M}$-parametrised morphisms in $\mathscr{C}$:

Definition 2 (Para). Let $\mathscr{C}$ be an $\mathscr{M}$-actegory. **Para**$_\bullet(\mathscr{C})$ is the bicategory whose

1. objects are the same of $\mathscr{C}$;

2. 1-cells $\phi \in$ **Para**$_\bullet(\mathscr{C})(X,Y)$ are given by a choice of object M (the *parameter*) in $\mathscr{M}$ and a choice of morphism $\phi : M \bullet X \to Y$ in $\mathscr{C}$, and we condense this data into the notation $\phi : X \xrightarrow{M} Y$;

3. 1-composition of $\phi : X \xrightarrow{M} Y$ and $\psi : Y \xrightarrow{N} Z$ is given by

$$\phi \,\fatsemi\, \psi : X \xrightarrow{N \odot M} Z \quad := \quad (N \odot M) \bullet X \xrightarrow{\delta} N \bullet (M \bullet X) \xrightarrow{N \bullet \phi} N \bullet Y \xrightarrow{\psi} Z$$

4. 2-cells $r : \phi \Rightarrow \phi'$ from $\phi : X \xrightarrow{M} Y$ to $\phi' : X \xrightarrow{M'} Y$ are given by morphisms $r : M \to M'$ in $\mathscr{M}$ such that $\phi = (r \bullet X) \,\fatsemi\, \phi'$

5. Identities and composition in the hom-cat **Para**$_\bullet(\mathscr{C})(X,Y)$ are the same of $\mathscr{M}$.

Proposition 3. **Para**$(-)$ *is a pseudomonad on the category $\mathscr{M}$-**Mod** of $\mathscr{M}$-actegories.*

Proof. The action of $\mathscr{M}$ on **Para**$(\mathscr{C})$ is inherited from $\mathscr{C}$ in the obvious way. Its unit is given by faithful, identity-on-objects functors sending a morphism ϕ to (J, ϕ). Its join sends a twice parametrised map (m, n, ϕ) to the once parametrised map $(m \odot n, \phi)$.   □

There's also a dual construction to that of **Para**$(-)$, which we call **CoPara**$(-)$, which produces a category of *coparametrised* morphisms, i.e. morphisms whose codomain is parametrised. A morphism $X \to Y$ in **CoPara**$(\mathscr{C})$ consists of an object M and a morphism $\phi : X \to M \bullet Y$. That is, **CoPara**$(\mathscr{C})$ is defined as **Para**$(\mathscr{C}^{\mathbf{op}})^{\mathbf{op}}$.

Morphisms in **Para**$(\mathscr{C})$ are more naturally denoted using string diagrams, as shown in figure 1. In these diagrams, objects of $\mathscr{M}$ are denoted by vertical strings, and objects of $\mathscr{C}$ by horizontal strings. Formally, these (and most other diagrams in this paper) can be considered as diagrams in a larger double category in which both **Para**$(\mathscr{C})$ and **CoPara**$(\mathscr{C})$ embed, and thus we can apply coherence for string diagrams in double categories [Mye18].
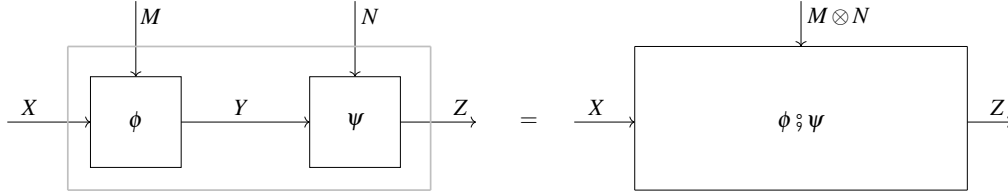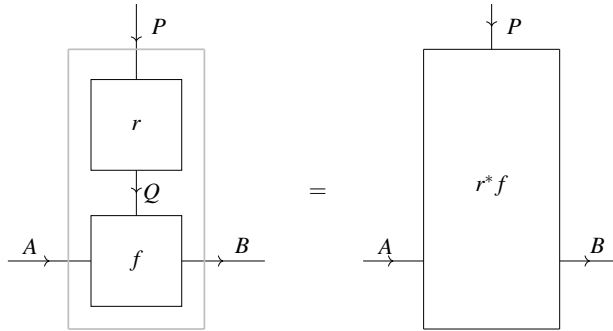

Figure 2: 1-cell composition in **Para**$(\mathscr{C})$

Figure 3: 2-cell reparametrisation in **Para**$(\mathscr{C})$

2.1 Monoidal structure on Para$(\mathscr{C})$.

If $\mathscr{C}$, in addition to an actegorical structure, has a symmetric monoidal structure (making it akin an *algebra* over a monoid or ring), we'd like to lift it to **Para**$(\mathscr{C})$. This is possible when the two structures are compatible:

Definition 4. Let $\mathscr{M}$ be a monoidal category. A symmetric monoidal $\mathscr{M}$-actegory is an $\mathscr{M}$-actegory $\mathscr{C}$ equipped with a symmetric monoidal structure, together with a natural isomorphism $\kappa_{M,X,Y} : M \bullet (X \otimes Y) \cong X \otimes (M \bullet Y)$, satisfying coherence laws reminiscent of the laws of a costrong comonad.

Given a symmetric monoidal $\mathscr{M}$-actegory, we can also define natural isomorphisms $\alpha : M \bullet (X \otimes Y) \cong (M \bullet X) \otimes Y$ and $\iota : (M \otimes N) \bullet (X \otimes Y) \cong (M \bullet X) \otimes (N \bullet Y)$, called the mixed associator and mixed interchanger, respectively.

As anticipated, when $\mathscr{C}$ is a symmetric monoidal $\mathscr{M}$-actegory, then **Para**$_\bullet(\mathscr{C})$ is itself a symmetric monoidal category, whose structure we still denote with I and $\otimes$. The tensor of two objects x, y is given

by the same tensor $x \otimes y$, whereas the tensor of two arrows $\phi : x \xrightarrow{m} y$ and $\psi : u \xrightarrow{n} v$ is given by

$$\phi \otimes \psi : X \otimes X' \xrightarrow{M \odot N} Y \otimes Y' \quad := \quad (M \odot N) \bullet X \otimes X' \xrightarrow{\iota} (M \bullet X) \otimes (N \bullet Y) \xrightarrow{\phi \otimes \psi} Y \otimes Y'$$

