J. Hedges
N. Ghani

A Compositional Approach to Economic Game Theory

Abstract. This paper aims to significantly impact the way we conceive of, reason about, and construct software for economic game theory. We achieve this by building an new and original model of economic games based upon the computer science idea of *compositionality*: concretely we i) give a number of operators for building up complex and irregular games from smaller and simpler games; and ii) show how the equilibria of these complex games can be defined recursively from the equilibria of their simpler components. We apply compositional reasoning to sophisticated games where agents must reason about how their actions affect future games and how those future games effect the utility they receive. This requires further originality — we augment the usual lexicon of games with a new concept of *coutility*. This dual notion to utility arises because, in order for games to accept utility, this utility must be generated by other games. A third source of originality is the representation of our games as string diagrams so as to give a clear visual interface to manipulate them. Our fourth, and final, source of originality is a categorical formalisation of these intuitive string diagrams which guarantees our model is mathematically rigourous.

Keywords: Game theory, Economic equilibria, String diagrams, Monoidal categories, Selection functions

1. Introduction

We must all grapple with this important question:

> *How can we ensure that our theories of the world scale from the small examples in our classrooms or on our blackboards, to the huge and complex systems that appear in the real world?*

One answer — the brute force answer — is to simply be very, very good at applying our theories. For example in computer science, many researchers are grappling with the phenomena of *big data* by building faster and faster computers. The equivalent within economic game theory might be to represent complex games as huge pay-off matrices of actions and associated utilities and use some of those very fast computers to compute the associated Nash equilibria of these games. An alternative to brute force techniques, widely regarded within computer science as being best practice, is *compositonality* where one sees complex systems as being built from smaller subsystems.

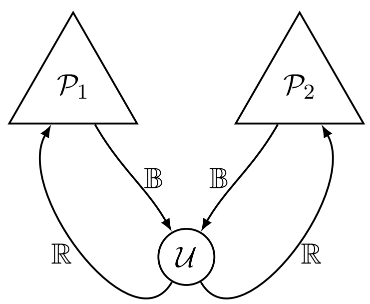
Presented by **Name of Editor**; *Received* December 1, 2005

Because these subsystems are smaller, they are easier to reason about. One then combines properties of these subsystems into properties of the overall system. Compositionality also promotes *reuse*: if a particular system is a subsystem of many other systems, then any results concerning that system do not need to be regenerated whenever compositional reasoning takes place on any super-system of that system: this further promotes efficient reasoning. Examples of compositional reasoning are numerous, e.g.

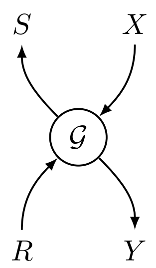
- Within logic, the truth of a formulae $A \wedge B$ is computed from the truth of A and B which, in turn, are computed recursively.
- Within programming languages, the effect of executing a programme p *then* q is computed by combining the effect of executing p with the effect of executing q. These, in turn, are computed recursively.
- Within concurrency, Milner's highly influential *Communicating Sequential Processes* proposes operators for building processes. As above, properties of processes are proved by combining properties of sub-processes.

Can the gains of compositionality within Computer Science be replicated within economic game theory? After all, not all reasoning can be put in compositional form, especially if there is significant emergent behaviour present in a large system which is not present in subsystems. And, such emergent behaviour is certainly present within economic games. For example, if σ is an optimal strategy for a game G, then is σ part of an optimal strategy for $G * H$ where H is another game and $G * H$ is some super-game built from G and H? Clearly the answer is not necessarily! More concretely, the iterated prisoner's dilemma has many equilibria (such as cooperative equilibria) that do not arise from repeatedly playing the Nash equilibrium of the stage game. Indeed, the difficulty in constructing a compositional model of economic game theory led one econometrician present at a recent talk on this work to exclaim "*This can't be done!*" while another described this as "*a conjuring trick*" in that it seemed to achieve the impossible!

So, how do we achieve the impossible? Our approach to cutting this particular Gordian knot is to observe that, at its essence, any compositional model of game theory must model not just each game, but the interaction of each game with all other games it might interact with. Of course, all current models of economic games do not attempt this endeavour with good reason — as just demonstrated, this interaction is non-trivial and a priori, there is a huge number of such games - indeed a proper class! Our solution — and the crucial technical insight underpinning our model — is that it suffices to replace the class of games which a given game might interact with the utility functions of those games. Put simply, standard models of game theory hardwire a specific utility function into the definition of a game while the novelty of our model is that each game is parameterised by the space of *all possible utility* functions. This is accomplished by embedding within each game "ports" where interaction with other games takes place in the form of sending the environment information about moves played and receiving utility from the environment. This leads to a second innovation within this paper: we introduce a new representation of games which complements the usual representation of games via trees or pay-off matrices. This new representation of games uses string diagrams because they enable us to clearly visualise the structure of complex games via the places where the ports of one game are glued together with the ports of other games. For example, a bimatrix game such as the Prisoner's Dilemma can be represented as the following string diagram



where the players are marked P_1 and P_2, the arrow out of each player represents their moves, and these are fed into $\mathcal{U}$ which computes utility and then feeds that utility back to each player. Switching to string diagrams - as demonstrated in the Prisoner's Dilemma example - gives an intuitively clear and simple presentation of the key components of a game and their interaction via the flow of information between them. Now, in a compositional model, games are built by operators combining simpler games which we call *pregames*. These pregames have the general form



where i) X is the type of the current state of the world; ii) Y is the type of possible actions or moves; iii) R is the type of utilities or payoffs; and iv) S is the type of the coutility extruded back to the environment. This concept of *coutility* is another of our contributions — coutility arises as, in a closed system, the utility of one player must have been generated by another player. For example, if I place a bet with a bookmaker and expect utility in the form of winnings, those winnings will have to come from the bookmaker. This we call the bookmakers coutility. Pregames also possess functions describing how moves are chosen, exactly how much coutility to extrude, and a relation describing its equilibria. Crucially, the equilibria relation can be *any relation* and need not be restricted to Nash equilibria. The scientific value of these pregames is that they form a compositional model of economic games in that we can provide operators for building complex pregames from simpler pregames and, in particular, for defining the equilibria of these complex pregames from the equilibria of their component pregames. Our final contribution is a categorical interpretation of pregames and their string diagram representations which ensures they have precise mathematical meanings and thus that our reasoning about them is mathematically rigorous.

Related Work: The context of this paper begins with [12], which for the first time approached game theory using ideas from program semantics. Nevertheless, the game theory developed in that paper is no more compositional than ordinary game theory. A key point of originality within our approach is the use of continuations to allow players to consider the possible outcomes of decisions and factor those possible outcomes into their decision making process. Continuations are implicit in the literature on selection functions (see [6] for an introduction). Pregames are also closely related to the 'partially defined games' of [10].

The relationship between string diagrams and monoidal categories has been developed extensively in quantum information theory [2] and bialgebra [4], and also applied to distributional semantics in linguistics [5]. There are many variants of monoidal categories, each with its own associated string language, surveyed in [15]. The language used in this paper is a fragment of the one for compact closed categories (section 4.8 of *loc. cit.*). We use a fragment because, within game theory, there is only a limited form of backward-causality (arising from rational agents reasoning about potential future values).

The paper is structured as follows: Section 2 introduces some of the notation used in this paper while Section 3 demonstrates the feasibility of a compositional model of game theory by considering a simple class on non-dynamic games. Section 4 introduces pregames and their representation as string diagrams, Section 5 defines the key operators of parallel and sequential composition of pregames, Section 6 gives a rigorous mathematical foundation for pregames and their representation as string diagrams, and Section 7 extends previous work to cover the infinite iteration of pregames. Finally, Section 8 discusses the choices inherent within our compositional model and directions for further research.

2. Preliminaries

We denoted by $\mathbb{R}$ the set of real numbers. We also denote by 1 the set containing only the element $*$ - when a function requires an input of type 1, we often elide that input to increase legibility. We use the λ-calculus notation $\lambda x.t$ to denote the function which takes x as input and returns the value of t — this will of course usually depend upon x. If f is a function in two variables, we often write $f(-,x)$ and $f(x,-)$ respectively for the function in one variable obtained by instantiating the first and second inputs respectively of f to be x. We also follow computer science notation and write $f\ x$ — as opposed to $f(x)$ — for the application of a function f to an input x. Composition of functions is written $f \circ g$ and projections are denoted $\pi_1 : A \times B \to A$ and $\pi_2 : A \times B \to B$. We denote by $\mathcal{P}$ the powerset functor on sets. At the centre of game theory is the desire to maximise utility and so we define, for any function $f : A \to B$ (where B is a preordered set), the set $\mathsf{argmax} f$ of inputs where f attains a maximal value.

$$\mathsf{argmax}\ f = \{a \in A \mid \forall a' \in A. f a \geq f a'\}$$

We assume familiarity with basic category theory and final coalgebras respectively as can be found in [**?**, **?**]. Readers not familiar with these topics can simply skip the technical details of Section 6 and Section 7 respectively. Where we use final coalgebras, we give concrete descriptions of them.

