

# Computable Algebra, I

Reed Solomon

Department of Mathematics  
University of Connecticut

[solomon@math.uconn.edu](mailto:solomon@math.uconn.edu)  
[www.math.uconn.edu/ solomon](http://www.math.uconn.edu/~solomon)

## Partial computable functions

$\varphi_0, \varphi_1, \dots$  denote partial computable functions (computer programs with no time or memory constraints)

Partial means  $\text{domain}(\varphi_e) \subseteq \omega$   
 $\varphi_e(n)$  may or may not halt.

$A$  is **computable**  $\Leftrightarrow \exists e \forall n (A(n) = \varphi_e(n))$

To show  $A$  is not computable meet requirements  $R_e$  that  $A \neq \varphi_e$  for all  $e \in \omega$ .

$A$  is **computably enumerable (c.e.)**  $\Leftrightarrow \exists e (A = \text{domain}(\varphi_e))$

$\text{Halt} = \{e \mid \varphi_e(e) \text{ halts}\}$  is c.e. but not computable.

## Working with oracles

$\varphi_e^X$  allows  $\varphi_e$  to ask “Is  $n \in X$ ?”  
( $\varphi_e$  has  $X$  on a hard drive)

$$A \leq_T B \Leftrightarrow \exists e \forall n (A(n) = \varphi_e^B(n))$$

$$A \equiv_T B \Leftrightarrow A \leq_T B \text{ and } B \leq_T A$$

$\equiv_T$  is equivalence relation on  $\mathcal{P}(\omega)$

$(\mathcal{P}(\omega) / \equiv_T, \leq_T)$  is **Turing degrees**

$\text{deg}(A) = \{B \mid B \equiv_T A\}$  will be our measure of the computational complexity of  $A$ .

## Turing degrees

Least degree  $\mathbf{0}$  consists of computable sets

$\text{deg}(\text{Halt}) = \mathbf{0}'$  satisfies  $\mathbf{0} <_T \mathbf{0}'$

Uncountable but with countable predecessor property

Degrees form an upper semi-lattice

$$A \oplus B = \{2n \mid n \in A\} \cup \{2n + 1 \mid n \in B\}$$

$$\text{deg}(A \oplus B) = \text{deg}(A) \vee \text{deg}(B)$$

Meets do not have to exist!

## Turing jump operator

Jump operator:  $A' = \{e \mid \varphi_e^A(e) \text{ halts}\}$

$\text{deg}(A) <_T \text{deg}(A')$

Iterating the jump gives the following picture:

Jump operator is not one-to-one!

$A$  is **low**  $\Leftrightarrow A' \equiv_T \mathbf{0}'$ .

$A$  is low means  $A$  is almost, but not quite computable.

$$A \text{ is } \mathbf{low}_2 \Leftrightarrow A'' \equiv_T 0''$$

$A$  is  $\mathbf{low}_2$  means  $A$  is almost computable but not quite as close to being computable as a low set.

## Set up for computable algebra

Let  $\mathcal{L}$  be a computable language.  
(Intuition:  $\mathcal{L}$  is finite.)

By Gödel numbering, formulas in  $\mathcal{L}$  can be viewed as numbers.

All structures in this tutorial are **countable**.

For any  $\mathcal{A}$ , we have  $|\mathcal{A}| \subseteq \omega$ .

(Never hurts to assume  $|\mathcal{A}| = \omega$ .)

Important to think not only of isomorphism type of  $\mathcal{A}$  but the actual copy (or coding or presentation) of  $\mathcal{A}$ .



## Assigning a degree to a structure

Fix a particular  $\mathcal{L}$  structure  $\mathcal{A}$ .

Let  $\mathcal{L}_{\mathcal{A}}$  be expansion with constants for each element of  $|\mathcal{A}|$ .

Atomic diagram of  $\mathcal{A} \subseteq \omega$ .

The **degree of**  $\mathcal{A}$  is the Turing degree of the atomic diagram of  $\mathcal{A}$ .

If  $\mathcal{L}$  is finite, then the degree of  $\mathcal{A}$  is

$$\text{deg}(|\mathcal{A}|) \vee \bigvee_{i=1}^k \text{deg}(f_i^{\mathcal{A}}) \vee \bigvee_{i=1}^l \text{deg}(R_i^{\mathcal{A}})$$

$\mathcal{A}$  is **computable**  $\Leftrightarrow$  all functions and relations in  $\mathcal{A}$  are (uniformly) computable.

## Degree spectrum

Because we assign  $\text{deg}(\mathcal{A})$  to a particular copy  $\mathcal{A}$ , we can have  $\mathcal{B} \cong \mathcal{A}$ ,  $\text{deg}(\mathcal{B}) \neq \text{deg}(\mathcal{A})$ .

We let the **degree spectrum** of  $\mathcal{A}$  be the set of degrees for all isomorphic copies of  $\mathcal{A}$

$$\text{DegSp}(\mathcal{A}) = \{d \mid \exists \mathcal{B} \cong \mathcal{A} (d = \text{deg}(\mathcal{B}))\}$$

**Question:** What can we say about  $\text{DegSp}(\mathcal{A})$ ?

**Theorem. (Knight)** Under very weak conditions,  $\text{DegSp}(\mathcal{A})$  is closed upwards.

**Example.** Let  $\mathcal{A}$  is linear order with  $|\mathcal{A}| = a_0, a_1, \dots$  and  $B$  be a set with  $\text{deg}(\mathcal{A}) \leq_T B$ .

Build  $\mathcal{C} \cong \mathcal{A}$  with  $\text{deg}(\mathcal{C}) \equiv_T B$ .

Domain of  $\mathcal{C}$  is  $c_0, c_1, \dots$

Define bijection  $f : |\mathcal{C}| \rightarrow |\mathcal{A}|$  and let

$$x \leq_{\mathcal{C}} y \Leftrightarrow f(x) \leq_{\mathcal{A}} f(y)$$

If  $n \in B \Rightarrow$ , we make  $c_{2n} <_{\mathcal{C}} c_{2n+1}$

If  $n \notin B \Rightarrow$ , we make  $c_{2n+1} <_{\mathcal{C}} c_{2n}$

Claim:  $\text{deg}(\mathcal{C}) \equiv_T B$ .

From  $B$ , compute  $\mathcal{A}$  and then  $f$  (gives  $\mathcal{C}$ )

From  $\mathcal{C}$ , compute  $n \in B \Leftrightarrow c_{2n} <_{\mathcal{C}} c_{2n+1}$ .

## Degrees for linear orders

Which linear orders have a computable copy?

When is  $\mathbf{0} \in \text{DegSp}(L)$ ?

No reasonable known answer.

$2^\omega$  many countable linear orders but  $\omega$  many computable linear orders.

Several positive results of the form:

**Theorem. (Downey, Knight)**  $L$  has a  $\Delta_2^0$  copy  $\Leftrightarrow (\mathbb{Q} + 2 + \mathbb{Q}) \cdot L$  has a computable copy.