Unitors and associator are borrowed from $\mathscr{C}$ through the immersion $\mathscr{C} \to \mathbf{Para}_\bullet(\mathscr{C})$.
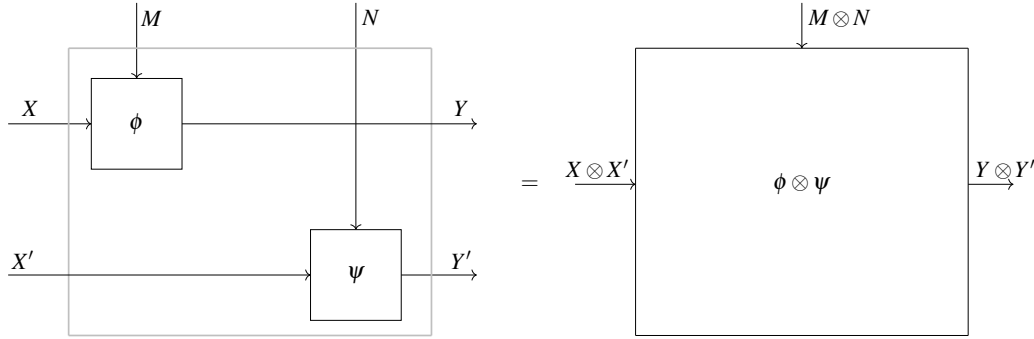

Figure 4: Monoidal product in $\mathbf{Para}(\mathscr{C})$

Given a monoidal $\mathscr{M}$-actegory $\mathscr{C}$, there is a functor $\mathscr{M} \to \mathscr{C}$ given by $M \mapsto M \bullet I$. We can think of the image of this functor as a 'representation' of $\mathscr{M}$ inside $\mathscr{C}$. States (morphisms from the monoidal unit) and costates (morphisms to the monoidal unit, sometimes known as effects) in $\mathbf{Para}(\mathscr{C})$ and $\mathbf{CoPara}(\mathscr{C})$ involve this representation. Namely, a state of X consists of an object M and a morphism $M \bullet I \to X$, and a costate of X in $\mathbf{CoPara}(\mathscr{C})$ consists of an object M and a morphism $X \to M \bullet I$. In the case of a monoidal category acting on itself, these are precisely the slices and coslices of X.

3 Optics

The story of dynamical systems is grounded in the notion of bidirectionality and feedback: processes do not just flow in one direction, but often compute values *going forward* and compute some notion of *feedback* or *rewards* going backward. In order of increasing generality, this is captured by lenses, optics, and mixed optics. In this paper we proceed with the latter notion and invoke special cases as needed.

Definition 5 (Optics, [CEG$^+$20, Def. 2.1]). Given a monoidal category $\mathscr{M}$ acting on categories $\mathscr{C}, \mathscr{D}$ through actions we both denote with $\bullet$ be two $\mathscr{M}$-actegories, we can define the category $\mathbf{Optic}_{\bullet,\bullet}(\mathscr{C}, \mathscr{D})$ whose objects are pair of sets and whose hom-sets are defined by the following coend

$$\mathbf{Optic}_{\bullet,\bullet}(\mathscr{C}, \mathscr{D})(\binom{X}{X'}, \binom{Y}{Y'}) = \int^M \mathscr{C}(X, M \cdot Y) \times \mathscr{D}(M \cdot Y', X')$$

We can think of optics as a bidirectional process taking in an X and computing an output $M \cdot Y$. Depending on the monoidal action used, this can unpacks to a number of different things. Most often it is a product of an output Y and the intermediate state M (called the residual). Off-screen, the environment takes in the output Y and returns a Y'. In the case of the monoidal action of the product, this Y' is used together with the previously saved M to produce a X' (Figure 5).

Formally, an optic $\binom{X}{X'} \rightleftarrows \binom{Y}{Y'}$ is an equivalence class of triples (M, v, u) where M is an object of $\mathscr{M}$, $v : X \xrightarrow{M} Y$ is a morphism of $\mathbf{CoPara}(\mathscr{C})$ and $u : Y' \xrightarrow{M} X'$ is a morphism of $\mathbf{Para}(\mathscr{C})$, modulo the equivalence relation generated by $(M, v, f^*u) \sim (M', f_*v, u)$ for morphisms $f : M \to M'$ of $\mathscr{M}$. This is a

small equivalent reformulation of the usual definition of optics via a coend. The coend in this formulation quotients out observationally indistinguishable reparametrisations, meaning that morphisms of residuals can be freely slid over back and forth between the forward and backward part.

Remark 6 (Specializations). If $\mathscr{M} = \mathscr{C} = \mathscr{D}$, we speak of *non-mixed optics* and we write $\mathbf{Optic}_\otimes(\mathscr{C})$, sometimes omitting the action as well. If $\mathscr{C}$ and $\mathscr{D}$ are symmetric monoidal, then $\mathbf{Optic}_{\bullet,\bullet}(\mathscr{C}, \mathscr{D})$ is itself symmetric monoidal. Also observe that $\mathbf{Lens}(\mathscr{C}) \cong \mathbf{Optic}_\times(\mathscr{C})$ for a cartesian monoidal category.
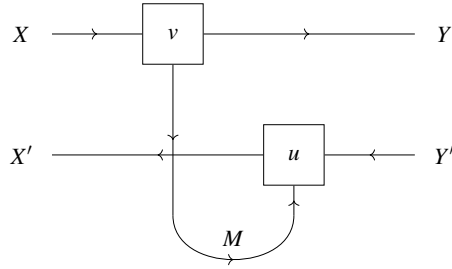

Figure 5: Graphical representation of an optic $\begin{pmatrix} X \\ X' \end{pmatrix} \rightleftarrows \begin{pmatrix} Y \\ Y' \end{pmatrix}$. For an animated version explaining the information flow, see [Gav21b, Gav21a]

Under certain conditions, we have a convenient description of the collection of **states** (maps from the monoidal unit) and **costates** (maps to the monoidal unit) in the category of optics. On the one hand, states of $\begin{pmatrix} X \\ X' \end{pmatrix}$ in $\mathbf{Optic}(\mathscr{C}, \mathscr{D})$ when $\mathscr{D}$ is semicartesian (hence its unit coincides with the terminal object) coincides with state of X in $\mathscr{C}$. On the other hand, costates of $\begin{pmatrix} X \\ X' \end{pmatrix}$ in non-mixed optics coincides with maps $X \to X'$ in $\mathscr{C}$. We note that the case when the forward and the backward categories are different can still be made precise, but now instead of a homomorphism $X \to X'$ in $\mathscr{C}$ we have a **heteromorphism** $X \to X'$ from $\mathscr{C}$ to $\mathscr{D}$ (mediated by a residual in $\mathscr{M}$).

4 Parametrised optics for cybernetics systems

The $\mathbf{Para}(-)$ construction lets us consider morphisms with "hidden" information, not available to the rest of the world. The $\mathbf{Optic}(-)$ construction lets us consider bidirectional morphisms in this category, allowing for constructions involving *feedback* and *update*. Applying the former to the latter results in a sophisticated construction which we unpack in this section.

