Computable Categoricity, and Topology in Reverse Mathematics

Doctoral Dissertation Defense

Java Darleen Villano

March 10, 2025

University of Connecticut

Outline

Part I: Computable categoricity relative to a degree

- Above $\mathbf{0}''$ and below $\mathbf{0}'$
- Beyond c.e. degrees
- Beyond directed graphs

Part II: Topology in reverse mathematics

- Coding topological spaces using $\mathbb{N}$
- The closure relation in the proof of Ginsburg-Sands
- Weak Ginsburg-Sands with closure

Part III: Sketch of construction for generic result

Part I: Computable categoricity relative to a degree

Definitions

Definition

A function $f : \mathbb{N} \to \mathbb{N}$ is a (partial) **computable** function if there exists an algorithm which computes the value of f on a given input.

Definition

A set $A \subseteq \mathbb{N}$ is **computable** if its characteristic function χ_A is a computable function.

Definition

A set A is **computably enumerable** (abbreviated as **c.e.**) if it is the range of a total computable function f.

Definition

A graph $\mathcal{G} = (G, E)$ is **computable** if its domain, G, is $\mathbb{N}$ and the edge relation E is a computable relation.

Recall that for a graph $\mathcal{G} = (G, E)$, the edge relation E is the set

$$\{(a, b) : a, b \in G \text{ and there is an edge connecting } a \text{ and } b\}.$$

Measuring the complexity of isomorphism types

Definition

A computable structure $\mathcal{A}$ is **computably categorical** if for every computable copy $\mathcal{B}$ of $\mathcal{A}$, there exists a computable isomorphism between $\mathcal{A}$ and $\mathcal{B}$.

For example, a linear order L is computably categorical if and only if it has finitely many adjacent pairs.

Measuring the complexity of isomorphism types

Definition

A computable structure $\mathcal{A}$ is **computably categorical** if for every computable copy $\mathcal{B}$ of $\mathcal{A}$, there exists a computable isomorphism between $\mathcal{A}$ and $\mathcal{B}$.

For example, a linear order L is computably categorical if and only if it has finitely many adjacent pairs.

Definition

A computable structure $\mathcal{A}$ is **relatively computably categorical** if for every copy (not necessarily computable) $\mathcal{B}$ of $\mathcal{A}$, there is a $\mathcal{B}$-computable isomorphism between $\mathcal{A}$ and $\mathcal{B}$.

These notions are not equivalent in general.

Relativizing categoricity

Definition

For a Turing degree $\mathbf{d}$, a computable structure $\mathcal{A}$ is **computably categorical relative to d** if for every $\mathbf{d}$-computable copy $\mathcal{B}$ of $\mathcal{A}$, there is a $\mathbf{d}$-computable isomorphism between $\mathcal{A}$ and $\mathcal{B}$.

Relativizing categoricity

Definition

For a Turing degree $\mathbf{d}$, a computable structure $\mathcal{A}$ is **computably categorical relative to d** if for every $\mathbf{d}$-computable copy $\mathcal{B}$ of $\mathcal{A}$, there is a $\mathbf{d}$-computable isomorphism between $\mathcal{A}$ and $\mathcal{B}$.

This is distinct from being $\mathbf{d}$-computably categorical.

Definition

A computable structure $\mathcal{A}$ is **d-computably categorical** if for all computable copies $\mathcal{B}$ of $\mathcal{A}$, there exists a $\mathbf{d}$-computable isomorphism between $\mathcal{A}$ and $\mathcal{B}$.

Relativizing categoricity

Definition

For a Turing degree $\mathbf{d}$, a computable structure $\mathcal{A}$ is **computably categorical relative to d** if for every **d**-computable copy $\mathcal{B}$ of $\mathcal{A}$, there is a $\mathbf{d}$-computable isomorphism between $\mathcal{A}$ and $\mathcal{B}$.

This is distinct from being $\mathbf{d}$-computably categorical.

Definition

A computable structure $\mathcal{A}$ is **d-computably categorical** if for all computable copies $\mathcal{B}$ of $\mathcal{A}$, there exists a $\mathbf{d}$-computable isomorphism between $\mathcal{A}$ and $\mathcal{B}$.

Definition

For a Turing degree $\mathbf{d}$, a computable structure $\mathcal{A}$ is **computably categorical relative to d** if for every $\mathbf{d}$-computable copy $\mathcal{B}$ of $\mathcal{A}$, there is a $\mathbf{d}$-computable isomorphism between $\mathcal{A}$ and $\mathcal{B}$.

It is also distinct from being relatively Δ^0_α-categorical.

Definition

A computable structure $\mathcal{A}$ is **relatively Δ^0_α-categorical** if for any copy $\mathcal{B}$ of $\mathcal{A}$, there is a $\Delta^0_\alpha(\mathcal{B})$-computable isomorphism between $\mathcal{A}$ and $\mathcal{B}$.

Definition

For a Turing degree **d**, a computable structure $\mathcal{A}$ is **computably categorical relative to d** if for every **d**-computable copy $\mathcal{B}$ of $\mathcal{A}$, there is a **d**-computable isomorphism between $\mathcal{A}$ and $\mathcal{B}$.

It is also distinct from being relatively Δ^0_α-categorical.

Definition

A computable structure $\mathcal{A}$ is **relatively Δ^0_α-categorical** if for any copy $\mathcal{B}$ of $\mathcal{A}$, there is a $\Delta^0_\alpha(\mathcal{B})$-computable isomorphism between $\mathcal{A}$ and $\mathcal{B}$.

Background

How does computable categoricity relative to a degree behave?

Background

How does computable categoricity relative to a degree behave?

Fact

A computable structure $\mathcal{A}$ is **relatively computably categorical** *if for all degrees* $\mathbf{d}$, *$\mathcal{A}$ is computably categorical relative to* $\mathbf{d}$.

Background

How does computable categoricity relative to a degree behave?

Fact

A computable structure $\mathcal{A}$ is **relatively computably categorical** *if for all degrees* $\mathbf{d}$, *$\mathcal{A}$ is computably categorical relative to* $\mathbf{d}$.

We begin with the following result.

Fact (Downey, Harrison-Trainor, Melnikov [DHTM21])

If $\mathcal{A}$ is a computable structure and it is computably categorical relative to some degree $\mathbf{d} \geq \mathbf{0}''$, then $\mathcal{A}$ is computably categorical relative to all $\mathbf{d} \geq \mathbf{0}''$.

Below $\mathbf{0}'$

In the c.e. degrees, being computably categorical relative to a degree is not monotonic.