3. Simple Games, Compositionality and Nash Equilibria

To ensure accessibility of our methods to non-computer scientists, we begin by treating simple games where players choose actions and utility is then generated. Without loss of generality, we consider just two players

Definition 1 (Simple 2-player Game). A simple two-player game G consists of a 5-tuple $(Y_1, Y_2, R_1, R_2, k : Y_1 \times Y_2 \to R_1 \times R_2)$, where Y_i are the sets of moves available to each player, R_i are the types of utility expected by each players, and k computes the utility associated to each pair of moves. Let $k_1 = \pi_1 \circ f$ and $k_2 = \pi_2 \circ f$. Then, the set of *Nash equilibria* $E_G \subseteq Y_1 \times Y_2$ is defined by

$$(y_1, y_2) \in E \quad \text{iff} \quad y_1 \in \mathsf{argmax}\, k_1(-, y_2) \\ \wedge\, y_2 \in \mathsf{argmax}\, k_2(y_1, -)$$

This definition of Nash equilibrium is not compositional: it is not derived from more primitive concepts but rather is postulated as being itself a primitive concept whose myriad of different applications, and conceptual simplicity, underpin its value. While of course accepting its usefulness, and conceptual simplicity, we do believe it can be derived in a compositional manner from the interaction of the first player with the second player. Substantiating this belief will show that a compositional model of economic game theory is possible and what it might look like. First, consider the players:

Definition 2 (Simple Game). A simple game consists of a set Y of moves, a set R of utilities and an equilibria function $E : (Y \to R) \to \mathcal{P}Y$. The set of simple players with actions Y and utilties R is written $\mathsf{SP}_R Y$

This definition leaves equilibria abstract - they may maximise utility from a choice of move or, as in say the El Farrol Bar game, they may reflect other criteria. (The functions E are precisely the multi-valued selection functions [8].) Thus, an equilibrium $y \in Ek$ means that if the utility in the game is given by the context k, then there is no move preferable to y. And, crucially, the equilibria of the game are not given for a specific utility function, but are given for every utility function! In computer science this is called 'continuation passing style'. Now, recall the essence of compositionality is to build new games from old in such a way that properties of the former can be derived from those of the later. We now introduce such an operator

Definition 3 (Monoidal Product of Simple Games). Let $G_1 \in \mathsf{SP}_{R_1} Y_1$ and $G_2 \in \mathsf{SP}_{R_2} Y_2$ be simple games. Their monoidal product is the simple game

$G_1 \otimes G_2 : \mathsf{SP}_{R_1 \times R_2} Y_1 \times Y_2$ with equilibrium function

$$(y_1, y_2) \in E_{G_1 \otimes G_2} k \quad \text{iff} \quad y_1 \in E_1(\pi_1 \circ k(-, y_2)) \\ \wedge\, y_2 \in E_2(\pi_2 \circ k(y_1, -))$$

Although the Nash equilibria of a simple 2-player game appeared initially to be a primitive and non-compositional concept, we can now show it arises compositionally. The proof is by unwinding definitions.

THEOREM 4. *Let $G = (Y_1, Y_2, R_1, R_2, k)$ be a simple 2-player game. Then $(y_1, y_2) \in E_G$ iff $(y_1, y_2) \in E_{G_1 \otimes G_2} k$ where G_1 and G_2 are the simple games $G_1 = (Y_1, R_1, \mathsf{argmax})$ and $G_2 = (Y_2, R_2, \mathsf{argmax})$.*

Note that in the simple games G_1 and G_2, the 'rational behaviour' of the players is to maximise (where 'rational' is taken to mean 'playing according to their equilibrium'). However the rational behaviour of players in $G_1 \otimes G_2$ may not be globally maximising – this is precisely what the Prisoner's Dilemma illustrates. Thus maximising behaviour is not compositional, so the generalisation from argmax to the equilibrium function E is essential. Also notice how in Theorem 4 it is essential to not hardwiring a utility function into a simple game but rather define the equilibria for *all* possible utility functions. Finally, notice how the equilibria of $G_1 \otimes G_2$ can be computed compositionally from only the equilibria of G_1 and G_2, even if G_1 and G_2 are highly complex games. That is, one need not delve into the definitions of G_1 and G_2 to see how they are built. In the rest of this paper, we generalise this basic compositional model economic game theory to a more sophisticated compositional model encompassing a much broader class of games.

4. Pregames

Section 3 defined a compositional model of economic game theory. That is, we defined i) the notion of a simple game ii) an operator for building new simple games from old; and iii) the equilibria for compound simple games in terms of the equilibria of their component simple games. However, simple games possess limited structure and hence support limited operators — this is a problem, as more operators enable more compositional reasoning! Therefore, we introduce a more sophisticated compositional model which finds a sweet spot of being expressive enough to model complex and highly irregular games while retaining enough simplicity to continue to deliver conceptual clarity. As motivation, consider me placing a bet with a bookmaker:

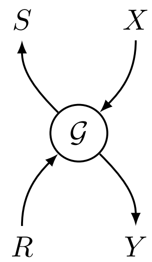
- My state informs my actions, e.g. my wealth affects the size of my bet. Secondly, I have different strategies, e.g. high-risk, long-odds bets, or low-risk, short-odds bets. Strategy and state are conceptually different as I can choose my strategy but not my state. Together, a state and a strategy determine the action taken, i.e. which bet I place. The environment, i.e. the bookmaker, returns my winnings.
- The bookmaker also has a state: in this case, the bet placed. They too have a number of strategies, e.g. to refuse the bet, to accept the bet, or to lay the bet. Again, their state and strategy determine their action. And, again, it is the environment that converts their actions into results and hence utility. Finally, the bookmaker will return winnings to the better — we call this *coutility* since it is the source of the better's utility.

The above example motivates our more general definition:

Definition 5 (Pregame). Let X, Y, R and S be sets. A *pregame* G over X, Y, R, S consists of the following data:

- A set $\Sigma_\mathcal{G}$ of *strategy profiles*
- A *play* function $P_\mathcal{G} : \Sigma_\mathcal{G} \to (X \to Y)$
- A *coutility* function $C_\mathcal{G} : \Sigma_\mathcal{G} \to (X \times R \to S)$
- An *equilibrium function* $E_\mathcal{G} : X \times (Y \to R) \to \mathcal{P}\Sigma_\mathcal{G}$.

For reasons that will become apparent later, we denote such a pregame $\mathcal{G} : (X, S) \to (Y, R)$. If $\sigma \in E_G(x, k)$, then we imagine there is no better strategy than σ if we are in state x and $k : Y \to R$ computes the utilities arising from each action. We also represent pregames as string diagrams containing ports where interaction with other games takes place.



Example 6 (Prisoners Dilemma and Other Simple Games). Let M be the set of moves in the Prisoners Dilemma game, i.e. $M = \{C, D\}^2$. Define the

pregame $\mathsf{PD} : (1, \mathbb{R} \times \mathbb{R}) \to (M, \mathbb{R} \times \mathbb{R})$ with strategies $\Sigma_{\mathsf{PD}} = M$ and with play function (eliding the unit type) choosing the strategy, i.e.

$$P_{\mathsf{PD}}\ m = m$$

Coutility is $C_{\mathsf{PD}}\ m\ r = r$, while

$$\begin{aligned}(\sigma_1, \sigma_2) \in E_{\mathsf{PD}} k \quad \mathsf{iff} \quad & \sigma_1 \in \mathsf{argmax}\ \pi_1 \circ k(-, \sigma_2) \\ & \wedge \sigma_2 \in \mathsf{argmax}\ \pi_2 \circ k(\sigma_1, -)\end{aligned}$$

The Prisoners Dilemma is a simple game (see Defn. 2) and in fact all simple games are pregames where $X = S = 1$, $\Sigma = Y$ and P is the identity function.

