Embeddings of Computable Structures

Asher M. Kach

(Joint work with Wesley Calvert, Oscar Levin, Joe Miller, and Reed Solomon)

University of Connecticut - Storrs
http://www.math.uconn.edu/~kach

University of Notre Dame
15 January 2009
Outline

1. Introduction and Motivation

2. Embeddings within Universal Classes
   - Directed Graphs
   - All Universal Structures

3. Embeddings within Non-Universal Classes
   - Linear Orders
   - Ordered Fields
   - Trees (Viewed as Posets)
   - Equivalence Relations
   - Boolean Algebras

4. Embeddings as Substructures

5. Summary and Questions
If \( \tau \) is an infinite order type, then \( \tau \) has a subset of order type \( \omega \) or \( \omega^* \).
Background

**Theorem**

If \( \tau \) is an infinite order type, then \( \tau \) has a subset of order type \( \omega \) or \( \omega^* \).

**Theorem (Denisov [3]; Tennenbaum [8])**

There is a computable presentation of the order type \( \omega + \omega^* \) having no computable subset of order type \( \omega \) or \( \omega^* \).
Theorem

If \( \tau \) is an infinite order type, then \( \tau \) has a subset of order type \( \omega \) or \( \omega^* \).

Theorem (Denisov [3]; Tennenbaum [8])

There is a computable presentation of the order type \( \omega + \omega^* \) having no computable subset of order type \( \omega \) or \( \omega^* \).

Theorem (Lerman [7]; Rosenstein [8])

If \( \mathcal{L} \) is a computable presentation of an infinite order type \( \tau \), then \( \mathcal{L} \) has a computable subset of order type \( \omega, \omega^*, \omega + \omega^*, \) or \( \omega + \eta \cdot \zeta + \omega^* \).

Moreover, all of these order types are necessary.
Theorem (Denisov [3]; Tennenbaum [8])

There is a computable presentation of the order type $\omega + \omega^*$ having no computable subset of order type $\omega$ or $\omega^*$.

Proof.

Construct a computable presentation of the order type $\omega + \omega^*$ meeting, for each $e$, the following requirement $R_e$.

$$R_e: \text{If } W_e \text{ is infinite, then } W_e \nsubseteq \omega \text{ and } W_e \nsubseteq \omega^*.$$

Meet $R_e$ by putting one element of $W_e$ into $\omega$ and one element into $\omega^*$. To facilitate this, maintain a virtual fence indicating the current boundary between $\omega$ and $\omega^*$. 
Theorem (Denisov [3]; Tennenbaum [8])

There is a computable presentation of the order type $\omega + \omega^*$ having no computable subset of order type $\omega$ or $\omega^*$.

Proof.

Construct a computable presentation of the order type $\omega + \omega^*$ meeting, for each $e$, the following requirement $R_e$.

$R_e$: If $W_e$ is infinite, then $W_e \not\subseteq \omega$ and $W_e \not\subseteq \omega^*$.

Meet $R_e$ by putting one element of $W_e$ into $\omega$ and one element into $\omega^*$. To facilitate this, maintain a virtual fence indicating the current boundary between $\omega$ and $\omega^*$.

Start 0 2 4 ... ... 5 3 1
Constructing $\omega + \omega^*$

**Theorem (Denisov [3]; Tennenbaum [8])**

There is a computable presentation of the order type $\omega + \omega^*$ having no computable subset of order type $\omega$ or $\omega^*$.

**Proof.**

Construct a computable presentation of the order type $\omega + \omega^*$ meeting, for each $e$, the following requirement $R_e$.

$R_e$: If $W_e$ is infinite, then $W_e \not\subseteq \omega$ and $W_e \not\subseteq \omega^*$.

Meet $R_e$ by putting one element of $W_e$ into $\omega$ and one element into $\omega^*$. To facilitate this, maintain a virtual *fence* indicating the current boundary between $\omega$ and $\omega^*$.
Theorem (Denisov [3]; Tennenbaum [8])

There is a computable presentation of the order type $\omega + \omega^*$ having no computable subset of order type $\omega$ or $\omega^*$.

Proof.

Construct a computable presentation of the order type $\omega + \omega^*$ meeting, for each $e$, the following requirement $R_e$.

$$R_e: \text{If } W_e \text{ is infinite, then } W_e \not\subseteq \omega \text{ and } W_e \not\subseteq \omega^*.$$  

Meet $R_e$ by putting one element of $W_e$ into $\omega$ and one element into $\omega^*$. To facilitate this, maintain a virtual *fence* indicating the current boundary between $\omega$ and $\omega^*$.

