Relating Structure to Power: from categorical semantics to descriptive complexity

Samson Abramsky

Department of Computer Science, University of Oxford

Structure *vs* Power: The Great Divide

Structure:

- compositionality, semantics
- How we can master the complexity of computer systems and software?

Power:

- expressiveness, complexity
- How we can harness the power of computation and recognize its limits?

Structure *vs* Power: The Great Divide

Structure:

- compositionality, semantics
- How we can master the complexity of computer systems and software?

Power:

- expressiveness, complexity
- How we can harness the power of computation and recognize its limits?

A *shocking fact*: the current state of the art is almost *disjoint communities* of researchers studying Structure and Power respectively, with no common technical language or tools.

Structure *vs* Power: The Great Divide

Structure:

- compositionality, semantics
- How we can master the complexity of computer systems and software?

Power:

- expressiveness, complexity
- How we can harness the power of computation and recognize its limits?

A *shocking fact*: the current state of the art is almost *disjoint communities* of researchers studying Structure and Power respectively, with no common technical language or tools.

This is a major obstacle to fundamental progress in Computer Science.

Structure *vs* Power: The Great Divide

Structure:

- compositionality, semantics
- How we can master the complexity of computer systems and software?

Power:

- expressiveness, complexity
- How we can harness the power of computation and recognize its limits?

A *shocking fact*: the current state of the art is almost *disjoint communities* of researchers studying Structure and Power respectively, with no common technical language or tools.

This is a major obstacle to fundamental progress in Computer Science.

An analogy: the Grothendieck program in algebraic geometry. The (very abstract) tools developed there were ultimately critical for concrete results, e.g. Wiles/FLT.

Structure *vs* Power: The Great Divide

Structure:

- compositionality, semantics
- How we can master the complexity of computer systems and software?

Power:

- expressiveness, complexity
- How we can harness the power of computation and recognize its limits?

A *shocking fact*: the current state of the art is almost *disjoint communities* of researchers studying Structure and Power respectively, with no common technical language or tools.

This is a major obstacle to fundamental progress in Computer Science.

An analogy: the Grothendieck program in algebraic geometry. The (very abstract) tools developed there were ultimately critical for concrete results, e.g. Wiles/FLT.

Mazur quoting Lenstra:

> *twenty years ago he was firm in his conviction that he DID want to solve Diophantine equations, and that he DID NOT wish to represent functors – and now he is amused to discover himself representing functors in order to solve Diophantine equations!*

The topic for this talk

We shall discuss a novel approach to relating categorical semantics, which exemplifies “Structure”, to finite model theory, which exemplifies “Power”.

The topic for this talk

We shall discuss a novel approach to relating categorical semantics, which exemplifies "Structure", to finite model theory, which exemplifies "Power".

Based on:

- The Pebbling Comonad in Finite Model Theory, SA, Anuj Dawar and Pengming Wang, LiCS 2017.
- Relating Structure to Power: comonadic semantics for computational resources, SA and Nihil Shah, CSL 2018. Extended version to appear in *Journal of Logic and Computation*. Available at `arXiv:2010.06496`.

The topic for this talk

We shall discuss a novel approach to relating categorical semantics, which exemplifies "Structure", to finite model theory, which exemplifies "Power".

Based on:

- The Pebbling Comonad in Finite Model Theory, SA, Anuj Dawar and Pengming Wang, LiCS 2017.
- Relating Structure to Power: comonadic semantics for computational resources, SA and Nihil Shah, CSL 2018. Extended version to appear in *Journal of Logic and Computation*. Available at `arXiv:2010.06496`.

Current EPSRC project with Anuj Dawar (Cambridge) on:
Resources and Co-resources: a junction between categorical semantics and descriptive complexity.

Post-docs Dan Marsden, Luca Reggio (Marie-Curie Fellow), Tomáš Jakl
Ph.D. students Tom Paine, Nihil Shah, Adam Ó Conghaile.

People

People

Anuj Dawar

Dan Marsden

Luca Reggio

Tomáš Jakl

Nihil Shah

Tom Paine

Adam Ó Conghaile

Model theory and Resources

Extensional category: relational structures

Model theory and Resources

Extensional category: relational structures

Extrinsic machinery:

Model theory and Resources

Extensional category: relational structures

Extrinsic machinery:

- Logical languages
  - Hierarchies and fragments: quantifier rank, number of variables, . . .

Model theory and Resources

Extensional category: relational structures

Extrinsic machinery:

- Logical languages
  - Hierarchies and fragments: quantifier rank, number of variables, ...
- Games
  - number of rounds
  - use of auxiliary resources, e.g. number of pebbles.

Model theory and Resources

Extensional category: relational structures

Extrinsic machinery:

- Logical languages
  - Hierarchies and fragments: quantifier rank, number of variables, . . .
- Games
  - number of rounds
  - use of auxiliary resources, e.g. number of pebbles.

Question

Can we give a more structural, intrinsic account of resources?

Model theory and Resources

Extensional category: relational structures

Extrinsic machinery:

- Logical languages
  - Hierarchies and fragments: quantifier rank, number of variables, ...
- Games
  - number of rounds
  - use of auxiliary resources, e.g. number of pebbles.

Question

Can we give a more structural, intrinsic account of resources?

Many potential benefits: generality, new connections, techniques and questions ...

Logic and resources: big picture

Logic and resources: big picture

- A key ingredient is the assignment of a *process structure*—an intensional description—to an extensional object, such as a function, a set, or a relational structure.

Logic and resources: big picture

- A key ingredient is the assignment of a *process structure*—an intensional description—to an extensional object, such as a function, a set, or a relational structure.
- It is this process structure, unfolding in space and time, to which a resource parameter can be applied, which can then be transferred to the extensional object.

Logic and resources: big picture

- A key ingredient is the assignment of a *process structure*—an intensional description—to an extensional object, such as a function, a set, or a relational structure.
- It is this process structure, unfolding in space and time, to which a resource parameter can be applied, which can then be transferred to the extensional object.
- At the basic level of computability, this happens when we assign a Turing machine description or a Gödel number to a recursive function. It is then meaningful to assign a complexity measure to the function.

Logic and resources: big picture

- A key ingredient is the assignment of a *process structure*—an intensional description—to an extensional object, such as a function, a set, or a relational structure.
- It is this process structure, unfolding in space and time, to which a resource parameter can be applied, which can then be transferred to the extensional object.
- At the basic level of computability, this happens when we assign a Turing machine description or a Gödel number to a recursive function. It is then meaningful to assign a complexity measure to the function.
- The same phenomenon arises in semantics: for example, the notion of *sequentiality* is applicable to a *process* computing a higher-order function. Reifying these processes in the form of *game semantics* led to a resolution of the famous full abstraction problem for PCF, and to a wealth of subsequent results.

Our approach: game comonads

Our approach: game comonads

- In our approach, we build tree-structured covers of a given, purely extensional relational structure.

Our approach: game comonads

- In our approach, we build tree-structured covers of a given, purely extensional relational structure.
- Such a tree cover will in general not have the full properties of the original structure, but be a "best approximation" in some resource-restricted setting.

Our approach: game comonads

- In our approach, we build tree-structured covers of a given, purely extensional relational structure.
- Such a tree cover will in general not have the full properties of the original structure, but be a “best approximation” in some resource-restricted setting.
- More precisely, this means that we have a (comonadic) adjunction, yielding the corresponding comonad.

Our approach: game comonads

- In our approach, we build tree-structured covers of a given, purely extensional relational structure.
- Such a tree cover will in general not have the full properties of the original structure, but be a "best approximation" in some resource-restricted setting.
- More precisely, this means that we have a (comonadic) adjunction, yielding the corresponding comonad.
- The objects of the category where the approximations live have an intrinsic tree structure, which can be captured axiomatically, as *arboreal categories*.