Parametrised optics. Given a symmetric monoidal category $\mathscr{M}$ acting on $\mathscr{C}$ and $\mathscr{D}$, we have shown above how to form the category $\mathbf{Optic}(\mathscr{C}, \mathscr{D})$. Moreover, since $\mathscr{M}$ is a symmetric monoidal category, it acts on itself by the monoidal product, so we can form the category $\mathbf{Optic}(\mathscr{M})$ of non-mixed optics, which is symmetric monoidal with pointwise monoidal product.

Proposition 7. *Let $\mathscr{M}$ be symmetric monoidal and $\mathscr{C}$ and $\mathscr{D}$ be $\mathscr{M}$-actegories. There is an action of* $\mathbf{Optic}_\odot(\mathscr{M})$ *on* $\mathbf{Optic}_{\bullet,\bullet}(\mathscr{C}, \mathscr{D})$, *defined on objects as* $\begin{pmatrix} M \\ M' \end{pmatrix} \circledast \begin{pmatrix} X \\ X' \end{pmatrix} := \begin{pmatrix} M \bullet X \\ M' \bullet X' \end{pmatrix}$

We thus can form the bicategory $\mathbf{Para}_\circledast(\mathbf{Optic}_{\bullet,\bullet}(\mathscr{C}, \mathscr{D}))$ for this action. Concretely, for objects X, Y of $\mathscr{C}$ and X', Y' of $\mathscr{D}$, a **parametrised optic** $\begin{pmatrix} X \\ X' \end{pmatrix} \rightleftarrows \begin{pmatrix} Y \\ Y' \end{pmatrix}$ consists of a choice of objects (P, Q) of $\mathscr{M}$, and an equivalence class of triples (M, v, u) where M is an object of $\mathscr{M}$, $v : P \bullet X \to M \bullet Y$ is a morphism of $\mathscr{C}$ and $u : M \bullet Y' \to Q \bullet X'$ is a morphism of $\mathscr{D}$. We depict such a morphism by a string diagram, shown in Figure 6(a).
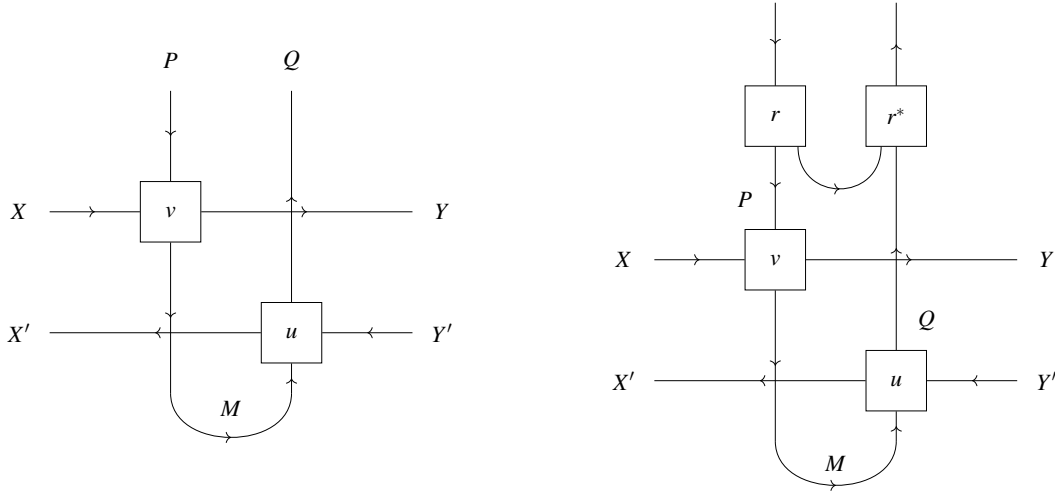
Figure 6: A morphism (a) and reparametrisation (b) in $\mathbf{Para}_{\circledast}(\mathbf{Optic}_{\bullet,\bullet}(\mathscr{C}, \mathscr{D}))$.

Cybernetic systems. The mathematical framework of parametrised optics extends the one already known for dynamical systems, in which different flavours of lenses are used to represent bidirectional information flow [VSL14, Mye20, MSN], in a direction first hinted at in [FST19]. Parametrised optics model cybernetic systems, namely dynamical systems *steered* by one or more agents. Then $\circledast$ represents *agency being exerted* on systems; in particular agents act in the world through the $\mathscr{M}$-actegorical structure on $\mathscr{C}$ and receive feedback through the $\mathscr{M}$-actegorical structure of $\mathscr{D}$. [1] $\mathbf{Optic}_{\odot}(\mathscr{M})$ is a category representing agents, and the framework of $\mathbf{Para}_{\circledast}(\mathbf{Optic}_{\bullet,\bullet}(\mathscr{C}, \mathscr{D}))$ is what allows us to interpret it as such. [2]

Graphically, vertical wires in $\mathbf{Para}_{\circledast}(\mathbf{Optic}_{\bullet,\bullet}(\mathscr{C}, \mathscr{D}))$ tell us how agents interact and compose. Reparametrisations (Figure 6(b)) denote agency dynamics as happening *over* the horizontal dynamical system (which can be called an **arena**, or a **protocol**) they act within. This point of view seamlessly unifies parameters and coparameters of a parametrised optic with residuals inside the optic, which are now representing the *private state* of agents controlling the given dynamical system. This state ferries information between the forward and the backward part of the dynamics (playing the role of 'memory') so that an agent's feedback is contingent on the action they brought about (see [Wat13, p.192]).[3] Furthermore, more complex protocols of interaction can be created, for instance, using optics for the coproduct (prisms), modeling control flow.

What's missing in the mathematical structure of $\mathbf{Para}(\mathbf{Optic}(-))$ is a feedback mechanism, central in cybernetics. In the next sections, we explore the current solutions adopted by the two most developed instantiations of the framework just described, namely *open learners* and *open games*.

---

[1] A happy coincidence of terminology: *agents acts through actions*.

[2] We remark agency is a property of the *model* of a system and not of the system *per se*. This is reflected in the mathematics by the fact that all morphisms in $\mathbf{Para}(\mathscr{C})$ are morphisms of $\mathscr{C}$, arranged in a different way.

[3] Further investigation in this direction is being pursued by the authors. In particular, it seems profitable to adopt a notion of *lax optics* in which the coend of Definition 5 is replaced by a lax coend [Lor20, Chapter 7]. Residuals and slidings would then be explicitly tracked as 2-dimensional structure on $\mathbf{Optic}(\mathscr{C}, \mathscr{D})$, as much as parameters in $\mathbf{Para}(\mathscr{C})$ are tracked in the 2-dimensional structure of that category. Lax fubini (ibid.) gives rise to an interesting duality between residuals and parameters, which the authors first heard of in a Zulip conversation started by Mitchell Riley [RC21].