After $2, 4 \in W_0$

$0 \ 2 \ \ldots \ \ldots \ 4 \ 5 \ 3 \ 1$
Theorem (Denisov [3]; Tennenbaum [8])

There is a computable presentation of the order type $\omega + \omega^*$ having no computable subset of order type $\omega$ or $\omega^*$.

Proof.

Construct a computable presentation of the order type $\omega + \omega^*$ meeting, for each $e$, the following requirement $R_e$.

$R_e$: If $W_e$ is infinite, then $W_e \not\subseteq \omega$ and $W_e \not\subseteq \omega^*$.

Meet $R_e$ by putting one element of $W_e$ into $\omega$ and one element into $\omega^*$. To facilitate this, maintain a virtual fence indicating the current boundary between $\omega$ and $\omega^*$.

Work towards $\omega + \omega^*$

0 2 7 9 \ldots \ldots 8 6 4 5 3 1
Constructing $\omega + \omega^*$

**Theorem (Denisov [3]; Tennenbaum [8])**

*There is a computable presentation of the order type $\omega + \omega^*$ having no computable subset of order type $\omega$ or $\omega^*$.***

**Proof.**

Construct a computable presentation of the order type $\omega + \omega^*$ meeting, for each $e$, the following requirement $R_e$.

$R_e$: If $W_e$ is infinite, then $W_e \not\subseteq \omega$ and $W_e \not\subseteq \omega^*$.

Meet $R_e$ by putting one element of $W_e$ into $\omega$ and one element into $\omega^*$. To facilitate this, maintain a virtual *fence* indicating the current boundary between $\omega$ and $\omega^*$.

See $3, 5 \in W_2$  

0 2 7 9 . . .  

... 8 6 4 5 3 1
Theorem (Denisov [3]; Tennenbaum [8])

There is a computable presentation of the order type $\omega + \omega^*$ having no computable subset of order type $\omega$ or $\omega^*$.

Proof.

Construct a computable presentation of the order type $\omega + \omega^*$ meeting, for each $e$, the following requirement $R_e$.

$R_e$: If $W_e$ is infinite, then $W_e \not\subseteq \omega$ and $W_e \not\subseteq \omega^*$.

Meet $R_e$ by putting one element of $W_e$ into $\omega$ and one element into $\omega^*$. To facilitate this, maintain a virtual fence indicating the current boundary between $\omega$ and $\omega^*$.

See $6, 8 \in W_2$

$0 \ 2 \ 7 \ 9 \ldots$

$\ldots \ 8 \ 6 \ 4 \ 5 \ 3 \ 1$
Constructing $\omega + \omega^*$

**Theorem (Denisov [3]; Tennenbaum [8])**

There is a computable presentation of the order type $\omega + \omega^*$ having no computable subset of order type $\omega$ or $\omega^*$.

**Proof.**

Construct a computable presentation of the order type $\omega + \omega^*$ meeting, for each $e$, the following requirement $R_e$.

$R_e$: If $W_e$ is infinite, then $W_e \not\subseteq \omega$ and $W_e \not\subseteq \omega^*$.

Meet $R_e$ by putting one element of $W_e$ into $\omega$ and one element into $\omega^*$. To facilitate this, maintain a virtual fence indicating the current boundary between $\omega$ and $\omega^*$.

After $6, 8 \in W_2$

\[0 \ 2 \ 7 \ 9 \ 8 \ldots \]

\[\ldots \ 6 \ 4 \ 5 \ 3 \ 1\]
Theorem (Denisov [3]; Tennenbaum [8])

There is a computable presentation of the order type $\omega + \omega^*$ having no computable subset of order type $\omega$ or $\omega^*$.

Proof.

Construct a computable presentation of the order type $\omega + \omega^*$ meeting, for each $e$, the following requirement $R_e$.

$R_e$: If $W_e$ is infinite, then $W_e \not\subseteq \omega$ and $W_e \not\subseteq \omega^*$.

Meet $R_e$ by putting one element of $W_e$ into $\omega$ and one element into $\omega^*$. To facilitate this, maintain a virtual fence indicating the current boundary between $\omega$ and $\omega^*$.

