A Coalgebraic Approach to Infinite Games

Ben Plummer, Corina Cîrstea

University of Southampton

April, 2026

Why infinite games?

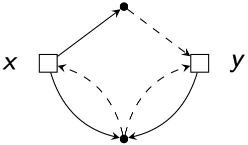
- Verification: does an implementation exhibit some desired behavour φ?
  - Implementation typically modelled as a *transition system*
  - Two (dual) flavours:
    - ($\forall$) Does *every* execution satisfy φ?
    - ($\exists$) Does *there exist* an execution satisfying φ?

Why infinite games?

- Verification: does an implementation exhibit some desired behaviour φ?
  - Implementation typically modelled as a *transition system*
  - Two (dual) flavours:
    - ($\forall$) Does *every* execution satisfy φ?
    - ($\exists$) Does *there exist* an execution satisfying φ?
- *Reactive synthesis*: can we *construct* an implementation which exhibits some desired behaviour φ?

Why infinite games?

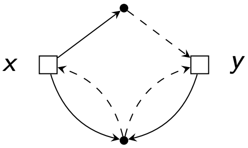
- Verification: does an implementation exhibit some desired behaviour φ?
  - Implementation typically modelled as a *transition system*
  - Two (dual) flavours:
    - ($\forall$) Does *every* execution satisfy φ?
    - ($\exists$) Does *there exist* an execution satisfying φ?
- *Reactive synthesis*: can we *construct* an implementation which exhibits some desired behaviour φ?
  - Two-player graph games model the interaction between a *controller* ($\square$) and an *environment* ($\bullet$)



Objective: "visit y infinitely often from x"

Why infinite games?

- Verification: does an implementation exhibit some desired behaviour φ?
  - Implementation typically modelled as a *transition system*
  - Two (dual) flavours:
    - ($\forall$) Does *every* execution satisfy φ?
    - ($\exists$) Does *there exist* an execution satisfying φ?
- *Reactive synthesis*: can we *construct* an implementation which exhibits some desired behaviour φ?
  - Two-player graph games model the interaction between a *controller* ($\Box$) and an *environment* ($\bullet$)

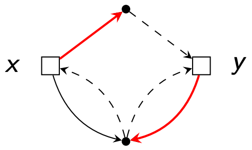


Objective: "visit y infinitely often from x"

  - ($\exists\forall$) Does *there exist* a controller strategy such that *every* possible execution satisfies φ?

Why infinite games?

- Verification: does an implementation exhibit some desired behaviour φ?
  - Implementation typically modelled as a *transition system*
  - Two (dual) flavours:
    - ($\forall$) Does *every* execution satisfy φ?
    - ($\exists$) Does *there exist* an execution satisfying φ?
- *Reactive synthesis*: can we *construct* an implementation which exhibits some desired behaviour φ?
  - Two-player graph games model the interaction between a *controller* ($\square$) and an *environment* ($\bullet$)



Objective: "visit y infinitely often from x"

  - ($\exists\forall$) Does *there exist* a controller strategy such that *every* possible execution satisfies φ?

Why infinite games? cont.

- Modern lines of research focus on *quantitative variants*:
  - Stochastic environments (i.e. Markov decision processes)
  - Non-deterministic and stochastic environments (i.e. *simple stochastic games*)
  - Energy costs and gains (i.e. energy games)

Why infinite games? cont.

- Modern lines of research focus on *quantitative variants*:
  - Stochastic environments (i.e. Markov decision processes)
  - Non-deterministic and stochastic environments (i.e. *simple stochastic games*)
  - Energy costs and gains (i.e. energy games)
- Given a type of arena (e.g. MDPs) and objective e.g. (reachability, ω-regular):
  - Is it possible to always construct an optimal winning strategy?
  - How much *memory* do optimal winning strategies require?
  - Can we solve this type of game *compositionally*?
  - In what manner can we combine objectives?

Why infinite games? cont.