We can also define a two round Prisoners Dilemma:

Example 7 (2-Round Prisoners Dilemma). Again, let $M = \{C, D\}^2$ be the set of moves in the Prisoners Dilemma game. We define a pregame $\mathsf{2PD} : (1, \mathbb{R} \times \mathbb{R}) \to (M \times M, \mathbb{R} \times \mathbb{R})$ which represents two rounds of the Prisoners Dilemma Game. The strategy set for the game ought to be a move for the first round together with a function which gives a round 2 move for each round 1 moves. That is $\Sigma_{\mathsf{2PD}} = M \times (M \to M)$. The play function is

$$P_{\mathsf{2PD}}\ (\sigma_1, \sigma_2) = (\sigma_1, \sigma_2 \sigma_1)$$

Coutility is similar to that above, while

$$\begin{aligned}(\sigma_1, \sigma_2) \in E_{\mathsf{2PD}}\ k\ \mathsf{iff}\ & \sigma_1 \in E_{\mathsf{PD}}\ k(-, \sigma_2 -) \quad \wedge \\ & \forall \sigma \in M.\ \sigma_2 \sigma \in E_{\mathsf{PD}}\ k(\sigma, -)\end{aligned}$$

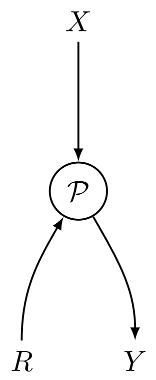
Note σ_2 must compute an optimal strategy for all round-1 plays — this accords with the no-incredible-threats principle of subgame perfect equilibria. Our definition of 2PD is compositional: subgame perfect equilibrium for 2PD consist of a Nash equilibrium in the first round (for a particular utility function defined using the second round strategies), together with a Nash equilibrium in the second round for each first round play (with another utility function defined by that first round play). We now show how we can define operators that build complex pregames from simpler pregames.

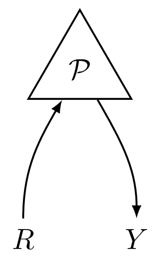
5. Operators on Pregames

We introduce operators for building pregames. The first three build atomic pregames while the last two build compound pregames from other pregames:

Definition 8 (Players). A player who observes a state of type X, makes a choice of type Y and optimises an outcome of type R is represented by a pregame $\mathcal{P} : (X, 1) \to (Y, R)$, where the set of strategies is $\Sigma_{\mathcal{P}} = X \to Y$, i.e. mappings from states to choices. The play function $P_{\mathcal{P}}(\sigma, x) = \sigma(x)$ applies the strategy to the state, while the coutility function is trivial. In general, the equilibrium function can be varied in different examples to give different goals to the player.

If the player does not make any observation then $X = 1$. In the two cases $X \neq 1$ and $X = 1$, a player is represented by a string diagram as





(triangles traditionally denote components of string diagrams that have connections on only one side). For an example of a classical, utility-maximising player, take $R = \mathbb{R}$ with the equilibrium relation being given by

$$E_{\mathcal{P}}\ (x, k) = \{\sigma : X \to Y \mid \sigma(x) \in \mathsf{argmax}\ k\}$$

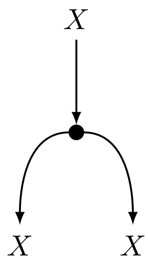
If $X = 1$, this simplifies to

$$E_{\mathcal{P}}\ (*, k) = \mathsf{argmax}\ k$$

Definition 9 (Computations). A function $f : X \to Y$ defines a two pregames, a 'covariant computation' $f : (X, 1) \to (Y, 1)$ and a 'contravariant computation' $f^* : (1, Y) \to (1, X)$. These are used to represent aspects of a game that are not players (more precisely, that cannot make strategic choices), e.g. utility functions. In both cases we have $\Sigma = 1$, and define E_f and E_{f^*} to always return $\{*\}$. In the covariant case, we set $P_f(*, x) = fx$, while in the contravariant case we set $C_{f^*}(*, *, x) = fx$.

Covariant and contravariant computations are respectively drawn as

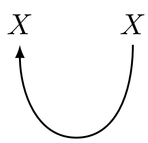
An important example of a covariant computation is copying $\Delta_X : (X, 1) \to (X \times X, 1)$ arising from the function $\Delta_X : X \to X \times X$ which copies its argument. Within games, this allows information to be used more than once, for example both being observed by a player, and used as input to a utility function.



Our final basic constructor for pregames is the *Teleological Counit* which mediates between the forward direction of information (chose strategy, play move) and the backward flow of information (a process generates coutility and passes it to a previous process as utility). More concretely, the teleological unit — in combination with a computation — allows us to specify a particular utility function to be used in the pregame. Of course, this curtails the possibility of interacting with other pregames by closing off the ports where that interaction can happen:

Definition 10 (Teleological counit). The pregame $\tau_X : (X, X) \to (1, 1)$ is given by $\Sigma_{\tau_X} = 1$, $C_{\tau_X}(*, x, *) = x$ and with E_{τ_X} always returning $\{*\}$.

We graphically represent the teleological counit by a *cup*



After introducing three basic operators for pregames, we now present two compound operators — the monoidal product which generalises the similar operator of Section 3, and sequential composition.

Definition 11 (Monoidal Product of Pregames). If $\mathcal{G} : (X_1, R_1) \to (Y_1, S_1)$ and $\mathcal{H} : (X_2, S_2) \to (Y_2, R_2)$ are pregames, their monoidal product

$$\mathcal{G} \otimes \mathcal{H} : (X_1 \times X_2, S_1 \times S_2) \to (Y_1 \times Y_2, R_1 \times R_2)$$

is defined by

- $\Sigma_{\mathcal{G}\otimes\mathcal{H}} = \Sigma_\mathcal{G} \times \Sigma_\mathcal{H}$
- $P_{\mathcal{G}\otimes\mathcal{H}}(\sigma_1, \sigma_2)(x_1, x_2) = (P_\mathcal{G}\ \sigma_1\ x_1, P_\mathcal{H}\ \sigma_2\ x_2)$
- $C_{\mathcal{G}\otimes\mathcal{H}}(\sigma_1, \sigma_2)((x_1, x_2), (r_1, r_2)) = (C_\mathcal{G}(\sigma_1(x_1, r_1), C_\mathcal{H}(\sigma_2)(x_2, r_2))$
- $(\sigma_1, \sigma_2) \in E_{\mathcal{G}\otimes\mathcal{H}}\ ((x_1, x_2),\ k)$ iff $\sigma_1 \in E_\mathcal{G}\ (x_1,\ k_1)$ and $\sigma_2 \in E_\mathcal{H}\ (x_2,\ k_2)$ where
$$k_1(y_1) = (\pi_1 \circ k)(y_1, P_\mathcal{H}(\sigma_2)(x_2))$$
$$k_2(y_2) = (\pi_2 \circ k)(P_\mathcal{G}(\sigma_1)(x_1), y_2)$$

Definition 12 (Sequential Composotion). Let $\mathcal{G} : (X, T) \to (Y, S)$ and $\mathcal{H} : (Y, S) \to (Z, R)$ be pregames. The composition $\mathcal{H} \circ \mathcal{G} : (X, T) \to (Z, R)$ is defined by

- $\Sigma_{\mathcal{H}\circ\mathcal{G}} = \Sigma_\mathcal{G} \times \Sigma_\mathcal{H}$
- $P_{\mathcal{H}\circ\mathcal{G}}(\sigma_1, \sigma_2) = P_\mathcal{H}(\sigma_2) \circ P_\mathcal{G}(\sigma_1)$
- $C_{\mathcal{H}\circ\mathcal{G}}(\sigma_1, \sigma_2)(x, r) = C_\mathcal{G}(\sigma_1)(x, C_\mathcal{H}(\sigma_2)(P_\mathcal{G}(\sigma_1)(x), r)))$
- $(\sigma_1, \sigma_2) \in E_{\mathcal{H}\circ\mathcal{G}}(x, k)$ iff
  1. $\sigma_1 \in E_\mathcal{G}(x, k')$, where $k'(y) = C_\mathcal{H}(\sigma_2)(y, k(P_\mathcal{H}(\sigma_2)(y)))$, and
  2. $\sigma_2 \in E_\mathcal{H}(P_\mathcal{G}(\sigma'_1, x), k)$ for all $\sigma'_1 \in \Sigma_\mathcal{G}$

The definition of sequential composition highlights two fundamental points

- We promised to explain why we write $\mathcal{G} : (X, S) \to (Y, R)$ if $\mathcal{G}$ has state X, moves Y, utility R and coutility S. When composing functions $f : A \to B$ and $g : B \to C$, the codomain of f must be the same as the domain of g because we feed the output of f into the input of g. Similarly, in $\mathcal{H} \circ \mathcal{G}$, we feed the output move $\mathcal{G}$ into the state of $\mathcal{H}$, and the coutility of $\mathcal{H}$ into the utility of $\mathcal{G}$. Thus i) the type of moves of $\mathcal{G}$ and the states of $\mathcal{H}$ must be the same; and ii) the type of utility of $\mathcal{G}$ and the coutility of $\mathcal{H}$ must also be the same. Thus, the domain of a game contains its state and coutility, while its codomain contains its moves and utility.

- To treat the composition $\mathcal{H} \circ \mathcal{G}$ correctly - as in the example of placing a bet with a bookmaker - $\mathcal{G}$ can only get its utility once H has its own utility. Technically, within the definition of equilibrium for a composed system, $\mathcal{G}$ must be in equilibrium for a utility function which depends upon what $\mathcal{H}$ feeds back to $\mathcal{G}$. This ability of the future system $\mathcal{H}$ to feed utility into a previous system $\mathcal{G}$ is exactly why we introduced the concept of coutility. Formally, the definition of equilibrium in the composition $E_{\mathcal{H}\circ\mathcal{G}}$ depends upon the existence of the coutility $C_{\mathcal{H}}$.