See $7, 9 \in W_1$

\[ 0 \ 2 \ 7 \ 9 \ 8 \ldots \]

\[ \ldots \ 6 \ 4 \ 5 \ 3 \ 1 \]
Constructing $\omega + \omega^*$

**Theorem (Denisov [3]; Tennenbaum [8])**

*There is a computable presentation of the order type $\omega + \omega^*$ having no computable subset of order type $\omega$ or $\omega^*$."

**Proof.**

Construct a computable presentation of the order type $\omega + \omega^*$ meeting, for each $e$, the following requirement $R_e$.

$R_e$: If $\mathcal{W}_e$ is infinite, then $\mathcal{W}_e \not\subseteq \omega$ and $\mathcal{W}_e \not\subseteq \omega^*$.

Meet $R_e$ by putting one element of $\mathcal{W}_e$ into $\omega$ and one element into $\omega^*$. To facilitate this, maintain a virtual *fence* indicating the current boundary between $\omega$ and $\omega^*$.

After $7, 9 \in \mathcal{W}_1$

\[
\begin{array}{cccccccccccc}
0 & 2 & 7 & \ldots & & & & & & & & \ldots & 9 & 8 & 6 & 4 & 5 & 3 & 1 \\
\end{array}
\]
The Questions

Question (#1)
Are there computable order types $\tau_1$ and $\tau_2$ having computable presentations $\mathcal{L}_1$ and $\mathcal{L}_2$ such that $\mathcal{L}_1$ does not computably embed into $\mathcal{L}_2$?

Question (#2)
Are there computable order types $\tau_1$ and $\tau_2$ such that $\mathcal{L}_1$ does not computably embed into $\mathcal{L}_2$ for any computable presentations $\mathcal{L}_1$ and $\mathcal{L}_2$?
Remark

Of course, there is no reason attention should be restricted to the context of linear orders.

Question

If $\mathcal{C}$ is a class of computable algebraic structures, are there $\overline{S}_1, \overline{S}_2 \in \mathcal{C}$ such that $S_1$ does not computably embed into $S_2$ for any computable presentations $S_1$ and $S_2$?
Outline

1. Introduction and Motivation

2. Embeddings within Universal Classes
   - Directed Graphs
   - All Universal Structures

3. Embeddings within Non-Universal Classes
   - Linear Orders
   - Ordered Fields
   - Trees (Viewed as Posets)
   - Equivalence Relations
   - Boolean Algebras

4. Embeddings as Substructures

5. Summary and Questions
Emmbeddings of Directed Graphs

**Theorem (Kach and Miller [6])**

If $\mathcal{C}$ is the class of computable directed graphs, then there are structures $S_1$ and $S_2$ in $\mathcal{C}$ such that for no hyperarithmetic presentations of $S_1$ and $S_2$ does $S_1$ hyperarithmetically embed into $S_2$.

**Definition**

If $T \subseteq \omega^{<\omega}$ is any tree, define $G_T$ to be the directed graph whose universe is

$$\{z_{\sigma} : \sigma \in T\} \cup \{x_{\sigma \upharpoonright i,0}, \cdots, x_{\sigma \upharpoonright i,i}, y_{\sigma \upharpoonright i,0}, \cdots, y_{\sigma \upharpoonright i,i} : \sigma \upharpoonright i \in T\}$$

and whose edge relations include $E(z_{\sigma}, x_{\sigma \upharpoonright i,0})$, $E(z_{\sigma}, y_{\sigma \upharpoonright i,0})$, $E(x_{\sigma \upharpoonright i,j}, x_{\sigma \upharpoonright i,j+1})$, $E(y_{\sigma \upharpoonright i,j}, y_{\sigma \upharpoonright i,j+1})$, $E(x_{\sigma \upharpoonright i,i}, z_{\sigma \upharpoonright i})$, and $E(y_{\sigma \upharpoonright i,i}, z_{\sigma \upharpoonright i})$ for $\sigma \in T$, $\sigma \upharpoonright i \in T$, and $0 \leq j < i$. 
Example

If $T$ is as on the left, then $G_T$ is as on the right.

Proof. Let $S_1$ be the graph of exactly one (directed) infinite path and let $S_2$ be the graph $G_T$ where $T \subseteq \omega < \omega$ is a computable tree with infinite paths but no hyperarithmetic paths. Then $S_1$ classically embeds into $S_2$, but there cannot be hyperarithmetic presentations with a hyperarithmetic embedding. For if there was a hyperarithmetic embedding $\pi$ between hyperarithmetic presentations, there would be a hyperarithmetic path in $T$. 
Example
If $T$ is as on the left, then $G_T$ is as on the right.