Theorem (Downey, Harrison-Trainor, Melnikov [DHTM21])

There is a computable structure $\mathcal{A}$ and c.e. degrees $\mathbf{0} = \mathbf{d}_0 <_T \mathbf{e}_0 <_T \mathbf{d}_1 <_T \mathbf{e}_1 <_T \dots$ such that

(1) *$\mathcal{A}$ is computably categorical relative to $\mathbf{d}_i$ for each i,*

(2) *$\mathcal{A}$ is not computably categorical relative to $\mathbf{e}_i$ for each i,*

(3) *$\mathcal{A}$ is computably categorical relative to $\mathbf{0}'$.*

Below $\mathbf{0}'$

Theorem (V.)

Let $P = (P, \leq)$ be a computable partially ordered set and let $P = P_0 \sqcup P_1$ be a computable partition. Then, there exists a computable directed graph $\mathcal{G}$ and an embedding h of P into the c.e. degrees where

(1) *$\mathcal{G}$ is computably categorical;*

(2) *$\mathcal{G}$ is computably categorical relative to each degree in $h(P_0)$; and*

(3) *$\mathcal{G}$ is not computably categorical relative to each degree in $h(P_1)$.*

We can also consider the version where $\mathcal{G}$ is made to be not computably categorical.

Definition

A degree **d** is **low for isomorphism** if for every pair of computable structures $\mathcal{A}$ and $\mathcal{B}$, $\mathcal{A}$ and $\mathcal{B}$ are **d**-computably isomorphic if and only if they are computably isomorphic.

In the generic degrees

Definition

A degree **d** is **low for isomorphism** if for every pair of computable structures $\mathcal{A}$ and $\mathcal{B}$, $\mathcal{A}$ and $\mathcal{B}$ are **d**-computably isomorphic if and only if they are computably isomorphic.

Theorem (Franklin, Solomon [FS14])

Every 2-generic degree is low for isomorphism.

This means that there *cannot* be a computable structure $\mathcal{A}$ which is not computably categorical but changes its mind when we relativize to a 2-generic degree **d**.

In the generic degrees

Definition

A degree **d** is **low for isomorphism** if for every pair of computable structures $\mathcal{A}$ and $\mathcal{B}$, $\mathcal{A}$ and $\mathcal{B}$ are **d**-computably isomorphic if and only if they are computably isomorphic.

Theorem (Franklin, Solomon [FS14])

Every 2-generic degree is low for isomorphism.

This means that there *cannot* be a computable structure $\mathcal{A}$ which is not computably categorical but changes its mind when we relativize to a 2-generic degree **d**.

Theorem (V.)

There exists a (properly) 1-generic G such that there is a computable directed graph $\mathcal{A}$ where $\mathcal{A}$ is not computably categorical but is computably categorical relative to G.

Beyond directed graphs

Question

For structures other than directed graphs, can you produce an example which witnesses the chaotic behavior in the poset result?

Beyond directed graphs

Question

For structures other than directed graphs, can you produce an example which witnesses the chaotic behavior in the poset result?

There are some results in the literature that give a negative result for Boolean algebras already.

Beyond directed graphs

Question

For structures other than directed graphs, can you produce an example which witnesses the chaotic behavior in the poset result?

There are some results in the literature that give a negative result for Boolean algebras already.

Theorem ([Gon77], [Rem81])

A computable Boolean algebra is computably categorical if and only if it has finitely many atoms.

Theorem (Bazhenov [Baz14])

For every degree $\mathbf{d} < \mathbf{0}'$, *a computable Boolean algebra is* $\mathbf{d}$*-computably categorical if and only if it is computably categorical.*

Beyond directed graphs

Corollary (from results in [Hir+02] and [Mil+18])

For the following classes of structures, there exists a computable example in each class which witnesses the chaotic behavior in the poset result:

(1) *symmetric, irreflexive graphs; partial orderings; lattices; rings with zero-divisors; integral domains of arbitrary characteristic; commutative semigroups; and 2-step nilpotent groups (Theorem 1.22 of [Hir+02])*

(2) *countable fields (Theorem 1.8 of [Mil+18])*

Currently, the full picture is yet to be determined for some classes of structures, such as linear orderings.

Part II: Topology in reverse mathematics

Brief overview of reverse mathematics

Question

When proving mathematical theorems, which logical axioms are sufficient and necessary to prove them?

Brief overview of reverse mathematics

Question

When proving mathematical theorems, which logical axioms are sufficient and necessary to prove them?

We make this question precise by measuring the logical strength of theorems using subsystems of second-order arithmetic. Our base theory is RCA_0.

Definition

The formal system RCA_0 consists of the following axioms and axiom schema:

(1) PA^-, i.e., axioms which describe a discrete ordered semiring;
(2) the Δ^0_1 comprehension scheme; and
(3) $\mathsf{I}\Sigma^0_1$ (the induction axiom restricted to Σ^0_1 formulas),

RCA_0 and beyond RCA_0

The following mathematical theorems are provable in RCA_0:

(1) The Baire category theorem;

(2) the existence of an algebraic closure of a countable field; and

(3) the Intermediate Value Theorem.

RCA_0 **and beyond** RCA_0

The following mathematical theorems are provable in RCA_0:

(1) The Baire category theorem;

(2) the existence of an algebraic closure of a countable field; and

(3) the Intermediate Value Theorem.

There are mathematical theorems which require more machinery, and so we can measure their logical strength using stronger subsystems such as ACA_0.

Definition

The formal system ACA_0 consists of RCA_0 and the comprehension scheme for all arithmetical formulas.

Other subsystems

Many well-known mathematical theorems are equivalent, over RCA_0, to one of the following subsystems, listed in order of increasing strength:

$$\mathsf{WKL}_0 \Leftarrow \mathsf{ACA}_0 \Leftarrow \mathsf{ATR}_0 \Leftarrow \Pi^1_1\text{-}\mathsf{CA}_0.$$

Other subsystems

Many well-known mathematical theorems are equivalent, over RCA_0, to one of the following subsystems, listed in order of increasing strength:

$$\mathsf{WKL}_0 \Leftarrow \mathsf{ACA}_0 \Leftarrow \mathsf{ATR}_0 \Leftarrow \Pi^1_1\text{-}\mathsf{CA}_0.$$

Other subsystems have arisen under the level of ACA_0 via combinatorial principles; one such principle is RT^2_2.

Ramsey's theorem for 2-colorings of pairs

Definition

Let $[\mathbb{N}]^n$ denote the collection of n-element subsets of $\mathbb{N}$. A k-**coloring** of $[\mathbb{N}]^n$ is a map $c : [\mathbb{N}]^n \to k$. A set $H \subseteq \mathbb{N}$ is **homogeneous** for c if there is an $i < k$ where $c(s) = i$ for all $s \in [H]^n$.