$\mathbf{0}' \in \text{DegSp}(L) \Leftrightarrow \mathbf{0} \in \text{DegSp}((\mathbb{Q} + 2 + \mathbb{Q}) \cdot L)$

$(\mathbb{Q} + 2 + \mathbb{Q}) \cdot L$  means replace each element of  $L$  by  $\mathbb{Q} + 2 + \mathbb{Q}$

**Theorem. (Feiner)** There is a linear order  $L$  with  $\text{deg}(L) \leq_T \mathbf{0}'$  such that  $L$  has no computable copy.

Code information in algebraic invariant of  $L$ .

$x_1 <_L \cdots <_L x_n$  form  $n$  block if

$\text{Block}(L) = \{n \mid L \text{ has an } n \text{ block}\}$

$\text{Block}(L)$  is  $\Sigma_3^0$  definable in  $L$ .

$L \cong \hat{L} \Rightarrow \text{Block}(L) = \text{Block}(\hat{L})$

**Claim:** For any  $\Sigma_3^0$  set  $V$ , there is computable  $L$  with  $\text{Block}(L) = V$ .

It suffices to prove the claim.

Relativize claim by one jump

For any  $\Sigma_4^0$  set  $\hat{V}$ , there is  $\hat{L}$  with  $\text{Deg}(\hat{L}) \leq_T \mathbf{0}'$  and  $\text{Block}(\hat{L}) = \hat{V}$ .

Let  $\hat{V}$  be  $\Sigma_4^0$  complete.

$\exists$  computable  $L \cong \hat{L} \Rightarrow \text{Block}(L) = \text{Block}(\hat{L})$ .

$\text{Block}(L)$  is  $\Sigma_3^0$  and  $\text{Block}(\hat{L})$  is  $\Sigma_4^0$  complete.  
Contradiction!

**Claim:** For any  $\Sigma_3^0$  set  $V$ , there is computable  $L$  with  $\text{Block}(L) = V$ .

$$V = \{y \mid \exists x \forall u \exists s R(y, x, u, s)\}$$

Build  $L$  in stages as  $L_s$ .

Stage 0:  $L_0 = \mathbb{Z} + \mathbb{Z} + \dots + \mathbb{Z} + \dots$

$x^{\text{th}}$  copy of  $\mathbb{Z}$ :  $\omega^* + (1) + (2) + \dots + (y) + \dots$

Use  $(y)$  in  $x^{\text{th}}$  copy of  $\mathbb{Z}$  to code  $\forall u \exists s R(y, x, u, s)$ .



Add two types of points:  $z$  points and  $w$  points

At each stage add two  $z$  points around  $(y)$  on the **outside**.

By themselves,  $z$  points make  $(y)$  into  $\mathbb{Z}$ .

When think  $\forall u \exists s R(y, x, u, s)$  is true, add two  $w$  points around  $(y)$  on **inside**.

Infinitely many  $w$  points  $\Rightarrow (y)$  becomes  $y$  block

Finitely many  $w$  points  $\Rightarrow (y)$  becomes  $\mathbb{Z}$

( $y$ ) becomes  $y$  block in  $x^{th}$  copy of  $\mathbb{Z}$   
 $\Leftrightarrow$  add infinitely many  $w$  points  
 $\Leftrightarrow \forall u \exists s R(y, x, u, s)$

How do we measure when  $\forall u \exists s R(y, x, u, s)$   
looks true at stage  $s$ ?

If  $u \leq s$  and  $\forall u' \leq u \exists s' \leq s R(y, x, u', s')$ , then have  
at least  $u$  many pairs of  $w$  points at stage  $s$ .

**Theorem. (Jockusch and Soare)** For any noncomputable c.e. degree  $d$ , there is a linear order  $L_d$  such that  $L_d$  has a copy of degree  $d$  but does not have a computable copy.

There is a low  $L$  which has no computable copy.

Recall:  $d$  is low  $\Leftrightarrow d' \equiv_T 0'$

Suppose  $L$  has copies of every noncomputable degree. Must  $L$  have a computable copy?

Can  $\text{DegSp}(L) = \{d \mid d \neq 0\}$ ?

**Theorem. (Slaman, Wehner)** There is a (graph) structure  $\mathcal{A}$  such that

$$\text{DegSp}(\mathcal{A}) = \{d \mid d \neq 0\}$$

**Theorem. (R. Miller)** There is a linear order  $L$  such that  $L$  has copies in every degree  $\leq_T 0'$  except  $0$ .

**Open Question.** Is there a linear order  $L$  with  $\text{DegSp}(L) = \{d \mid d \neq 0\}$ ?

## Degrees of isomorphism types

Suppose  $\text{DegSp}(\mathcal{A})$   
has a least element  $\mathbf{a}$ .

Then  $\mathbf{a}$  is **the degree of the isomorphism type of  $\mathcal{A}$** .

In general,  $\text{DegSp}(\mathcal{A})$  need not have least element.

For any degree  $\mathbf{a}$ , there is a countable  $\mathcal{A}_a$  such that

$$\text{DegSp}(\mathcal{A}_a) = \{\mathbf{d} \mid \mathbf{a} \leq_T \mathbf{d}\}$$

**Theorem. (Richter)** For any linear order  $L$ , if  $\text{DegSp}(L)$  has a least degree, it is  $\mathbf{0}$ .

Richter constructed copies  $L_0$  and  $L_1$  such that  $\text{deg}(L_0)$  and  $\text{deg}(L_1)$  form a minimal pair.

$$\text{deg}(L_0) \vee \text{deg}(L_1) = \mathbf{0}$$

Any linear order without computable copy has no simplest presentation.

Most general possible behavior with respect to degree spectra does not occur within linear orders.

Maybe problem is that coding requires several quantifiers.

Richter's Theorem says that we cannot code a noncomputable set into the atomic diagram of the isomorphism type of a linear order.

Instead of looking at  $\text{deg}(L)$ , look at  $\text{deg}(L)'$  and ask

If  $0' \leq_T a$ , is there  $L$  such that

$$\{\text{deg}(\hat{L})' \mid \hat{L} \cong L\} = \{d \mid a \leq_T d\}?$$

If so, we say  $a$  is the **jump degree** of  $L$ .



**Theorem. (Knight)** If a linear order has a jump degree, then it is  $0'$ .

Ask similar questions for larger number of jump.

The  $n^{\text{th}}$  **jump degree** of  $\mathcal{A}$  is the least element of  $\{\text{deg}(\mathcal{B})^{(n)} \mid \mathcal{B} \cong \mathcal{A}\}$  if it exists.

**Theorem. (Ash, Downey, Jockusch, Knight)** For  $n \geq 2$  and any  $\mathbf{d} \geq_T 0^{(n)}$ , there is linear order with proper  $n$  degree  $\mathbf{d}$ .

Given two jumps, can code any degree.

## Boolean algebras

**Theorem. (Feiner)** There is a boolean algebra with  $\Delta_2^0$  copy but no computable copy.

**Theorem. (Downey and Jockush)** Every low boolean algebra has a computable copy.

If  $\text{DegSp}(\mathcal{B})$  contains all noncomputable degrees, then it contains  $\mathbf{0}$ .

Most general behavior for degree spectra is not possible in boolean algebras.