Proof.
Let $S_1$ be the graph of exactly one (directed) infinite path and let $S_2$ be the graph $G_T$ where $T \subseteq \omega^\omega$ is a computable tree with infinite paths but no hyperarithmetic paths.

Then $S_1$ classically embeds into $S_2$, but there cannot be hyperarithmetic presentations with a hyperarithmetic embedding. For if there was a hyperarithmetic embedding $\pi$ between hyperarithmetic presentations, there would be a hyperarithmetic path in $T$. 
Example

If $T$ is as on the left, then $\mathcal{G}_T$ is as on the right.

Proof.

Let $\overline{S}_1$ be the graph of exactly one (directed) infinite path and let $\overline{S}_2$ be the graph $\mathcal{G}_T$ where $T \subseteq \omega^\omega$ is a computable tree with infinite paths but no hyperarithmetic paths.

Then $\overline{S}_1$ classically embeds into $\overline{S}_2$, but there cannot be hyperarithmetic presentations with a hyperarithmetic embedding. For if there was a hyperarithmetic embedding $\pi$ between hyperarithmetic presentations, there would be a hyperarithmetic path in $T$. 
Example

If $T$ is as on the left, then $G_T$ is as on the right.

Proof.

Let $S_1$ be the graph of exactly one (directed) infinite path and let $S_2$ be the graph $G_T$ where $T \subseteq \omega^{<\omega}$ is a computable tree with infinite paths but no hyperarithmetic paths.

Then $S_1$ classically embeds into $S_2$, but there cannot be hyperarithmetic presentations with a hyperarithmetic embedding. For if there was a hyperarithmetic embedding $\pi$ between hyperarithmetic presentations, there would be a hyperarithmetic path in $T$. 

\[ \Box \]
Example

If $T$ is as on the left, then $G_T$ is as on the right.

Proof.

Let $\mathcal{S}_1$ be the graph of exactly one (directed) infinite path and let $\mathcal{S}_2$ be the graph $G_T$ where $T \subseteq \omega^{<\omega}$ is a computable tree with infinite paths but no hyperarithmetic paths.

Then $\mathcal{S}_1$ classically embeds into $\mathcal{S}_2$, but there cannot be hyperarithmetic presentations with a hyperarithmetic embedding. For if there was a hyperarithmetic embedding $\pi$ between hyperarithmetic presentations, there would be a hyperarithmetic path in $T$. 

Asher M. Kach (UConn)
Corollary (With Hirschfeldt, Khoussainov, Shore, and Slinko [4])

If $\mathcal{C}$ is the class of computable

- Directed graphs,
- Undirected graphs,
- Commutative rings,
- Two-step nilpotent groups,
- Integral domains, or
- Commutative semigroups,

then there are structures $\overline{S}_1$ and $\overline{S}_2$ in $\mathcal{C}$ such that for no hyperarithmetic presentations $S_1$ and $S_2$ does $S_1$ hyperarithmetically embed into $S_2$. 
Outline

1. Introduction and Motivation

2. Embeddings within Universal Classes
   - Directed Graphs
   - All Universal Structures

3. Embeddings within Non-Universal Classes
   - Linear Orders
   - Ordered Fields
   - Trees (Viewed as Posets)
   - Equivalence Relations
   - Boolean Algebras

4. Embeddings as Substructures

5. Summary and Questions
Remark

Of particular (and natural) interest are the special cases when $\mathcal{L}_1 = \eta$ and $\mathcal{L}_1 = \omega^*$. 
Remark
Of particular (and natural) interest are the special cases when $\mathcal{L}_1 = \eta$ and $\mathcal{L}_1 = \omega^*$. 

Proposition

*The order type* $\eta$ *is computably categorical.*

*The standard presentation of the order type* $\omega^*$ *computably embeds into any computable presentation of the order type* $\omega^*$. 
There is a computable non-scattered order type $\tau_\eta$ that is intrinsically hyperarithmetically scattered, i.e., there is a computable order type $\tau_\eta$ into which the rationals embed, but for which the rationals do not hyperarithmetically embed into any hyperarithmetic presentation of $\tau_\eta$.