Definition

RT^2_2 is the statement that every 2-coloring $c : [\mathbb{N}]^2 \to 2$ admits an infinite homogeneous set H.

Consequences of RT^2_2

The following consequences of RT^2_2 are important to the results in my thesis.

Definition (Chain/antichain principle)

CAC is the statement that every infinite partial order $(P, \leq_P)$ has an infinite chain or antichain.

Definition (Ascending/descending sequence principle)

ADS is the statement that every infinite linear order has an infinite ascending sequence or an infinite descending sequence.

The Ginsburg-Sands theorem

Theorem (Ginsburg, Sands [GS79])

Every infinite topological space contains one of the following five spaces, with $\mathbb{N}$ as the underlying set, as a subspace:

(i) *discrete: all subsets of $\mathbb{N}$ are open;*

(ii) *indiscrete: the only open sets are $\mathbb{N}$ and $\emptyset$;*

(iii) *cofinite: the open sets are $\mathbb{N}$, $\emptyset$, and all subsets of $\mathbb{N}$ with finite complement;*

(iv) *initial segment: the open sets are $\mathbb{N}$, $\emptyset$, and all sets of the form $[0, n] = \{k \in \mathbb{N} : k \leq n\}$;*

(v) *final segment: the open sets are $\mathbb{N}$, $\emptyset$, and all sets of the form $[n, \infty) = \{k \in \mathbb{N} : n \leq k\}$.*

Countable second-countable spaces

Definition

X is said to be **second-countable** (or is a **second-countable space**) if there is a countable collection $\mathcal{U} = \{U_i\}_{i \in \mathbb{N}}$ of open subsets of X that form a basis for the topology on X.

Countable second-countable spaces

Definition

X is said to be **second-countable** (or is a **second-countable space**) if there is a countable collection $\mathcal{U} = \{U_i\}_{i \in \mathbb{N}}$ of open subsets of X that form a basis for the topology on X.

Definition (Dorais [Dor11])

A **countable second-countable (CSC)** space is a tuple $\langle X, \mathcal{U}, k \rangle$ as follows:

(1) X is a subset of $\mathbb{N}$;

(2) $\mathcal{U} = \langle U_n : n \in \mathbb{N} \rangle$ is a family of subsets of X such that every $x \in X$ belongs to U_n for some $n \in \mathbb{N}$;

(3) $k : X \times \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ is a function such that for every $x \in X$ and all $m, n \in \mathbb{N}$, if $x \in U_m \cap U_n$ then $x \in U_{k(x,m,n)} \subseteq U_n \cap U_m$.

Building CSC spaces

Proposition ([Dor11])

The following is provable in RCA_0. *Given a set* $X \subseteq \mathbb{N}$ *and a collection* $\langle V_n : n \in \mathbb{N} \rangle$ *of subsets of* X, *there exists a CSC space* $\langle X, \mathcal{U}, k \rangle$ *with* $\mathcal{U} = \{U_n : n \in \mathbb{N}\}$ *as follows:*

(1) *for every* $n \in \mathbb{N}$, $V_n \in \mathcal{U}$;

(2) *for every* $m \in \mathbb{N}$, $U_m = \bigcap_{n \in F} V_n$, *where* F *is the finite set coded by* m.

We say that a CSC built up by specifying a sequence $\langle V_n : n \in \mathbb{N} \rangle$ is **generated** by that sequence.

The closure relation

The first part of the proof of Ginsburg-Sands involves defining the following equivalence relation on an infinite topological space X:

$$x \sim y \iff \text{cl}\{x\} = \text{cl}\{y\}.$$

The closure relation

The first part of the proof of Ginsburg-Sands involves defining the following equivalence relation on an infinite topological space X:

$$x \sim y \iff \mathrm{cl}\{x\} = \mathrm{cl}\{y\}.$$

If each equivalence class is finite, then there must be infinitely many such classes, and so we can form an infinite T_0 subspace. On this subspace, we define the following partial order:

$$x \leq y \iff x \in \mathrm{cl}\{y\}.$$

The closure relation

The first part of the proof of Ginsburg-Sands involves defining the following equivalence relation on an infinite topological space X:

$$x \sim y \iff \mathrm{cl}\{x\} = \mathrm{cl}\{y\}.$$

If each equivalence class is finite, then there must be infinitely many such classes, and so we can form an infinite T_0 subspace. On this subspace, we define the following partial order:

$$x \leq y \iff x \in \mathrm{cl}\{y\}.$$

Question

For a CSC space X, how much logical strength do we need to define the closure of a point?

The closure relation

Definition

The **closure relation** cl_X on a topological space X is the binary relation defined by

$$(y, x) \in \mathrm{cl}_X \iff y \in \mathrm{cl}\{x\}.$$

The closure relation

Definition

The **closure relation** cl_X on a topological space X is the binary relation defined by

$$(y, x) \in \mathrm{cl}_X \iff y \in \mathrm{cl}\{x\}.$$

In RCA_0, we formalize cl_X for a CSC space X in the following way.

Definition (RCA_0)

The **closure relation** cl_X on a CSC space X is the binary relation defined by

$$(y, x) \in \mathrm{cl}_X \iff (\forall n)(y \in U_n \to x \in U_n).$$

Results regarding the closure relation

Theorem (RCA_0)

The following are equivalent:

(1) ACA_0.

(2) *For a CSC space* $\langle X, \mathcal{U}, k\rangle$, *the closure relation* cl_X *exists.*

Theorem (RCA_0)

The following are equivalent:

(1) ACA_0.

(2) *For a CSC space* $\langle X, \mathcal{U}, k\rangle$, *the closure relation* cl_X *exists.*

The next question we can ask is for a CSC space with the closure relation given as part of its description, is CAC necessary to prove Ginsburg-Sands?

Weakening Ginsburg-Sands

Definition

$\mathrm{wGS^{cl}}$: Let $\langle X, \mathcal{U}, k \rangle$ be an infinite CSC space with a closure relation cl_X. Then, X has one of the following:

(i) an infinite T_1 subspace;

(ii) an infinite indiscrete subspace;

(iii) an infinite subspace homeomorphic to $\mathbb{N}$ with the initial segment topology;

(iv) an infinite subspace homeomorphic to $\mathbb{N}$ with the final segment topology.

This is a weakening of the original Ginsburg-Sands theorem since the infinite discrete subspace or infinite subspace with the cofinite topology cases are collapsed into (i).

Topological characterization of CAC

Theorem (RCA$_0$)

The following are equivalent:

(1) CAC.

(2) $\mathsf{wGS}^{\mathsf{cl}}$.