- Modern lines of research focus on *quantitative variants*:
  - Stochastic environments (i.e. Markov decision processes)
  - Non-deterministic and stochastic environments (i.e. *simple stochastic games*)
  - Energy costs and gains (i.e. energy games)
- Given a type of arena (e.g. MDPs) and objective e.g. (reachability, ω-regular):
  - Is it possible to always construct an optimal winning strategy?
  - How much *memory* do optimal winning strategies require?
  - Can we solve this type of game *compositionally*?
  - In what manner can we combine objectives?
- We are looking for a mathematical framework for games
  - *Parametric* in the type of arena/objective
  - Benefits: gain insight into limitations, generic solutions to general classes of games, transfer of results and algorithms

Our coalgebraic approach

- First point of call: *coalgebra*
- Coalgebraic techniques have already given us generic methods in automata theory:
  - E.g. abstract determinisation procedures, up-to techniques, behavioural metrics, model checking techniques,...

Our coalgebraic approach

- First point of call: *coalgebra*
- Coalgebraic techniques have already given us generic methods in automata theory:
  - E.g. abstract determinisation procedures, up-to techniques, behavioural metrics, model checking techniques,...
- Initial steps towards treating *finite* two-player games coalgebraically [Plummer and Cirstea, 2025]:
  - Strategies are the fundamental objects
  - Strategy semantics $\sim$ linear-time semantics in games
  - For linear-time semantics, we require *monads*

Our coalgebraic approach

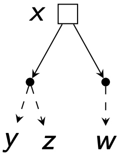
- First point of call: *coalgebra*
- Coalgebraic techniques have already given us generic methods in automata theory:
  - E.g. abstract determinisation procedures, up-to techniques, behavioural metrics, model checking techniques,...
- Initial steps towards treating *finite* two-player games coalgebraically [Plummer and Cirstea, 2025]:
  - Strategies are the fundamental objects
  - Strategy semantics $\sim$ linear-time semantics in games
  - For linear-time semantics, we require *monads*
- We develop, in *infinite games*, the coalgebraic theory of:
  - Strategy outcomes: the set of plays which conform to a strategy
  - Game outcomes: the set of strategies in the game
    - Game outcome at a state x: set of strategies starting from x

The game arena as a coalgebra

- We are concerned with the game arena
  - I.E. a coalgebra $\delta : X \to PP(X)$
  - Each $U \in \delta(x)$ is controller move
  - Each $u \in U$ is an environment move

The game arena as a coalgebra

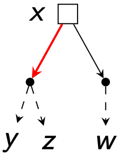
- We are concerned with the game arena
  - I.E. a coalgebra $\delta : X \to PP(X)$
  - Each $U \in \delta(x)$ is controller move
  - Each $u \in U$ is an environment move



$$\delta(x) = \{\{y, z\}, \{w\}\}$$

The game arena as a coalgebra

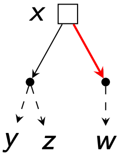
- We are concerned with the game arena
  - I.E. a coalgebra $\delta : X \to PP(X)$
  - Each $U \in \delta(x)$ is controller move
  - Each $u \in U$ is an environment move



$$\delta(x) = \{\{y, z\}, \{w\}\}$$

The game arena as a coalgebra

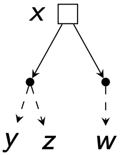
- We are concerned with the game arena
  - I.E. a coalgebra $\delta : X \to PP(X)$
  - Each $U \in \delta(x)$ is controller move
  - Each $u \in U$ is an environment move



$$\delta(x) = \{\{y, z\}, \{w\}\}$$

The game arena as a coalgebra

- We are concerned with the game arena
  - I.E. a coalgebra $\delta : X \to PP(X)$
  - Each $U \in \delta(x)$ is controller move
  - Each $u \in U$ is an environment move

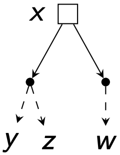


$$\delta(x) = \{\{y, z\}, \{w\}\}$$

- Almost a monad structure on PP:

The game arena as a coalgebra

- We are concerned with the game arena
  - I.E. a coalgebra $\delta : X \to PP(X)$
  - Each $U \in \delta(x)$ is controller move
  - Each $u \in U$ is an environment move

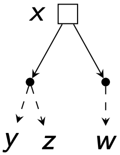


$$\delta(x) = \{\{y, z\}, \{w\}\}$$

- Almost a monad structure on PP:
  - Let $\mathcal{G}(X) \subseteq PP(X)$ be set of subsets *closed under arbitrary non-empty union*