**Theorem. (Thurber; Knight and Stob)**  
Every  $\text{low}_4$  boolean algebra has a computable copy.

**Theorem. (Remmel, Vaught)** Let  $B$  be a boolean algebra with infinitely many atoms. If  $\hat{B}$  is formed from  $B$  by splitting each atom finitely many times, then  $\hat{B} \cong B$ .

**Open question.** Is every  $\text{low}_n$  boolean algebra isomorphic to a computable one?

**Theorem. (Richter)** If  $\text{DegSp}(B)$  has a least degree, it is 0.

Jump degree results also known for boolean algebras.

## Rank 1 torsion free abelian groups

Torsion free means for any  $g \in G$

$$g \neq 0_G \Rightarrow ng \neq 0_g$$

Rank 1 means  $G$  embeds into  $(\mathbb{Q}, +)$ .

Divisible closure of  $G$  is 1 dimensional vector space over  $\mathbb{Q}$ .

In general,  $\text{DegSp}(G)$  may not have least element.

**Theorem. (Downey)** For every degree  $\mathbf{a}$ , there is rank 1 torsion free abelian group  $G_{\mathbf{a}}$  such that

$$\text{DegSp}(G_{\mathbf{a}}) = \{\mathbf{d} \mid \mathbf{a} \leq_T \mathbf{d}\}$$

Let  $G$  be subgroup of  $(\mathbb{Q}, +)$  and fix  $g \neq 0_G$ .

Let  $p_0, p_1, \dots$  denote primes. **Type** of  $G$  is

$\chi(G) = \langle i_0, i_1, \dots, i_k, \dots \rangle$  where

$i_k =$  greatest  $m$  such that  $p_k^m$  divides  $g$

$i_k = \infty$  if  $p_k^m$  divides  $g$  for all  $m$ .

$\chi(G) =^* \chi(H) \Leftrightarrow$  sequences differ in finitely many places and never when  $\infty$ .

**Theorem. (Baer)**  $G \cong H \Leftrightarrow \chi(G) =^* \chi(H)$ .

Approximate  $\chi(G)$  by  $S(G) =$  all  $\langle m, k \rangle$  with  $m \leq i_k$ .  $S(G)$  is c.e. in  $G$ .

**Theorem. (Downey)** For every degree  $\mathbf{a}$ , there is rank 1 torsion free abelian group  $G_{\mathbf{a}}$  such that

$$\text{DegSp}(G_{\mathbf{a}}) = \{\mathbf{d} \mid \mathbf{a} \leq_T \mathbf{d}\}$$

Fix  $A \in \mathbf{a}$ .

$$A \oplus \bar{A} = \{2n \mid n \in A\} \cup \{2n + 1 \mid n \in \bar{A}\}$$

$A \oplus \bar{A}$  is c.e. in  $X \Rightarrow A \leq_T X$ .

Let  $G$  have  $i_k = 1$  if  $k \in A \oplus \bar{A}$  and  $i_k = 0$  if  $k \notin A \oplus \bar{A}$ .

$G$  has copy computable in  $A$ .

If  $H \cong G$ , then  $S(G)$  (and hence  $A \oplus \bar{A}$ ) is c.e. in  $H$ . Therefore,  $A \leq_T H$

What about  $n$  degrees for rank 1 torsion free abelian groups?

$G$  has finite type if  $i_k < \infty$  for all  $k$ .

**Theorem. (Coles, Downey and Slaman)**

Every rank 1 torsion free abelian group  $G$  has 2 degree. If  $G$  has finite type then  $G$  has 1 degree.

(Soskov has given alternate proof using enumeration degrees.)

**Theorem. (Downey and Jockusch)** For every  $\mathbf{d} \geq_T \mathbf{0}''$ , there is rank 1 torsion free abelian group with proper 2 degree  $\mathbf{d}$ .

## Continuous functions on computable Polish spaces

J. Miller introduced the continuous degrees.

- sit between Turing and enumeration degrees
- every continuous function has a continuous degree
- every continuous degree is the degree of some continuous function
- if the continuous degree is not a Turing degree, then the continuous functions with that degree do not have least degree among their Turing degree representations.



## Summary

$\text{deg}(\mathcal{A}) =$  Turing degree of atomic diagram

$\text{DegSp}(\mathcal{A}) = \{\mathbf{d} \mid \exists \mathcal{B} \cong \mathcal{A} (\mathbf{d} = \text{deg}(\mathcal{B}))\}$

$\text{DegSp}(\mathcal{A})$  closed upwards

Sometimes  $\text{DegSp}(\mathcal{A})$  has least element

$\forall \mathbf{a} \exists \mathcal{A}_a (\text{DegSp}(\mathcal{A}_a) = \{\mathbf{d} \mid \mathbf{a} \leq_T \mathbf{d}\})$

Occurs in rank 1 torsion free abelian groups,  
but not in linear orders or boolean algebras

Most general behavior with respect to de-  
gree spectra does not occur in linear orders  
or boolean algebras.

If  $\mathcal{A}$  has a copy which is close to computable, must it have a computable copy? No.

There is low linear order with no computable copy.

$$\exists \mathcal{A} (\text{DegSp}(\mathcal{A}) = \{\mathbf{d} \mid \mathbf{0} <_T \mathbf{d}\})$$

Does not occur in boolean algebras because if  $B$  has low copy then  $B$  has computable copy.

Given some finite number of jumps, can we do more coding?

If  $n \geq 1$  and  $\mathbf{0}^{(n)} \leq_T \mathbf{a}$ , is there  $\mathcal{A}$  with proper  $n^{\text{th}}$  jump degree  $\mathbf{a}$ ?

For linear orders, yes if  $n \geq 2$ .

For rank 1 torsion free abelian groups, only if  $n \leq 2$ .

# Computable Algebra, II

Reed Solomon

Department of Mathematics  
University of Connecticut

[solomon@math.uconn.edu](mailto:solomon@math.uconn.edu)  
[www.math.uconn.edu/~solomon](http://www.math.uconn.edu/~solomon)

## Relations on computable structures

Focus on computable structures.

Consider a computable linear order  $L$ .

The **adjacency relation** on  $L$  is

$$\text{adj}(L) = \{\langle x, y \rangle \mid x \text{ immediately precedes } y\}$$

Can have computable  $\hat{L} \cong L$  such that

$$\text{deg}(\text{adj}(L)) \neq \text{deg}(\text{adj}(\hat{L}))$$

Is there always computable  $\hat{L} \cong L$  with  $\text{adj}(\hat{L})$  computable?

Is there an upper bound on  $\text{adj}(\hat{L})$  for computable  $\hat{L} \cong L$ ?

What are the possible sets of degrees for  $\text{adj}(\hat{L})$  when  $\hat{L}$  varies over all computable  $\hat{L} \cong L$ ?

Fix class of algebraic structures and fix algebraic object associated with this class.

- adjacency relation on linear orders
- atom relation on boolean algebras
- center of a group
- basis for vector space
- order on torsion free abelian group

Is there a computable structure in class for which object is not computable?

If so, how badly not computable is the object?

What are the possible degrees of the object?

## Adjacency relation on linear order

Is adjacency computable for computable  $L$ ?

**Theorem.** There is a computable linear order in which the adjacency relation is not computable.

Domain of  $L$  is  $\omega$  and  $\leq_L$  defined in stages.