Topological characterization of CAC

Theorem (RCA_0)

The following are equivalent:

(1) CAC.

(2) $\mathsf{wGS}^{\mathrm{cl}}$.

This gives us a topological characterization of the combinatorial principle CAC.

Part III: Sketch of construction for generic result

Requirements

We have the following requirements:

- R_j : $(\exists\sigma \subseteq G)(\sigma \in W_j \vee (\forall\tau \supseteq \sigma)(\tau \notin W_j))$,
- P_e : $\Phi_e : \mathcal{A} \to \mathcal{B}$ is not an isomorphism, and
- S_i : if $\mathcal{A} \cong \mathcal{M}_i^G$, then there exists a G-computable isomorphism $f_i^G : \mathcal{A} \to \mathcal{M}_i^G$.

Building $\mathcal{A}$ in stages

We build the computable directed graph $\mathcal{A}$ in stages.

Building $\mathcal{A}$ in stages

We build the computable directed graph $\mathcal{A}$ in stages.

At stage $s = 0$, we set the domain of $\mathcal{A}$ to be empty.

Building $\mathcal{A}$ in stages

We build the computable directed graph $\mathcal{A}$ in stages.

At stage $s = 0$, we set the domain of $\mathcal{A}$ to be empty.

At stage $s > 0$, we add two new connected components by adding a_{2s} and a_{2s+1} as root nodes. We attach 2-loop to each node. Then, we attach a $(5s + 1)$-loop to a_{2s} and a $(5s + 2)$-loop to a_{2s+1}.

Building $\mathcal{A}$ in stages

We build the computable directed graph $\mathcal{A}$ in stages.

At stage $s = 0$, we set the domain of $\mathcal{A}$ to be empty.

At stage $s > 0$, we add two new connected components by adding a_{2s} and a_{2s+1} as root nodes. We attach 2-loop to each node. Then, we attach a $(5s+1)$-loop to a_{2s} and a $(5s+2)$-loop to a_{2s+1}.

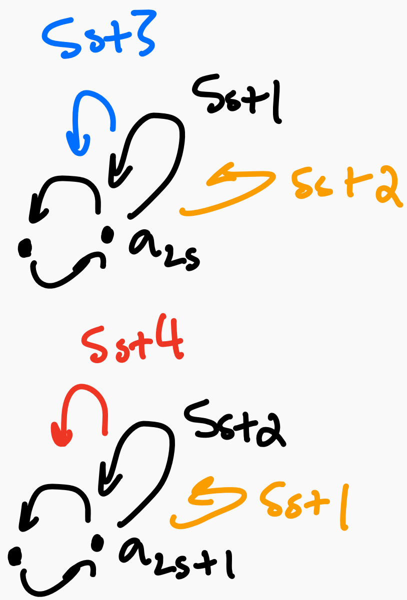
Definition

The root node a_{2s} in our graph $\mathcal{A}$ with its loops is the $2s$**th connected component** or just the $2s$th component of $\mathcal{A}$.

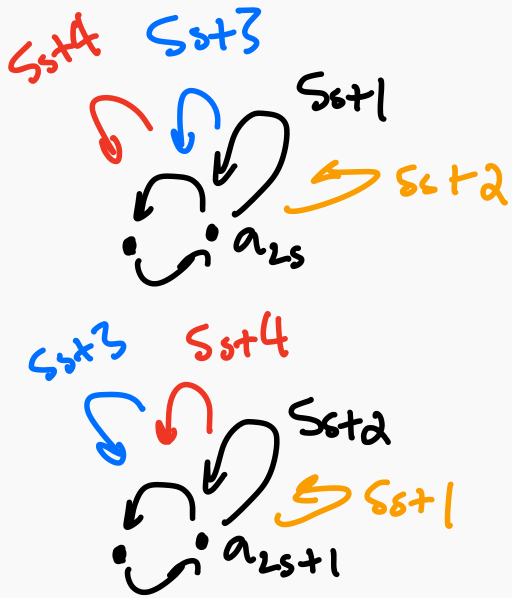
I


Ss+1
a2s
Ss+2
a2s+1

II


Ss+3
Ss+1
Ss+2
a2s
Ss+4
Ss+2
Ss+1
a2s+1

III


Ss+4
Ss+3
Ss+1
Ss+2
a2s
Ss+3
Ss+4
Ss+2
Ss+1
a2s+1

Basic strategies: P_e

This is our basic strategy to satisfy all P_e requirements.

Let s be the current stage of the construction and let α be a P_e-strategy.

1. If α is first eligible to act at stage s, it defines its witness n_α to be a large unused number. Let $n = n_\alpha$.

Basic strategies: P_e

This is our basic strategy to satisfy all P_e requirements.

Let s be the current stage of the construction and let α be a P_e-strategy.

1. If α is first eligible to act at stage s, it defines its witness n_α to be a large unused number. Let $n = n_\alpha$.
2. Check if Φ_e maps the $2n$th and $(2n+1)$st components of $\mathcal{A}$ to the $2n$th and $(2n+1)$st components of $\mathcal{B}$, respectively.

Basic strategies: P_e

This is our basic strategy to satisfy all P_e requirements.

Let s be the current stage of the construction and let α be a P_e-strategy.

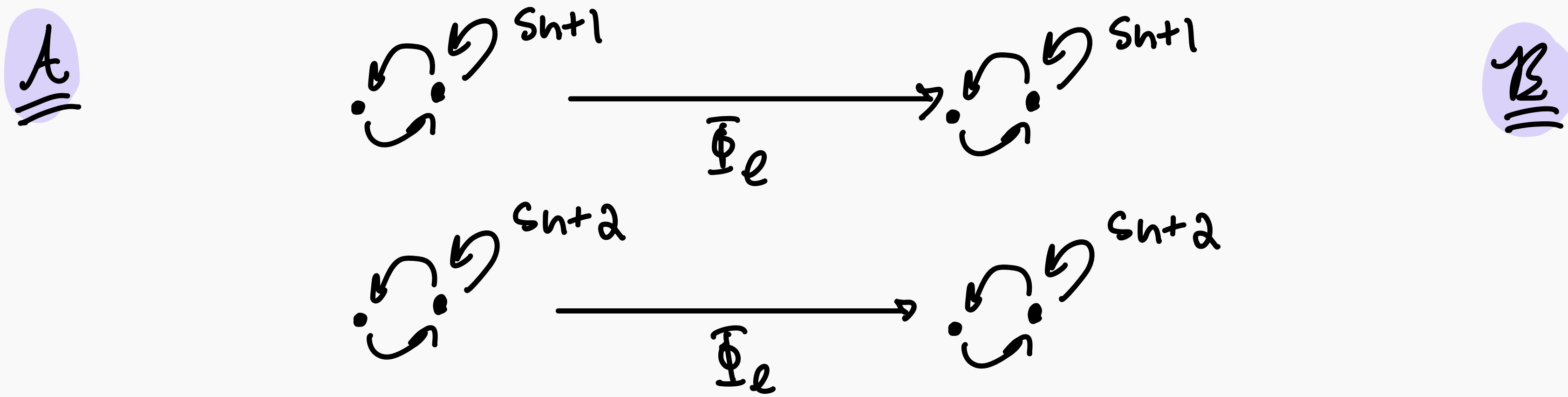
1. If α is first eligible to act at stage s, it defines its witness n_α to be a large unused number. Let $n = n_\alpha$.
2. Check if Φ_e maps the $2n$th and $(2n+1)$st components of $\mathcal{A}$ to the $2n$th and $(2n+1)$st components of $\mathcal{B}$, respectively. If not, α takes no action at stage s.