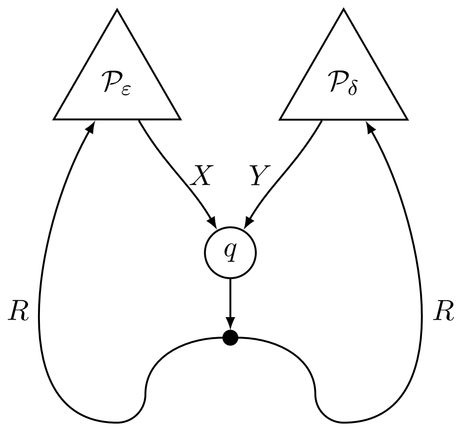
EXAMPLE 13. A two-player context-dependent game is defined in [8] to consist of the following data:

- Sets X, Y of choices for each player, and R of outcomes
- Multivalued selection functions $\varepsilon : (X \to R) \to \mathcal{P}X$, $\delta : (Y \to R) \to \mathcal{P}Y$
- An outcome function $q : X \times Y \to R$
- A strategy profile is simply a pair $(\sigma_1, \sigma_2) : X \times Y$. A strategy profile is called a *selection equilibrium* if

$$\sigma_1 \in \varepsilon(q(-, \sigma_2))$$
$$\sigma_2 \in \delta(q(\sigma_1, -))$$

Context dependent games generalise the simple games of Def 2 in that the selection functions need not be argmax. For example, in [8] it is shown that selection functions returning a set of fixpoints or non-fixpoints gives an elegant model of coordination or differentiation goals of players, respectively. The selection equilibria of context dependent games can be characterised compositionally.

THEOREM 14. *The selection equilibria of this game are precisely the equilibria of the string diagram*



where q is the lifting of the outcome function as a covariant computation, and $\mathcal{P}_\varepsilon$ and $\mathcal{P}_\delta$ are players, whose equilibrium functions are respectively

$$E_{\mathcal{P}_\varepsilon}(*, k) = \varepsilon k$$
$$E_{\mathcal{P}_\delta}(*, k) = \delta k$$

Proof. The above string diagram is clearly definable within our grammar of pregames, for example, by the expression

$$\mathcal{G} = (\tau_R \otimes \tau_R) \circ (\mathrm{id}_R^* \otimes (\Delta_R \circ q) \otimes \mathrm{id}_R^*) \circ (\mathcal{P}_\varepsilon \otimes \mathcal{P}_\delta) : (1,1) \to (1,1)$$

Unwinding the definitions, we have

$$\begin{aligned} & (\sigma_1, \sigma_2) \in E_{\mathcal{G}}(*, *) \\ \iff & (\sigma_1, \sigma_2) \in E_{\mathcal{P}_\varepsilon \otimes \mathcal{P}_\delta}(*, k) \wedge * \in E_{(\tau_R \otimes \tau_R) \circ (\mathrm{id}_R^* \otimes (\Delta_R \circ q) \otimes \mathrm{id}_R^*)}(*, *) \end{aligned}$$

The second term of this is found to be vacuously true, and so it is equivalent to

$$\begin{aligned} \iff & (\sigma_1, \sigma_2) \in E_{\mathcal{P}_\varepsilon \otimes \mathcal{P}_\delta}(*, k) \\ \iff & \sigma_1 \in E_{\mathcal{P}_\varepsilon}(*, k_1) \wedge \sigma_2 \in E_{\mathcal{P}_\delta}(*, k_2) \\ \iff & \sigma_1 \in \varepsilon k_1 \wedge \sigma_2 \in \delta k_2 \end{aligned}$$

It is straightforward but tedious to verify that, according to the above definitions, the continuations k, k_1 and k_2 are given as follows:

$$\begin{aligned} k(x, y) &= C_{(\tau_R \otimes \tau_R) \circ (\mathrm{id}_R^* \otimes (\Delta_R \circ q) \otimes \mathrm{id}_R^* \otimes \mathrm{id}_R^*)}(*)((x, y), *) \\ &= (q(x, y), q(x, y)) \end{aligned}$$

$$k_1(x) = (\pi_1 \circ k)(x, P_{\mathcal{P}_\delta}(\sigma_2, *)) = (\pi_1 \circ k)(x, \sigma_2) = q(x, \sigma_2)$$
$$k_2(y) = (\pi_2 \circ k)(P_{\mathcal{P}_\varepsilon}(\sigma_1, *), y) = (\pi_2 \circ k)(\sigma_1, y) = q(\sigma_1, y)$$

Thus, we have proved that $E_{\mathcal{G}}(*, *)$ is exactly the set of (σ_1, σ_2) that satisfy the conditions of a selection equilibrium. ■

Context dependent games and selection equilibria include ordinary normal form games and Nash equilibria as a special case, by taking $R = \mathbb{R}^2$, and the selection function ε and δ to be the argmax operators for the first and second coordinates:

$$\varepsilon(k) = \{x \in X \mid (\pi_1 \circ k)(x) \geq (\pi_1 \circ k)(x') \text{ for all } x' \in X\}$$
$$\delta(k) = \{y \in Y \mid (\pi_2 \circ k)(y) \geq (\pi_2 \circ k)(y') \text{ for all } y' \in Y\}$$

A second class of games that are subsumed by our grammar are generalised sequential games, which subsume classical games of perfect information.

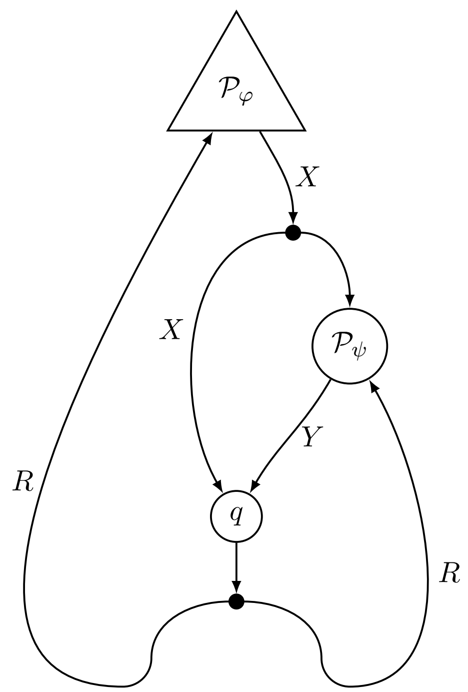
DEFINITION 15. A two-player sequential game is defined in [6] to consist of the following data:

- Sets X, Y of choices for each player, and R of outcomes
- Multivalued quantifiers $\varphi : (X \to R) \to \mathcal{P}R$, $\psi : (Y \to R) \to \mathcal{P}R$
- An outcome function $q : X \times Y \to R$
- A strategy profile for this game consists of a move $\sigma_1 : X$ for the first player and a contingent strategy $\sigma_2 : X \to Y$ for the second player. A strategy profile is called *optimal* if

$$q(\sigma_1, \sigma_2(\sigma_1)) \in \varphi \lambda x.q(x, \sigma_2(x))$$
$$q(x, \sigma_2(x)) \in \psi \lambda y.q(x, y) \quad \text{for all } x \in X$$

Note that the difference between selection functions and quantifiers is relatively unimportant: we could equally well define simultaneous games using quantifiers, and sequential games using selection functions.