Our approach: game comonads

- In our approach, we build tree-structured covers of a given, purely extensional relational structure.
- Such a tree cover will in general not have the full properties of the original structure, but be a "best approximation" in some resource-restricted setting.
- More precisely, this means that we have a (comonadic) adjunction, yielding the corresponding comonad.
- The objects of the category where the approximations live have an intrinsic tree structure, which can be captured axiomatically, as *arboreal categories*.
- The tree encodes a process for generating (parts of) the relational structure, to which resource notions can be applied.

Our approach: game comonads

- In our approach, we build tree-structured covers of a given, purely extensional relational structure.
- Such a tree cover will in general not have the full properties of the original structure, but be a "best approximation" in some resource-restricted setting.
- More precisely, this means that we have a (comonadic) adjunction, yielding the corresponding comonad.
- The objects of the category where the approximations live have an intrinsic tree structure, which can be captured axiomatically, as *arboreal categories*.
- The tree encodes a process for generating (parts of) the relational structure, to which resource notions can be applied.
- This allows us to apply resource notions to the objects of the extensional category via the adjunction.

The setting: relational structures

The setting: relational structures

A relational vocabulary σ is a family of relation symbols R, each of some arity $n > 0$.

The setting: relational structures

A relational vocabulary σ is a family of relation symbols R, each of some arity $n > 0$.

A relational structure for σ is $\mathcal{A} = (A, \{R^A \mid R \in \sigma\})$, where $R^{\mathcal{A}} \subseteq A^n$.

The setting: relational structures

A relational vocabulary σ is a family of relation symbols R, each of some arity $n > 0$.

A relational structure for σ is $\mathcal{A} = (A, \{R^A \mid R \in \sigma\})$, where $R^{\mathcal{A}} \subseteq A^n$.

A homomorphism of σ-structures $f : \mathcal{A} \to \mathcal{B}$ is a function $f : A \to B$ such that, for each relation $R \in \sigma$ of arity n and $(a_1, \ldots, a_n) \in A^n$:

$$(a_1, \ldots, a_n) \in R^{\mathcal{A}} \;\Rightarrow\; (f(a_1), \ldots, f(a_n))) \in R^{\mathcal{B}}.$$

The setting: relational structures

A relational vocabulary σ is a family of relation symbols R, each of some arity $n > 0$.

A relational structure for σ is $\mathcal{A} = (A, \{R^A \mid R \in \sigma\})$, where $R^{\mathcal{A}} \subseteq A^n$.

A homomorphism of σ-structures $f : \mathcal{A} \to \mathcal{B}$ is a function $f : A \to B$ such that, for each relation $R \in \sigma$ of arity n and $(a_1, \ldots, a_n) \in A^n$:

$$(a_1, \ldots, a_n) \in R^{\mathcal{A}} \;\Rightarrow\; (f(a_1), \ldots, f(a_n))) \in R^{\mathcal{B}}.$$

There notions are pervasive in

- logic (model theory),
- computer science (databases, constraint satisfaction, finite model theory)
- combinatorics (graphs and graph homomorphisms).

The setting: relational structures

A relational vocabulary σ is a family of relation symbols R, each of some arity $n > 0$.

A relational structure for σ is $\mathcal{A} = (A, \{R^A \mid R \in \sigma\})$, where $R^{\mathcal{A}} \subseteq A^n$.

A homomorphism of σ-structures $f : \mathcal{A} \to \mathcal{B}$ is a function $f : A \to B$ such that, for each relation $R \in \sigma$ of arity n and $(a_1, \ldots, a_n) \in A^n$:

$$(a_1, \ldots, a_n) \in R^{\mathcal{A}} \;\Rightarrow\; (f(a_1), \ldots, f(a_n))) \in R^{\mathcal{B}}.$$

There notions are pervasive in

- logic (model theory),
- computer science (databases, constraint satisfaction, finite model theory)
- combinatorics (graphs and graph homomorphisms).

Our setting will be $\mathcal{R}(\sigma)$, the category of relational structures and homomorphisms.
This will be the *extensional category*.

First example

A *forest* is a poset $(F, \leq)$ such that, for all $x \in F$, the set of predecessors is a finite chain.

A forest morphism preserves roots and the covering relation.

First example

A *forest* is a poset $(F, \leq)$ such that, for all $x \in F$, the set of predecessors is a finite chain.
A forest morphism preserves roots and the covering relation.

A *forest-ordered σ-structure* $(\mathcal{A}, \leq)$ is a σ-structure $\mathcal{A}$ with a forest order $\leq$ on A.

First example

A *forest* is a poset $(F, \leq)$ such that, for all $x \in F$, the set of predecessors is a finite chain.
A forest morphism preserves roots and the covering relation.

A *forest-ordered σ-structure* $(\mathcal{A}, \leq)$ is a σ-structure $\mathcal{A}$ with a forest order $\leq$ on A.

This must satisfy condition (E): elements of A which are adjacent in the Gaifman graph of $\mathcal{A}$ must be comparable in the order.

First example

A *forest* is a poset $(F, \leq)$ such that, for all $x \in F$, the set of predecessors is a finite chain.
A forest morphism preserves roots and the covering relation.

A *forest-ordered* σ*-structure* $(\mathcal{A}, \leq)$ is a σ-structure $\mathcal{A}$ with a forest order $\leq$ on A.

This must satisfy condition (E): elements of A which are adjacent in the Gaifman graph of $\mathcal{A}$ must be comparable in the order.

The *mininum height* of such a forest order on $\mathcal{A}$ is the *tree-depth* of $\mathcal{A}$ (Nešetřil and Ossona de Mendez).

First example

A *forest* is a poset $(F, \leq)$ such that, for all $x \in F$, the set of predecessors is a finite chain.
A forest morphism preserves roots and the covering relation.

A *forest-ordered σ-structure* $(\mathcal{A}, \leq)$ is a σ-structure $\mathcal{A}$ with a forest order $\leq$ on A.

This must satisfy condition (E): elements of A which are adjacent in the Gaifman graph of $\mathcal{A}$ must be comparable in the order.

The *mininum height* of such a forest order on $\mathcal{A}$ is the *tree-depth* of $\mathcal{A}$ (Nešetřil and Ossona de Mendez).

Important combinatorial parameter, used extensively by Rossman in his Homomorphism Preservation Theorems.

Resource cover and adjunction

There is a category $\mathcal{R}^{E}(\sigma)$, and an evident forgetful functor $U : \mathcal{R}^{E}(\sigma) \rightarrow \mathcal{R}(\sigma)$.

Resource cover and adjunction

There is a category $\mathcal{R}^E(\sigma)$, and an evident forgetful functor $U : \mathcal{R}^E(\sigma) \to \mathcal{R}(\sigma)$.

For each $k > 0$, if we restrict to forest orders of height $\leq k$, we get a sub-category $\mathcal{R}^E_k(\sigma)$, and a functor $U_k : \mathcal{R}^E_k(\sigma) \to \mathcal{R}(\sigma)$.

Resource cover and adjunction

There is a category $\mathcal{R}^E(\sigma)$, and an evident forgetful functor $U : \mathcal{R}^E(\sigma) \to \mathcal{R}(\sigma)$.

For each $k > 0$, if we restrict to forest orders of height $\leq k$, we get a sub-category $\mathcal{R}^E_k(\sigma)$, and a functor $U_k : \mathcal{R}^E_k(\sigma) \to \mathcal{R}(\sigma)$.

This functor has a right adjoint G_k, giving rise to a comonad $\mathbb{E}_k = U_k G_k$ on $\mathcal{R}(\sigma)$.

Resource cover and adjunction

There is a category $\mathcal{R}^E(\sigma)$, and an evident forgetful functor $U : \mathcal{R}^E(\sigma) \to \mathcal{R}(\sigma)$.

