Two-to-One Structures

Douglas Cenzer, Department of Mathematics
University of Florida, Gainesville, FL 32611
email: cenzer@math.ufl.edu

Valentina Harizanov, Department of Mathematics
George Washington University, Washington, DC 20052
email: harizanv@gwu.edu

Jeffrey B. Remmel, Department of Mathematics
University of California-San Diego, La Jolla, CA 92093
email: jremmel@ucsd.edu

Abstract

We investigate computability-theoretic properties of computable structures with single unary functions f such that, for every x in the image, $f^{-1}(x)$ has exactly two elements, which we call 2:1 structures. We also investigate structures for which $f^{-1}(x)$ has either exactly two or zero elements, which we call (2,0):1 structures. In particular, we are interested in the complexity of isomorphisms between these structures. We prove that a computable 2:1 structure $\mathcal{A}$ is computably categorical if and only if $\mathcal{A}$ has only finitely many $\mathbb{Z}$-chains. We show that every computable 2:1 structure is Δ^0_2-categorical. We further investigate computable and higher level categoricity of various natural subclasses of (2,0):1 structures, including highly computable and locally finite strufctures.

Keywords: computability theory, two-to-one functions, injections, effective categoricity, locally finite structures, trees, chains

1 Introduction and Preliminaries

Computable model theory uses the concepts and methods of computability theory to explore algorithmic content of constructions in various areas of classical mathematics. In this paper we are interested in the complexity of isomorphisms between a computable structure and its isomorphic copies. The main notion in this area of investigation is that of computable categoricity. We say that a computable structure $\mathcal{A}$ is *computably categorical* if for every computable structure $\mathcal{B}$ isomorphic to $\mathcal{A}$, there exists a computable isomorphism from $\mathcal{A}$ onto $\mathcal{B}$. This

* Cenzer was partially supported by the NSF grant DMS-1101123, Harizanov by the NSF grant DMS-1202328 and by the GWU Columbian College Facilitating Fund.

concept has been part of computable model theory since the mid-1950s. Here we continue our investigation of computable and higher level categoricity begun in [4, 5], where we investigated computable structures with single one-to-one functions. We first review some notation.

Let $\mathbb{N} = \{0, 1, 2, \ldots\}$ denote the natural numbers and $\mathbb{Z} = \{0, \pm 1, \pm 2, \ldots\}$ denote the integers. We let ω denote the order type of $\mathbb{N}$ under the usual ordering and Z denote the order type of $\mathbb{Z}$ under the usual ordering. In what follows, we restrict our attention to countable structures for computable languages. Hence, if a structure is infinite, we can assume that its universe is $\mathbb{N}$. We recall some basic definitions. If $\mathcal{A}$ is a structure with universe A for a language $\mathcal{L}$, then $\mathcal{L}^A$ is the language obtained by expanding $\mathcal{L}$ by constants for all elements of A. The *atomic diagram* of $\mathcal{A}$ is the set of all quantifier-free sentences of $\mathcal{L}^A$ true in $\mathcal{A}$. A structure $\mathcal{A}$ is *computable* if its atomic diagram is computable. We call two structures *computably isomorphic* if there is a computable function that is an isomorphism between them. A computable structure $\mathcal{A}$ is *relatively computably isomorphic* to a possibly noncomputable structure $\mathcal{B}$ if there is an isomorphism between them that is computable in the atomic diagram of $\mathcal{B}$. A computable structure $\mathcal{A}$ is *computably categorical* if every computable structure that is isomorphic to $\mathcal{A}$ is computably isomorphic to $\mathcal{A}$. A computable structure $\mathcal{A}$ is *relatively computably categorical* if every structure that is isomorphic to $\mathcal{A}$ is relatively computably isomorphic to $\mathcal{A}$. A structure $\mathcal{A}$ is relatively computably categorical if and only if $\mathcal{A}$ has a c.e. Scott family consisting of only existential formulas. A *Scott family* for a structure $\mathcal{A}$ is a countable family Ψ of $L_{\omega_1\omega}$-formulas with finitely many fixed parameters from A such that: (i) each finite tuple in $\mathcal{A}$ satisfies some $\psi \in \Psi$; and (ii) if $\overline{a}$, $\overline{b}$ are tuples in $\mathcal{A}$, of the same length, satisfying the same formula in Ψ, then there is an automorphism of $\mathcal{A}$, which maps $\overline{a}$ to $\overline{b}$. See [1] for details.