THEOREM 16. *Every optimal strategy profile for this game is an equilibrium of the string diagram*



where P_ε and P_δ are players with equilibrium functions

$$\begin{aligned}\sigma_1 \in E_{P_\varphi}(*, k_1) &\iff k_1(\sigma_1) \in \varphi(k_1)\\ \sigma_2 \in E_{P_\psi}(x, k_2) &\iff k_2(\sigma_2(x)) \in \psi(k_2)\end{aligned}$$

Proof. Again, this string diagram falls within our grammar being given algebraically by

$$(\tau_R \otimes \tau_R) \circ (\mathrm{id}_R^* \otimes (((\Delta_R \circ q) \otimes \mathrm{id}_R^*) \circ (\mathrm{id}_X \otimes \mathcal{P}_\psi) \circ \Delta_X)) \circ \mathcal{P}_\varphi : (1,1) \to (1,1)$$

Unwinding the definition, we have

$$\begin{aligned} & (\sigma_1, \sigma_2) \in E_{(\tau_R \otimes \tau_R) \circ (\mathrm{id}_R^* \otimes (((\Delta_R \circ q) \otimes \mathrm{id}_R^*) \circ (\mathrm{id}_X \otimes \mathcal{P}_\psi) \circ \Delta_X)) \circ \mathcal{P}_\varphi}(*, *)\\ \iff & (\sigma_1, \sigma_2) \in E_{(\mathrm{id}_R^* \otimes (((\Delta_R \circ q) \otimes \mathrm{id}_R^*) \circ (\mathrm{id}_X \otimes \mathcal{P}_\psi) \circ \Delta_X)) \circ \mathcal{P}_\varphi}(*, \mathrm{id}_{R \times R})\\ \iff & \sigma_1 \in E_{\mathcal{P}_\varphi}(*, k_1) \quad \wedge \quad \forall \sigma_1' \in X.\ \ \sigma_2 \in E_{\mathcal{G}}(P_{\mathcal{P}_\varphi}(\sigma_1')(*), \mathrm{id}_{R \times R})\end{aligned}$$

where

$$\mathcal{G} = \mathrm{id}_R^* \otimes (((\Delta_R \circ q) \otimes \mathrm{id}_R^*) \circ (\mathrm{id}_X \otimes \mathcal{P}_\psi) \circ \Delta_X) : (X, R) \to (R \times R, R \times R)$$

and

$$k_1(x) = C_{\mathcal{G}}(\sigma_2)(x, \mathrm{id}_{R \times R}(P_{\mathcal{G}}(\sigma_2)(x))) = q(x, \sigma_2(x))$$

The first condition is therefore equivalent to

$$q(\sigma_1, \sigma_2(\sigma_1)) \in \varphi(\lambda x. q(x, \sigma_2(x)))$$

which is the first part of the definition of an optimal strategy.

Continuing, the second condition for a particular $\sigma_1' \in X$ is

$$\begin{aligned}\sigma_2 \in E_{\mathcal{G}}(P_{\mathcal{P}_\varepsilon}(\sigma_1')(*), \mathrm{id}_{R \times R}) &\iff \sigma_2 \in E_{\mathcal{G}}(\sigma_1', \mathrm{id}_{R \times R})\\ &\iff \sigma_2 \in E_{((\Delta_R \circ q) \otimes \mathrm{id}_R^*) \circ (\mathrm{id}_X \otimes \mathcal{P}_\psi) \circ \Delta_X}(\sigma_1', \pi_2)\\ &\iff \sigma_2 \in E_{((\Delta_R \circ q) \otimes \mathrm{id}_R^*) \circ (\mathrm{id}_X \otimes \mathcal{P}_\psi)}((\sigma_1', \sigma_1'), \pi_2)\\ &\iff \sigma_2 \in E_{\mathrm{id}_X \otimes \mathcal{P}_\psi}((\sigma_1', \sigma_1'), k_2)\\ &\iff \sigma_2 \in E_{\mathcal{P}_\psi}(\sigma_1', k_3)\end{aligned}$$

where

$$\begin{aligned} k_2(x, y) &= C_{(\Delta_R \circ q) \otimes \mathrm{id}_R^*}(*)((x, y), (\pi_2 \circ P_{(\Delta_R \circ q) \otimes \mathrm{id}_R^*}(*))(x, y)))\\ &= q(x, y)\\ k_3(y) &= k_2(\sigma_1', y) = q(\sigma_1', y)\end{aligned}$$

Thus, the second condition is that for all $\sigma'_1 \in X$,

$$k_3(\sigma_2(\sigma'_1)) \in \psi(k_3)$$

which is

$$q(\sigma'_1, \sigma_2(\sigma'_1)) \in \psi(\lambda y.q(\sigma'_1, y))$$

which is the second part of the definition of an optimal strategy. ∎

6. The Monoidal Category of Pregames

The mathematical treatment of pregames above delivers a bottom-up collection of operators for defining complex pregames from simple pregames and computing the equilibria of those complex games from their more simple components. Furthermore, pregames naturally posses a graphical form as a string diagram showing the flow of information around the game. This makes pregames easier to visualise and therefore comprehend. However, this mathematical treatment is so far insufficient for a number of reasons:

- Reasoning about string diagrams needs us to formally define diagrams — this can be a very intricate and messy process. Category theory offers a treatment of string diagrams known as *monoidal category theory* which has been successfully applied to a number of areas, eg network theory, quantum physics, concurrency theory.
- We want our operators to capture fundamental structure. Category theory is used to find fundamental structures across mathematics and computer science, eg one such is the monoidal product which arises in both simple games and pregames.
- To reason effectively, we need to know when two different pregames are the same, eg the associativity of composition or the distributivity of contra variant computations over composition: $(g \circ f)^* = f^* \circ g^*$. Category theory gives canonical laws one can expect to hold between the canonical operators that arise in category theory.
- Category theory provides universal properties for structuring and reasoning. A key example are final coalgebras which we use to model infinite iteration of pregames.

For these reasons (and for others more related to future work), we introduce a categorical model of pregames. The first guess is to take as objects pairs of sets and define the morphisms from (X, S) to (Y, R) to be the pregames $(X, S) \to (Y, R)$. However this doesn't quite work, e.g. composition would fail to be associative. The problem is that games which have isomorphic sets of strategies ought really to be the same and hence we quotient pregames by such isomorphisms. This is in tune with general categorical principles where one allows objects to be isomorphic but asks morphisms to either be equal or not. A similar approach underlies the construction of free cartesian closed categories where one quotients well typed λ-calculus terms by $\beta\eta$-equality.

THEOREM 17. *There is a category* Pregame *whose objects are pairs of sets* (X, S)*. Morphisms from* (X, S) *to* (Y, R) *are pregames* $(X, S) \to (Y, R)$ *quotiented by the relation which identifies two pregames if their sets of strategy profiles are isomorphic and their play, coutility and equilibrium functions are the same under this isomorphism.*

PROOF. To ensure readability, our proof treats pregames and we omit the entirely trivial checks that our arguments preserve the equivalence relation on pregames. On the other hand, since morphisms/pregames contain a lot of data, we give the rest of the proof in some detail. The identity on the object (X, S) is the pregame $\mathrm{id}_X \otimes \mathrm{id}_S^*$. More concretely, this is the pregame

- $\Sigma_{\mathrm{id}_{(X,S)}} = 1$
- $P_{\mathrm{id}_{(X,S)}}(*) = \mathrm{id}_X$
- $C_{\mathrm{id}_{(X,S)}}(*)(x, r) = r$
- $* \in E_{\mathrm{id}_{(X,S)}}(x, k)$ for every x and k

Left identity Let $\mathcal{G} : (X, S) \to (Y, R)$. We prove that $\mathrm{id}_{(Y,R)} \circ \mathcal{G} = \mathcal{G}$. Then

- $\Sigma_{\mathrm{id}_{(Y,R)} \circ \mathcal{G}} = \Sigma_{\mathrm{id}_{(Y,R)}} \times \Sigma_\mathcal{G} = 1 \times \Sigma_\mathcal{G} = \Sigma_\mathcal{G}$
- $P_{\mathrm{id}_{(Y,R)} \circ \mathcal{G}}(\sigma) = P_{\mathrm{id}_{(Y,R)}}(*) \circ P_\mathcal{G}(\sigma) = \mathrm{id}_Y \circ P_\mathcal{G}(\sigma) = P_\mathcal{G}(\sigma)$
- $C_{\mathrm{id}_{(Y,R)} \circ \mathcal{G}}(\sigma)(x, r) = C_\mathcal{G}(\sigma)(x, C_{\mathrm{id}_{(Y,R)}}(*)(P_\mathcal{G}(\sigma)(x), r)) = C_\mathcal{G}(\sigma)(x, r)$
- $\sigma \in E_{\mathrm{id}_{(Y,R)} \circ \mathcal{G}}(x, k) \iff \sigma \in E_\mathcal{G}(x, k') \wedge * \in E_{\mathrm{id}_{(Y,R)}}(P_\mathcal{G}(\sigma)(x), k) \iff \sigma \in E_\mathcal{G}(x, k')$ where

$$k'(y) = C_{\mathrm{id}_{(Y,R)}}(*)(y, k(P_{\mathrm{id}_{(Y,R)}}(*)(y))) = k(P_{\mathrm{id}_{(Y,R)}}(*)(y)) = k(y)$$

Right identity We prove that $\mathcal{G} \circ \mathrm{id}_{X,S} = \mathcal{G}$. We have

- $\Sigma_{\mathcal{G} \circ \mathrm{id}_{(X,S)}} = \Sigma_\mathcal{G} \times \Sigma_{\mathrm{id}_{(X,S)}} = \Sigma_\mathcal{G} \times 1 = \Sigma_\mathcal{G}$
- $P_{\mathcal{G} \circ \mathrm{id}_{(X,S)}}(\sigma) = P_\mathcal{G}(\sigma) \circ P_{\mathrm{id}_{(X,S)}}(*) = P_\mathcal{G}(\sigma) \circ \mathrm{id}_X = P_\mathcal{G}(\sigma)$