To determine if  $n \leq_L m$ , run construction until both  $n$  and  $m$  placed in  $L$ .

Requirement  $R_e$ :  $\varphi_e$  does not compute adjacency relation.

Stage 0: Place all even numbers into  $L$ :

$$2 <_L 4 <_L 6 <_L 8 <_L 10 <_L 12 <_L \dots$$

Stage 0:

$$2 <_L 4 <_L 6 <_L 8 <_L 10 <_L 12 <_L \dots$$

Stage  $s$ : For each  $e \leq s$ , run  $\varphi_e(a_e, b_e)$  for  $s$  steps.

Take least undefeated  $e$  with  $\varphi_e(a_e, b_e) = 1$ .  
( $\varphi_e$  thinks  $a_e$  and  $b_e$  are adjacent.)

Put least unused odd number between  $a_e$  and  $b_e$  in  $L$ . Declare  $\varphi_e$  defeated.

Note:  $(L, \leq_L) \cong (\omega, \leq)$ !

What are the possible degrees for adjacency relation on a computable copy of  $(\omega, \leq)$ ?

Since adjacency is definable by  $\Pi_1^0$  formula, it must have c.e. Turing degree.

The **degree spectrum** of relation  $R$  on computable  $\mathcal{A}$  is

$$\text{DegSp}(\mathcal{A}, R) = \{d \mid \exists \text{ comp } \mathcal{B} \cong \mathcal{A} (R^{\mathcal{B}} \equiv_T d)\}$$

$$\text{DegSp}((\omega, \leq), \text{Adj}) \subseteq \{d \mid d \text{ is c.e.}\}$$

How badly noncomputable can the adjacency relation be on a computable copy of  $(\omega, \leq)$ ?

At worst, it can have degree  $0'$ .



**Theorem.** For any c.e. set  $C$ , there is a computable copy  $(L, \leq_L)$  of  $(\omega, \leq)$  such that  $\text{Adj}(L) \equiv_T C$ .

$$2 <_L 4 <_L 6 <_L 8 <_L 10 <_L 12 <_L \dots$$

At stage  $s$ , if  $n$  enters  $C$  then place next unused odd number between  $a_n$  and  $b_n$ .

$n \in C \Leftrightarrow a_n$  and  $b_n$  are not adjacent.

Let  $C$  be the halting problem: adjacency on computable copy of  $(\omega, \leq)$  can be  $0'$  (as bad as possible).

**Corollary.** There is computable linear order  $L$  such that

$$\text{DegSp}(L, \text{Adj}) = \{d \mid d \text{ is c.e.}\}$$

Does every computable linear order  $L$  have a computable copy  $\hat{L}$  such that adjacency in  $\hat{L}$  is computable?

**Theorem. (Downey, Moses)** There is a computable linear order  $L$  such that  $\text{DegSp}(L, \text{Adj}) = \{0'\}$ .

**Theorem. (Downey)** There is a computable linear order  $L$  such that  $\text{DegSp}(L, \text{Adj}) =$  all noncomputable c.e. degrees.

One extreme situation:

**Theorem. (Downey, Moses)** There is computable  $L$  with  $\text{DegSp}(L, \text{Adj}) = \{0'\}$ .

Opposite extreme: If  $\text{DegSp}(\mathcal{A}, R) = \{0\}$ , then we say  $R$  is **intrinsically computable**.

When is a computable relation  $R$  intrinsically computable for linear order  $L$ ?

**Theorem. (Hirschfeldt, Moses)** A computable relation on a computable linear order is intrinsically computable  $\Leftrightarrow$  it is definable by a quantifier free formula with constants from  $L$ .

Furthermore, if  $R$  is not intrinsically computable, then  $\text{DegSp}(L, R)$  is infinite.

This is different from the general situation for computable structures.

**Theorem. (Hirschfeldt)** For any two c.e. degrees  $\mathbf{a}$  and  $\mathbf{b}$ , there is a computable  $\mathcal{A}$  and a relation  $R$  on  $\mathcal{A}$  such that  $\text{DegSp}(\mathcal{A}, R) = \{\mathbf{a}, \mathbf{b}\}$ .

Most general situation with respect to degree spectra of relations cannot occur in linear orders.

## Atoms in a boolean algebra

There are computable boolean algebras  $\hat{B} \cong B$  with

$$\deg(\text{atom}(\hat{B})) \neq \deg(\text{atom}(B))$$

**Theorem. (Goncharov)** There is a computable boolean algebra for which the atoms are not computable in any computable copy.

**Theorem. (Downey)** For every computable boolean algebra  $B$ , there is a computable  $\hat{B} \cong B$  such that the set of atoms in  $\hat{B}$  is  $<_T 0'$ .

Cannot have  $\text{DegSp}(B, \text{Atom}) = \{0'\}$ !

Let  $B$  be computable boolean algebra.

There is computable linear order  $L$  such that  $B \cong \text{IntAlg}(L)$ .

Atoms of  $B$  correspond to adjacencies in  $L$ .

**Theorem. (Downey, Goncharov, Hirschfeldt)** A computable relation  $R$  on a computable boolean algebra is intrinsically computable  $\Leftrightarrow$  it is definable by a quantifier free formula with constants from  $B$ .

Furthermore, if  $R$  is not intrinsically computable, then  $\text{DegSp}(B, R)$  is infinite.

Most general behavior with respect to degree spectra of relations does not occur in boolean algebras.

## Basis for torsion free abelian groups

Let  $G$  be torsion free abelian group.

$\{g_1, \dots, g_k\}$  are **linearly independent** if for any  $n_1, \dots, n_k \in \mathbb{Z}$

$$\sum n_i g_i = 0_G \Rightarrow \forall i (n_i = 0)$$

A **basis** for  $G$  is maximal independent set and the **rank** of  $G$  is the size of a basis.

If  $G$  is divisible, then  $G$  is vector space over  $\mathbb{Q}$ . These terms have same meaning as vector space terms in this context.

There is a computable torsion free abelian group  $G$  with no computable basis.

Unlike adjacency or atom relation:

**Theorem. (Dobritsa)** Every computable torsion free abelian group  $G$  has a computable copy with a computable basis.

Some copies of  $G$  may not have computable basis, but if the copy is chosen carefully, there will be a computable basis.

**Proof sketch.** Assume  $G$  has infinite rank.

Build  $H$  and isomorphism  $f : H \rightarrow G$  in stages

$$H_0 \subseteq H_1 \subseteq \cdots \subseteq H_s \subseteq \cdots$$

and  $f_s : H_s \rightarrow G$  such that  $H = \cup H_s$  and  $f = \lim_s f_s$ .



Define basis  $\{b_0, b_1, b_2, \dots\}$  for  $H$  during construction.

Stage  $s+1$ : Have  $\{b_0, \dots, b_s\}$  and need to define  $b_{s+1}$ .

Suppose  $f_s(b_i) = c_i^s \in G$ .

Want:  $\lim_s c_i^s = c_i$  and  $c_0, c_1, \dots$  is basis for  $G$ .

Problem: Discover new relations between the  $c_i^s$  elements, so must redefine  $f_s$  preserving addition facts defined so far.