4.1 Neural Networks

A special case of this construction is studied in [CGG$^+$21], in which the authors instantiate this construction in the context of machine learning, when the base is set to **Smooth**, the category of Euclidean spaces and smooth maps. They first start by reframing backpropagation in terms of optic composition, via the functor $R : \mathbf{Smooth} \to \mathbf{Optic}(\mathbf{Smooth})$ which augments maps with their backward pass (though they use lenses, as defined in Remark 6). Then, using the action of $\mathbf{Para}(-)$ on morphisms (Proposition 3), they lift this functor to its $\mathbf{Para}(-)$ counterpart. We now unpack this in more detail.

Consider a morphism in $(\mathbb{R}^p, f) : \mathbf{Para}(\mathbf{Smooth})(\mathbb{R}^n, \mathbb{R}^m)$. It consists of a *parameter space* $\mathbb{R}^p$ and a smooth function $f : \mathbb{R}^p \times \mathbb{R}^n \to \mathbb{R}^m$. For instance, this could be the forward pass of a complex, multi-layered neural network. It takes in a parameter value $p : \mathbb{R}^p$, an input $x : \mathbb{R}^n$ and computes a *prediction* $f(p,x) : \mathbb{R}^m$. Composition of morphisms in $\mathbf{Para}(\mathbf{Smooth})$ reduces to composition of forward passes of neural networks, coherently tracking which subnetworks each incoming parameter needs to be relayed to. This becomes important as the authors apply a functor

$$\mathbf{Para}(R) : \mathbf{Para}(\mathbf{Smooth}) \to \mathbf{Para}(\mathbf{Optic}(\mathbf{Smooth})), \tag{1}$$

augmenting this neural network with the *backward pass*. Acting on the morphism f, this results in a forward and a backward component displayed (Figure 6(a)). The forward component consists of an input parameter space $\mathbb{R}^p$ and a morphism $v : \mathbb{R}^p \times \mathbb{R}^n \to \mathbb{R}^m \times \mathbb{R}^n$ which in addition to computing f also copies the input $\mathbb{R}^n$ and saves it as the *residual* used on the backward pass. The backward pass consists of an output parameter space $\mathbb{R}^p$ (interpreted as the space of *changes*), and a map $u : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n \times \mathbb{R}^p$ which takes in an *change in the output*, and together with the saved residual computes a change in the parameters and change in inputs. The latter is then subsequently used in the previous learner as its incoming change in outputs. This complex data flow is automatic: it falls out of the machinery of both $\mathbf{Para}(-)$ and $\mathbf{Optic}(\mathbf{Smooth})$. The former deals with parameter spaces, and the latter knows how to perform backpropagation.

These parameter spaces become important as neural networks can be coherently *reparametrised* within this framework, where reparametrisations in $\mathbf{Para}(\mathbf{Optic}(\mathbf{Smooth}))$ are themselves optics. For instance, consider the parameter port $(\mathbb{R}^p, \mathbb{R}^p)$ of a 1-cell: it takes in a parameter value and produces a *change in that parameter*. By using the optic of gradient descent we can reparametrise this learner and express the idea that a learner is *minimizing* a particular objective function.

Construction 8 (Gradient Descent, compare [CGG$^+$21, Example 3.15])**.** Consider a lens on the base $\mathscr{C} = \mathbf{Smooth}$. Fix $\alpha : \mathbb{R}$. The notion of α-gradient descent is a lens $gd_\alpha : \left(\begin{smallmatrix}\mathbb{R}\\\mathbb{R}\end{smallmatrix}\right) \rightleftarrows \left(\begin{smallmatrix}\mathbb{R}\\\mathbb{R}\end{smallmatrix}\right)$ whose forward map is identity, and the backward map $(p, \nabla p) \mapsto p - \alpha \nabla p$ computes a new parameter value by moving in the α-scaled direction of the change in the parameter ∇p. For negative choices of α we obtain gradient *ascent*, and denote it with ga_α.

The archetypal example of two neural networks optimizing *different goals* is Generative Adversarial Networks (GANs) [GPAM$^+$14, ACB17]. We hereby present the first categorical formalization of this neural network architecture and interpret its game-theoretic behaviour in terms of opposing reparametrisations: gradient *descent* and gradient *ascent*. [4] A GAN is composed of two networks: a *generator (g)* and a *discriminator (d)* network in $\mathbf{Para}(\mathbf{Optic}(\mathbf{Smooth}))$, obtained as the image under the functor $\mathbf{Para}(R)$ (Eq. 1). The discriminator (an element of $\mathbf{Para}(\mathbf{Smooth})(\mathbb{R}^x, \mathbb{R})$) is tasked with assigning a $\mathbb{R}$-valued *cost* to each point of a particular "image space" (denoted with $\mathbb{R}^x$) as a measure of how much it *looks like* it belongs to a particular predefined dataset, where low cost equals realistic looking samples.

---

[4]We focus on WGAN for simplicity, and ignore the Lipschitz regularization condition.

The generator (an element of **Para**(**Smooth**)($\mathbb{R}^z, \mathbb{R}^x$)) is tasked with generating realistic looking samples from this dataset, by taking in some "latent vector" in $\mathbb{R}^z$ and producing a sample in $\mathbb{R}^x$ (often called the "fake" sample). The user's goal is to train the generator such that, as we vary the latent vector input $\mathbb{R}^z$, we obtain different realistically looking samples in $\mathbb{R}^x$. This is done via the *adversarial training* regime and a carefully chosen composition of these networks (Figure 7). This regime feeds in real and fake samples to the discriminator, training it to distinguish between them. At the same time, it trains the generator with an opposing goal: producing samples that *fool* the discriminator, making it assign a low cost to them.
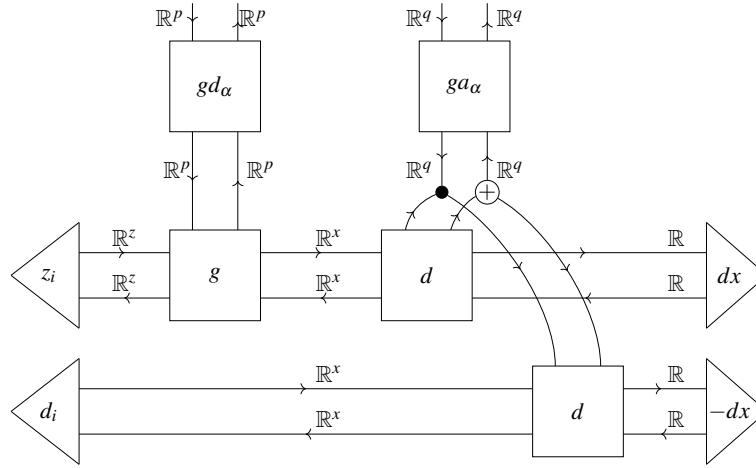

Figure 7: A generative adversarial network as a closed system.