Basic strategies: P_e

This is our basic strategy to satisfy all P_e requirements.

Let s be the current stage of the construction and let α be a P_e-strategy.

1. If α is first eligible to act at stage s, it defines its witness n_α to be a large unused number. Let $n = n_\alpha$.
2. Check if Φ_e maps the $2n$th and $(2n+1)$st components of $\mathcal{A}$ to the $2n$th and $(2n+1)$st components of $\mathcal{B}$, respectively. If not, α takes no action at stage s. If so, α takes action by adding new loops to these components in $\mathcal{A}$ and $\mathcal{B}$.

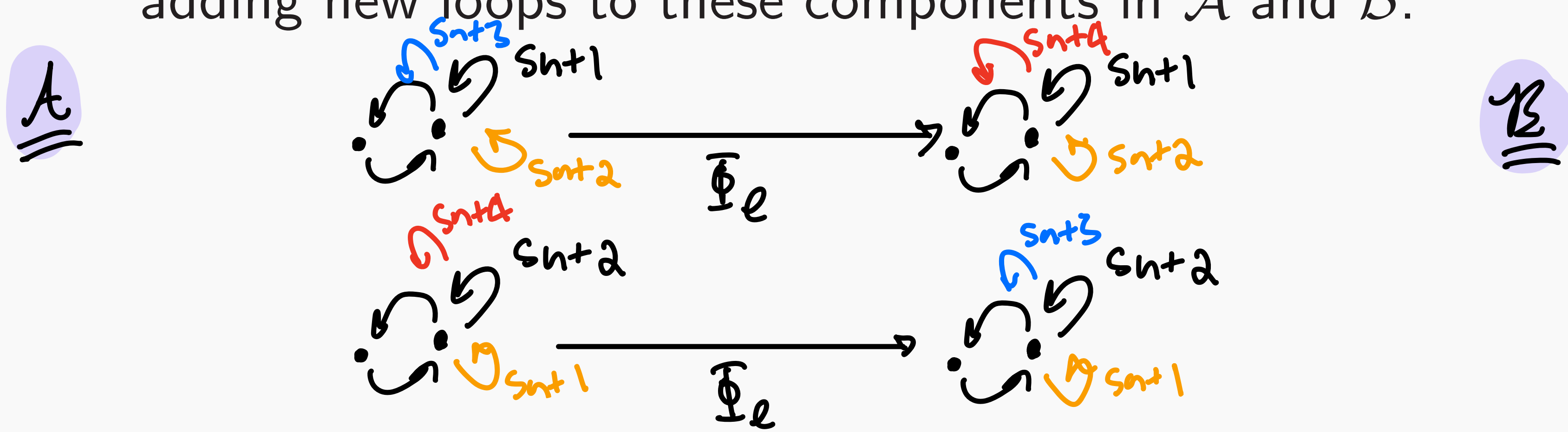


Basic strategies: P_e

This is our basic strategy to satisfy all P_e requirements.

Let s be the current stage of the construction and let α be a P_e-strategy.

1. If α is first eligible to act at stage s, it defines its witness n_α to be a large unused number. Let $n = n_\alpha$.
2. Check if Φ_e maps the $2n$th and $(2n+1)$st components of $\mathcal{A}$ to the $2n$th and $(2n+1)$st components of $\mathcal{B}$, respectively. If not, α takes no action at stage s. If so, α takes action by adding new loops to these components in $\mathcal{A}$ and $\mathcal{B}$.



Basic strategies: S_i

This is our basic strategy for the S_i requirements. Let α be an S_i-strategy.

Basic strategies: S_i

This is our basic strategy for the S_i requirements. Let α be an S_i-strategy.

For each n, we try to find copies of the $2n$th and $(2n+1)$st components of $\mathcal{A}$ in $\mathcal{M}_i^G$.

Basic strategies: S_i

This is our basic strategy for the S_i requirements. Let α be an S_i-strategy.

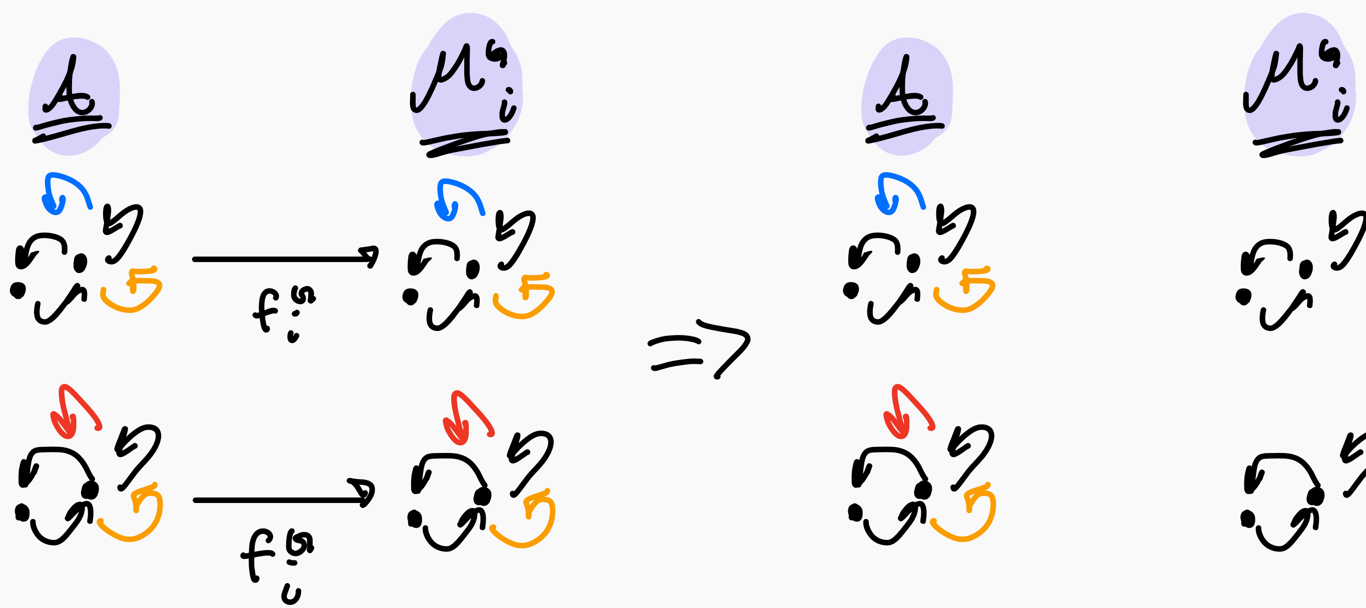
For each n, we try to find copies of the $2n$th and $(2n+1)$st components of $\mathcal{A}$ in $\mathcal{M}_i^G$. Initial segments of G can change throughout the construction, and so loops in $\mathcal{M}_i^G$ or embeddings using certain initial segments of G can disappear or reappear.