For each $k > 0$, if we restrict to forest orders of height $\leq k$, we get a sub-category $\mathcal{R}^E_k(\sigma)$, and a functor $U_k : \mathcal{R}^E_k(\sigma) \to \mathcal{R}(\sigma)$.

This functor has a right adjoint G_k, giving rise to a comonad $\mathbb{E}_k = U_k G_k$ on $\mathcal{R}(\sigma)$.

- Given a structure $\mathcal{A}$, the universe of $G_k\mathcal{A}$ is $A^{\leq k}$, the non-empty sequences of length $\leq k$.

Resource cover and adjunction

There is a category $\mathcal{R}^E(\sigma)$, and an evident forgetful functor $U : \mathcal{R}^E(\sigma) \to \mathcal{R}(\sigma)$.

For each $k > 0$, if we restrict to forest orders of height $\leq k$, we get a sub-category $\mathcal{R}^E_k(\sigma)$, and a functor $U_k : \mathcal{R}^E_k(\sigma) \to \mathcal{R}(\sigma)$.

This functor has a right adjoint G_k, giving rise to a comonad $\mathbb{E}_k = U_k G_k$ on $\mathcal{R}(\sigma)$.

- Given a structure $\mathcal{A}$, the universe of $G_k\mathcal{A}$ is $A^{\leq k}$, the non-empty sequences of length $\leq k$.
- This is forest-ordered by the prefix order.

Resource cover and adjunction

There is a category $\mathcal{R}^E(\sigma)$, and an evident forgetful functor $U : \mathcal{R}^E(\sigma) \to \mathcal{R}(\sigma)$.

For each $k > 0$, if we restrict to forest orders of height $\leq k$, we get a sub-category $\mathcal{R}^E_k(\sigma)$, and a functor $U_k : \mathcal{R}^E_k(\sigma) \to \mathcal{R}(\sigma)$.

This functor has a right adjoint G_k, giving rise to a comonad $\mathbb{E}_k = U_k G_k$ on $\mathcal{R}(\sigma)$.

- Given a structure $\mathcal{A}$, the universe of $G_k\mathcal{A}$ is $A^{\leq k}$, the non-empty sequences of length $\leq k$.
- This is forest-ordered by the prefix order.
- The counit map $\varepsilon_{\mathcal{A}} : A^{\leq k} \to A$ sends a sequence $[a_1, \ldots, a_n]$ to a_n.

Resource cover and adjunction

There is a category $\mathcal{R}^E(\sigma)$, and an evident forgetful functor $U : \mathcal{R}^E(\sigma) \to \mathcal{R}(\sigma)$.

For each $k > 0$, if we restrict to forest orders of height $\leq k$, we get a sub-category $\mathcal{R}^E_k(\sigma)$, and a functor $U_k : \mathcal{R}^E_k(\sigma) \to \mathcal{R}(\sigma)$.

This functor has a right adjoint G_k, giving rise to a comonad $\mathbb{E}_k = U_k G_k$ on $\mathcal{R}(\sigma)$.

- Given a structure $\mathcal{A}$, the universe of $G_k\mathcal{A}$ is $A^{\leq k}$, the non-empty sequences of length $\leq k$.
- This is forest-ordered by the prefix order.
- The counit map $\varepsilon_{\mathcal{A}} : A^{\leq k} \to A$ sends a sequence $[a_1, \ldots, a_n]$ to a_n.
- How do we lift the relations on $\mathcal{A}$ to $G_k\mathcal{A}$?

Resource cover and adjunction

There is a category $\mathcal{R}^E(\sigma)$, and an evident forgetful functor $U : \mathcal{R}^E(\sigma) \to \mathcal{R}(\sigma)$.

For each $k > 0$, if we restrict to forest orders of height $\leq k$, we get a sub-category $\mathcal{R}^E_k(\sigma)$, and a functor $U_k : \mathcal{R}^E_k(\sigma) \to \mathcal{R}(\sigma)$.

This functor has a right adjoint G_k, giving rise to a comonad $\mathbb{E}_k = U_k G_k$ on $\mathcal{R}(\sigma)$.

- Given a structure $\mathcal{A}$, the universe of $G_k\mathcal{A}$ is $A^{\leq k}$, the non-empty sequences of length $\leq k$.
- This is forest-ordered by the prefix order.
- The counit map $\varepsilon_{\mathcal{A}} : A^{\leq k} \to A$ sends a sequence $[a_1, \ldots, a_n]$ to a_n.
- How do we lift the relations on $\mathcal{A}$ to $G_k\mathcal{A}$?
- Given e.g. a binary relation R, we define $R^{G_k\mathcal{A}}$ to the set of pairs (s, t) such that

Resource cover and adjunction

There is a category $\mathcal{R}^E(\sigma)$, and an evident forgetful functor $U : \mathcal{R}^E(\sigma) \to \mathcal{R}(\sigma)$.

For each $k > 0$, if we restrict to forest orders of height $\leq k$, we get a sub-category $\mathcal{R}^E_k(\sigma)$, and a functor $U_k : \mathcal{R}^E_k(\sigma) \to \mathcal{R}(\sigma)$.

This functor has a right adjoint G_k, giving rise to a comonad $\mathbb{E}_k = U_k G_k$ on $\mathcal{R}(\sigma)$.

- Given a structure $\mathcal{A}$, the universe of $G_k\mathcal{A}$ is $A^{\leq k}$, the non-empty sequences of length $\leq k$.
- This is forest-ordered by the prefix order.
- The counit map $\varepsilon_{\mathcal{A}} : A^{\leq k} \to A$ sends a sequence $[a_1, \ldots, a_n]$ to a_n.
- How do we lift the relations on $\mathcal{A}$ to $G_k\mathcal{A}$?
- Given e.g. a binary relation R, we define $R^{G_k\mathcal{A}}$ to the set of pairs (s, t) such that
  - $s \sqsubseteq t$ or $t \sqsubseteq s$ (in prefix order)

Resource cover and adjunction

There is a category $\mathcal{R}^E(\sigma)$, and an evident forgetful functor $U : \mathcal{R}^E(\sigma) \to \mathcal{R}(\sigma)$.

For each $k > 0$, if we restrict to forest orders of height $\leq k$, we get a sub-category $\mathcal{R}^E_k(\sigma)$, and a functor $U_k : \mathcal{R}^E_k(\sigma) \to \mathcal{R}(\sigma)$.

This functor has a right adjoint G_k, giving rise to a comonad $\mathbb{E}_k = U_k G_k$ on $\mathcal{R}(\sigma)$.

- Given a structure $\mathcal{A}$, the universe of $G_k\mathcal{A}$ is $A^{\leq k}$, the non-empty sequences of length $\leq k$.
- This is forest-ordered by the prefix order.
- The counit map $\varepsilon_{\mathcal{A}} : A^{\leq k} \to A$ sends a sequence $[a_1, \ldots, a_n]$ to a_n.
- How do we lift the relations on $\mathcal{A}$ to $G_k\mathcal{A}$?
- Given e.g. a binary relation R, we define $R^{G_k\mathcal{A}}$ to the set of pairs (s, t) such that
  - $s \sqsubseteq t$ or $t \sqsubseteq s$ (in prefix order)
  - $R^{\mathcal{A}}(\varepsilon_{\mathcal{A}}(s), \varepsilon_{\mathcal{A}}(t))$.