This adversarial component is in our framework captured by reparametrisation, allowing to specify agent preferences *internal to the agent*. Consider the composite $R(g) \, ; R(d)$ in **Para**(**Optic**(**Smooth**)) (middle part of Figure 7). At some time step i this neural network takes a latent vector z_i as input, produces a *fake* element x_i of $\mathbb{R}^x$ using g, and assigns a $\mathbb{R}$-valued score to it using d. In this setting we want the discriminator to assign an even higher score next time (as the sample was fake) and the generator to produce a sample which will have a lower score next time (as it is trying to fool the discriminator). We can achieve this by reparametrising both the generator and the discriminator, but with different, "opposing" optics. We reparametrise the generator with the gradient descent (making it move in the negative direction of the gradient, minimizing the cost the discriminator assigns to the fake sample), and the discriminator with the gradient *ascent* (making it move in the positive direction of the gradient, increasing the cost discriminator assigns to the fake image). This allows both of these agents to have unchanged external behaviour during one time-step of updating, and use the same training signal to enact a different update rule internally. [5]

Another benefit of reparametrisation is that it allows us to recast two 1-cells with the same parameter object as *the same player*. Consider the discriminator in Figure 7. It appears twice: once valuating samples coming from the generator (x_i) , and once evaluating samples coming from the actual dataset (d_i). The parameter port of the parallel product (Def. 4) $R[d] \times R[d]$ takes in two parameters $\mathbb{R}^q \times \mathbb{R}^q$ as input and produces *two changes* $\mathbb{R}^q \times \mathbb{R}^q$ in these parameters as output. By first reparametrising this

---

[5]The lens costates on the right of Fig. 7 are aptly called dx. Their backward maps are constant at $1 : \mathbb{R}$, the usually invisible "initial" factor of 1 in backpropagation.

product using the image of the copy map $\Delta_{\mathbb{R}^q} : \mathbb{R}^q \to \mathbb{R}^q \times \mathbb{R}^q$ under the functor R, we make sure that the same parameter is copied to both discriminators in the forward pass, and their gradients are summed up on the backward pass. Also called *weight tying*, this couples together two different neural networks and treats them as a single unit.

5 Selection relations

In cybernetics, agents provide parameters which are evaluated by the environment through a dynamical process and fed back to them. Crucially, agents then use this information to update the parameters they provide, closing the loop. When agents are satisfied with the outcome of the interaction, an equilibrium is established. In machine learning, parameter updating is explicitly modelled (Construction 8), but in game theory we directly seek the equilibrium. We propose here a framework for equilibrium selection for parametrised optics based on a generalization of selection functions.

A selection relation, defined in [HOS$^+$17a] as "multi-valued selection functions", is a relation of type $\varepsilon \subseteq X \times (X \to R)$. Selection relations provide an abstraction of the "personality" or "goals" of game-theoretic agents. Specifically, we consider that a function $\varepsilon \subseteq X \times (X \to R)$ describes an agent who can make a choice from a set X, with outcomes in a set R, in which $(x, k) \in \varepsilon$ means that the agent considers the move x to be 'good' in the context k, mapping possible moves x' to their outcome $k(x')$.

Selection relations are a variant of the better-known single-valued selection functions [EO11], which are better behaved mathematically (forming a monad) but are less flexible as descriptions of agents. For example, the very common *utility-maximizing agents* can be described as a multivalued selection function argmax, where $(x, k) \in \arg\max$ iff x attains the global maximum of k.

The functor of selection relations. The starting observation of the following mathematical treatment of selection relations is that a point $x \in X$ is exactly a state of $\binom{X}{R}$ in the monoidal category **Lens**(**Set**), while a function $k : X \to R$ is exactly a costate of $\binom{X}{R}$. Therefore a selection relation can be seen as a relation $\varepsilon \subseteq \textbf{Lens}(\textbf{Set})(I, \binom{X}{R}) \times \textbf{Lens}(\textbf{Set})(\binom{X}{R}, I)$.

Definition 9. Let $\mathscr{M}$ be a monoidal category. We define the functor $\mathbb{S}_{\mathscr{M}} : \mathscr{M} \to \textbf{Cat}$ as follows:

1. $\mathbb{S}_{\mathscr{M}}(X)$ is the poset $\mathscr{P}(\mathscr{M}(I, X) \times \mathscr{M}(X, I))$, ordered by inclusion. When $\varepsilon \in \mathbb{S}(X)$, we write $\varepsilon(x, k)$ if $(x, k) \in \varepsilon$, emphasizing our view of ε as a predicate. We call the elements of $\mathbb{S}_{\mathscr{M}}(X)$ *selection relations on X*.

2. For $f : X \to Y, E \in \mathbb{S}_{\mathscr{M}}(X)$, we define $Sel_{\mathscr{M}}(f)(\varepsilon) \in \mathbb{S}_{\mathscr{M}}(Y)$ by $\mathbb{S}_{\mathscr{M}}(f)(\varepsilon) = \{(x \,\mathring{\,}\, f, k) \mid \varepsilon(x, f \,\mathring{\,}\, k)\}$

We simply write $\mathbb{S}$ rather than $\mathbb{S}_{\mathscr{M}}$ when the category under consideration is obvious from context. It is straightforward to verify that the $\mathbb{S}(f)$ are functors (i.e. monotone maps), that $\mathbb{S}_{\mathscr{M}}(f \,\mathring{\,}\, g) = \mathbb{S}(f) \,\mathring{\,}\, \mathbb{S}(g)$, and $\mathbb{S}(1_X) = 1_{\mathbb{S}(X)}$. To simplify notation, we write f_* for $\mathbb{S}_{\mathscr{M}}(f)$.

Although we write this definition over an arbitrary monoidal category, we are mostly interested in the case where $\mathscr{M} = \textbf{Optic}(\mathscr{M}')$, where $\mathscr{M}'$ is a semicartesian symmetric monoidal category. Then, equivalently, we have that $\mathbb{S}_{\textbf{Optic}(\mathscr{M}')}\binom{X}{X'} \cong \mathscr{P}(\mathscr{M}'(I, X) \times \mathscr{M}'(X, X'))$.

Example 10. *Let $\mathscr{M} = \textbf{Lens}(\textbf{Set})$. Then for any set X, there is a selection relation $\arg\max_X$ on $\binom{X}{\mathbb{R}}$ defined by $\arg\max_X(x, k)$ iff $k(x) \geq k(x')$ for all $x' \in X$. Analogous selection relations also exist over many other suitable categories.*

The Nash product. There is a composition law for selection functions called the *Nash product*, generalising a construction that appeared in [Hed18] as the "sum of selection functions". Given a context on a tensor product $k : X \otimes Y \to I$, we can imagine two agents, one of whom controls the state $x : I \to X$ and

one who controls the state $y : I \to Y$. Suppose their decisions are governed respectively by the selection functions ε, δ. Given y, we can compose it with k to form $k_y := (1_X \otimes y) \, \mathbin{\hat{,}} \, k : X \to I$, and ask whether the first player is satisfied with their choice in this context. Analogously, we can ask whether the second player is satisfied with their choice in the context k_x given by x and k. If both are satisfied, the composite state $x \otimes y : I \to X \otimes Y$ is said to be an *equilibrium* — nobody wants to unilaterally deviate from it.