- $C_{\mathcal{G}\circ\mathrm{id}_{(X,S)}}(\sigma)(x,r) = C_{\mathrm{id}_{(X,S)}}(*)(x, C_{\mathcal{G}}(\sigma)(P_{\mathrm{id}_{(X,S)}}(*)(x), r)) = C_{\mathcal{G}}(\sigma)(P_{\mathrm{id}_{(X,S)}}(*)(x), r) = C_{\mathcal{G}}(\sigma)(x,r)$
- $\sigma \in E_{\mathcal{G}\circ\mathrm{id}_{(X,S)}}(x,k) \iff * \in E_{\mathrm{id}_{(X,S)}}(x,k') \wedge \sigma \in E_{\mathcal{G}}(P_{\mathrm{id}_{(X,S)}}(*)(x), k) \iff \sigma \in E_{\mathcal{G}}(x,k)$

Associativity Let $\mathcal{G} : (X,U) \to (Y,T)$, $\mathcal{H} : (Y,T) \to (Z,S)$ and $\mathcal{I} : (Z,S) \to (W,R)$. We have

$$\Sigma_{(\mathcal{I}\circ\mathcal{H})\circ\mathcal{G}} = \Sigma_{\mathcal{G}} \times \Sigma_{\mathcal{I}\circ\mathcal{H}} = \Sigma_{\mathcal{G}} \times \Sigma_{\mathcal{H}} \times \Sigma_{\mathcal{I}} = \Sigma_{\mathcal{H}\circ\mathcal{G}} \times \Sigma_{\mathcal{I}} = \Sigma_{\mathcal{I}\circ(\mathcal{H}\circ\mathcal{G})}$$

For the play function we have

$$\begin{aligned}
P_{(\mathcal{I}\circ\mathcal{H})\circ\mathcal{G}}(\sigma_1,\sigma_2,\sigma_3) &= P_{\mathcal{I}\circ\mathcal{H}}(\sigma_2,\sigma_3) \circ \mathbb{P}_{\mathcal{G}}(\sigma_1) \\
&= P_{\mathcal{I}}(\sigma_3) \circ P_{\mathcal{H}}(\sigma_2) \circ P_{\mathcal{G}}(\sigma_1) \\
&= P_{\mathcal{I}}(\sigma_3) \circ P_{\mathcal{H}\circ\mathcal{G}}(\sigma_1,\sigma_2) \\
&= P_{\mathcal{I}\circ(\mathcal{H}\circ\mathcal{G})}(\sigma_1,\sigma_2,\sigma_3)
\end{aligned}$$

and for the coutility function have

$$\begin{aligned}
&C_{(\mathcal{I}\circ\mathcal{H})\circ\mathcal{G}}(\sigma_1,\sigma_2,\sigma_3)(x,r) \\
&= C_{\mathcal{G}}(\sigma_1)(x, C_{\mathcal{I}\circ\mathcal{H}}(\sigma_2,\sigma_3)(P_{\mathcal{G}}(\sigma_1)(x), r)) \\
&= C_{\mathcal{G}}(\sigma_1)(x, C_{\mathcal{H}}(\sigma_2)(P_{\mathcal{G}}(\sigma_1)(x), C_{\mathcal{I}}(\sigma_3)(P_{\mathcal{H}}(\sigma_2)(P_{\mathcal{G}}(\sigma_1)(x), r)))) \\
&= C_{\mathcal{G}}(\sigma_1)(x, C_{\mathcal{H}}(\sigma_2)(P_{\mathcal{G}}(\sigma_1)(x), C_{\mathcal{I}}(\sigma_3)(P_{\mathcal{H}\circ\mathcal{G}}(\sigma_1,\sigma_2)(x), r))) \\
&= C_{\mathcal{H}\circ\mathcal{G}}(\sigma_1,\sigma_2)(x, C_{\mathcal{I}}(\sigma_3)(P_{\mathcal{H}\circ\mathcal{G}}(\sigma_1,\sigma_2)(x), r)) \\
&= C_{\mathcal{I}\circ(\mathcal{H}\circ\mathcal{G})}(\sigma_1,\sigma_2,\sigma_3)(x,r)
\end{aligned}$$

For the equilibrium condition we have

$$\begin{aligned}
& (\sigma_1,\sigma_2,\sigma_3) \in E_{(\mathcal{I}\circ\mathcal{H})\circ\mathcal{G}}(x,k_3) \\
\iff & \sigma_1 \in E_{\mathcal{G}}(x,k_1) \wedge \forall \sigma_1' \in \Sigma_{\mathcal{G}}.(\sigma_2,\sigma_3) \in E_{\mathcal{I}\circ\mathcal{H}}(P_{\mathcal{G}}(\sigma_1')(x), k_3) \\
\iff & \sigma_1 \in E_{\mathcal{G}}(x,k_1) \wedge \forall \sigma_1' \in \Sigma_{\mathcal{G}}.[\sigma_2 \in E_{\mathcal{H}}(P_{\mathcal{G}}(\sigma_1')(x), k_2) \\
& \wedge \forall \sigma_2' \in \Sigma_{\mathcal{H}}.\sigma_3 \in E_{\mathcal{I}}(P_{\mathcal{H}}(\sigma_2')(P_{\mathcal{G}}(\sigma_1')(x)), k_3)] \\
\iff & \sigma_1 \in E_{\mathcal{G}}(x,k_1) \wedge [\forall \sigma_1' \in \Sigma_{\mathcal{G}}.\sigma_2 \in E_{\mathcal{H}}(P_{\mathcal{G}}(\sigma_1')(x), k_2)] \\
& \wedge \forall (\sigma_1',\sigma_2') \in \Sigma_{\mathcal{H}\circ\mathcal{G}}.\sigma_3 \in E_{\mathcal{I}}(P_{\mathcal{H}\circ\mathcal{G}}(\sigma_1',\sigma_2')(x), k_3) \\
\iff & (\sigma_1,\sigma_2) \in E_{\mathcal{H}\circ\mathcal{G}}(x,k_2) \\
& \wedge \forall (\sigma_1',\sigma_2') \in \Sigma_{\mathcal{H}\circ\mathcal{G}}.\sigma_3 \in E_{\mathcal{I}}(P_{\mathcal{H}\circ\mathcal{G}}(\sigma_1',\sigma_2')(x), k_3) \\
\iff & (\sigma_1,\sigma_2,\sigma_3) \in E_{\mathcal{I}\circ(\mathcal{H}\circ\mathcal{G})}(x,k_3)
\end{aligned}$$

where

$$
\begin{aligned}
k_1(y) &= C_{\mathcal{I}\circ\mathcal{H}}(\sigma_2,\sigma_3)(y, k_3(P_{\mathcal{I}\circ\mathcal{H}}(\sigma_2,\sigma_3)(x))) \\
&= C_{\mathcal{I}\circ\mathcal{H}}(\sigma_2,\sigma_3)(y, k_3(P_{\mathcal{I}}(\sigma_3)(P_{\mathcal{H}}(\sigma_2)(x)))) \\
&= C_{\mathcal{H}}(\sigma_2)(y, C_{\mathcal{I}}(\sigma_3)(P_{\mathcal{H}}(\sigma_2)(y), k_3(P_{\mathcal{I}}(\sigma_3)(P_{\mathcal{H}}(\sigma_2)(x))))) \\
&= C_{\mathcal{H}}(\sigma_2)(y, k_2(P_{\mathcal{H}}(\sigma_2)(x))) \\
k_2(z) &= C_{\mathcal{I}}(\sigma_3)(z, k_3(P_{\mathcal{I}}(\sigma_3)(z)))
\end{aligned}
$$

This completes the proof. ∎

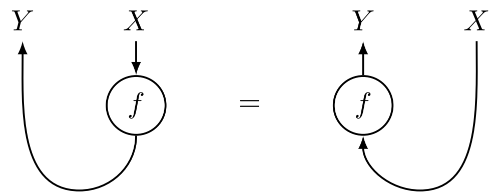
Theorem 17 demonstrates how our composition operator is a fundamental algebraic operation. The same is true of our operation for putting games in parallel.