The counIversal property

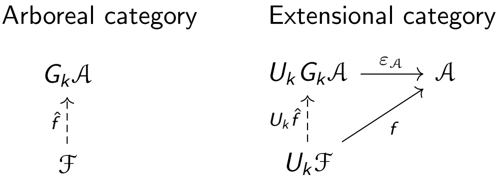
Arboreal category

$$
\begin{array}{c}
G_k\mathcal{A} \\
\hat{f} \Big\uparrow \\
\mathcal{F}
\end{array}
$$

Extensional category

$$
\begin{array}{ccc}
U_k G_k \mathcal{A} & \xrightarrow{\varepsilon_{\mathcal{A}}} & \mathcal{A} \\
U_k \hat{f} \Big\uparrow & \nearrow_{f} & \\
U_k \mathcal{F} & &
\end{array}
$$

The couniversal property



Moreover, the adjunction is *comonadic*, meaning that the category of coalgebras for $\mathbb{E}_k$ is exactly $\mathcal{R}_k^E(\sigma)$.

Consequences

Remember that the right adjoint is uniquely determined by the forgetful functor, and the comonad by the adjunction.

Consequences

Remember that the right adjoint is uniquely determined by the forgetful functor, and the comonad by the adjunction.

As we shall now see, this structure gives us directly:

- The Ehrenfeucht-Fraïssé game
- The quantifier-rank indexed fragments of FOL
- Equivalences of structures induced by:
  - the full fragment of q.r. $\leq k$
  - the existential positive part of the fragment
  - the extension of the fragment with counting quantifiers
- We also recover the important tree-depth combinatorial parameter from the coalgebras of the comonad

Consequences

Remember that the right adjoint is uniquely determined by the forgetful functor, and the comonad by the adjunction.

As we shall now see, this structure gives us directly:

- The Ehrenfeucht-Fraïssé game
- The quantifier-rank indexed fragments of FOL
- Equivalences of structures induced by:
  - the full fragment of q.r. $\leq k$
  - the existential positive part of the fragment
  - the extension of the fragment with counting quantifiers
- We also recover the important tree-depth combinatorial parameter from the coalgebras of the comonad

Moreover, this template can be used to give similar analyses of a wealth of other logical and combinatorial notions.

Consequences

Remember that the right adjoint is uniquely determined by the forgetful functor, and the comonad by the adjunction.

As we shall now see, this structure gives us directly:

- The Ehrenfeucht-Fraïssé game
- The quantifier-rank indexed fragments of FOL
- Equivalences of structures induced by:
  - the full fragment of q.r. $\leq k$
  - the existential positive part of the fragment
  - the extension of the fragment with counting quantifiers
- We also recover the important tree-depth combinatorial parameter from the coalgebras of the comonad

Moreover, this template can be used to give similar analyses of a wealth of other logical and combinatorial notions.

This general pattern has been axiomatised in *Arboreal Categories and Resources*, SA and Luca Reggio, to appear in ICALP 2021, available at `arXiv:2102.08109`.

CoKleisli maps are strategies

Intuitively, an element of $A^{\leq k}$ represents a play in $\mathcal{A}$ of length $\leq k$.

CoKleisli maps are strategies

Intuitively, an element of $A^{\leq k}$ represents a play in $\mathcal{A}$ of length $\leq k$.

A coKleisli morphism $\mathbb{E}_k \mathcal{A} \to \mathcal{B}$ represents a Duplicator strategy for the existential Ehrenfeucht-Fraïssé game with k rounds:

CoKleisli maps are strategies

Intuitively, an element of $A^{\leq k}$ represents a play in $\mathcal{A}$ of length $\leq k$.

A coKleisli morphism $\mathbb{E}_k\mathcal{A} \to \mathcal{B}$ represents a Duplicator strategy for the existential Ehrenfeucht-Fraïssé game with k rounds:

Spoiler plays only in $\mathcal{A}$, and $b_i = f[a_1, \ldots, a_i]$ represents Duplicator's response in $\mathcal{B}$ to the i'th move by Spoiler.

CoKleisli maps are strategies

Intuitively, an element of $A^{\leq k}$ represents a play in $\mathcal{A}$ of length $\leq k$.

A coKleisli morphism $\mathbb{E}_k \mathcal{A} \to \mathcal{B}$ represents a Duplicator strategy for the existential Ehrenfeucht-Fraïssé game with k rounds:

Spoiler plays only in $\mathcal{A}$, and $b_i = f[a_1, \ldots, a_i]$ represents Duplicator's response in $\mathcal{B}$ to the i'th move by Spoiler.

The winning condition for Duplicator in this game is that, after k rounds have been played, the induced relation $\{(a_i, b_i) \mid 1 \leq i \leq k\}$ is a partial homomorphism from $\mathcal{A}$ to $\mathcal{B}$.

CoKleisli maps are strategies

Intuitively, an element of $A^{\leq k}$ represents a play in $\mathcal{A}$ of length $\leq k$.

A coKleisli morphism $\mathbb{E}_k\mathcal{A} \to \mathcal{B}$ represents a Duplicator strategy for the existential Ehrenfeucht-Fraïssé game with k rounds:

Spoiler plays only in $\mathcal{A}$, and $b_i = f[a_1, \ldots, a_i]$ represents Duplicator's response in $\mathcal{B}$ to the i'th move by Spoiler.

The winning condition for Duplicator in this game is that, after k rounds have been played, the induced relation $\{(a_i, b_i) \mid 1 \leq i \leq k\}$ is a partial homomorphism from $\mathcal{A}$ to $\mathcal{B}$.

Theorem

The following are equivalent:

1. *There is a homomorphism* $\mathbb{E}_k\mathcal{A} \to \mathcal{B}$.
2. *Duplicator has a winning strategy for the existential Ehrenfeucht-Fraïssé game with* k *rounds, played from* $\mathcal{A}$ *to* $\mathcal{B}$.
3. *For every existential positive sentence* φ *with quantifier rank* $\leq k$, $\mathcal{A} \models \varphi \;\Rightarrow\; \mathcal{B} \models \varphi$.

Open pathwise embeddings and back-and-forth equivalences

How do we capture back-and-forth equivalences, and hence the whole logic rather than just the existential positive part?

Open pathwise embeddings and back-and-forth equivalences

How do we capture back-and-forth equivalences, and hence the whole logic rather than just the existential positive part?

The key idea is to work in the arboreal category $\mathcal{R}_k^E(\sigma)$, where we have enough process structure to make game and bisimulation notions meaningful.

Open pathwise embeddings and back-and-forth equivalences

How do we capture back-and-forth equivalences, and hence the whole logic rather than just the existential positive part?

The key idea is to work in the arboreal category $\mathcal{R}_k^E(\sigma)$, where we have enough process structure to make game and bisimulation notions meaningful.

The key notions are

- *paths*, *i.e.* objects of $\mathcal{R}_k^E(\sigma)$ in which the order is linear (so the forest is a single branch), and
- *path embeddings*, *i.e.* forest morphisms with paths as domains which are embeddings of relational structures.

Open pathwise embeddings and back-and-forth equivalences

How do we capture back-and-forth equivalences, and hence the whole logic rather than just the existential positive part?

The key idea is to work in the arboreal category $\mathcal{R}_k^E(\sigma)$, where we have enough process structure to make game and bisimulation notions meaningful.

The key notions are

- *paths*, *i.e.* objects of $\mathcal{R}_k^E(\sigma)$ in which the order is linear (so the forest is a single branch), and
- *path embeddings*, *i.e.* forest morphisms with paths as domains which are embeddings of relational structures.

These are special cases of notions which are axiomatised in the arboreal categories setting in great generality.

Pathwise embeddings and open maps

A morphism $f\colon X \to Y$ in $\mathcal{R}_k^E(\sigma)$ is a *pathwise embedding* if, for all path embeddings $m\colon P \rightarrowtail X$, the composite $f \circ m$ is a path embedding.

Pathwise embeddings and open maps

A morphism $f\colon X \to Y$ in $\mathcal{R}_k^E(\sigma)$ is a *pathwise embedding* if, for all path embeddings $m\colon P \rightarrowtail X$, the composite $f \circ m$ is a path embedding.