In a general (non-cartesian) monoidal category, there may be states $s : I \to X \otimes Y$ which do not have the form $x \otimes y$. We think of states of the form $x \otimes y$ as *independent*. The Nash product models a situation where players make their choices without communication, and so we conservatively rule out any non-independent states as Nash equilibria.

Proposition 11. *The functor* $\mathbb{S} : \mathscr{M} \to \mathbf{Cat}$ *admits a laxator* $\boxtimes : \mathbb{S}(X) \times \mathbb{S}(Y) \to \mathbb{S}(X \otimes Y)$ *given by*

$$\varepsilon \boxtimes \delta = \{(x \otimes y, k) \mid \varepsilon(x, k_y) \text{ and } \delta(y, k_x)\}$$

where $k_x : Y \cong I \otimes Y \xrightarrow{x \otimes I} X \otimes Y \xrightarrow{k} I$ *and* $k_y : X \cong X \otimes I \xrightarrow{I \otimes y} X \otimes Y \xrightarrow{k} I$.

Proof. It's straightforward to verify associativity and unitality of the lax monoidal structure, as well as commutativity. The nontrivial step is to verify that $\boxtimes$ is a natural transformation at all. To that end, let $f : X \to Y, f' : X' \to Y'$, and let $\varepsilon \in \mathbb{S}(X), \delta \in \mathbb{S}(X')$. Then we must verify $f_* \varepsilon \boxtimes f'_* \delta = (f \otimes f')_* (\varepsilon \boxtimes \delta)$.

This statement boils down to the fact that existential quantifiers commute. On the one hand, we have $(f \otimes f')_* (\varepsilon \boxtimes \delta)(s, k)$ if and only if there exists a factorization $s = t \, \mathbin{\hat{,}} \, (f \otimes f')$ and a further factorization $t = x \otimes x'$ so that $\varepsilon(x, (1_x \otimes x') \, \mathbin{\hat{,}} \, (f \otimes f') \, \mathbin{\hat{,}} \, k)$, and analogously for x'. On the other hand, we can ask that s factor as $y \otimes y'$ so that $f_* \varepsilon(y, (1_y \otimes y') \, \mathbin{\hat{,}} \, k)$ (and analogously the other one). This in turn means that y factors as $x \, \mathbin{\hat{,}} \, f$, and that $y' = x' \, \mathbin{\hat{,}} \, f'$, with $\varepsilon(x, f \, \mathbin{\hat{,}} \, (1_y \otimes (x' \, \mathbin{\hat{,}} \, f')) \, \mathbin{\hat{,}} \, k)$, which, applying the equations of a symmetric monoidal category, is the same as the condition above. Hence the two selection relations are equivalent as desired. $\square$

Example 12. *Let X, Y be sets. Then* $\mathrm{argmax}_X \boxtimes \mathrm{argmax}_Y$ *is a relation between $X \times Y$ and $X \times Y \to \mathbb{R}^2$. Specifically,* $(\mathrm{argmax}_X \boxtimes \mathrm{argmax}_Y)((x, y), k)$ *if and only if (x, y) is a pure strategy Nash equilibrium for the 2-player normal form game with payoff matrix k.*

Therefore, the functor $\mathbb{S} : \mathscr{M} \to \mathbf{Cat}$ is an indexed monoidal category. Such an indexed category is turned into a (strong) monoidal functor $\int \mathbb{S} \to \mathscr{M}$ by the monoidal Grothendieck construction [MV20]:

Definition 13. The category $\mathscr{M}_{\mathbb{S}} := \int \mathbb{S}$ has

1. objects given by pairs (X, ε) of an object of $\mathscr{M}$ together with a selection function $\varepsilon \in \mathbb{S}(X)$ on it.

2. morphisms $(X, \varepsilon) \to (Y, \delta)$ given by morphisms $f : X \to Y$ in $\mathscr{M}$ with the property that for all $h : I \to X$ and $k : Y \to I$, if $(h, f \, \mathbin{\hat{,}} \, k) \in \varepsilon$ then $(h \, \mathbin{\hat{,}} \, f, k) \in \delta$.

It is monoidal with unit $(I, \top_I)$ and product $(X, \varepsilon) \otimes (Y, \delta) = (X \otimes Y, \varepsilon \boxtimes \delta)$. The projection $\pi : \mathscr{M}_{\mathbb{S}} \to \mathscr{M}$ is trivially strong monoidal.

6   Open Games

In this section we will equip parametrised optics with selection relations on the parameters. This results in a category in which we can do 'compositional game theory', which refines open games [GHWZ18] by adding an explicit account of agents. More detail on this perspective can be found in [CGLF21].

Let $\mathscr{C}$ be an $\mathscr{M}$-actegory. Since the forgetful functor $\pi : \mathscr{M}_{\mathbb{S}} \to \mathscr{M}$ is strong monoidal, $\mathscr{C}$ is also an $\mathscr{M}_{\mathbb{S}}$-actegory with action given by $(X, \varepsilon) \bullet_{\mathbb{S}} Y := X \bullet Y$. We can then use this action to form the

symmetric monoidal bicategory $\mathbf{Para}_{\bullet_{\mathbb{S}}}(\mathscr{C})$ as per Definition 2. In particular, morphisms of this category will be parametrised not just by objects of $\mathscr{M}$ but also by selection relations on them, which specify preferences on the parameter space. Now let $\mathscr{C}$ and $\mathscr{D}$ be monoidal $\mathscr{M}$-actegories. Then $\mathbf{Optic}(\mathscr{C}, \mathscr{D})$ is monoidal and instantiating the above construction for the action of $\circledast$ described in Proposition 7 gives us a monoidal category $\mathbf{Para}_{\circledast_{\mathbb{S}}}(\mathbf{Optic}(\mathscr{C}, \mathscr{D}))$. We think of this as a category of "open games". Although it is closely related to existing definitions of open games [GHWZ18, BHZ19, AGG$^+$20], it differs in several key ways.

The parameter object M is an object of $\mathbf{Optic}(\mathscr{M})_{\mathbb{S}}$, hence a pair $\left(\begin{smallmatrix}\Omega \\ \mho\end{smallmatrix}\right)$[6], equipped with a selection relation $\varepsilon \in \mathbb{S}_{\mathscr{M}}\left(\begin{smallmatrix}\Omega \\ \mho\end{smallmatrix}\right)$. Ω is the object of 'strategy profiles' found in other definitions of open games (usually called Σ). We refer to $\mho$ as the set of 'rewards', or 'intrinsic utilities', that the agents playing a game actually optimise. Despite having such a clear game-theoretic reading, agents' rewards were left implicit in previous formulations of open games.