Each  $h \in H_s$  has dependence relation

$$\alpha h = \alpha_0 b_0 + \cdots + \alpha_s b_s$$

with  $|\alpha|, |\alpha_i| \leq s$ .

For  $t \geq s$ , need

$$\alpha f_t(h) = \alpha_0 f_t(b_0) + \cdots + \alpha_s f_t(b_s)$$

When redefining  $f_{s+1}(b_i)$ , make sure

$$\alpha x = \alpha_0 f_{s+1}(b_0) + \cdots + \alpha_s f_{s+1}(b_s)$$

has a solution.

We set  $f_t(h)$  to be this solution.

Defining  $f_{s+1}(b_0), \dots, f_{s+1}(b_{s+1})$ :

Set  $c_{s+1}^s = 0_G$  and look at

$$c_0^s, c_1^s, \dots, c_s^s, c_{s+1}^s$$

Let  $I$  be least such that  $c_0^s, \dots, c_I^s$  is  $s + 1$  dependent.

Let  $c'_I, \dots, c'_{s+1}$  be such that

$$c_0^s, c_1^s, \dots, c_{I-1}^s, c'_I, \dots, c'_{s+1}$$

is  $s + 1$  independent.

Set  $c_i^{s+1} = c_i^s$  for  $i < I$ .

Set  $c_j^{s+1} = s!c'_j + c_j^s$  for  $j \geq I$ .

If  $h \in H_s$  assigned dependence relation

$$\alpha h = \alpha_0 b_0 + \cdots + \alpha_s b_s$$

(with  $|\alpha|, |\alpha_i| \leq s$ ), then

$$\alpha x = \alpha_0 c_0^s + \cdots + \alpha_s c_s^s$$

has a solution and so does

$$\alpha x = \alpha_0 c_0^{s+1} + \cdots + \alpha_s c_s^{s+1}$$

since this is the same as

$$\alpha x = \alpha_0 c_0^s + \cdots + \alpha_s c_s^s + s!(\alpha_0 c_0' + \cdots + \alpha_s c_s')$$

and  $|\alpha| \leq s$ .

Once  $f_{s+1}(b_i)$  defined, we extend definition by solving equations.

Make sure  $f$  is onto.

## Orders on torsion free abelian groups

$(G, \leq)$  is an ordered group if  $G$  is group and for all  $g, h, k \in G$

$$g \leq h \Rightarrow gk \leq hk \wedge kg \leq kh$$

Abelian  $G$  is orderable  $\Leftrightarrow G$  is torsion free.

**Theorem. (Downey and Kurtz)** There is a computable torsion free abelian group  $G$  with no computable order.

In their proof,  $G \cong \bigoplus_{\omega} \mathbb{Z}$  under complicated coding. This does have computable copy with computable order.

**Corollary to Dobritsa's Theorem.** Every computable torsion free abelian group  $G$  has a computable copy with a computable order.

Take a computable copy  $H$  of  $G$  which has a computable basis and use this basis to define an order on  $H$ .

For adjacency and atom: no computable copy with computable relation

For basis and order: always computable copy with computable relation

Is difference due to fact that basis and order are not unique?

## Archimedean classes in ordered groups

Let  $G$  be ordered abelian group.

$$a \ll b \Leftrightarrow \forall n (|na| < |b|)$$

$$a \equiv b \Leftrightarrow \exists n (|a| \leq |nb| \wedge |b| \leq |na|)$$

$A \subseteq G$  is set of **archimedean representatives**  
if

$$\forall a, b \in A (a \neq b \rightarrow a \not\equiv b)$$

$$\forall g \in G \exists a \in A (g \equiv a)$$

Sets of archimedean representatives behave like adjacency and atoms.

**Theorem. (Solomon)** There is a computable ordered abelian group  $G$  such that no computable copy of  $G$  has a computable set of archimedean representatives.

Code Feiner's linear order  $L$  which has  $\Delta_2^0$  copy but no computable copy into the archimedean classes so that a copy of  $L$  is computable from any set of archimedean representatives.



**Question.** Does every computable ordered abelian group  $(G, \leq_G)$  have a computable copy with a computable basis?

$G$  is torsion free, so Dobritsa's Theorem says there is computable  $H \cong G$  such that  $H$  has computable basis.

Is there a computable  $\leq_H$  on  $H$  such that  $(H, \leq_H) \cong (G, \leq_G)$ ?

**Theorem. (Goncharov, Lempp, Solomon)**  
If  $(G, \leq_G)$  is a computable ordered abelian group with finitely many archimedean classes, then  $(G, \leq_G)$  has a computable copy with a computable basis.

**Open question:** What happens if  $(G, \leq_G)$  has infinitely many archimedean classes?

## Computable dimension

$\exists$  computable copies  $L_0$  and  $L_1$  of  $(\omega, \leq)$

- adjacency in  $L_0$  is computable
- adjacency in  $L_1$  is not computable

$L_0$  and  $L_1$  are not computably isomorphic.

The **computable dimension** of a computable structure  $\mathcal{A}$  is the number of computable copies of  $\mathcal{A}$  up to computable isomorphism.

Notice:  $\text{CompDim}(\mathcal{A}) \in \omega$ .

Intuition: If  $\text{CompDim}(\mathcal{A}) = 1$ , then computational properties do not depend on which computable copy of  $\mathcal{A}$  you consider.

$\mathcal{A}$  is **computably categorical (c.c.)**  $\Leftrightarrow$

$\text{CompDim}(\mathcal{A}) = 1 \Leftrightarrow$

$\forall$  computable  $\mathcal{B} \cong \mathcal{A}$ , there is computable isomorphism from  $\mathcal{A}$  to  $\mathcal{B}$

If a linear order is computably categorical, then the adjacency relation has the same degree in every computable copy.

Goal: Classify which structures are computably categorical within various classes of structures.

## Classifying computable categoricity

Linear order is c.c.  $\Leftrightarrow$  finite number of adjacencies

Boolean algebra is c.c.  $\Leftrightarrow$  finite number of atoms

Algebraically closed field is c.c.  $\Leftrightarrow$  finite transcendence degree

Torsion free abelian group is c.c.  $\Leftrightarrow$  finite rank

Ordered abelian group is c.c.  $\Leftrightarrow$  finite rank

Conditions also known for abelian  $p$ -groups and trees.

**Open question:** What about fields?

## Possible computable dimensions

The following classes of structures admit only computable dimension 1 or  $\omega$ :

Linear orders, boolean algebras, algebraically closed fields, real closed fields, abelian  $p$ -groups, torsion free abelian groups, ordered abelian groups, trees

(Dzgoev, Goncharov, Larouche, Lempp, McCoy, Metakides, R. Miller, Nerode, Nurtazin, Remmel, Smith, Solomon)

This is not the most general possible situation.

**Theorem. (Goncharov)** For each  $1 < n < \omega$ , there is a computable structure  $\mathcal{A}_n$  with  $\text{CompDim}(\mathcal{A}_n) = n$ .

Which classes of structures admit finite computable dimensions other than 1?