Similar definitions arise for other naturally definable classes of structures and their isomorphisms. For example, for any $n \in \omega$, a structure is Δ^0_n if its atomic diagram is Δ^0_n, two Δ^0_n structures are Δ^0_n-*isomorphic* if there is a Δ^0_n isomorphism between them, and a computable structure $\mathcal{A}$ is Δ^0_n-*categorical* if every computable structure that is isomorphic to $\mathcal{A}$ is Δ^0_n- isomorphic to $\mathcal{A}$. The notions and notation of computability theory are standard and as in Soare [11].

Among the simplest nontrivial structures are equivalence structures, i.e., structures of the form $\mathcal{A} = (\omega, E)$ where E is an equivalence relation. The study of the complexity of isomorphisms between computable equivalence structures was carried out by Calvert, Cenzer, Harizanov, and Morozov in [2] where they characterized computably categorical and also relatively Δ^0_2-categorical equivalence structures. Cenzer, LaForte, and Remmel [6] extended this work by investigating equivalence structures in the Ershov hierarchy. More recently, Cenzer, Harizanov and Remmel [3] studied Σ^0_1 and Π^0_1 equivalence structures.

For any equivalence structure $\mathcal{A}$, we let $Fin(\mathcal{A})$ denote the set of elements of $\mathcal{A}$ that lie in finite equivalence classes. For equivalence structures, it is natural to consider the different sizes of the equivalence classes of the elements in $Fin^{\mathcal{A}}$ since such sizes code information into the equivalence relation. The *character* of an equivalence structure $\mathcal{A}$ is the set

$$\chi(\mathcal{A}) = \{(k,n) : n,k > 0 \text{ and } \mathcal{A} \text{ has at least } n \text{ equivalence classes of size } k\}.$$

This set provides a kind of skeleton for $Fin(\mathcal{A})$. Any set $K \subseteq (\omega-\{0\})\times(\omega-\{0\})$ such that for all $n > 0$ and k, $(k, n+1) \in K$ implies $(k,n) \in K$, is called a *character*. We say a character K is *bounded* if there is some finite k_0 such that for all $(k,n) \in K$, $k < k_0$. Khisamiev [9] introduced the concepts of s-functions and s_1-functions as a means of computably approximating the characters of equivalence relations.

Definition 1.1. Let $f : \omega^2 \to \omega$. The function f is an *s-function* if the following hold:
1. for every $i, s \in \omega$, $f(i,s) \leq f(i, s+1)$;
2. for every $i \in \omega$, the limit $m_i = lim_s f(i,s)$ exists.

We say that f is an s_1-*function* if, in addition:
3. for every $i \in \omega$, $m_i < m_{i+1}$.

Calvert, Cenzer, Harizanov and Morozov [2] gave conditions under which a given character K can be the character of a computable equivalence structure. In particular, they observed that if K is a bounded character and $\alpha \leq \omega$, then there is a computable equivalence structure with character K and exactly α infinite equivalence classes. To prove the existence of computable equivalence structures for unbounded characters K, they needed additional information given by s- and s_1-functions. They showed that if K is a Σ^0_2 character, $r < \omega$, and either
(a) there is an s-function f such that

$$(k,n) \in K \Leftrightarrow card(\{i : k = \lim_{s\to\infty} f(i,s)\}) \geq n \text{ or}$$

(b) there is an s_1-function f such that for every $i \in \omega$, $(\lim_s f(i,s), 1) \in K$,
then there is a computable equivalence structure with character K and exactly r infinite equivalence classes.