When $\mathscr{C} = \mathscr{D} = \mathscr{M}$ is a semicartesian monoidal category, acting on itself by monoidal product, then a scalar (morphism $I \to I$) in $\mathbf{Para}_{\otimes_{\mathbb{S}}}(\mathbf{Optic}(\mathscr{M}))$ consists of a pair of parameter objects $\left(\begin{smallmatrix}\Omega \\ \mho\end{smallmatrix}\right)$, a selection relation $\varepsilon \subseteq \mathscr{M}(I, \Omega) \times \mathscr{M}(\Omega, \mho)$, and a morphism $k : \mathscr{M}(\Omega, \mho)$. We can then form the set $\{\omega \in \mathscr{M}(I, \Omega) \mid \varepsilon(\omega, k)\}$. We think of this as the *solution set* of the game: it is the set of strategy profiles that the agents accept in the context given by the game they are playing.

The simplest possible case is when $\mathscr{C} = \mathscr{D} = \mathscr{M} = \mathbf{Set}$, acting on itself by cartesian product. In this case, a morphism $\left(\begin{smallmatrix}X \\ S\end{smallmatrix}\right) \xrightarrow{(\Omega, \mho)} \left(\begin{smallmatrix}Y \\ R\end{smallmatrix}\right)$ consists of a pair of sets $\left(\begin{smallmatrix}\Omega \\ \mho\end{smallmatrix}\right)$, a selection function $\varepsilon \subseteq \Omega \times (\Omega \to \mho)$, and a lens $\mathscr{G} : \left(\begin{smallmatrix}\Omega \\ \mho\end{smallmatrix}\right) \otimes \left(\begin{smallmatrix}X \\ S\end{smallmatrix}\right) \to \left(\begin{smallmatrix}Y \\ R\end{smallmatrix}\right)$. This in turn consists of a "play function" $P : \Omega \times X \to Y$, and a function $\Omega \times X \times R \to \mho \times S$ that splits into a "coplay function" $C : \Omega \times X \times R \to S$ and an "intrinsic utility function" $U : \Omega \times X \times R \to \mho$.[7] This data determines an open game in $\mathbf{OG}\left(\left(\begin{smallmatrix}X \\ S\end{smallmatrix}\right), \left(\begin{smallmatrix}Y \\ R\end{smallmatrix}\right)\right)$, where $\mathbf{OG}$ is the usual category if open games in the sense of [GHWZ18]. Specifically, we keep the set of strategy profiles, the play function and the coplay function the same. It remains to choose an equilibrium function $E : X \times (Y \to R) \to \mathscr{P}(\Omega)$. We define it by $E(h, k) = \{\omega \mid \varepsilon(\omega, K_{h,k})\}$, where $K_{h,k}$ is given by the following composition in $\mathbf{Lens}(\mathbf{Set})$:

$$\begin{pmatrix}\Omega \\ \mho\end{pmatrix} \xrightarrow{\cong} \begin{pmatrix}\Omega \\ \mho\end{pmatrix} \otimes I \xrightarrow{\left(\begin{smallmatrix}\Omega \\ \mho\end{smallmatrix}\right) \otimes h} \begin{pmatrix}\Omega \\ \mho\end{pmatrix} \otimes \begin{pmatrix}X \\ S\end{pmatrix} \xrightarrow{\mathscr{G}} \begin{pmatrix}Y \\ R\end{pmatrix} \xrightarrow{k} I$$

This construction defines a strong monoidal functor $\mathbf{Para}_{\circledast_{\mathbb{S}}}(\mathbf{Lens}(\mathbf{Set})) \to \mathbf{OG}$, and can also be carried out for Bayesian open games and other more general formulations of open games.

As a worked example, we will demonstrate how this formulation of game theory improves on open games by describing a prisoner's dilemma and then modifying it so that both decisions are made by the same agent. For simplicity, our base will again be $\mathbf{Set}$ acting on itself by cartesian product, which describes deterministic games and pure strategy Nash equilibria. Given sets X and Y, a decision to choose an element of Y after observing an element of X and with a real number payoff is described by a morphism $\left(\begin{smallmatrix}X \\ 1\end{smallmatrix}\right) \to \left(\begin{smallmatrix}Y \\ \mathbb{R}\end{smallmatrix}\right)$ in $\mathbf{Para}(\mathbf{Optic}(\mathbf{Set}))$, whose parameter sets are $\left(\begin{smallmatrix}X \to Y \\ \mathbb{R}\end{smallmatrix}\right)$, where the play function is function application $(X \to Y) \times X \to Y$, and the intrinsic utility function is the projection $(X \to Y) \times X \times \mathbb{R} \to \mathbb{R}$. Note that we have separated the concept of a decision in a game, where a choice is made and a real number payoff is obtained, from the goal of the agent to maximise that payoff, or the mechanism by which they do so. This aligns with classical game theory, but is in contrast to open games, in which agents and decisions are conflated. In the case that X has one element, describing a decision with no

---

[6] We pronounce the symbol $\mho$ as "coméga".

[7] In [FST19] this function is called 'update'.

observation, decisions are particularly simple and can be described by a bending wire, as illustrated in figure 8.
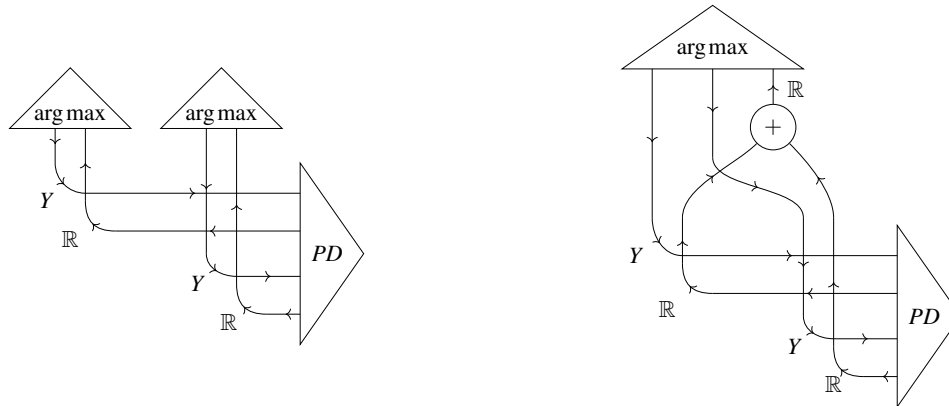

Figure 8: (a) Standard prisoner's dilemma   (b) Modified prisoner's dilemma

We take $Y = \{C, D\}$ to be the set of moves in prisoner's dilemma. We take the tensor product of two decisions, describing that they are made in parallel. We then postcompose with the morphism $\binom{Y^2}{\mathbb{R}^2} \to I$ in $\mathbf{Para}_{\times_{\mathbb{S}}}(\mathbf{Lens}(\mathbf{Set}))$ given by lifting the payoff function $PD : Y^2 \to \mathbb{R}^2$ of prisoner's dilemma to a costate in $\mathbf{Optic}(\mathbf{Set})$, taking the parameter sets to be $\binom{\Omega}{\mho} = (1, 1) = I$. This results in a scalar $I \to I$ in $\mathbf{Para}_{\times_{\mathbb{S}}}(\mathbf{Optic}(\mathbf{Set}))$, with parameters $\binom{Y^2}{\mathbb{R}^2}$, described by the lower half of figure 8(a). Ultimately, this determines a costate $\binom{Y^2}{\mathbb{R}^2} \to I$ in $\mathbf{Lens}(\mathbf{Set})$, namely another copy of the payoff function PD.