Let $\tau_T$ be the order type $\tau_\epsilon$, where $\epsilon$ denotes the empty string.
Embeddings of Linear Orders (Non-Scattered)

**Definition**

\[
\tau_\sigma = \omega + f(\sigma) + \zeta + \left( \sum_{i \in \omega \cap i \in T} \tau_\sigma \cap i \right)^* + \left( \sum_{i \in \omega \cap i \in T} \tau_\sigma \cap i \right) + \zeta + f(\sigma) + \omega^*.
\]
Embeddings of Linear Orders (Non-Scattered)

**Definition**

\[ \tau_\sigma = \omega + f(\sigma) + \zeta + \left( \sum_{i \in \omega} \tau_\sigma \cap i \right)^* + \left( \sum_{i \in \omega} \tau_\sigma \cap i \right) + \zeta + f(\sigma) + \omega^*. \]

\[ \tau_T = \tau_\epsilon = \omega + f(\epsilon) + \zeta + \left( \tau_0 + \tau_1 \right)^* + \left( \tau_0 + \tau_1 \right) + \zeta + f(\epsilon) + \omega^*. \]
Embeddings of Linear Orders (Non-Scattered)

**Definition**

\[ \tau_\sigma = \omega + f(\sigma) + \zeta + \left( \sum_{i \in \omega} \tau_{\sigma \land i} \right)^* + \left( \sum_{i \in \omega} \tau_{\sigma \land i} \right)^* + \zeta + f(\sigma) + \omega^*. \]

\[ \tau_T = \tau_\epsilon = \omega + f(\epsilon) + \zeta + \left( \tau_0 + \tau_1 \right)^* + \left( \tau_0 + \tau_1 \right)^* + \zeta + f(\epsilon) + \omega^*. \]

\[ \tau_0 = \omega + f(0) + \zeta + \left( \right)^* + \left( \right)^* + \zeta + f(0) + \omega^*. \]
Embeddings of Linear Orders (Non-Scattered)

**Definition**

\[
\tau_\sigma = \omega + f(\sigma) + \zeta + \left( \sum_{i\in\omega} \tau_\sigma \cap i \right)^* + \left( \sum_{i\in\omega} \tau_\sigma \cap i \right) + \zeta + f(\sigma) + \omega^*.
\]

\[
\tau_T = \tau_\varepsilon = \omega + f(\varepsilon) + \zeta + \left( \tau_0 + \tau_1 \right)^* + \left( \tau_0 + \tau_1 \right) + \zeta + f(\varepsilon) + \omega^*
\]

\[
\tau_0 = \omega + f(0) + \zeta + \left( \right)^* + \left( \right) + \zeta + f(0) + \omega^*
\]

\[
\tau_1 = \omega + f(1) + \zeta + \left( \tau_{10} + \tau_{12} \right)^* + \left( \tau_{10} + \tau_{12} \right) + \zeta + f(1) + \omega^*
\]
Theorem (Kach and Miller [6])

There is a computable non-scattered order type $\tau_\eta$ that is intrinsically hyperarithmetically scattered.

Proof.

Let $T \subseteq \omega^{<\omega}$ be a computable tree with infinite paths but no hyperarithmetic paths. Then $\tau_T$ is computable as the definition of $\tau_\sigma$ depended only on knowing whether $\sigma \upharpoonright i \in T$. Also $\tau_T$ is non-scattered as $T$ had an infinite path. Finally $\tau_T$ is intrinsically hyperarithmetically scattered as a $(\Delta^0_4(\mathcal{L}_T) \oplus \pi)$-computable path in $T$ can be recovered from a hyperarithmetic embedding $\pi : \eta \rightarrow \mathcal{L}_T$. $\square$
**Theorem (Kach and Miller [6])**

There is a computable, non-well-ordered order type $\tau_{\omega^*}$ that is intrinsically computably well-ordered, i.e., there is a computable order type $\tau_{\omega^*}$ into which the negative integers embed, but for no computable presentation of $\tau_{\omega^*}$ do the negative integers computably embed.

**Definition (Montalbán)**

If $F : \omega \to \omega$ is any function, define $\tau_F$ to be the order type

$$\cdots + \omega^n \cdot F(n) + \cdots + \omega^2 \cdot F(2) + \omega \cdot F(1) + F(0)$$

**Proof (Sketch).**

Show that, for carefully chosen $\emptyset^{(\omega)}$-computable functions $F$, the order type $\tau_F$ is not computable but $\tau_{\omega^*} := \omega^\omega + \tau_F$ is computable.
Embeddings of Linear Orders (Non-Well-Ordered) (Continued...)

Proof (Sketch).

Demonstrate that a function $F : \omega \to \omega$ is $\Delta^0_{2n+1}$-limit infimum (equivalently $\Delta^0_{2n+2}$-limitwise monotonic) if and only if the linear order $\omega^\omega + \mathcal{L}_F$ is computable. Note that the forwards direction is difficult; the backwards direction is relatively straightforward.