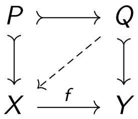
To capture the "back" or p-morphism condition, we introduce a notion of *open map* (Joyal-Nielsen-Winskel) that, combined with the concept of pathwise embedding, will allow us to define an appropriate notion of bisimulation.

Pathwise embeddings and open maps

A morphism $f : X \to Y$ in $\mathcal{R}^E_k(\sigma)$ is a *pathwise embedding* if, for all path embeddings $m : P \rightarrowtail X$, the composite $f \circ m$ is a path embedding.

To capture the "back" or p-morphism condition, we introduce a notion of *open map* (Joyal-Nielsen-Winskel) that, combined with the concept of pathwise embedding, will allow us to define an appropriate notion of bisimulation.

A morphism $f : X \to Y$ in $\mathcal{R}^E_k(\sigma)$ is said to be *open* if it satisfies the following path-lifting property: Given any commutative square



with P, Q paths, there exists a diagonal filler $Q \to X$ (*i.e.* an arrow $Q \to X$ making the two triangles commute).

Bisimulations

A *bisimulation* between objects X, Y of $\mathcal{R}_k^E(\sigma)$ is a span of open pathwise embeddings

$$\begin{array}{ccc} & R & \\ \swarrow & & \searrow \\ X & & Y \end{array}$$

If such a bisimulation exists, we say that X and Y are *bisimilar*.

Theorem

$G_k\mathcal{A}$ and $G_k\mathcal{B}$ are bisimilar in $\mathcal{R}_k^E(\sigma)$ iff Duplicator has a winning strategy in the k-round Ehrenfeucht-Fraïssé game between $\mathcal{A}$ and $\mathcal{B}$.

Bisimulations

A *bisimulation* between objects X, Y of $\mathcal{R}_k^E(\sigma)$ is a span of open pathwise embeddings

$$X \longleftarrow R \longrightarrow Y$$

If such a bisimulation exists, we say that X and Y are *bisimilar*.

Theorem

$G_k\mathcal{A}$ and $G_k\mathcal{B}$ are bisimilar in $\mathcal{R}_k^E(\sigma)$ iff Duplicator has a winning strategy in the k-round Ehrenfeucht-Fraïssé game between $\mathcal{A}$ and $\mathcal{B}$.

Note that we use the *resource category* $\mathcal{R}_k^E(\sigma)$ to study logical properties of objects of the *extensional category* $\mathcal{R}(\sigma)$.

Connection to logic

Fragments of first-order logic:

- $\mathcal{L}_k$ is the fragment of quantifier-rank $\leq k$.
- $\exists\mathcal{L}_k$ is the existential positive fragment of $\mathcal{L}_k$
- $\mathcal{L}_k(\#)$ is the extension of $\mathcal{L}_k$ with counting quantifiers $\exists_{\leq n}$, $\exists_{\geq n}$

Connection to logic

Fragments of first-order logic:

- $\mathcal{L}_k$ is the fragment of quantifier-rank $\leq k$.
- $\exists\mathcal{L}_k$ is the existential positive fragment of $\mathcal{L}_k$
- $\mathcal{L}_k(\#)$ is the extension of $\mathcal{L}_k$ with counting quantifiers $\exists_{\leq n}$, $\exists_{\geq n}$

We define three resource-indexed equivalences on structures $\mathcal{A}$, $\mathcal{B}$ of $\mathcal{R}(\sigma)$ using the resource categories $\mathcal{R}^E_k(\sigma)$:

- $\mathcal{A} \rightleftarrows_k \mathcal{B}$ iff there are morphisms $G_k\mathcal{A} \to G_k\mathcal{B}$ and $G_k\mathcal{B} \to G_k\mathcal{A}$. Note that there need be no relationship between these morphisms.
- $\mathcal{A} \leftrightarrow_k \mathcal{B}$ iff $G_k\mathcal{A}$ and $G_k\mathcal{B}$ are bisimilar in $\mathcal{R}^E_k(\sigma)$.
- $\mathcal{A} \cong_k \mathcal{B}$ iff $G_k\mathcal{A}$ and $G_k\mathcal{B}$ are isomorphic in $\mathcal{R}^E_k(\sigma)$.

Connection to logic

Fragments of first-order logic:

- $\mathcal{L}_k$ is the fragment of quantifier-rank $\leq k$.
- $\exists\mathcal{L}_k$ is the existential positive fragment of $\mathcal{L}_k$
- $\mathcal{L}_k(\#)$ is the extension of $\mathcal{L}_k$ with counting quantifiers $\exists_{\leq n}$, $\exists_{\geq n}$

We define three resource-indexed equivalences on structures $\mathcal{A}$, $\mathcal{B}$ of $\mathcal{R}(\sigma)$ using the resource categories $\mathcal{R}_k^E(\sigma)$:

- $\mathcal{A} \rightleftarrows_k \mathcal{B}$ iff there are morphisms $G_k\mathcal{A} \to G_k\mathcal{B}$ and $G_k\mathcal{B} \to G_k\mathcal{A}$.
  Note that there need be no relationship between these morphisms.
- $\mathcal{A} \leftrightarrow_k \mathcal{B}$ iff $G_k\mathcal{A}$ and $G_k\mathcal{B}$ are bisimilar in $\mathcal{R}_k^E(\sigma)$.
- $\mathcal{A} \cong_k \mathcal{B}$ iff $G_k\mathcal{A}$ and $G_k\mathcal{B}$ are isomorphic in $\mathcal{R}_k^E(\sigma)$.

Theorem

For structures $\mathcal{A}$ and $\mathcal{B}$:

$$
\begin{array}{llll}
(1) & \mathcal{A} \equiv^{\exists\mathcal{L}_k} \mathcal{B} & \Longleftrightarrow & \mathcal{A} \rightleftarrows_k \mathcal{B}. \\
(2) & \mathcal{A} \equiv^{\mathcal{L}_k} \mathcal{B} & \Longleftrightarrow & \mathcal{A} \leftrightarrow_k \mathcal{B}. \\
(3) & \mathcal{A} \equiv^{\mathcal{L}_k(\#)} \mathcal{B} & \Longleftrightarrow & \mathcal{A} \cong_k \mathcal{B}.
\end{array}
$$

Coalgebra number and tree-depth

A coalgebra for a comonad (G, ε, δ) is a morphism $\alpha : A \to GA$ such that the following diagrams commute:

$$
\begin{array}{ccc}
A & \xrightarrow{\alpha} & GA \\
 & \searrow^{\mathrm{id}_A} & \downarrow{\scriptstyle \varepsilon_A} \\
 & & A
\end{array}
\qquad\qquad
\begin{array}{ccc}
A & \xrightarrow{\alpha} & GA \\
{\scriptstyle \alpha}\downarrow & & \downarrow{\scriptstyle \delta_A} \\
GA & \xrightarrow{G\alpha} & G^2 A
\end{array}
$$

Coalgebra number and tree-depth

A coalgebra for a comonad (G, ε, δ) is a morphism $\alpha : A \to GA$ such that the following diagrams commute:

$$
\begin{array}{ccc}
A & \xrightarrow{\alpha} & GA \\
& \searrow^{\mathrm{id}_A} & \downarrow \varepsilon_A \\
& & A
\end{array}
\qquad\qquad
\begin{array}{ccc}
A & \xrightarrow{\alpha} & GA \\
\alpha \downarrow & & \downarrow \delta_A \\
GA & \xrightarrow{G\alpha} & G^2 A
\end{array}
$$

We should only expect a coalgebra structure $\alpha : \mathcal{A} \to \mathbb{E}_k \mathcal{A}$ to exist when the k-local information on $\mathcal{A}$ is sufficient to determine the structure of $\mathcal{A}$.