In order to describe the standard prisoner's dilemma, we compose this diagram with a pair of arg max operators, describing the situation where a pair of players independently optimise their own payoff, without communicating. The lack of communication can be seen in the 'air gap' in the top part of figure 8(a).[8] The composition of a pair of arg max operators involves the Nash product, and in the end applying the costate PD to the selection function $\arg\max_Y \boxtimes \arg\max_Y$ determines the solution set, namely the set $\{(D, D)\}$, which is the unique Nash equilibrium of the prisoner's dilemma.

Now consider the modified prisoner's dilemma denoted by figure 8(b). This can be described in two different ways, which are equivalent but not the same: namely as the prisoner's dilemma pulled back along an optic on parameters, coupled to the arg max selection function describing a classical optimising player; or alternatively as the ordinary prisoner's dilemma coupled to a selection function obtained by pushing forwards arg max along the same optic, describing a player who makes two choices to maximise the sum of two payoffs. Such a strategy profile is known as Hicks optimal, a strengthening of Pareto optimality. For the prisoner's dilemma, the solution set of this example is $\{(C, C)\}$. Notably this is disjoint to the set of Nash equilibria.

Although this is only a very simple example, separating an open game into these 'horizontal' and 'vertical' parts provides a very flexible language for describing a wide variety of game-theoretic situations, such as those described in [CGLF21].

---

[8]The top parts of figure 8 are currently informal: we are drawing the selection functions as though they are states in $\mathbf{Lens}(\mathbf{Set})$, but they are not. It may be possible, by thinking of them as 'generalised states', to embed (perhaps by co-Yoneda) into a larger category in which they are states; alternatively the top parts can be thought of as 'just' a decoration of the top boundary.

References

[ACB17]   Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein GAN. *arXiv e-prints*, page arXiv:1701.07875, January 2017.

[AGG⁺20]   Robert Atkey, Bruno Gavranović, Neil Ghani, Clemens Kupke, Jérémy Ledent, and Fredrik Nordvall Forsberg. Compositional game theory, compositionally. Presented at ACT2020, 2020.

[BHZ19]   Joe Bolt, Jules Hedges, and Philipp Zahn. Bayesian open games. arXiv:1910.03656, 2019.

[CEG⁺20]   Bryce Clarke, Derek Elkins, Jeremy Gibbons, Fosco Loregian, Bartosz Milewski, Emily Pillmore, and Mario Román. Profunctor optics: A categorical update. arXiv:2001.07488, 2020.

[CGG⁺21]   G. S. H. Cruttwell, Bruno Gavranović, Neil Ghani, Paul Wilson, and Fabio Zanasi. Categorical foundations of gradient-based learning, 2021.

[CGLF21]   Matteo Capucci, Neil Ghani, Jérémy Ledent, and Fredrik Nordvall Forsberg. Translating extensive form games to open games with agency. Forthcoming, 2021.

[EO11]   Martin Escardó and Paulo Oliva. Sequential games and optimal strategies. *Proceedings of the Royal Society A*, 467:1519–1545, 2011.

[FST19]   Brendan Fong, David Spivak, and Rémy Tuyéras. Backprop as functor: A compositional perspective on supervised learning. In *Proceedings of Logic in Computer Science (LiCS) 2019*. ACM, 2019.

[Gav21a]   Bruno Gavranović. Animation of optic composition. `https://twitter.com/bgavran3/status/1366202140788731908`, 2021.

[Gav21b]   Bruno Gavranović. Animation of optics. `https://twitter.com/bgavran3/status/1364644337968103428`, 2021.

[GHWZ18]   Neil Ghani, Jules Hedges, Viktor Winschel, and Philipp Zahn. Compositional game theory. In *Proceedings of Logic in Computer Science (LiCS) 2018*, pages 472–481. ACM, 2018.

[GPAM⁺14] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[Hed18]   Jules Hedges. Backward induction for repeated games. In *Proceedings of Mathematically Structured Functional Programming (MSFP) 2018*, volume 275 of *Electronic Proceedings in Theoretical Computer Science*, pages 35–52, 2018.

[HOS⁺17a] Jules Hedges, Paulo Oliva, Evguenia Shprits, Viktor Winschel, and Philipp Zahn. Higher-order decision theory. In Jörg Rothe, editor, *Algorithmic Decision Theory*, volume 10576 of *Lecture Notes in Artificial Intelligence*, pages 241–254. Springer, 2017.

[HOS⁺17b] Jules Hedges, Paulo Oliva, Evguenia Shprits, Viktor Winschel, and Philipp Zahn. Selection equilibria of higher-order games. In *Practical aspects of declaritive languages*, volume 10137 of *Lecture Notes in Computer Science*, pages 136–151. Springer, 2017.

[Lor20]   Fosco Loregian. *Coend calculus*. 2020. arXiv:1501.02503.

[MSN]   David Jaz Myers, David I Spivak, and Nelson Niu. *Categorical Systems Theory*.

[MV20]   Joe Moeller and Christina Vasilakopoulou. Monoidal Grothendieck construction. *Theory and applications of categories*, 35(31):1159–1207, 2020.

[Mye18]   David Jaz Myers. String diagrams for double categories and equipments. arXiv:1612.02762, 2018.

[Mye20]   David Jaz Myers. Double categories of open dynamical systems. arXiv:2005.05956, 2020.

[RC21]   Mitchell Riley and The CT Zulip Community. An involution on 'learn'. `https://mattecapu.github.io/ct-zulip-archive/stream/229156-practice:-applied-ct/topic/An.20involution.20on.20'Learn'.html#235034768`, 2021.

[VSL14]   Dmitry Vagner, David I Spivak, and Eugene Lerman. Algebras of open dynamical systems on the operad of wiring diagrams. *arXiv preprint arXiv:1408.1598*, 2014.

[Wat13]   Joel Watson. *Strategy: An introduction to game theory*. W. W. Norton, 2013.