## Finite computable dimension

**Theorem. (Goncharov, Molokov, Romanovskii)** For each  $1 < n < \omega$ , there is a computable nilpotent group  $G_n$  with  $\text{CompDim}(G_n) = n$ .

Furthermore,  $G_n$  can be chosen to be torsion free.

Similar results for partially ordered sets, graphs and lattices.

Which classes of structures admit the most general possible behavior with respect to all of the notions from computable algebra mentioned so far?

## Strong Coding

A class  $\mathcal{C}$  of algebraic structures admits **strong coding** if for any countable graph  $G$ , there is a structure  $\mathcal{A}_G$  from  $\mathcal{C}$  such that

- $\text{DegSp}(G) = \text{DegSp}(\mathcal{A}_G)$
- $\text{CompDim}_d(G) = \text{CompDim}_d(\mathcal{A}_G)$  for any degree  $d$
- For any  $R \subseteq G$ , there is  $U \subseteq |\mathcal{A}|$  such that  $\text{DegSp}(G, R) = \text{DegSp}(\mathcal{A}_G, U)$

If  $\mathcal{C}$  admits strong coding, then the most general computable model theoretic behavior is realized inside  $\mathcal{C}$ .

Not strong	Strong
boolean algebras linear orders trees	lattices partial orders graphs
$ACF_0$ $RCF_0$	rings integral domains
abelian $p$ -groups  torsion free abelian groups  ordered abelian groups	nilpotent groups  torsion free nilpotent groups

(Hirschfeldt, Khoussainov, Shore and Slinko building on coding techniques of Rabin)

**Open Question:** What about fields?



If  $\mathcal{C}$  admits strong coding, then the algebraic behavior within  $\mathcal{C}$  does not interfere with the computational properties.

If not, then the algebraic behavior within  $\mathcal{C}$  interacts non-trivially with the computational properties.

## Summary

Relation  $R$  on computable  $\mathcal{A}$

$$\text{DegSp}(\mathcal{A}, R) = \{d \mid \exists \text{comp. } \mathcal{B} \cong \mathcal{A} (d = \text{deg}(R^{\mathcal{B}}))\}$$

For adjacency in linear orders, atom in boolean algebras and archimedean equivalence in ordered abelian groups, there are structures in which they are noncomputable in all computable copies.

For basis or order in torsion free abelian group, there is always computable copy in which relation is computable.

Computable dimension and computable categoricity

Which computable dimensions are possible?

In which classes of structures do they occur?

Classify the computably categorical structures in various classes.

Strong coding

Isolate classes of structures in which the most general possible computational behavior occurs.

# Computable Algebra, III

Reed Solomon

Department of Mathematics  
University of Connecticut

[solomon@math.uconn.edu](mailto:solomon@math.uconn.edu)  
[www.math.uconn.edu/ solomon](http://www.math.uconn.edu/~solomon)

## Relative computable categoricity

A computable structure  $\mathcal{A}$  is

**computably categorical (c.c.)** if for **every computable**  $\mathcal{B} \cong \mathcal{A}$ , there is a computable isomorphism from  $\mathcal{A}$  to  $\mathcal{B}$ .

**relatively computably categorical** if for **every**  $\mathcal{B} \cong \mathcal{A}$ , there is an isomorphism from  $\mathcal{A}$  to  $\mathcal{B}$  which is computable in  $\text{Deg}(\mathcal{B})$ .

Relative c.c.  $\Rightarrow$  c.c.

In many examples, these notions are the same.

For linear orders

c.c.  $\Leftrightarrow$  relatively c.c.  $\Leftrightarrow$   
finite number of adjacencies.

In general, computable categoricity does not imply relative computable categoricity.

**Theorem. (Goncharov)** There is a computable  $\mathcal{A}$  which is computably categorical but not relatively so.

**Theorem. (Goncharov)** If  $\mathcal{A}$  is  $\Sigma_2$  decidable (the  $\forall\exists$  diagram is decidable), then  $\mathcal{A}$  is computably categorical  $\Leftrightarrow$  it is relatively computably categorical.

**Theorem. (Kudinov)** There is a  $\Sigma_1$  decidable  $\mathcal{A}$  which is computably categorical but not relatively so.

Why work with relative computable categoricity?

## Scott families

A **Scott family** for  $\mathcal{A}$  is a set of formulas  $\Phi$  (using a fixed finite set of parameters from  $\mathcal{A}$ ) such that

- $\forall \bar{a} \in \mathcal{A} \exists \varphi \in \Phi (\mathcal{A} \models \varphi(\bar{a}))$
- $\forall \bar{a}, \bar{b} \in \mathcal{A} [\exists \varphi \in \Phi (\mathcal{A} \models \varphi(\bar{a}) \wedge \varphi(\bar{b})) \rightarrow (\mathcal{A}, \bar{a}) \cong (\mathcal{A}, \bar{b})]$

$\mathcal{A}$  countable  $\Rightarrow \mathcal{A}$  has Scott family  $\Phi$  in  $\mathcal{L}_{\omega_1, \omega}$ .

A Scott family  $\Phi$  is **formally**  $\Sigma_1^0$  if each formula in  $\Phi$  is finitary  $\exists$  formula and the set (of Gödel numbers for formulas in)  $\Phi$  is c.e.

## Characterization of relative c.c.

**Theorem. (Ash, Knight, Mannasse, Slaman and Chisholm)** A computable  $\mathcal{A}$  is relatively computably categorical  $\Leftrightarrow$  it has a formally  $\Sigma_1^0$  Scott family.

Using back and forth relations, there is a simpler condition describing the existence of Scott families of finitary  $\Sigma_1^0$  formulas.

Using this definition, the relation  $\mathcal{A}$  is relatively computably categorical becomes  $\Sigma_3^0$ .



## Intrinsically c.e. relations

Let  $R$  be a relation on computable  $\mathcal{A}$ .

$R$  is **intrinsically c.e.** if for every **computable**  $\mathcal{B} \cong \mathcal{A}$ ,  $R^{\mathcal{B}}$  is c.e.

$R$  is **relatively intrinsically c.e.** if for **every**  $\mathcal{B} \cong \mathcal{A}$ ,  $R^{\mathcal{B}}$  is c.e. relative to  $\text{deg}(\mathcal{B})$ .

A set  $X$  is c.e. relative to  $Y$  if there is an index  $e$  such that

$$X = \{n \mid \varphi_e^Y(n) \text{ halts}\}$$

There is procedure which enumerates  $X$  when given oracle  $Y$ .

## Characterizing intrinsically c.e. relations

A formula  $\varphi$  is **computable  $\Sigma_1$**  if it is a c.e. infinitary disjunction  $\bigvee \psi_i$  in which each  $\psi_i$  is a finitary  $\Sigma_1^0$  formula.

**Theorem. (Ash, Nerode)** Under some effectiveness hypotheses,  $R$  is intrinsically c.e.  $\Leftrightarrow R$  is definable by a computable  $\Sigma_1$  formula (with finitely many parameters).

**Theorem. (Ash, Knight, Mannasse, Slaman and Chisholm)**  $R$  is relatively intrinsically c.e.  $\Leftrightarrow R$  is definable by a computable  $\Sigma_1$  formula (with finitely many parameters).