In [4] and [5], we studied *injection structures*. Here an injection is just a one-to-one (1:1) function and an injection structure $\mathcal{A} = (A, f)$ consists of a set A and an injection $f : A \to A$. $\mathcal{A}$ is a *permutation structure* if f is a permutation of A. Given $a \in A$, the orbit $\mathcal{O}_f(a)$ of a under f is

$$\mathcal{O}_f(a) = \{b \in A : (\exists n \in \mathbb{N})(f^n(a) = b \ \vee \ f^n(b) = a)\}.$$

The order $|a|_f$ of a under f is $card(\mathcal{O}_f(a))$. Clearly, the isomorphism type of a permutation structure $\mathcal{A}$ is determined by the number of orbits of size k for $k = 1, 2, \ldots, \omega$. By analogy with characters of equivalence structures, we define the *character* $\chi(\mathcal{A})$ of an injection structure $\mathcal{A} = (A, f)$ by

$$\chi(\mathcal{A}) = \{(n,k) : \mathcal{A} \text{ has at least } n \text{ orbits of size } k\}.$$

Injection structures (A, f) may have two types of infinite orbits, $\mathbb{Z}$-orbits which are isomorphic to $(\mathbb{Z}, S)$ in which every element is in the range of f, and ω-orbits, which are isomorphic to (ω, S) and have the form $\mathcal{O}_f(a) = \{f^n(a) : n \in \mathbb{N}\}$ for some $a \notin ran(f)$. Thus injection structures are characterized by the number of orbits of size k for each finite k and by the number of orbits of types $\mathbb{Z}$ and ω.

It is clear from the definitions above that any computable injection structure (A, f) will induce a Σ^0_1 equivalence structure (A, E) in which the equivalence classes are simply the orbits of (A, f).

In [4], we investigated algorithmic properties of computable injection structures and their characters, characterized computably categorical injection structures, and showed that they are all relatively computably categorical. We proved that a computable injection structure $\mathcal{A}$ is computably categorical if and only if it has finitely many infinite orbits. We also characterized Δ^0_2-categorical injection structures as those with finitely many orbits of type ω, or with finitely many orbits of type $\mathbb{Z}$. We showed that they coincide with the relatively Δ^0_2-categorical structures. Finally, we proved that every computable injection structure is relatively Δ^0_3-categorical.

In this paper, we consider structures of the form $\mathcal{A} = (A, f)$ where $f : A \to A$ is a function such that $card(f^{-1}(x)) = 2$ for all $x \in A$, which we call 2:1 structures or where $card(f^{-1}(x)) \in \{0, 2\}$ for all x, which we call (2,0):1 structures.

We shall often identify a structure $\mathcal{A} = (A, f)$ with its directed graph $G(A, f)$ which has vertex set A and where the edge set consists of all pairs $(i, f(i))$ for $i \in A$. Given any $a \in A$, we let the orbit of $O_{\mathcal{A}}(a)$ consist of the set of all points in A which lie in the connected component of $G(A, f)$ containing a. Thus $O_{\mathcal{A}}(a) = \{y \in A : (\exists n)(f^n(y) = a) \ \vee \ (\exists m, n)(f^n(y) = f^m(a))\}$.

Let B be the infinite complete binary tree with all edges directed toward the root. In fact, it will be useful for later proofs to have a canonical version of B in mind. We shall think of B as a directed graph on the vertex set $\mathbb{N} - \{0\}$. The root of B will be 1 and the nodes at the height n, will be $2^n, 2^n+1, \ldots, 2^{n+1}-1$. For $n \geq 1$, the $(2k)$-th and $(2k+1)$st nodes at height n will have edges to the k-th element of height $n-1$. Thus the first few levels of the tree B are pictured in Figure 1.

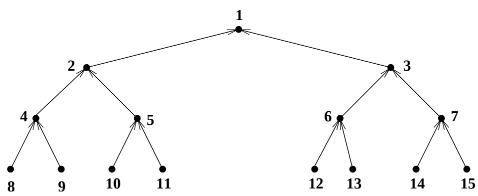

Figure 1: The canonical infinite binary tree B.

It is easy to see that there are two types of orbits in a 2:1 structure $\mathcal{A} = (A, f)$. That is, there are $\mathbb{Z}$-chains as pictured in Figure 2 and there are cycles as pictured in Figure 3. Here a $\mathbb{Z}$-chain in a 2:1 structure consists of $\mathbb{Z}$-chain where there is a copy of the binary tree B attached to each point in the $\mathbb{Z}$-chain. A k-cycle consists of a directed cycle of size k where there is a copy of the binary tree attached to each element in the cycle.

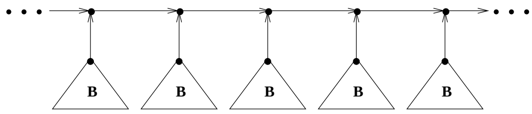

Figure 2: A $\mathbb{Z}$-chain of a 2:1 function.