The game arena as a coalgebra

- We are concerned with the game arena
  - I.E. a coalgebra $\delta : X \to PP(X)$
  - Each $U \in \delta(x)$ is controller move
  - Each $u \in U$ is an environment move

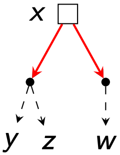


$$\delta(x) = \{\{y, z\}, \{w\}\}$$

- Almost a monad structure on PP:
  - Let $\mathcal{G}(X) \subseteq PP(X)$ be set of subsets *closed under arbitrary non-empty union*
    - Add in "convex choices" for the controller

The game arena as a coalgebra

- We are concerned with the game arena
  - I.E. a coalgebra $\delta : X \to PP(X)$
  - Each $U \in \delta(x)$ is controller move
  - Each $u \in U$ is an environment move

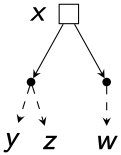


$$\delta(x) = \{\{y, z\}, \{w\}\}$$

- Almost a monad structure on PP:
  - Let $\mathcal{G}(X) \subseteq PP(X)$ be set of subsets *closed under arbitrary non-empty union*
    - Add in "convex choices" for the controller

The game arena as a coalgebra

- We are concerned with the game arena
  - I.E. a coalgebra $\delta : X \to PP(X)$
  - Each $U \in \delta(x)$ is controller move
  - Each $u \in U$ is an environment move



$$\delta(x) = \{\{y, z\}, \{w\}\}$$

- Almost a monad structure on PP:
  - Let $\mathcal{G}(X) \subseteq PP(X)$ be set of subsets *closed under arbitrary non-empty union*
    - Add in "convex choices" for the controller
  - There is a monad structure on $\mathcal{G}$

Strategies as a chain of relations

- A relation $f : X \nrightarrow Y$ is
  - *right-total* when every point y has some x with $y \in f(x)$
  - *separating* when $f(x) \cap f(x') = \emptyset$ for disinct x, x'
  - separating and right-total iff $f^\dagger$ is a function $Y \to X$

Strategies as a chain of relations

- A relation $f : X \nrightarrow Y$ is
  - *right-total* when every point y has some x with $y \in f(x)$
  - *separating* when $f(x) \cap f(x') = \emptyset$ for disinct x, x'
  - separating and right-total iff $f^\dagger$ is a function $Y \to X$
- A strategy is a chain of right-total relations:

$$\begin{array}{ccccccccc} 1 & \xrightarrow{\sigma_*} & S_0 & \xrightarrow{\sigma_0} & S_1 & \xrightarrow{\sigma_1} & S_2 & \xrightarrow{\sigma_2} & \cdots \\ & & \downarrow & & \downarrow & & \downarrow & & \\ & & X & & X^2 & & X^3 & & \end{array}$$

  - such that:
    - $\sigma_n(x_0 \ldots x_n)$ comes from a choice in the game
    - each element of $\sigma_n(x_0 \ldots x_n)$ extends $x_0 \ldots x_n$

Strategies as a chain of relations

- A relation $f : X \nrightarrow Y$ is
  - *right-total* when every point y has some x with $y \in f(x)$
  - *separating* when $f(x) \cap f(x') = \emptyset$ for disinct x, x'
  - separating and right-total iff $f^\dagger$ is a function $Y \to X$
- A strategy is a chain of right-total relations:

$$\begin{array}{ccccccccc} 1 & \xrightarrow{\sigma_*} & S_0 & \xrightarrow{\sigma_0} & S_1 & \xrightarrow{\sigma_1} & S_2 & \xrightarrow{\sigma_2} & \cdots \\ & & \downarrow & & \downarrow & & \downarrow & & \\ & & X & & X^2 & & X^3 & & \end{array}$$

  - such that:
    - $\sigma_n(x_0 \ldots x_n)$ comes from a choice in the game
    - each element of $\sigma_n(x_0 \ldots x_n)$ extends $x_0 \ldots x_n$
  - Consequence: each σ_n is separating