Coalgebra number and tree-depth

A coalgebra for a comonad (G, ε, δ) is a morphism $\alpha : A \to GA$ such that the following diagrams commute:

$$\begin{array}{ccc} A & \xrightarrow{\alpha} & GA \\ & \searrow^{\mathrm{id}_A} & \downarrow \varepsilon_A \\ & & A \end{array} \qquad\qquad \begin{array}{ccc} A & \xrightarrow{\alpha} & GA \\ \alpha \downarrow & & \downarrow \delta_A \\ GA & \xrightarrow{G\alpha} & G^2 A \end{array}$$

We should only expect a coalgebra structure $\alpha : \mathcal{A} \to \mathbb{E}_k \mathcal{A}$ to exist when the k-local information on $\mathcal{A}$ is sufficient to determine the structure of $\mathcal{A}$.

Our use of indexed comonads $\mathbb{C}_k$ opens up a new kind of question for coalgebras. Given a structure $\mathcal{A}$, we can ask: what is the least value of k such that a $\mathbb{C}_k$-coalgebra exists on $\mathcal{A}$? We call this the *coalgebra number* of $\mathcal{A}$.

Coalgebra number and tree-depth

A coalgebra for a comonad (G, ε, δ) is a morphism $\alpha : A \to GA$ such that the following diagrams commute:

$$
\begin{array}{ccc}
A & \xrightarrow{\alpha} & GA \\
& \searrow^{\mathrm{id}_A} & \downarrow{\scriptstyle \varepsilon_A} \\
& & A
\end{array}
\qquad\qquad
\begin{array}{ccc}
A & \xrightarrow{\alpha} & GA \\
{\scriptstyle \alpha}\downarrow & & \downarrow{\scriptstyle \delta_A} \\
GA & \xrightarrow{G\alpha} & G^2 A
\end{array}
$$

We should only expect a coalgebra structure $\alpha : \mathcal{A} \to \mathbb{E}_k\mathcal{A}$ to exist when the k-local information on $\mathcal{A}$ is sufficient to determine the structure of $\mathcal{A}$.

Our use of indexed comonads $\mathbb{C}_k$ opens up a new kind of question for coalgebras. Given a structure $\mathcal{A}$, we can ask: what is the least value of k such that a $\mathbb{C}_k$-coalgebra exists on $\mathcal{A}$? We call this the *coalgebra number* of $\mathcal{A}$.

Theorem

For the Ehrenfeucht-Fraïssé comonad $\mathbb{E}_k$, the coalgebra number of $\mathcal{A}$ corresponds precisely to the tree-depth of $\mathcal{A}$.

Coalgebra number and tree-depth

A coalgebra for a comonad (G, ε, δ) is a morphism $\alpha : A \to GA$ such that the following diagrams commute:

$$
\begin{array}{ccc}
A & \xrightarrow{\alpha} & GA \\
 & \searrow^{\mathrm{id}_A} & \downarrow \varepsilon_A \\
 & & A
\end{array}
\qquad\qquad
\begin{array}{ccc}
A & \xrightarrow{\alpha} & GA \\
\alpha \downarrow & & \downarrow \delta_A \\
GA & \xrightarrow{G\alpha} & G^2 A
\end{array}
$$

We should only expect a coalgebra structure $\alpha : \mathcal{A} \to \mathbb{E}_k \mathcal{A}$ to exist when the k-local information on $\mathcal{A}$ is sufficient to determine the structure of $\mathcal{A}$.

Our use of indexed comonads $\mathbb{C}_k$ opens up a new kind of question for coalgebras. Given a structure $\mathcal{A}$, we can ask: what is the least value of k such that a $\mathbb{C}_k$-coalgebra exists on $\mathcal{A}$? We call this the *coalgebra number* of $\mathcal{A}$.

Theorem

For the Ehrenfeucht-Fraïssé comonad $\mathbb{E}_k$, the coalgebra number of $\mathcal{A}$ corresponds precisely to the tree-depth of $\mathcal{A}$.

This follows from the comonadicity of the adjunction.

Pebble Games

Pebble Games

Similar but subtly different to EF-games

Pebble Games

Similar but subtly different to EF-games

Spoiler moves by placing one from a fixed set of pebbles on an element of $\mathcal{A}$ or $\mathcal{B}$; Duplicator responds by placing their matching pebble on an element of the other structure.

Pebble Games

Similar but subtly different to EF-games

Spoiler moves by placing one from a fixed set of pebbles on an element of $\mathcal{A}$ or $\mathcal{B}$; Duplicator responds by placing their matching pebble on an element of the other structure.

Duplicator wins if after each round, the relation defined by the current positions of the pebbles is a partial isomorphism

Pebble Games

Similar but subtly different to EF-games

Spoiler moves by placing one from a fixed set of pebbles on an element of $\mathcal{A}$ or $\mathcal{B}$; Duplicator responds by placing their matching pebble on an element of the other structure.

Duplicator wins if after each round, the relation defined by the current positions of the pebbles is a partial isomorphism

Thus there is a "sliding window" on the structures, of fixed size. It is this size which bounds the resource, not the length of the play.

Pebble Games

Similar but subtly different to EF-games

Spoiler moves by placing one from a fixed set of pebbles on an element of $\mathcal{A}$ or $\mathcal{B}$; Duplicator responds by placing their matching pebble on an element of the other structure.

Duplicator wins if after each round, the relation defined by the current positions of the pebbles is a partial isomorphism

Thus there is a "sliding window" on the structures, of fixed size. It is this size which bounds the resource, not the length of the play.

Whereas the k-round EF game corresponds to bounding the quantifier rank, k-pebble games correspond to bounding the number of variables which can be used in a formula.

Pebble Games

Similar but subtly different to EF-games

Spoiler moves by placing one from a fixed set of pebbles on an element of $\mathcal{A}$ or $\mathcal{B}$; Duplicator responds by placing their matching pebble on an element of the other structure.

Duplicator wins if after each round, the relation defined by the current positions of the pebbles is a partial isomorphism

Thus there is a "sliding window" on the structures, of fixed size. It is this size which bounds the resource, not the length of the play.

Whereas the k-round EF game corresponds to bounding the quantifier rank, k-pebble games correspond to bounding the number of variables which can be used in a formula.

Just as for EF-games, there is an existential-positive version, in which Spoiler only plays in $\mathcal{A}$, and Duplicator responds in $\mathcal{B}$.

The pebbling adjunction

We define a *k-pebble forest-ordered σ-structure* $(\mathcal{A}, \leq, p)$ to be a σ-structure $\mathcal{A}$ with a forest order $\leq$ on A, and a pebbling function $p : A \to \{1, \ldots, k\}$.

The pebbling adjunction

We define a *k-pebble forest-ordered σ-structure* $(\mathcal{A}, \leq, p)$ to be a σ-structure $\mathcal{A}$ with a forest order $\leq$ on A, and a pebbling function $p : A \to \{1, \ldots, k\}$.

In addition to condition (E), it must also satisfy the following condition:

(P) if a is adjacent to b in the Gaifman graph of A, and $a < b$ in the forest order, then for all $x \in (a, b]$, $p(a) \neq p(x)$.

The pebbling adjunction

We define a *k-pebble forest-ordered σ-structure* $(\mathcal{A}, \leq, p)$ to be a σ-structure $\mathcal{A}$ with a forest order $\leq$ on A, and a pebbling function $p : A \to \{1, \ldots, k\}$.

In addition to condition (E), it must also satisfy the following condition:

(P) if a is adjacent to b in the Gaifman graph of A, and $a < b$ in the forest order, then for all $x \in (a, b]$, $p(a) \neq p(x)$.

Morphisms of these structures are morphisms of forest-ordered structures which additionally preserve the pebbling function.