Also demonstrate the existence of such a function $F$ for which $\mathcal{L}_F$ is not computable by diagonalizing against all linear orders that look “like” $\mathcal{L}_F$ for some function $F$. Note that higher priority strategies have access to more powerful oracles, and can thus determine the success or failure of lower priority strategies.
Corollary

For each computable ordinal $\alpha$, there is a computable, non-well-ordered order type that is intrinsically $\emptyset(\alpha)$-computably well-ordered.

Proof.

Relativizing the construction of $\tau_{\omega^*}$, build a $\emptyset(\alpha)$-computable presentation that is intrinsically $\emptyset(\alpha)$-computably well-ordered. Then $\omega^\alpha \cdot \tau_{\omega^*}$ is computable and still intrinsically $\emptyset(\alpha)$-computably well-ordered.
Theorem (Harrison)

If a presentation of a computable, non-well-ordered linear order has no hyperarithmetic descending sequence, then it has the form $\omega_1^{CK} (1 + \eta) + \beta$ for some computable ordinal $\beta$.

Corollary

There is no computable, non-well-ordered linear order that is intrinsically hyperarithmetically well-ordered.
Theorem (Calvert, Kach, Levin, and Solomon [2])

If \( \mathcal{C} \) is the class of computable ordered fields, then there are structures \( \overline{S}_1 \) and \( \overline{S}_2 \) in \( \mathcal{C} \) such that for no hyperarithmetic presentations \( S_1 \) and \( S_2 \) does \( S_1 \) hyperarithmetically embed into \( S_2 \).
Theorem (Calvert, Kach, Levin, and Solomon [2])

If \( \mathcal{C} \) is the class of computable ordered fields, then there are structures \( \bar{S}_1 \) and \( \bar{S}_2 \) in \( \mathcal{C} \) such that for no hyperarithmetic presentations \( S_1 \) and \( S_2 \) does \( S_1 \) hyperarithmetically embed into \( S_2 \).

Definition

An ordered field \( \mathcal{F} = (F : +, \cdot, 0, 1, <) \) is a field \( (F : +, \cdot, 0, 1) \) with an order \( < \) that behaves as it should.
Embeddings of Ordered Fields

Theorem (Calvert, Kach, Levin, and Solomon [2])

If $C$ is the class of computable ordered fields, then there are structures $S_1$ and $S_2$ in $C$ such that for no hyperarithmetic presentations $S_1$ and $S_2$ does $S_1$ hyperarithmetically embed into $S_2$.

Definition

An ordered field $\mathcal{F} = (F : +, \cdot, 0, 1, <)$ is a field $(F : +, \cdot, 0, 1)$ with an order $<$ that behaves as it should.

Definition

If $\tau$ is any order type, define $\overline{\mathcal{F}}_{\tau}$ to be the ordered field whose universe is generated by $\mathbb{Q} \cup \{x_i : i \in \tau\}$ and whose order is generated by $x_i^m <_{\overline{\mathcal{F}}_{\tau}} x_j^n$ if $i <_{\tau} j$ as well as $q < x_i$. 
Theorem (Calvert, Kach, Levin, and Solomon [2])

If $\mathcal{C}$ is the class of computable ordered fields, then there are structures $\overline{S}_1$ and $\overline{S}_2$ in $\mathcal{C}$ such that for no hyperarithmetic presentations $S_1$ and $S_2$ does $S_1$ hyperarithmetically embed into $S_2$.

Proof.

The ordered fields $\overline{F}_\eta$ and $\overline{F}_\tau_\eta$ suffice. In order to (somewhat) effectively recover an embedding of $\eta$ into $\tau_\eta$ from a hyperarithmetic embedding of $\overline{F}_\eta$ into $\overline{F}_\tau_\eta$, use Archimedean power classes.
Definition

A tree is a partial order \((T : \preceq)\) with least element such that for all \(x \in T\), the set \(\{y \in T : y \preceq x\}\) is finite and linearly ordered.

Theorem (Binns, Kjos-Hanssen, Lerman, Schmerl, Solomon [1])

There is an infinite computable binary branching tree \(S\) with no isolated paths such that any nontrivial self-embedding computes \(0''\).