Basic strategies: S_i

This is our basic strategy for the S_i requirements. Let α be an S_i-strategy.

For each n, we try to find copies of the $2n$th and $(2n+1)$st components of $\mathcal{A}$ in $\mathcal{M}_i^G$. Initial segments of G can change throughout the construction, and so loops in $\mathcal{M}_i^G$ or embeddings using certain initial segments of G can disappear or reappear.

e.g.



Basic strategies: S_i

When α is next eligible to act at stage s, it can check if an initial segment of G has changed up to some previously defined use for an f_i^G-computation at that point in the construction.

Basic strategies: S_i

When α is next eligible to act at stage s, it can check if an initial segment of G has changed up to some previously defined use for an f_i^G-computation at that point in the construction.

This will determine what parameter, n_α, α will work with when trying to match $\mathcal{A}[s]$-components with their copies (if any) in $\mathcal{M}_i^G[s]$.

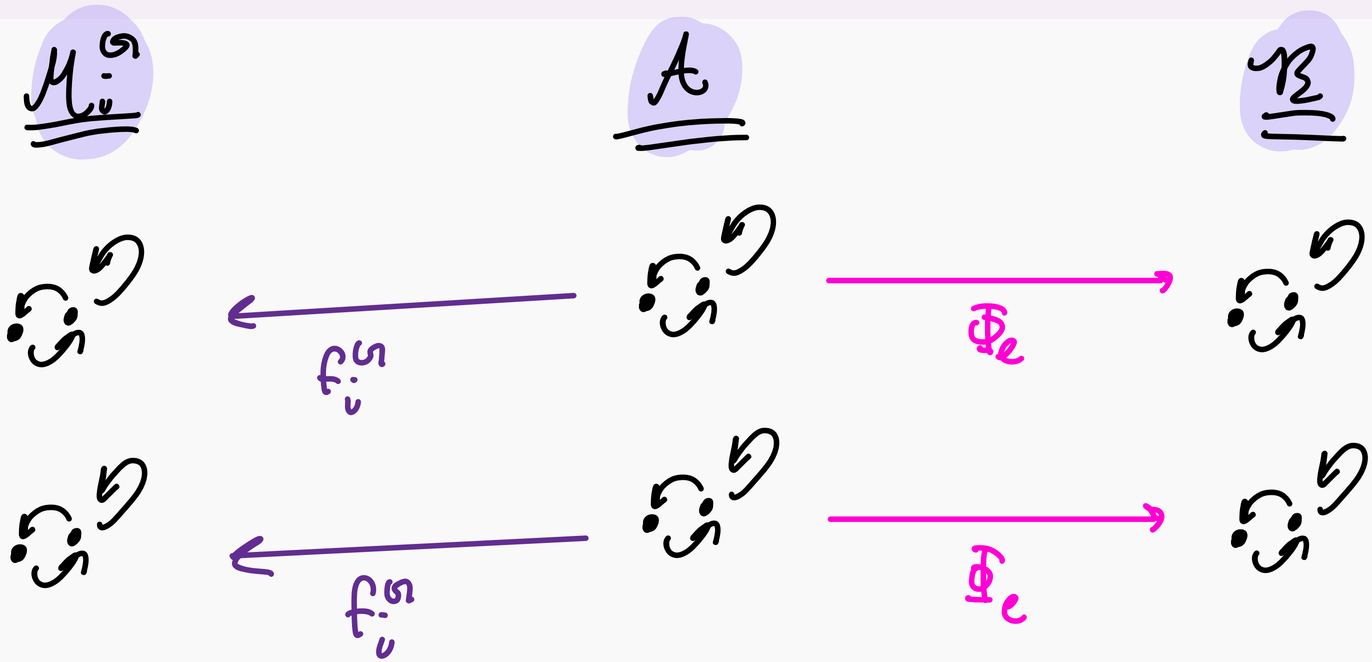
Interactions between strategies

There are several interactions and conflicts to keep note of in the construction.

There are several interactions and conflicts to keep note of in the construction.