The pebbling adjunction

We define a *k-pebble forest-ordered σ-structure* $(\mathcal{A}, \leq, p)$ to be a σ-structure $\mathcal{A}$ with a forest order $\leq$ on A, and a pebbling function $p : A \to \{1, \ldots, k\}$.

In addition to condition (E), it must also satisfy the following condition:

(P) if a is adjacent to b in the Gaifman graph of A, and $a < b$ in the forest order, then for all $x \in (a, b]$, $p(a) \neq p(x)$.

Morphisms of these structures are morphisms of forest-ordered structures which additionally preserve the pebbling function.

These define a category $\mathcal{R}^P_k(\sigma)$ (where k bounds the number of pebbles, rather than the height of the forest order), and there is an evident forgetful functor $V_k : \mathcal{R}^P_k(\sigma) \to \mathcal{R}(\sigma)$.

The pebbling adjunction

We define a *k-pebble forest-ordered σ-structure* $(\mathcal{A}, \leq, p)$ to be a σ-structure $\mathcal{A}$ with a forest order $\leq$ on A, and a pebbling function $p : A \to \{1, \ldots, k\}$.

In addition to condition (E), it must also satisfy the following condition:

(P) if a is adjacent to b in the Gaifman graph of A, and $a < b$ in the forest order, then for all $x \in (a, b]$, $p(a) \neq p(x)$.

Morphisms of these structures are morphisms of forest-ordered structures which additionally preserve the pebbling function.

These define a category $\mathcal{R}_k^P(\sigma)$ (where k bounds the number of pebbles, rather than the height of the forest order), and there is an evident forgetful functor $V_k : \mathcal{R}_k^P(\sigma) \to \mathcal{R}(\sigma)$.

Theorem

For each $k > 0$, the functor V_k has a right adjoint H_k.

The corresponding comonad is $\mathbb{P}_k$, the pebbling comonad.

The pebbling comonad

Given a structure $\mathcal{A}$, the universe of $\mathbb{P}_k\mathcal{A}$ is $(\mathsf{k} \times A)^+$, the set of finite non-empty sequences of moves (p, a). Note this will be infinite even if $\mathcal{A}$ is finite.

The pebbling comonad

Given a structure $\mathcal{A}$, the universe of $\mathbb{P}_k\mathcal{A}$ is $(\mathsf{k} \times A)^+$, the set of finite non-empty sequences of moves (p, a). Note this will be infinite even if $\mathcal{A}$ is finite.

The counit map $\varepsilon_{\mathcal{A}} : \mathbb{P}_k\mathcal{A} \to \mathcal{A}$ sends a sequence $[(p_1, a_1), \ldots, (p_n, a_n)]$ to a_n.

The pebbling comonad

Given a structure $\mathcal{A}$, the universe of $\mathbb{P}_k\mathcal{A}$ is $(\mathsf{k} \times A)^+$, the set of finite non-empty sequences of moves (p, a). Note this will be infinite even if $\mathcal{A}$ is finite.

The counit map $\varepsilon_{\mathcal{A}} : \mathbb{P}_k\mathcal{A} \to \mathcal{A}$ sends a sequence $[(p_1, a_1), \ldots, (p_n, a_n)]$ to a_n.

To lift the relations on $\mathcal{A}$ to $\mathbb{P}_k\mathcal{A}$ we have the following condition in addition to those for $\mathbb{E}_k$:

The pebbling comonad

Given a structure $\mathcal{A}$, the universe of $\mathbb{P}_k\mathcal{A}$ is $(\mathsf{k} \times A)^+$, the set of finite non-empty sequences of moves (p, a). Note this will be infinite even if $\mathcal{A}$ is finite.

The counit map $\varepsilon_{\mathcal{A}} : \mathbb{P}_k\mathcal{A} \to \mathcal{A}$ sends a sequence $[(p_1, a_1), \ldots, (p_n, a_n)]$ to a_n.

To lift the relations on $\mathcal{A}$ to $\mathbb{P}_k\mathcal{A}$ we have the following condition in addition to those for $\mathbb{E}_k$:

- If $s \sqsubseteq t$, then the pebble index of the last move in s does not appear in the suffix of s in t.

Same same . . .

We can now run exactly the same script as for the Ehrenfeucht-Fraïssé case:

- We can define paths, pathwise embeddings, open maps, bisimilarity in $\mathcal{R}_k^P(\sigma)$ in exactly the same fashion as we did for $\mathcal{R}_k^E(\sigma)$.
- Hence we can define bisimulations between object of the extensional category $\mathcal{R}(\sigma)$ using the resource category $\mathcal{R}_k^P(\sigma)$.
- We can define the equivalence relations $\mathcal{A} \rightleftarrows_k \mathcal{B}$, $\mathcal{A} \leftrightarrow_k \mathcal{B}$, $\mathcal{A} \cong_k \mathcal{B}$ with respect to $\mathcal{R}_k^P(\sigma)$.

Same same . . .

We can now run exactly the same script as for the Ehrenfeucht-Fraïssé case:

- We can define paths, pathwise embeddings, open maps, bisimilarity in $\mathcal{R}_k^P(\sigma)$ in exactly the same fashion as we did for $\mathcal{R}_k^E(\sigma)$.
- Hence we can define bisimulations between object of the extensional category $\mathcal{R}(\sigma)$ using the resource category $\mathcal{R}_k^P(\sigma)$.
- We can define the equivalence relations $\mathcal{A} \rightleftarrows_k \mathcal{B}$, $\mathcal{A} \leftrightarrow_k \mathcal{B}$, $\mathcal{A} \cong_k \mathcal{B}$ with respect to $\mathcal{R}_k^P(\sigma)$.

We now take $\mathcal{L}_k$ to be the k-variable fragment of first-order logic. $\exists\mathcal{L}_k$ is the existential-positive part of this fragment, $\mathcal{L}_k(\#)$ its extension with counting quantifiers.

Same same ...

We can now run exactly the same script as for the Ehrenfeucht-Fraïssé case:

- We can define paths, pathwise embeddings, open maps, bisimilarity in $\mathcal{R}_k^P(\sigma)$ in exactly the same fashion as we did for $\mathcal{R}_k^E(\sigma)$.
- Hence we can define bisimulations between object of the extensional category $\mathcal{R}(\sigma)$ using the resource category $\mathcal{R}_k^P(\sigma)$.
- We can define the equivalence relations $\mathcal{A} \rightleftarrows_k \mathcal{B}$, $\mathcal{A} \leftrightarrow_k \mathcal{B}$, $\mathcal{A} \cong_k \mathcal{B}$ with respect to $\mathcal{R}_k^P(\sigma)$.

We now take $\mathcal{L}_k$ to be the k-variable fragment of first-order logic. $\exists\mathcal{L}_k$ is the existential-positive part of this fragment, $\mathcal{L}_k(\#)$ its extension with counting quantifiers.

With this notation, we get verbatim the same result as before:

Theorem

For structures $\mathcal{A}$ and $\mathcal{B}$:

$$
\begin{array}{llcl}
(1) & \mathcal{A} \equiv^{\exists\mathcal{L}_k} \mathcal{B} & \Longleftrightarrow & \mathcal{A} \rightleftarrows_k \mathcal{B}. \\
(2) & \mathcal{A} \equiv^{\mathcal{L}_k} \mathcal{B} & \Longleftrightarrow & \mathcal{A} \leftrightarrow_k \mathcal{B}. \\
(3) & \mathcal{A} \equiv^{\mathcal{L}_k(\#)} \mathcal{B} & \Longleftrightarrow & \mathcal{A} \cong_k \mathcal{B}.
\end{array}
$$

Coalgebra number and tree-width

We can define the coalgebra number for the pebbling comonad exactly as done before for the Ehrenfeucht-Fraïssé comonad.
A slightly more subtle argument is needed to show:

Theorem

For the pebbling comonad $\mathbb{P}_k$, the coalgebra number of $\mathcal{A}$ corresponds precisely to the tree-width of $\mathcal{A}$.