## Application of relative c.c.

The definition of computable categoricity gives a  $\Pi_1^1$  condition for determining if  $\mathcal{A}$  is c.c.

When classifying which structures in given class are c.c., we want a condition that is simpler than  $\Pi_1^1$ .

For linear order,  $L$  is c.c.  $\Leftrightarrow L$  has finite number of adjacencies.

There is  $n$  such that for all choices of more than  $n$  many pairs of elements of  $L$ , at least one pair of these elements is not adjacent.

At worst, computable categoricity for  $L$  is  $\Sigma_3^0$ .

## Computationally categoricity for trees

**Theorem. (R. Miller)** Every computable infinite height tree has computable dimension  $\omega$ .

**Theorem. (Lempp, McCoy, R. Miller, Solomon)** The following hold for computable finite height tree  $T$ .

- $T$  has computable dimension 1 or  $\omega$ .
- $T$  is c.c.  $\Leftrightarrow T$  has finite type.
- $T$  is c.c.  $\Leftrightarrow T$  is relatively c.c.

Let  $(T, \leq)$  be a finite height tree and  $x \in T$  have immediate successors  $\{x_i \mid i \in I\}$ . Let  $T[x_i] = \{y \in T \mid x_i \leq y\}$ . We say that  $x$  is **of strongly finite type** if

- $\exists$  only finitely many isomorphism types in  $\{T[x_i] \mid i \in I\}$ , each of which is of strongly finite type; and
- $\forall j, k \in I$ , if  $T[x_j] \hookrightarrow T[x_k]$ , then either  $T[x_j] \cong T[x_k]$  or the isomorphism type of  $T[x_k]$  appears only finitely often in  $\{T[x_i] \mid i \in I\}$ .

$T$  is **of strongly finite type** if the root node is of strongly finite type.

Using the same notation, we say that  $x$  is **of finite type** if

- $\exists$  only finitely many isomorphism types in  $\{T[x_i] \mid i \in I\}$ , each of which is of finite type; and
- Every isomorphism type which appears infinitely often in  $\{T[x_i] \mid i \in I\}$  is of strongly finite type; and
- $\forall j, k \in I$ , if  $T[x_j] \hookrightarrow T[x_k]$ , then either  $T[x_j] \cong T[x_k]$  or the isomorphism type of  $T[x_j]$  appears only finitely often in  $\{T[x_i] \mid i \in I\}$ , or the isomorphism type of  $T[x_k]$  appears only finitely often in  $\{T[x_i] \mid i \in I\}$ .

$T$  itself is **of finite type** if every node in  $T$  is of finite type.

A finite height tree  $T$  is c.c.  $\Leftrightarrow T$  has finite type.

Definition of finite type is analytic, so does not seem to be a reasonable classification.

$T$  is c.c.  $\Leftrightarrow T$  is relatively c.c.

Now we have  $\Sigma_3^0$  condition to determine c.c. for finite height trees.

Sketch a proof of

**Theorem. (Goncharov)** There is a computable  $\mathcal{A}$  which is computably categorical but not relatively so.

## Enumerations of families of sets

Let  $S \subseteq \mathcal{P}(\omega)$  be countable family of sets.

An **enumeration** of  $S$  is a set  $\mu \subseteq \omega \times \omega$  such that

$$S = \{\mu(i) \mid i \in \omega\}$$

where  $\mu(i) = \{x \mid \langle i, x \rangle \in \mu\}$ .

$\mu$  is **computable** if it is computable as a set of ordered pairs.

$\mu$  is 1-to-1 if  $\forall i \neq j (\mu(i) \neq \mu(j))$ .



If  $\mu$  and  $\nu$  are computable enumerations of  $S$ , then

$$\nu \leq \mu \Leftrightarrow \exists \text{ comp. } f (\nu(i) = \mu(f(i)))$$

$$\nu \equiv \mu \Leftrightarrow \nu \leq \mu \text{ and } \mu \leq \nu$$

If  $\nu \equiv \mu$ , we say  $\nu$  and  $\mu$  are the same up to **computable equivalence**.

Let  $S$  have a unique 1-to-1 computable enumeration up to computable equivalence.

For computable 1-to-1  $\nu$  and  $\mu$ ,  $\exists$  computable permutation  $f$  of  $\omega$  with  $\mu(i) = \nu(f(i))$ .

## Computability part of proof

$S \subseteq \mathcal{P}(\omega)$  is **discrete** if for each  $A \in S$  there is a finite string  $\sigma \in 2^{<\omega}$  such that for all  $B \in S$

$$\sigma \subset B \Leftrightarrow B = A$$

$S$  is **effectively discrete** if there is a c.e. set  $E$  of finite strings such that

$$\forall A \in S \exists \sigma \in E (\sigma \subset A)$$

$$\forall \sigma \in E \forall A, B \in S (\sigma \subset A, B \rightarrow A = B)$$

**Theorem. (Selivanov)** There is  $S \subseteq \mathcal{P}(\omega)$  which has unique 1-to-1 computable enumeration and is discrete but not effectively discrete.

## Coding part of proof

Code  $S$  into directed rigid graph  $G(S)$ .

Place infinitely many coding locations  $c$  with  $c \rightarrow c$ .

Use coding location to code a set  $A$  from  $S$  with each set coded exactly once.

$n \in A \Rightarrow$  add cycle of length  $2n$

$n \notin A \Rightarrow$  add cycle of length  $2n + 1$

If  $H$  is computable copy of  $G(S)$ , then there is associated computable 1-to-1 enumeration  $\mu_H$  of  $S$ .

If  $\mu$  is computable 1-to-1 enumeration of  $S$ , then there is associated computable copy  $G_\mu$  of  $G(S)$ .

If  $\mu \equiv \nu$ , then  $G_\mu$  and  $G_\nu$  are computably isomorphic.

If  $H, K$  are computably isomorphic copies of  $G(S)$ , then  $\mu_H \equiv \mu_K$ .

$\text{CompDim}(G(S)) =$  number of 1-to-1 computable enumerations of  $S$ .

If  $S$  has unique computable 1-to-1 enumeration, then  $G(S)$  is computably categorical.

**Theorem. (Goncharov)** There is a computable  $\mathcal{A}$  which is computably categorical but not relatively so.

**Theorem. (Selivanov)** There is  $S \subseteq \mathcal{P}(\omega)$  which has unique 1-to-1 computable enumeration and is discrete but not effectively discrete.

$G(S)$  is computably categorical.

Let  $S$  have unique 1-to-1 computable enumeration and be discrete but not effectively discrete.

Then  $S$  does not have a formally  $\Sigma_1^0$  Scott family.

Can translate a formally  $\Sigma_1^0$  Scott family into an effectively discrete set for  $S$  by considering coding locations and the formulas defining them in the Scott family.

**Theorem. (Goncharov)** There is a computable  $\mathcal{A}$  which is computably categorical but not relatively so.

Proof: Take family  $S$  from Selivanov's Theorem and consider  $G(S)$ .

$G(S)$  is computably categorical since  $S$  has unique 1-to-1 computable enumeration.

$G(S)$  has no formally  $\Sigma_1^0$  Scott family since  $S$  is discrete but not effectively discrete.