Interaction 1

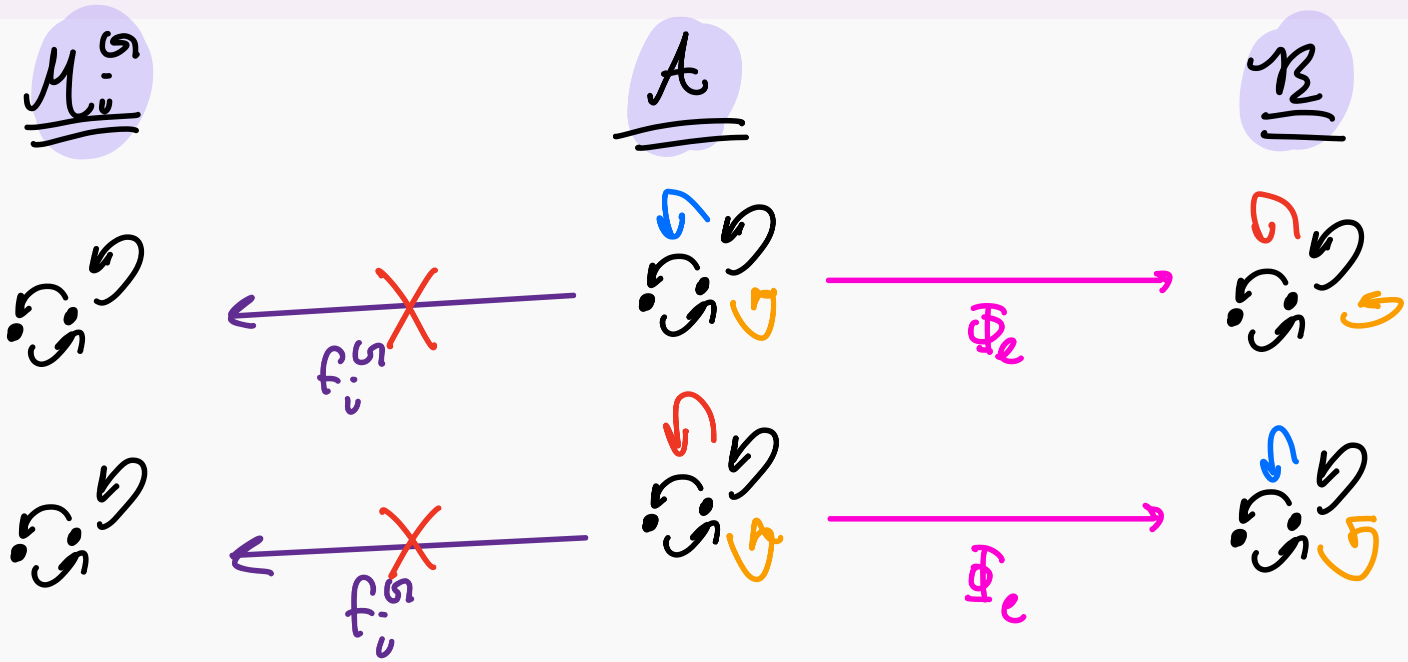
The P_e requirement wants to diagonalize while the S_i requirements want to build embeddings: this can primarily be resolved by having P_e "wait" for higher priority S_i requirements



There are several interactions and conflicts to keep note of in the construction.

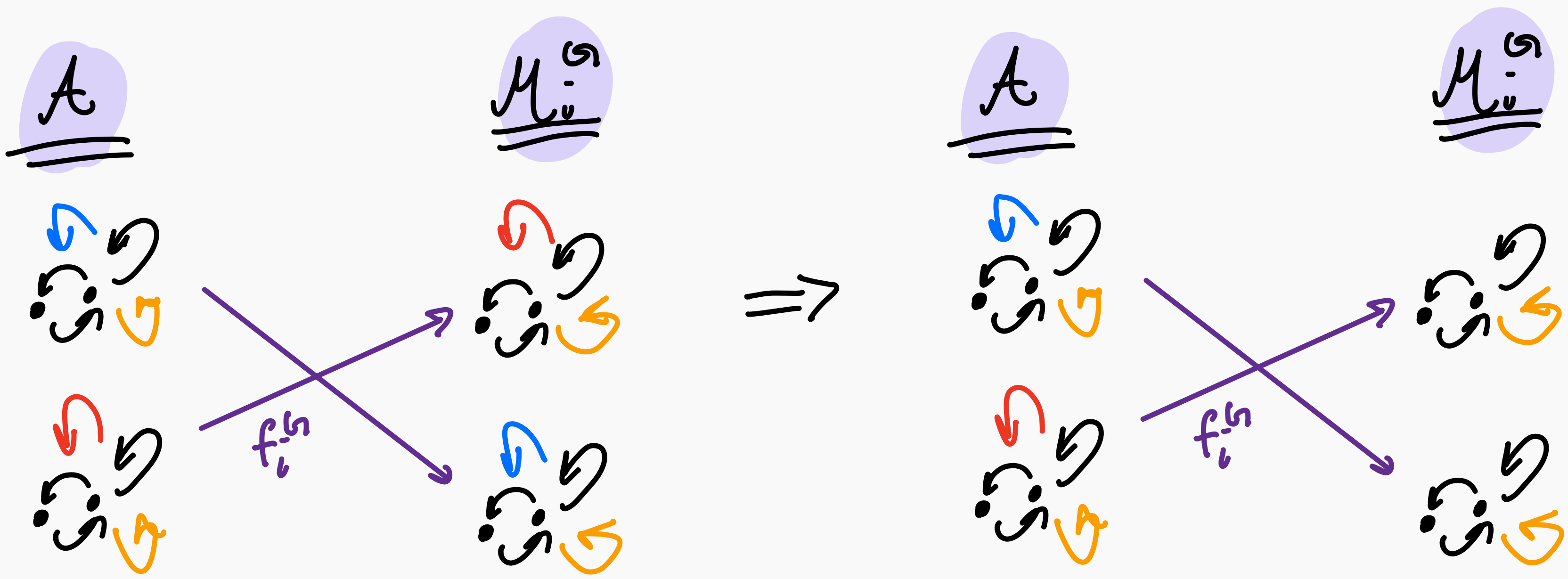
Interaction 1

The P_e requirement wants to diagonalize while the S_i requirements want to build embeddings: this can primarily be resolved by having P_e "wait" for higher priority S_i requirements



Interaction 2

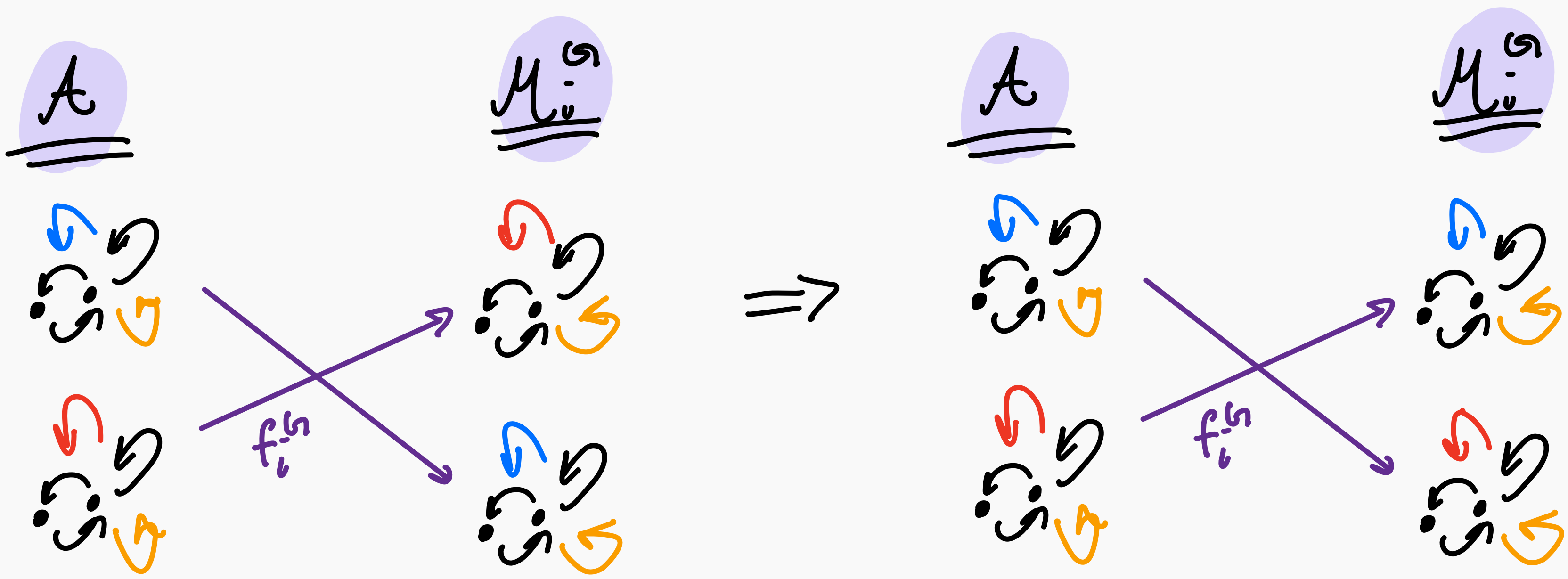
Changes in initial segments of G can make computations which disappeared reappear again: this can be resolved by making pairs of $\mathcal{A}$-components indistinct