The modal comonad

We can replay the whole script again for basic modal logic.

- In this case, the modal comonad $\mathbb{M}_k$ corresponds to k-level *unravelling* of a Kripke structure.
- Open pathwise embedding bisimulation recovers standard modal bisimulation.
- The logical equivalences are the modal versions of those previously considered:
  - full modal logic of depth $\leq k$,
  - the diamond-only positive fragment, and
  - *graded modal logic* for the counting case.
- The coalgebra number in this case recovers the *property* of being a synchronization tree of height $\leq k$.
- The fact that it is a property rather than a structure in this case follows from the fact that this comonad is idempotent, and hence corresponds to a coreflective subcategory.

Where we are

We now have a considerable number of examples of game comonads corresponding to various notions of model comparison game:

- pebbling comonad
- EF comonad
- modal comonad
- comonads for hybrid logic and other extensions of basic modal logic
- guarded quantifier comonads (atom, loose and clique guards)
- generalized quantifier comonads
- pathwidth comonad
- hypertree comonad

Where we are

We now have a considerable number of examples of game comonads corresponding to various notions of model comparison game:

- pebbling comonad
- EF comonad
- modal comonad
- comonads for hybrid logic and other extensions of basic modal logic
- guarded quantifier comonads (atom, loose and clique guards)
- generalized quantifier comonads
- pathwidth comonad
- hypertree comonad

In each case, we have tight connections with logical fragments, and with combinatorial invariants.

Where we are

We now have a considerable number of examples of game comonads corresponding to various notions of model comparison game:

- pebbling comonad
- EF comonad
- modal comonad
- comonads for hybrid logic and other extensions of basic modal logic
- guarded quantifier comonads (atom, loose and clique guards)
- generalized quantifier comonads
- pathwidth comonad
- hypertree comonad

In each case, we have tight connections with logical fragments, and with combinatorial invariants.

The connections involve both the coKleisli categories and the Eilenberg-Moore categories of the comonads.

Where we are

We now have a considerable number of examples of game comonads corresponding to various notions of model comparison game:

- pebbling comonad
- EF comonad
- modal comonad
- comonads for hybrid logic and other extensions of basic modal logic
- guarded quantifier comonads (atom, loose and clique guards)
- generalized quantifier comonads
- pathwidth comonad
- hypertree comonad

In each case, we have tight connections with logical fragments, and with combinatorial invariants.

The connections involve both the coKleisli categories and the Eilenberg-Moore categories of the comonads.

We get direct descriptions of the coalgebras in terms of *comonadic forgetful functors*. These are important both for formulating bisimulation, and for the connection with combinatorial invariants.

Summary table

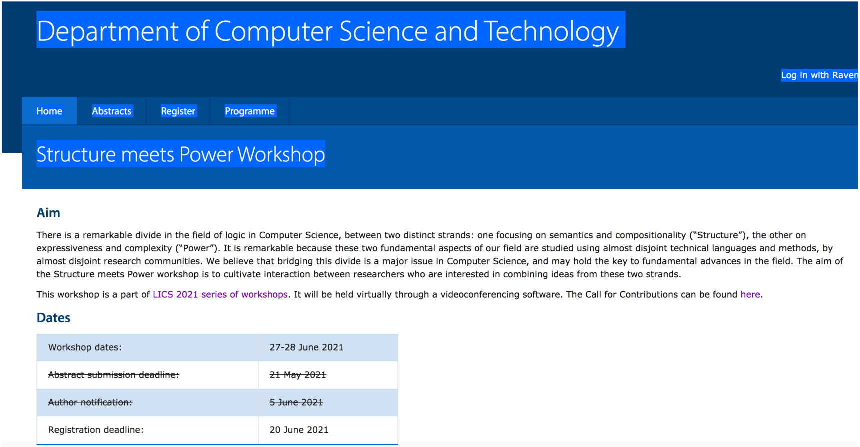
$\mathbb{C}_k$	**Logic**	$\kappa^{\mathbb{C}}$	$\rightarrow_k^{\mathbb{C}}$	$\leftrightarrow_k^{\mathbb{C}}$	$\cong_k^{\mathbb{C}}$
$\mathbb{E}_k$ [AS20]	**FOL** w/ qr $\leq k$	tree-depth	✓	✓	✓
$\mathbb{P}_k$ [ADW17]	k-variable logic	treewidth $+1$	✓	✓	✓
$\mathbb{M}_k$ [AS20]	**ML** w/ md $\leq k$	sync. tree-depth	✓	✓	✓
$\mathbb{G}_k^{\mathfrak{g}}$ [AM20]	$\mathfrak{g}$-guarded logic w/ width $\leq k$	guarded treewidth	✓	✓	?
$\mathbb{H}_{n,k}$ [CD20]	k-variable logic w/ $\mathbf{Q}_n$-quantifiers	n-ary general treewidth	✓	✓	✓
$\mathbb{PR}_k$	k-variable logic restricted-$\wedge$	pathwidth $+1$	✓	?	?
$\mathbb{LG}_k$	k-conjunct guarded logic	hypertree-width	✓	?	?

Further developments

- Arboreal categories [AR21]: axiomatic development
- General versions of model-theoretic results such as van Benthem-Rosen theorems, preservation theorems
- Lovasz-type theorems on counting homomorphisms [DJR21]
- Combinatorial parameters: concrete cases, axiomatic approach via density comonads

Advertisement

Structure meets Power workshop affiliated with LiCS 2021
`https://www.cst.cam.ac.uk/conference/structure-meets-power-2021/programme`



Department of Computer Science and Technology

Log in with Raven

Home | Abstracts | Register | Programme

Structure meets Power Workshop

Aim

There is a remarkable divide in the field of logic in Computer Science, between two distinct strands: one focusing on semantics and compositionality ("Structure"), the other on expressiveness and complexity ("Power"). It is remarkable because these two fundamental aspects of our field are studied using almost disjoint technical languages and methods, by almost disjoint research communities. We believe that bridging this divide is a major issue in Computer Science, and may hold the key to fundamental advances in the field. The aim of the Structure meets Power workshop is to cultivate interaction between researchers who are interested in combining ideas from these two strands.

This workshop is a part of LICS 2021 series of workshops. It will be held virtually through a videoconferencing software. The Call for Contributions can be found here.

Dates

Workshop dates:	27-28 June 2021
~~Abstract submission deadline:~~	~~21 May 2021~~
~~Author notification:~~	~~5 June 2021~~
Registration deadline:	20 June 2021

Samson Abramsky, Anuj Dawar, and Pengming Wang.
The pebbling comonad in finite model theory.
In *Logic in Computer Science (LICS), 2017 32nd Annual ACM/IEEE Symposium on*, pages 1–12. IEEE, 2017.

Samson Abramsky and Dan Marsden.
Comonadic semantics for guarded fragments, 2020.
To appear in LiCS 2021.

Samson Abramsky and Luca Reggio.
Arboreal categories: An axiomatic theory of resources.
arXiv preprint arXiv:2102.08109, 2021.
To appear in ICALP 2021.

Samson Abramsky and Nihil Shah.
Relating structure and power: Comonadic semantics for computational resources, 2020.
To appear in *Journal of Logic and Computation*.

Adam Ó Conghaile and Anuj Dawar.
Game comonads & generalised quantifiers, 2020.

Anuj Dawar, Tomáš Jakl, and Luca Reggio.
Lovász-type theorems and game comonads.

arXiv preprint arXiv:2105.03274, 2021.
To appear in LiCS 2021.