## Lifting Goncharov's Theorem

Let  $\alpha$  be a computable ordinal.

$\alpha$  is a countable ordinal which has a computable copy.

There is a computable linear order  $L$  such that  $L \cong \alpha$ .

Really work with ordinal notations.

$$X \leq_T \mathbf{0}' \Leftrightarrow X \text{ is } \Delta_2^0$$

$$X \leq_T \mathbf{0}'' \Leftrightarrow X \text{ is } \Delta_3^0$$

For finite  $n$ ,  $X \leq_T \mathbf{0}^{(n)} \Leftrightarrow X \text{ is } \Delta_{n+1}^0$ .

Using ordinal notations, we can extend this analysis to computable ordinals and work with sets which are  $\Delta_\alpha^0$ .

A computable structure  $\mathcal{A}$  is

**$\Delta_\alpha^0$  categorical** if every computable  $\mathcal{B} \cong \mathcal{A}$  is isomorphic to  $\mathcal{A}$  by a  $\Delta_\alpha^0$  map.

**relatively  $\Delta_\alpha^0$  categorical** if every  $\mathcal{B} \cong \mathcal{A}$  is isomorphic to  $\mathcal{A}$  by a map which is  $\Delta_\alpha^0(\text{deg}(\mathcal{B}))$

## Characterizing relative $\Delta_\alpha^0$ categoricity

**Theorem. (Ash, Knight, Mannasse, Slaman and Chisholm)** Computable  $\mathcal{A}$  is relatively  $\Delta_\alpha^0$  categorical  $\Leftrightarrow \mathcal{A}$  has a formally  $\Sigma_\alpha^0$  Scott family.

A formally  $\Sigma_\alpha^0$  Scott family is a Scott family  $\Phi$  in which the set  $\Phi$  is  $\Sigma_\alpha^0$  and each formula  $\varphi \in \Phi$  is a computable  $\Sigma_\alpha$  formula.

To define computable  $\Sigma_\alpha$  formulas, work in  $\mathcal{L}_{\omega_1, \omega}$  but only allow c.e. conjunctions and disjunctions.

**Theorem. (Goncharov, Harizanov, Knight, McCoy, Miller, Solomon)** For each computable successor ordinal, there is a computable  $\mathcal{A}$  which is  $\Delta_\alpha^0$  categorical but not relatively so.

There is computable  $\mathcal{A}$  with a relation  $R$  which is intrinsically  $\Sigma_\alpha^0$  but not relatively so.

For each finite  $n$ , there is computable  $\mathcal{A}$  with  $\Delta_\alpha^0$  dimension  $n$ .

There is computable  $\mathcal{A}$  such that

$$\text{DegSp}(\mathcal{A}) = \{\mathbf{d} \mid \Delta_\alpha^0(\mathbf{d}) \neq \Delta_\alpha^0\}$$

**Open question:** What happens at limit ordinals?

## Idea of proof

Relativize Selivanov's Theorem to  $\Delta_\alpha^0$  and code family into  $\Delta_\alpha^0$  graph  $G$  such that

$G$  has unique  $\Delta_\alpha^0$  copy up to  $\Delta_\alpha^0$  isomorphism

$G$  has no  $\Sigma_\alpha^0$  Scott family of finitary existential formulas

We want computable  $G^*$  such that

$G^*$  has one computable copy up to  $\Delta_\alpha^0$  isomorphism

$G^*$  has no  $\Sigma_\alpha^0$  Scott family of computable  $\Sigma_\alpha^0$  families.

## Computable categoricity and Scott families

A computable  $\mathcal{A}$  is relatively computably categorical  $\Leftrightarrow \mathcal{A}$  has a formally  $\Sigma_1^0$  Scott family.

Formally  $\Sigma_1^0$  Scott family is set of formulas  $\Phi$

each  $\phi \in \Phi$  is a finitary  $\Sigma_1^0$  formula  
(restrict complexity of formulas in family)

and the set  $\Phi$  is c.e.  
(restrict complexity of set of formulas)

$\mathcal{A}$  has a formally  $\Sigma_1^0$  Scott family  
 $\Rightarrow \mathcal{A}$  is relatively computably categorical  
 $\Rightarrow \mathcal{A}$  is computably categorical.

How complex is Scott family for c.c.  $\mathcal{A}$ ?

If  $\mathcal{A}$  is 2 decidable, then  $\mathcal{A}$  is c.c.  $\Leftrightarrow$

$\mathcal{A}$  is relatively c.c.  $\Leftrightarrow$

$\mathcal{A}$  has a formally  $\Sigma_1^0$  Scott family.

What is  $\mathcal{A}$  is not 2 decidable?

Can we place bounds on complexity of formulas in Scott family and on the complexity of set of formulas?

Does  $\mathcal{A}$  have a Scott family of finitary formulas?

## Computable stability

A computable  $\mathcal{A}$  is **computably stable**  $\Leftrightarrow$  for every computable  $\mathcal{B} \cong \mathcal{A}$ , every isomorphism between  $\mathcal{A}$  and  $\mathcal{B}$  is computable.

Example. If  $\mathcal{A}$  is computably categorical and rigid, then  $\mathcal{A}$  is computably stable.

Example. If  $\mathcal{A}$  is finite dimensional vector space over  $\mathbb{Q}$  then  $\mathcal{A}$  is computably stable.

Does every computably stable  $\mathcal{A}$  have a Scott family of finitary formulas? No.

**Open question:** How complex is the index set of computably categorical structures? Is it as bad as  $\Pi_1^1$  complete?



**Theorem. (Cholak, Shore, Solomon)**

There is a computably stable  $\mathcal{A}$  such that  $\mathcal{A}$  has no Scott family of finitary formulas.

The proof uses the same coding idea but a different enumeration theorem.

There is a partial computable function  $\psi$  and a family of c.e. sets  $\mathcal{A}$  such that  $\psi$  gives a 1-to-1 enumeration of  $\mathcal{A}$  as

$$\{A_i \mid i \in \omega\}$$

( $n \in A_i$  if and only if  $\langle n, i \rangle \in \text{domain}(\psi)$ )

with the following properties

1.  $\mathcal{A}$  has a single 1-to-1 c.e. enumeration
  
2.  $\exists^\infty d \exists^\infty s \exists m \neq d (A_{d,s} \subseteq A_{m,s+1})$ . For each such  $d$ 
  - (a)  $A_d$  is infinite.
  
  - (b)  $\forall k$ , there is stage  $s_k$  such that  $\forall t > s_k$  and all indices  $m \neq d$ , if  $A_{d,t} \subseteq A_{m,t+1}$ , then  $A_{m,t}$  contains no numbers  $\leq k$ .
  
  - (c)  $\forall$  indices  $z$ , there is stage  $s_z$  such that for all stages  $t > s_z$  and all indices  $m \neq d$ , if  $A_{d,t} \subseteq A_{m,t+1}$ , then  $m > z$ .
  
  - (d)  $\forall m \neq d$ , if there is stage  $s$  such that we enumerate the elements of  $A_{d,s}$  into  $A_{m,s+1}$  to cause  $A_{d,s} \subseteq A_{m,s+1}$ , then  $A_{m,s+1} = A_m$ .