THEOREM 18. *The category* **Pregame** *is symmetric monoidal.*

PROOF. The monoidal product acts on objects (X, R) and (X', R') by taking their product componentwise giving $(X \times X', R \times R')$. The action of the monoidal product on morphisms is given by parallel composition. The unit of this monoidal structure is $(1, 1)$, while the symmetry is inherited from that of the product on the category of sets. As the proof that this data does indeed form a symmetric monoidal category is straightforward, containing none of the detail of Theorem 17, we leave the proof as an exercise. ∎

One final piece of categorical structure bears a striking resemblance to graphical reasoning in a compact closed category:

THEOREM 19. *If $f : X \to Y$ is a computation then $\tau_Y \circ (f \otimes \mathrm{id}_Y^*) = \tau_X \circ (\mathrm{id}_X \otimes f^*)$. In diagrams,*



PROOF. Firstly note that we have

$$
\begin{aligned}
P_{f\otimes \mathrm{id}_Y^*}(*)(x) &= f(x) & \qquad C_{f\otimes \mathrm{id}_Y^*}(*)(x,y) &= y \\
P_{\mathrm{id}_X \otimes f^*}(*)(x) &= x & \qquad C_{\mathrm{id}_X \otimes f^*}(*)(x,x') &= f(x')
\end{aligned}
$$

We have $\Sigma_{\tau_Y \circ (f \otimes \mathrm{id}_Y^*)} \cong 1 \cong \Sigma_{\tau_X \circ (\mathrm{id}_X \otimes f^*)}$. Both play functions are the unique function $X \to 1$. For the coutility function,

$$C_{\tau_Y \circ (f \otimes \mathrm{id}_Y^*)}(*)(x,*) = f(x) = C_{\tau_X \circ (\mathrm{id}_X \otimes f^*)}(*)(x,*)$$

Finally for both cases we have $* \in E(x,*)$ for all x. ∎

7. Iteration

While we have so far demonstrated the possibility of a compositional theory of Nash equilibria, more operators are needed to cover a compelling variety of games. As there will likely never be a set of operators capturing all possible games, we seek a balance between i) the desire for more operators to model more games; and ii) fewer (but perhaps better) operators to ensure the collection of operators remains small and hence tractable. Obtaining this balance is a substantial endeavour requiring a follow-up paper. Nevertheless we make a start here, primarily to showcase how compositional Nash equilibria can be defined for games more sophisticated than those studied already. To that end, we consider IPD - the Iterated Prisoners' Dilemma.

Informally, IPD consists of playing PD infinitely often. Thus, one might conjecture IPD is defined recursively by the equation $\mathsf{IPD} = \mathsf{IPD} \circ \mathsf{PD}$. However, this is not quite right as the definition of composition means that a strategy for IPD would consist of a strategy for PD and another strategy for IPD. However, a strategy for IPD actually consists of a strategy for PD and - for each move of PD - a strategy for IPD to be followed if that move were played. Thus the correct definition is

Definition 20 (Infinite Iteration). Let $\mathcal{G} : (X,R) \to (Y,R)$ be a pregame with strategies, play, coutility and equilibria given by $\Sigma_\mathcal{G}, P_\mathcal{G}, C_\mathcal{G}$ and $E_\mathcal{G}$ respectively. Further, let $\phi : Y \to X$ be a function which updates state after a move is played. Define the iterated pregame $\mathcal{G}^\omega_\phi : (X,R) \to (Y^\omega, R)$, where Y^ω is the type of infinite lists over Y, as follows

$$\begin{array}{rcl}
\Sigma_{\mathcal{G}^\omega} & = & \nu Z.\, \Sigma_\mathcal{G} \times (Y \to Z) \\
P_{\mathcal{G}^\omega_\phi}\ (\sigma, f)\ x & = & \mathsf{let}\ y = P_\mathcal{G}\ \sigma\ x\ \mathsf{in}\ y : P_{\mathcal{G}^\omega_\phi}\ (f\ y)\ (\phi\ y) \\
C_{\mathcal{G}^\omega_\phi}\ (\sigma, f)\ x\ r & = & \mathsf{let}\ y = P_\mathcal{G}\ \sigma\ x\ \mathsf{in}\ C_\mathcal{G}\ \sigma\ x\ (C_{\mathcal{G}^\omega_\phi}(k\ y)\ (\phi\ y)\ r) \\
(\sigma, f) \in E_{\mathcal{G}^\omega_\phi}\ x\ k & \mathsf{iff} & \sigma \in E_\mathcal{G}\ x\ k' \wedge (\forall y \in Y)\ fy \in E_{\mathcal{G}^\omega_\phi}\ (\phi\ y)\ k \\
 & & \text{where } k'\ y = C_{\mathcal{G}^\omega_\phi}\ (fy)\ (\phi y)\ (k(y : P_{\mathcal{G}^\omega_\phi}\ (fy)\ (\phi y))
\end{array}$$

The above definition is somewhat complex and hence needs explanation. First note that $\nu Z.\ \Sigma_{\mathcal{G}} \times (Y \to Z)$ is a final coalgebra that denotes the set of infinite depth trees with nodes labelled by an element of $\Sigma_{\mathcal{G}}$ and where each node has Y-children. As a result, an element of $\Sigma_{\mathcal{G}^\omega_\phi}$ can be written uniquely as a pair (σ, f) where $\sigma \in \Sigma_{\mathcal{G}}$ and $f : Y \to \Sigma_{\mathcal{G}^\omega_\phi}$ maps each element of Y to its associated child-tree. Next, note that the symbol : is the 'cons' operator which takes a stream of data an a piece of data and adds the data to the front of the stream. The play function of $\mathcal{G}^\omega_\phi$ is defined by computing the first move and then adding that to the recursive computation of subsequent moves. Note how the coutility function is similar to the coutility function for composition reflecting how coutility from subsequent moves played later in the game is fed into the coutility function for the first move. Finally, a strategy is in equilibrium iff the strategy to be played in the first game is in equilibrium for the one-step game, and - for each potential move - the associated strategies for the remaining subgame are also in equilibrium.

As an example, we apply the above definition to the Prisoners Dilemma, thereby producing the Iterated Prisoners Dilemma. Recalling the Prisoners Dilemma PD is a pregame $\mathsf{PD} : (1, S) \to (M, S)$ where $M = \{C, D\}^2$ and $S = \mathbb{R}^2$. The strategies, play, coplay and utility functions are all given in Example 6. Applying our iteration combinator with the (unique) state function $\phi : M \to 1$, we get the Iterated Prisoners Dilemma $\mathsf{IPD} = \mathsf{PD}^\omega_\phi$ as follows:

Example 21. The Iterated Prisoners Dilemma is the game $\mathsf{IPD} : (1, \mathbb{R} \times \mathbb{R}) \to (M^\omega, \mathbb{R}^2)$ with strategies, play, coutility and equilibria given - once again after removing any unit types - by

$$\begin{array}{rcl} \Sigma_{\mathsf{IPD}} & = & \nu Z.\ M \times (M \to Z) \\ P_{\mathsf{IPD}}\,(\sigma, f) & = & \sigma : P_{\mathsf{IPD}}\,(f\sigma) \\ (\sigma, f) \in E_{\mathsf{IPD}}\,k & \text{iff} & \sigma \in E_{\mathsf{PD}}\,(k(P_{\mathsf{IPD}}\,(-, f))) \\ & & \wedge \\ & & (\forall \sigma \in M) f\sigma \in E_{\mathsf{IPD}}\,k \end{array}$$

with the trivial coutility function.

Note that the strategies Σ_{IPD} consist of infinite depth trees whose nodes are labelled by elements of M and each node has M-children. The element at each node describes the strategy to be used in the first round and each child describes the strategy to be used if the corresponding move were to have been played. These are precisely the pure strategies for the ordinary iterated prisoner's dilemma. If we choose $k : M^\omega \to \mathbb{R}^2$ to be a utility function for the iterated prisoner's dilemma (perhaps given by a discounted sum, but perhaps not) then $E_{\mathsf{IPD}}\ k$ exactly characterises those strategies in Σ_{IPD} that are subgame-perfect equilibria of the iterated prisoner's dilemma. Intuitively, the condition says that i) the first move is a Nash equilibrium of the one-shot game PD but, crucially, for a utility function that incorporates information about future moves via P_{IPD}; and ii) for every potential first move, the associated subsequent strategy for the remaining subgame (which is still an infinitely repeated game) is also in equilibrium. Formally, for each k, the equilibirum predicate $E_{\mathsf{IPD}}k$ is a *coalgebraic predicate* [**?**] which therefore supports the following reasoning principle called *coinduction*

Theorem 22. *Let* $Q \subseteq \Sigma_{\mathsf{IPD}}$ *and* $k : M^\omega \to \mathbb{R}^2$*. Further, if*

$$\forall(\sigma, f) \in Q.\ \sigma \in E_{\mathsf{PD}}\ (k(P_{\mathsf{IPD}}\ (-, f))) \quad\wedge\quad (\forall \sigma \in M) f\sigma \in Q$$

then $Q \subseteq E_{\mathsf{IPD}}\ k$*.*

We can use the above theorem to show certain strategies are in E_{IPD} for various utility functions. The point is not that the result below is anything but obvious, but rather to demonstrate the nature of the coinductive proof technique to econometricians who may not be familiar with it.