Strategies as a chain of relations

- A relation $f : X \nrightarrow Y$ is
  - *right-total* when every point y has some x with $y \in f(x)$
  - *separating* when $f(x) \cap f(x') = \emptyset$ for disinct x, x'
  - separating and right-total iff $f^\dagger$ is a function $Y \to X$
- A strategy is a chain of right-total relations:

$$\begin{array}{ccccccccc} 1 & \overset{\sigma_*}{\longrightarrow\!\!\!\!\!\!\!+\;\;} & S_0 & \overset{\sigma_0}{\longrightarrow\!\!\!\!\!\!\!+\;\;} & S_1 & \overset{\sigma_1}{\longrightarrow\!\!\!\!\!\!\!+\;\;} & S_2 & \overset{\sigma_2}{\longrightarrow\!\!\!\!\!\!\!+\;\;} & \cdots \\ & & \downarrow & & \downarrow & & \downarrow & & \\ & & X & & X^2 & & X^3 & & \end{array}$$

  - such that:
    - $\sigma_n(x_0 \ldots x_n)$ comes from a choice in the game
    - each element of $\sigma_n(x_0 \ldots x_n)$ extends $x_0 \ldots x_n$
  - Consequence: each σ_n is separating
- Thus $(-)^\dagger$ maps the chain into a cochain in **Set**:

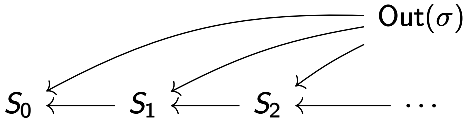
$$S_0 \longleftarrow S_1 \longleftarrow S_2 \longleftarrow \cdots$$

Strategies as a chain of relations

- A relation $f : X \nrightarrow Y$ is
  - *right-total* when every point y has some x with $y \in f(x)$
  - *separating* when $f(x) \cap f(x') = \emptyset$ for disinct x, x'
  - separating and right-total iff $f^\dagger$ is a function $Y \to X$
- A strategy is a chain of right-total relations:

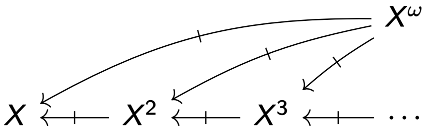
$$\begin{array}{ccccccccc} 1 & \overset{\sigma_*}{\nrightarrow} & S_0 & \overset{\sigma_0}{\nrightarrow} & S_1 & \overset{\sigma_1}{\nrightarrow} & S_2 & \overset{\sigma_2}{\nrightarrow} & \cdots \\ & & \downarrow & & \downarrow & & \downarrow & & \\ & & X & & X^2 & & X^3 & & \end{array}$$

  - such that:
    - $\sigma_n(x_0 \dots x_n)$ comes from a choice in the game
    - each element of $\sigma_n(x_0 \dots x_n)$ extends $x_0 \dots x_n$
  - Consequence: each σ_n is separating
- Thus $(-)^\dagger$ maps the chain into a cochain in **Set**:



Strategy outcomes: alternative characterisations

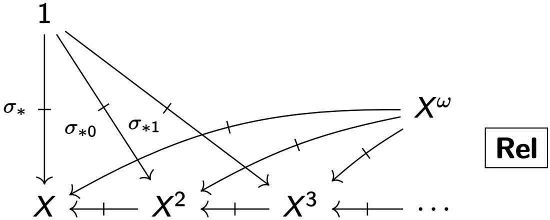
Consider the functor $X \times (-) : \mathbf{Set} \to \mathbf{Set}$
Largest mediating map:



Rel

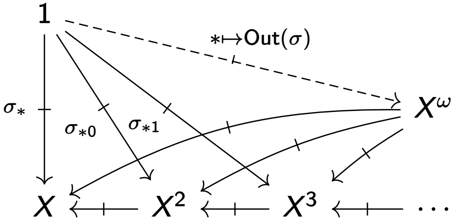
Strategy outcomes: alternative characterisations

Consider the functor $X \times (-) : \mathbf{Set} \to \mathbf{Set}$
Largest mediating map:



Strategy outcomes: alternative characterisations

Consider the functor $X \times (-) : \mathbf{Set} \to \mathbf{Set}$
Largest mediating map:

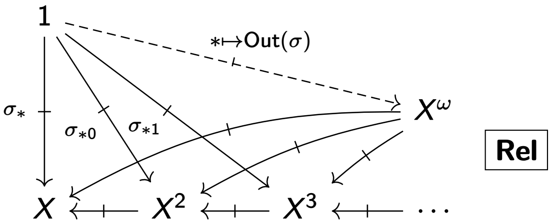


Rel

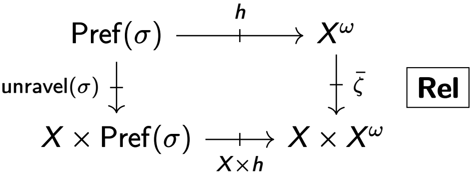
Strategy outcomes: alternative characterisations

Consider the functor $X \times (-) : \mathbf{Set} \to \mathbf{Set}$

Largest mediating map:

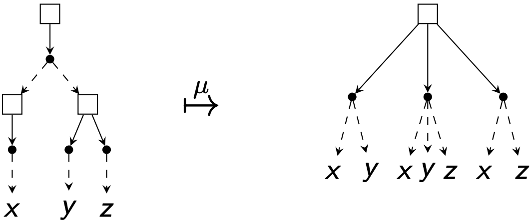


Largest homomorphism:



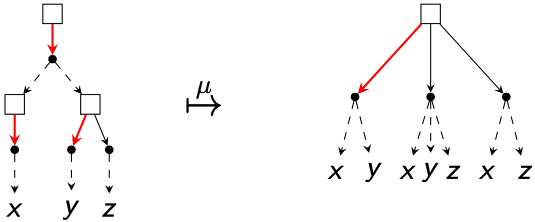
Games are in $\mathbf{Kl}(\mathcal{G})$

- Composition in $\mathbf{Kl}(\mathcal{G})$ *computes strategies*:



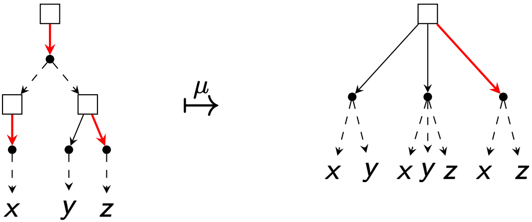
Games are in $\mathbf{Kl}(\mathcal{G})$

- Composition in $\mathbf{Kl}(\mathcal{G})$ *computes strategies*:



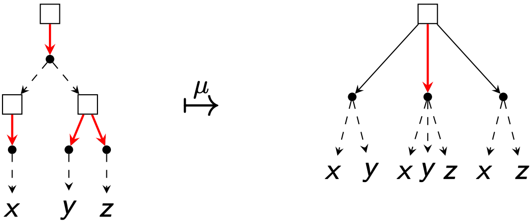
Games are in $\mathbf{Kl}(\mathcal{G})$

- Composition in $\mathbf{Kl}(\mathcal{G})$ *computes strategies*:



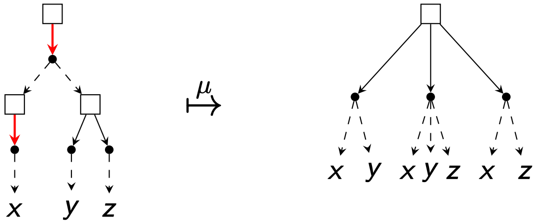
Games are in **Kl**($\mathcal{G}$)

- Composition in **Kl**($\mathcal{G}$) *computes strategies*:



Games are in $\mathbf{Kl}(\mathcal{G})$

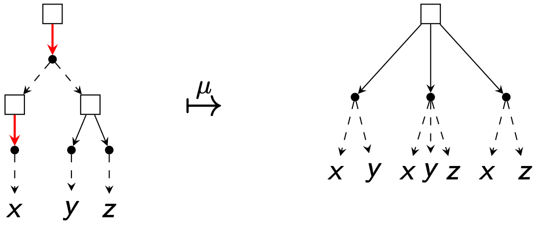
- Composition in $\mathbf{Kl}(\mathcal{G})$ *computes strategies*:



- Given a game $\delta : X \to \mathcal{G}(X)$

Games are in $\mathbf{Kl}(\mathcal{G})$

- Composition in $\mathbf{Kl}(\mathcal{G})$ *computes strategies*:

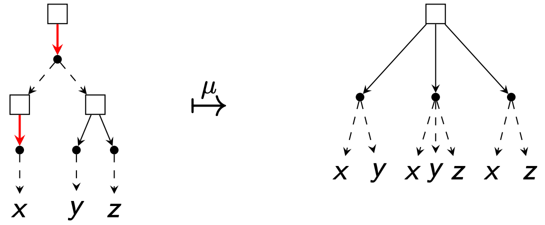


- Given a game $\delta : X \to \mathcal{G}(X)$
- We use a $\gamma : X \to \mathcal{G}(X \times X)$ which *records state information*:

$$\gamma(x) := \{\{(x, x') \mid x' \in U\} \mid U \in \delta(x)\}$$

Games are in $\mathbf{Kl}(\mathcal{G})$

- Composition in $\mathbf{Kl}(\mathcal{G})$ *computes strategies*:



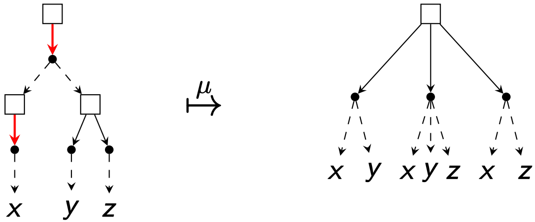
- Given a game $\delta : X \to \mathcal{G}(X)$
- We use a $\gamma : X \to \mathcal{G}(X \times X)$ which *records state information*:

$$\gamma(x) := \{\{(x, x') \mid x' \in U\} \mid U \in \delta(x)\}$$

- Define $\gamma_n : X \to \mathcal{G}(X^n \times X)$ as iterating γ in $\mathbf{Kl}(\mathcal{G})$

Games are in $\mathbf{Kl}(\mathcal{G})$

- Composition in $\mathbf{Kl}(\mathcal{G})$ *computes strategies*:



- Given a game $\delta : X \to \mathcal{G}(X)$
- We use a $\gamma : X \to \mathcal{G}(X \times X)$ which *records state information*:

$$\gamma(x) := \{\{(x, x') \mid x' \in U\} \mid U \in \delta(x)\}$$

- Define $\gamma_n : X \to \mathcal{G}(X^n \times X)$ as iterating γ in $\mathbf{Kl}(\mathcal{G})$
  - We can show $\gamma_n(x)$ consists of n-step strategies starting from x

The game outcome as a limit

- Define $\gamma_n : X \nrightarrow F^n(X)$ by iterating $\gamma : X \nrightarrow F(X)$ in $\mathbf{Kl}(\mathcal{G})$
- Define $\mathrm{Im}(\gamma_n) = \bigcup_{x \in X} \gamma_n(x)$.

The game outcome as a limit

- Define $\gamma_n : X \nrightarrow F^n(X)$ by iterating $\gamma : X \nrightarrow F(X)$ in $\mathbf{Kl}(\mathcal{G})$
- Define $\mathrm{Im}(\gamma_n) = \bigcup_{x \in X} \gamma_n(x)$.
- Think of: $U \in \mathrm{Im}(\gamma_n)$ as an *n-step strategy*
  - (Using $\mu^{\mathcal{G}}$ to compute strategies)

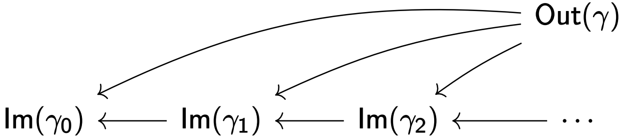
The game outcome as a limit

- Define $\gamma_n : X \twoheadrightarrow F^n(X)$ by iterating $\gamma : X \twoheadrightarrow F(X)$ in $\mathbf{Kl}(\mathcal{G})$
- Define $\mathrm{Im}(\gamma_n) = \bigcup_{x \in X} \gamma_n(x)$.
- Think of: $U \in \mathrm{Im}(\gamma_n)$ as an *n-step strategy*
  - (Using $\mu^{\mathcal{G}}$ to compute strategies)
- We can define the outcome of γ as a categorical limit:

$$\mathrm{Im}(\gamma_0) \longleftarrow \mathrm{Im}(\gamma_1) \longleftarrow \mathrm{Im}(\gamma_2) \longleftarrow \cdots$$