Corollary

If \(\mathcal{C}\) is the class of computable trees, then there are structures \(\overline{S}_1\) and \(\overline{S}_2\) in \(\mathcal{C}\) such that for no computable presentations \(S_1\) and \(S_2\) does \(S_1\) computably embed into \(S_2\).
Embeddings of Equivalence Relations

**Theorem (Calvert, Kach, Levin, and Solomon [2])**

If \( C \) is the class of computable equivalence structures, for all structures \( \mathcal{E}_1 \) and \( \mathcal{E}_2 \) in \( C \), there are computable presentations \( \mathcal{E}_1 \) and \( \mathcal{E}_2 \) such that \( \mathcal{E}_1 \) computably embeds into \( \mathcal{E}_2 \).

**Proof.**

If \( \mathcal{E}_2 \) has bounded character, then \( \mathcal{E}_1 \) has bounded character and the result is immediate.

If \( \mathcal{E}_2 \) has unbounded character but only finitely many infinite equivalence classes, a finite injury argument suffices to build computable presentations \( \mathcal{E}_1 \) and \( \mathcal{E}_2 \) and a computable embedding between them.

Finally, if \( \mathcal{E}_2 \) has infinitely many infinite equivalence classes, then the addition of countably more infinite equivalence classes provides a place for the image of \( \mathcal{E}_1 \).
Theorem (Calvert, Kach, Levin, and Solomon [2])

If $C$ is the class of computable Boolean algebras, for all structures $B_1$ and $B_2$ in $C$, there are computable presentations $B_1$ and $B_2$ such that $B_1$ computably embeds into $B_2$.

Proof.

If $B_2$ is superatomic, then $B_1$ is superatomic and the result is immediate.

If $B_2$ is not superatomic, then there is a computable ordinal $\alpha$ such that

$$B_2 \cong B_2 \oplus \text{IntAlg}(\omega^\alpha(1 + \eta))$$

as a consequence of work by Ketonen. There is a nice presentation of the latter into which the countable atomless Boolean algebra (and thus $B_1$) computably embeds.
Outline

1. Introduction and Motivation

2. Embeddings within Universal Classes
   - Directed Graphs
   - All Universal Structures

3. Embeddings within Non-Universal Classes
   - Linear Orders
   - Ordered Fields
   - Trees (Viewed as Posets)
   - Equivalence Relations
   - Boolean Algebras

4. Embeddings as Substructures

5. Summary and Questions
Remark

Thus far, we have been considering pairs of structures $\overline{S}_1$ and $\overline{S}_2$ such that $\overline{S}_1$ classically embeds into $\overline{S}_2$. Until now, we have concerned ourselves with attempting to make sure no embedding is computable for any computable presentations $S_1$ and $S_2$.

If this is possible, it is natural to ask what further restrictions are necessary before such a phenomena is no longer possible.
Remark

Thus far, we have been considering pairs of structures $\overline{S}_1$ and $\overline{S}_2$ such that $\overline{S}_1$ classically embeds into $\overline{S}_2$. Until now, we have concerned ourselves with attempting to make sure no embedding is computable for any computable presentations $S_1$ and $S_2$.

If this is possible, it is natural to ask what further restrictions are necessary before such a phenomena is no longer possible.

By restricting the computable presentations to fixed computable presentations but allowing the embedding to vary, we arrive at Question #1.
Remark

Thus far, we have been considering pairs of structures $\overline{S}_1$ and $\overline{S}_2$ such that $\overline{S}_1$ classically embeds into $\overline{S}_2$. Until now, we have concerned ourselves with attempting to make sure *no* embedding is computable for *any* computable presentations $S_1$ and $S_2$.

If this is possible, it is natural to ask what further restrictions are necessary before such a phenomena is no longer possible.

By restricting the computable presentations to *fixed* computable presentations but allowing the embedding to vary, we arrive at Question #1.

By restricting the embedding to a *fixed* embedding but allowing the presentations to vary, we arrive at new questions.
More Questions (Cont...)

Question (#3)
If \( \mathcal{C} \) is a class of computable algebraic structures, are there structures \( \overline{S}_1 \) and \( \overline{S}_2 \) in \( \mathcal{C} \) and presentations \( S_1 \subseteq S_2 \) such that for no automorphism \( \pi : S_2 \rightarrow S_2 \) is \( \pi(S_1) \) computably enumerable?