Interactions between strategies

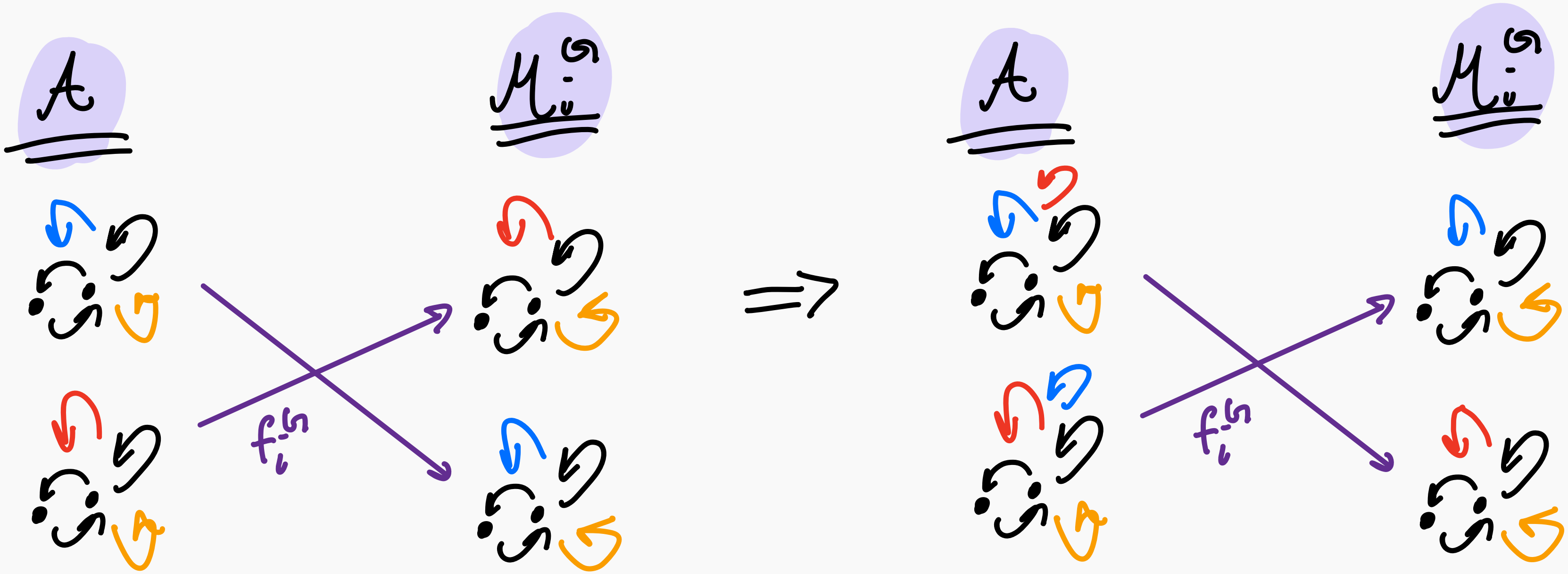
Interaction 2

Changes in initial segments of G can make computations which disappeared reappear again: this can be resolved by making pairs of $\mathcal{A}$-components indistinct



Interaction 2

Changes in initial segments of G can make computations which disappeared reappear again: this can be resolved by making pairs of $\mathcal{A}$-components indistinct



Thanks

Thank you for your attention!

I'd be happy to answer any questions.

References

[Baz14] N. A. Bazhenov. **“Δ^0_2-Categoricity of Boolean Algebras”.** *Journal of Mathematical Sciences* 203.4 (2014), pp. 444–454.

[DHTM21] R. Downey, M. Harrison-Trainor, and A. Melnikov. **“Relativizing computable categoricity”.** *Proc. Amer. Math. Soc.* 149.9 (2021), pp. 3999–4013. ISSN: 0002-9939.

[Dor11] F. G. Dorais. **Reverse mathematics of compact countable second-countable spaces.** 2011. arXiv: 1110.6555 [math.LO].

[FS14] J. N. Y. Franklin and R. Solomon. **“Degrees that Are Low for Isomorphism”.** *Computability* 3 (2014), pp. 73–89.

[Gon77] S. S. Gončarov. **“The quantity of nonautoequivalent constructivizations”.** *Algebra and Logic* 16.3 (May 1977), pp. 169–185. ISSN: 1573-8302.

[GS79] J. Ginsburg and B. Sands. **“Minimal Infinite Topological Spaces”.** *The American Mathematical Monthly* 86.7 (1979), pp. 574–576. ISSN: 00029890, 19300972. URL: http://www.jstor.org/stable/2320588 (visited on 10/20/2023).

[Hir+02] D. R. Hirschfeldt et al. **“Degree spectra and computable dimensions in algebraic structures”.** *Annals of Pure and Applied Logic* 115.1 (2002), pp. 71–113. ISSN: 0168-0072.

[Mil+18] R. Miller et al. **“A Computable Functor from Graphs to Fields”.** *The Journal of Symbolic Logic* 83.1 (2018), 326–348.

[Rem81] J. B. Remmel. **“Recursive isomorphism types of recursive Boolean algebras”.** *Journal of Symbolic Logic* 46.3 (1981), 572–594. DOI: 10.2307/2273757.