Example 23. Define $\sigma : \Sigma_{\mathsf{IPD}}$ by $\sigma_0 = ((D, D), \lambda m.\sigma)$. That is σ is the strategy in which both players defect upon their first turn and no matter what moves are played, decide to defect forever after. Further, define $k : M \to \mathbb{R} \times \mathbb{R}$ to be the utility function for PD defined by

$$k(C, C) = (-1, -1) \qquad k(D, C) = (0, -4)$$
$$k(C, D) = (-4, 0) \qquad k(D, D) = (-3, -3)$$

Such a k is a classic utility function for which the only Nash equilibria are for both players to defect. Now, lets define a utility function $k^\omega : M^\omega \to \mathbb{R} \times \mathbb{R}$ for IPD by discounting: $k^\omega(m : ms) = km \oplus k^\omega ms$ where $\oplus$ is componentwise addition. To show $\sigma \in E_{\mathsf{IPD}}$ we use the above theorem with $Q = \{\sigma\}$. Thus, we need to show

$$(D, D) \in E_{\mathsf{PD}}\ ((k-) \oplus k^\omega d^\omega) \quad\wedge\quad \sigma \in Q$$

where $d^\omega = (D, D) : d^\omega$ is the sequence of plays where both players keep on defecting. The first subgoal is clear as one can easily check that the only equilibria of the Prisoners Dilemma with utility function $(k-) \oplus k^\omega d^\omega$ is for both players to defect. The second subgoal is true by assumption.

We conclude with a question of utility. We were asked by friends "*What questions could your work help those interested in economic game theory solve*". Such questions are always difficult for a paper aiming at concepts and ideas, but nevertheless, we ask readers the following questions

1. Can you develop a model of equilibria in iterated games that does not insist that the utility of an iterated game is obtained by discounting? The above work infact is exactly such a model as nowhere do we assume discounting. This is in the tradition of best scientific practice where one ought not to make assumptions (here discounting) unless one need to.
2. Lets say we want to enter some software in a tournament for games playing Iterated Prisoners Dilemma. We would thus have to play the iterated Prisoners' Dilemma over an over again. Can you formalise that game, the associated strategies, moves and equilibria? With out work, it is simply the iterated version of the iterated prisoners dilemma, i.e. $\mathsf{IIPD} = \mathsf{IPD}^\omega = (\mathsf{PD}^\omega)^\omega$. It is thus simply a matter of reading off the right definitions from our formulas above.
3. Can you find conditions for a general class of iterated games which ensure repeating a strategy which is in equilibria for each state game is in equilibrium for the iterated variant of the stage game? Our work suggests exactly such a generic answer.

8. Conclusions and Future Research

We finish the paper with a discussion about what we have achieved and what we intend to do in the future.

Achievements: This paper's most significant contribution is a new way to think about economic games using the idea of *compositionality* from computer science. This involved the development of a new model of economic games which had several original features: i) games do not come with a specific utility function; ii) games can reason about how current actions effect future utility via the new concept of *coutility*; iii) games posses a clear visual interface as string diagrams; and iv) have a fully rigorous treatment within monoidal category theory. The compositionality of this model is demonstrated by the operators that can be defined on arbitrary games and used to build, reason about and even implement larger games from smaller games. Before discussing future work, we analyse decisions implicitly taken is the design of our model and their validity.

- To facilitate accessibility of our paper we chose to work internally to the category of sets as opposed to internally to an ambient symmetric monoidal category. The choice between clarity and mathematical sophistication is important - and we believe we have it right - but is not inherently a scientific one but rather a subjective matter of presentation.
- We made pregames the morphisms of a category to highlight the inputs and outputs of games which was a prerequisite to formalising the idea of *coutility*. This also enabled a string diagram interpretation of pregames thereby giving rigour to our pregame constructors and relating them to constructions on string diagrams. However, ought we to have made pregames the objects of a category? Certainly making pregames objects would have enabled universal properties to have been developed for pregame operators. Actually, both perspectives are valid. Just as the category of sets, relations and relation preserving morphisms is naturally a 2-category, so the category of pregames is naturally a 2-category. For this paper, the 2-dimensional structure was not relevant so we focussed on the 1-dimensional structure of pregames. As commented below, in future work we will need to turn to the 2-dimensional structure of pregames.

Future Work: The potential applications of a compositional, graphical game theory are numerous, especially in economics, and this paper also raises some interesting theoretical questions. We conclude by broadly giving some future research directions and questions:

- Possibly the most important theoretical concepts missing from this paper are incomplete information games, which are ubiquitous in economic applications. This requires the 2-dimensional structure of pre-games alluded to above. Examples of such games show that delicate choices need to be made about whether data should be treated covariantly, contravariantly or relationally. Nevertheless, we have initial thoughts and hope to progress them soon.
- Another important aspect of game theory that cannot be modelled by the operators above is the ability of a pregame to depend on a previous move. Influence diagrams also suffer from this problem. For example in a market entry game a firm decides whether to enter a new market, and then subsequently a particular game is played only if the market was entered. This is also commonly used with the 'moves of nature' in the standard approach to incomplete information. Our approach will be to use dependent types and fibred monoidal categories to underpin dependently typed string diagrams.

- Computer support is vital. The first author has developed a Haskell implementation, but it is extremely awkward because the Haskell type system does not unify types like $X \sim 1 \times X$ and $X \sim 1 \to X$, and so the user must manually track these isomorphisms. As an intermediate step, a code generator for a domain specific language similar to Haskell's arrows [11] would be useful. Unfortunately, for technical reasons it does not seem to be possible to use GHC's built-in arrow preprocessor. Ultimately a graphical interface would be invaluable for these ideas to become accessible to working economists.

- As a by-product of obtaining a compositional theory, we have the ability to model preferences of agents which are extremely different to utility maximisation or preference relations. This extends a line of work begun in [8], which uses fixpoint selection functions to model coordination and differentiation. Obvious next steps include modelling bounded rationality [14] and social concerns.

- As described above, a potentially very powerful dimension is to vary the underlying category from Sets to another category. The use of ordinary (possibilistic) nondeterminism in game theory is explored in [12, 3, 7] and [9, chapter 9], and work in progress by the author suggests that the order structure on possibilistic strategies is important. We also have experimental evidence that correlated equilibria [1] appear as a special case by using a commutative monad transformer stack in which a reader monad gives players read-only access to a shared randomising device. This is strong evidence that side effects in the sense of programming languages can also be a unifying idea in game theory.

- Using noncommutative side-effects is potentially even more rewarding. In this case, the category of pregames may be premonoidal. A major aim is to use strategies with mutable states to model learning, and individual rationality relations to specify that a strategy can be subjectively rational with respect to the current epistemic state, for example using methods of epistemic game theory [13].

References

[1] Aumann, Robert, 'Subjectivity and correlation in randomized strategies', *Journal of mathematical economics*, 1 (1974), 67–96.

[2] Baez, John, and Mike Stay, 'Physics, topology, logic and computation: a Rosetta stone', in Bob Coecke, (ed.), *New structures for physics*, Springer, 2010, pp. 95–172.

[3] Blumensath, Achim, and Viktor Winschel, 'A coalgebraic framework for games in economics', , 2013. Working paper.

[4] Bonchi, Filippo, Pawel Sobocinski, and Fabio Zanasi, 'Interacting bialgebras are Frobenius', in *Proceedings of FoSSaCS 2014*, vol. 8412, Springer, 2014, pp. 351–365.

[5] Coecke, Bob, Edward Grefenstette, and Mehrnoosh Sadrzadeh, 'Lambek vs. Lambek: functorial vector space semantics and string diagrams for Lambek calculus', *Annals of pure and applied logic*, (2013).

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

[7] Hedges, Jules, 'Monad transformers for backtracking search', in *Proceedings of MSFP'14*, EPTCS, 2014, pp. 31–50.

[8] Hedges, Jules, Paulo Oliva, Evguenia Sprits, Viktor Winschel, and Philipp Zahn, 'Higher-order game theory', , 2015. Submitted.

[9] LaValle, Steven M., *Planning algorithms*, Cambridge University Press, 2006.

[10] Oliva, Paulo, and Thomas Powell, 'A constructive interpretation of Ramsey's theorem via the product of selecion functions', *Mathematical structures in computer science*, (2014).

[11] Paterson, Ross, 'A new notation for arrows', in *Proceedings of the sixth ACM SIGPLAN international conference on functional programming*, ACM, 2001, pp. 229–240.

[12] Pavlovic, Dusko, 'A semantical approach to equilibria and rationality', in *Algebra and coalgebra in computer science*, vol. 5728 of *Lectures notes in computer science*, Springer, 2009, pp. 317–334.

[13] Perea, Andrés, *Epistemic game theory*, Cambridge University Press, 2012.

[14] Rubinstein, Ariel, *Modelling bounded rationality*, MIT Press, 1998.

[15] Selinger, Peter, 'A survey of graphical languages for monoidal categories', in Bob Coecke, (ed.), *New structures for physics*, Springer, 2011, pp. 289–355.