Question (#4)
If \( \mathcal{C} \) is a class of computable algebraic structures, are there structures \( \overline{S}_1 \) and \( \overline{S}_2 \) in \( \mathcal{C} \) and presentations \( S_1 \subseteq S_2 \) such that for no automorphism \( \pi : S_2 \rightarrow S_2 \) is \( \pi(S_1) \) computable?
Remark

Note that a positive answer to Question #2 trivially implies a positive answer to Question #3 and Question #4.

Remark

If a class of computable algebraic structures has a positive answer to Question #3 or Question #4 but a negative answer to Question #2, then it “is not” possible to code into isomorphism types but it “is” possible to code within how an isomorphism type fits inside another isomorphism type in a fixed manner.
Proposition (Calvert, Kach, Levin, and Solomon [2])

If \( \mathcal{C} \) is the class of computable equivalence structures, Boolean algebras, or abelian \( p \)-groups (of length below \( \omega^2 \)), then \( \mathcal{C} \) has a positive answer to Question \#3 and a negative answer to Question \#2.

Proof.

For equivalence structures, let \( \overline{S}_1 \) and \( \overline{S}_2 \) be the equivalence structure with exactly one class of every finite size and no infinite equivalence class. Have \( S_1 \) be the substructure of \( S_2 \) where the class of size \( i \) in \( S_1 \) is a subset of the class of size \( 2i \) in \( S_2 \) if \( i \in S \) and a subset of the class of size \( 2i + 1 \) otherwise.

For Boolean algebras, let \( \overline{S}_1 \) be \( \text{IntAlg}(\omega) \) and \( \overline{S}_2 \) be \( \text{IntAlg}(\omega^2) \). Have \( S_1 \) be a substructure of \( S_2 \) such that there is an atom of \( S_1 \) bounding exactly \( i \) atoms in \( S_2 \) if and only if \( i \in S \).
Outline

1. Introduction and Motivation
2. Embeddings within Universal Classes
   - Directed Graphs
   - All Universal Structures
3. Embeddings within Non-Universal Classes
   - Linear Orders
   - Ordered Fields
   - Trees (Viewed as Posets)
   - Equivalence Relations
   - Boolean Algebras
4. Embeddings as Substructures
5. Summary and Questions
The (computable) embedding phenomena happens when $C$ is the class of computable
- Directed graphs (or any universal class).
- Linear orders.
- Ordered fields.
- Trees.

The embedding phenomena fails to happen when $C$ is the class of computable
- Equivalence structures.
- Boolean algebras.
Questions

Question

Does Question #2 have a positive answer when $C$ is the class of computable fields? When $C$ is the class of computable abelian $p$-groups?

Does Question #2 have a positive answer at the hyperarithmetic level when $C$ is the class of computable trees?
Questions

Question

Does Question \#2 have a positive answer when $C$ is the class of computable fields? When $C$ is the class of computable abelian $p$-groups?

Does Question \#2 have a positive answer at the hyperarithmetic level when $C$ is the class of computable trees?

Question

(Conjecture) Is there a computable, non-scattered linear order that is intrinsically computably well-ordered?
Questions

Question
Does Question \#2 have a positive answer when $C$ is the class of computable fields? When $C$ is the class of computable abelian $p$-groups?

Does Question \#2 have a positive answer at the hyperarithmetic level when $C$ is the class of computable trees?

Question
(Conjecture) Is there a computable, non-scattered linear order that is intrinsically computably well-ordered?

Question
Is there, for every computable order type $\tau_1$, a computable order type $\tau_2$ such that for no computable presentations $\mathcal{L}_1$ and $\mathcal{L}_2$ does $\mathcal{L}_1$ computably embed into $\mathcal{L}_2$?
Stephen Binns, Bjorn Kjos-Hanssen, Manuel Lerman, James H. Schmerl, and Reed Solomon.
Self embeddings of computable trees.

Wesley Calvert, Asher M. Kach, Oscar Levin, and D. Reed Solomon.
Embeddings of computable structures.
In preparation.

S. S. Gončarov and A. T. Nurtazin.
Constructive models of complete decidable theories.

Denis R. Hirschfeldt, Bakhadyr Khoussainov, Richard A. Shore, and Arkadii M. Slinko.
Degree spectra and computable dimensions in algebraic structures.

Denis R. Hirschfeldt and Richard A. Shore.
Combinatorial principles weaker than Ramsey's theorem for pairs.

Asher M. Kach and Joseph S. Miller.
Embeddings of computable linear orders.
In preparation.

Manuel Lerman.
On recursive linear orderings.

Joseph G. Rosenstein.
*Linear orderings*, volume 98 of *Pure and Applied Mathematics*.