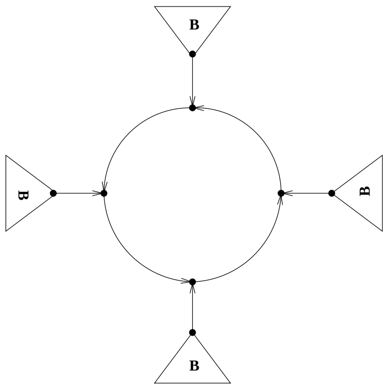

Figure 3: A 4-cycle of a 2:1 function.

The orbits of (2,0):1 structures are similar, except there are now three types of orbits. There are $\mathbb{Z}$-chains, like those pictured in Figure 4, except now a tree B_i, attached to a node of a $\mathbb{Z}$-chain, can be any binary tree with all edges directed to the root. There are k-cycles, like those pictured in Figure 5, except now a tree B_i can be any binary tree with all edges directed toward the root. Finally, there are ω-chains, like those pictured in Figure 6, each consisting of an ω-chain where all but the first element in each orbit has some binary tree B_i attached.

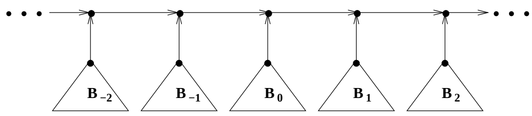

Figure 4: A $\mathbb{Z}$-chain of a (2,0):1 function.

If $\mathcal{A} = (A, f)$ is a 2:1 structure or a (2,0):1 structure and $a \in A$, then we let $tree_{\mathcal{A}}(a) = \{y \in A : (\exists n)(f^n(y) = x)\}$ and $Tree_{\mathcal{A}}(a)$ be the graph whose vertex set is $tree_{\mathcal{A}}(a)$ and whose edge set consists of the set of $(x, f(x))$ such that both x and $f(x)$ are in $tree_{\mathcal{A}}(a)$. We let $tree_{\mathcal{A}}(a, m) = \{y \in A : (\exists n \leq m)(f^n(y) = x)\}$ and $Tree_{\mathcal{A}}(a, m)$ denote the graph of $Tree_{\mathcal{A}}(a)$ restricted to the vertex set $tree_{\mathcal{A}}(a, m)$. In a 2:1 structure $\mathcal{A} = (A, f)$, $Tree_{\mathcal{A}}(a)$ is always isomorphic to the infinite complete binary tree B, unless a is an element of a k-cycle, in which

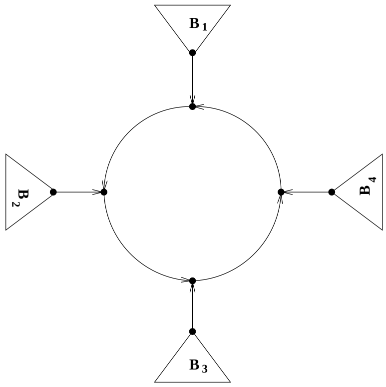

Figure 5: A cycle of a (2,0):1 function.

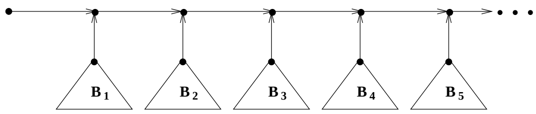

Figure 6: A ω-chain of a (2,0):1 function.

case $tree_{\mathcal{A}}(a) = O_{\mathcal{A}}(a)$. It is clear from the definitions that if $\mathcal{A} = (A, f)$ is a computable structure, then $O_{\mathcal{A}}(a)$ and $tree_{\mathcal{A}}(a)$ are Σ^0_1 sets.