The game outcome as a limit

- Define $\gamma_n : X \nrightarrow F^n(X)$ by iterating $\gamma : X \nrightarrow F(X)$ in $\mathbf{Kl}(\mathcal{G})$
- Define $\mathrm{Im}(\gamma_n) = \bigcup_{x \in X} \gamma_n(x)$.
- Think of: $U \in \mathrm{Im}(\gamma_n)$ as an *n-step strategy*
  - (Using $\mu^{\mathcal{G}}$ to compute strategies)
- We can define the outcome of γ as a categorical limit:



Alternative characterisation (1): greatest homomorphism

- There exists a greatest homomorphism $f : X \to \mathcal{G}(X^\omega)$:

$$
\begin{array}{ccc}
X & \xrightarrow{\ f\ } & X^\omega \\
{\scriptstyle\gamma}\downarrow & & \downarrow{\scriptstyle\bar{\bar{\zeta}}} \\
X \times X & \xrightarrow[X \times f]{} & X \times X^\omega
\end{array}
\qquad \boxed{\mathbf{Kl}(\mathcal{G})}
$$

Alternative characterisation (1): greatest homomorphism

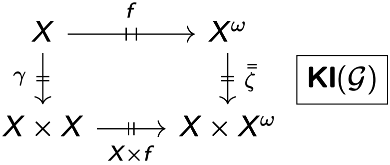
- There exists a greatest homomorphism $f : X \to \mathcal{G}(X^\omega)$:

$$\begin{array}{ccc} X & \xrightarrow{\;f\;} & X^\omega \\ {\scriptstyle\gamma}\big\downarrow & & \big\downarrow{\scriptstyle\bar{\bar{\zeta}}} \\ X \times X & \xrightarrow[X \times f]{} & X \times X^\omega \end{array} \qquad \boxed{\mathbf{Kl}(\mathcal{G})}$$

- Unfortunately, the greatest homomorphism contains too many subsets

Alternative characterisation (1): greatest homomorphism

- There exists a greatest homomorphism $f : X \to \mathcal{G}(X^\omega)$:



- Unfortunately, the greatest homomorphism contains too many subsets
- It is possible to restrict homomorphisms such that each $V \in f(x)$ is *limit closed* [Emerson, 1983]
  - Roughly means V contains every possible infinite play which conforms to finite approximations of V

Alternative characterisation (1): greatest homomorphism

- There exists a greatest homomorphism $f : X \to \mathcal{G}(X^\omega)$:

$$\begin{array}{ccc} X & \xrightarrow{\;f\;} & X^\omega \\ {\scriptstyle\gamma}\downarrow & & \downarrow{\scriptstyle\bar{\bar{\zeta}}} \\ X \times X & \xrightarrow[X \times f]{} & X \times X^\omega \end{array} \qquad \boxed{\mathbf{Kl}(\mathcal{G})}$$

- Unfortunately, the greatest homomorphism contains too many subsets
- It is possible to restrict homomorphisms such that each $V \in f(x)$ is *limit closed* [Emerson, 1983]
  - Roughly means V contains every possible infinite play which conforms to finite approximations of V
- Theorem: the greatest homomorphism f which contains limit-closed subsets recovers the outcome of a game.
  - $f(x)$ gives us the strategies which start in x

$$f(x) = \mathrm{Out}(\gamma)(x)$$

Alternative characterisation (2): least homomorphism

- Assume the game $\gamma : X \twoheadrightarrow \mathcal{G}(X \times X)$ has no controller deadlocks

Alternative characterisation (2): least homomorphism

- Assume the game $\gamma : X \nrightarrow \mathcal{G}(X \times X)$ has no controller deadlocks
- There is *maximally permissive* strategy from every state $x \in X$

Alternative characterisation (2): least homomorphism

- Assume the game $\gamma : X \twoheadrightarrow \mathcal{G}(X \times X)$ has no controller deadlocks
- There is *maximally permissive* strategy from every state $x \in X$
  - Denote by $Z_{\gamma,x} \subseteq X^\omega$ this outcome

Alternative characterisation (2): least homomorphism

- Assume the game $\gamma : X \twoheadrightarrow \mathcal{G}(X \times X)$ has no controller deadlocks
- There is *maximally permissive* strategy from every state $x \in X$
  - Denote by $Z_{\gamma,x} \subseteq X^\omega$ this outcome
- The space of maps $X \to \mathcal{G}(X^\omega)$ can be restricted to those f s.t.

$$Z_{\gamma,x} \in f(x)$$

Alternative characterisation (2): least homomorphism

- Assume the game $\gamma : X \twoheadrightarrow \mathcal{G}(X \times X)$ has no controller deadlocks
- There is *maximally permissive* strategy from every state $x \in X$
  - Denote by $Z_{\gamma,x} \subseteq X^\omega$ this outcome
- The space of maps $X \to \mathcal{G}(X^\omega)$ can be restricted to those f s.t.

$$Z_{\gamma,x} \in f(x)$$

- The least such map $\bot_\gamma$ maps $x \mapsto \{Z_{\gamma,x}\}$

Alternative characterisation (2): least homomorphism

- Assume the game $\gamma : X \twoheadrightarrow \mathcal{G}(X \times X)$ has no controller deadlocks
- There is *maximally permissive* strategy from every state $x \in X$
  - Denote by $Z_{\gamma,x} \subseteq X^\omega$ this outcome
- The space of maps $X \to \mathcal{G}(X^\omega)$ can be restricted to those f s.t.

$$Z_{\gamma,x} \in f(x)$$

- The least such map $\perp_\gamma$ maps $x \mapsto \{Z_{\gamma,x}\}$
- Theorem: let $f : X \to \mathcal{G}(X^\omega)$ be the least homomorphism $(X, \gamma) \to (Z, \overline{\overline{\zeta}})$ such that $Z_{\gamma,x} \in f(x)$, if we close $f(x)$ under intersections of descending chains of subsets, we obtain the map which assigns states to their outcomes.

$$\text{close}(f(x)) = \text{Out}(\gamma)(x)$$

Conclusion

- Strategy outcomes are captured by:
  - Categorical limits
  - Largest homomorphisms
  - Largest mediating maps

Conclusion

- Strategy outcomes are captured by:
  - Categorical limits
  - Largest homomorphisms
  - Largest mediating maps
- Game outcomes are captured by:
  - Categorical limits
  - Largest homomorphisms
  - Least homomorphisms + closing

Conclusion

- Strategy outcomes are captured by:
  - Categorical limits
  - Largest homomorphisms
  - Largest mediating maps
- Game outcomes are captured by:
  - Categorical limits
  - Largest homomorphisms
  - Least homomorphisms + closing
- In the paper we treat linear functors $A \times (-) + B$

Conclusion

- Strategy outcomes are captured by:
  - Categorical limits
  - Largest homomorphisms
  - Largest mediating maps
- Game outcomes are captured by:
  - Categorical limits
  - Largest homomorphisms
  - Least homomorphisms + closing
- In the paper we treat linear functors $A \times (-) + B$
- Future work:
  - Expand the picture to:
    - Stochastic environments (MDPs)
    - Stochastic + nondeterministic environments (SSGs)
    - Weighted transitions (energy games)
  - Add objectives:
    - Coalgebraic approach to Büchi/parity conditions [Urabe et al., 2016]
    - Product construction of a game with an automaton
  - Generic techniques for computing/approximating solutions

References I

Emerson, E. (1983).
Alternative semantics for temporal logics.
Theoretical Computer Science, 26(1):121–130.

Plummer, B. and Cirstea, C. (2025).
Traces via strategies in two-player games.
Electronic Notes in Theoretical Informatics and Computer Science, Volume 5 - Proceedings of MFPS XLI.

Urabe, N., Shimizu, S., and Hasuo, I. (2016).
Coalgebraic Trace Semantics for Buechi and Parity Automata.
In Desharnais, J. and Jagadeesan, R., editors, *27th International Conference on Concurrency Theory (CONCUR 2016)*, volume 59 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 24:1–24:15, Dagstuhl, Germany. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.