We shall often identify finite sets $S \subseteq \mathbb{N}$ with their canonical indices $can(S)$ where $can(\emptyset) = 0$ and $can(S) = 2^{x_1} + \cdots + 2^{x_k}$ if $S = \{x_1 < \cdots < x_k\}$. Thus when we write that $S = tree_{\mathcal{A}}(a, n)$ is a 2:1 structure or a (2,0):1 structure $\mathcal{A} = (A, f)$, we mean that $can(S) = can(tree_{\mathcal{A}}(a, m))$. In a computable 2:1 structure $\mathcal{A} = (A, f)$, the predicate $S = tree_{\mathcal{A}}(a, m)$ is a computable predicate if S a finite set and $m \geq 1$. That is, in a 2:1 structure, $Tree_{\mathcal{A}}(a, m)$ is always a complete binary tree of height m if a is not an element of a k-cycle where $k \leq m$, so that we can enumerate all the pairs $(i, f(i))$ with $i \in A$ until we find all the elements of $tree_{\mathcal{A}}(a, m)$. If a is part of a k-cycle $(a, f(a), \ldots, f^{k-1}(a))$ with $k \leq m$, then let d_i be the unique element which in not in the cycle such that $f(d_i) = f^i(a)$ for $i = 0, \ldots, k-1$. Then $tree(a, m)$ consists of the elements $a, f(a), \ldots, f_{k-1}(a)$ plus the elements the trees $Tree(d_0, m-1), Tree(f^{k-1}, m-2), Tree(f^{k-2}, m-3), \ldots, Tree(f(a), m-k)$. Thus the sets $tree_{\mathcal{A}}(a, m)$ are uniformly computable. Hence we can effectively decide if $S = tree_{\mathcal{A}}(a, m)$. However, in a computable (2,0):1 structure, $S = tree_{\mathcal{A}}(a, m)$ is a Π^0_1 predicate. That is, $S = tree(a, m)$ if and only if

$$(\forall y \in S)(\exists n \leq m)(f^n(y) = a) \ \wedge \ (\forall y)(\forall n \leq m)(f^n(y) = a \Rightarrow y \in S).$$

In Section 2, we characterize computably categorical 2:1 structures as those that have finitely many $\mathbb{Z}$-chains. We show that every computable 2:1 structure is Δ^0_2-categorical. In Section 3, we investigate natural classes of computable (2,0):1 structures that are computably categorical. In Section 4, we investigate those that are not computably categorical. We show that, while every computable locally finite (2,0):1 structure is Δ^0_3-categorical, every such structure with only finitely many ω-chains is Δ^0_2-categorical.

2 Computable Categoricity of 2:1 Structures

Let $\mathcal{A} = (A, f)$ be a countably infinite 2:1 structure. The *character* $\chi(\mathcal{A})$ of (A, f) is the set of all (k, n) such that either $k = 0$ and $\mathcal{A}$ has $\geq n$ $\mathbb{Z}$-orbits or $k \geq 1$, and $\mathcal{A}$ has $\geq n$ orbits which are k-cycles.

Lemma 2.1. *Let $\mathcal{A} = (A, f)$ be a computable 2:1 structure.*

1. *The predicate "$O_{\mathcal{A}}(a)$ is a k-cycle" is Σ^0_1. and*
2. *the predicate "$O_{\mathcal{A}}(a)$ is a $\mathbb{Z}$-chain" is Π^0_1.*
3. *$\chi(\mathcal{A})$ is a Σ^0_1 set.*

Proof. For (1), note that $O_{\mathcal{A}}(a)$ is a k-cycle if and only if there exists an $n \geq 0$ such that $f^{n+k}(a) = f^n(a)$ and $f^{n+j}(a) \neq f^n(a)$ for $1 \leq j < k$. Thus the predicate "$O_{\mathcal{A}}(a)$ is a k-cycle " is Σ^0_1.

For (2), note that $O_{\mathcal{A}}(a)$ is a $\mathbb{Z}$-chain if and only if it is not the case that there exists $n \geq 0$ and $k > 0$ such that $f^{n+k}(a) = f^n(a)$. Thus the predicate "$O_{\mathcal{A}}(a)$ is a $\mathbb{Z}$-chain" is Π^0_1.

For (3), first note that $\{(0, n) : \mathcal{A} \text{ has } \geq n\ \mathbb{Z} \text{ orbits}\}$ is either $\{0\} \times \omega$ or is $\{0\} \times \{0, 1, \ldots, n\}$ for some finite n, and that this set is computable in either case.

For $k, n > 0$, note that in any k-cycle there is a unique finite set $\{a_0, a_1, \ldots, a_{k-1}\}$ such that $f(a_i) = a_{i+1}$ for $i < k-1$ and $f(a_{k-1}) = a_0$. Thus $\mathcal{A}$ has at least n k-cycles provided that there exist $b_1, b_2, \ldots, b_n$ such that
(i) For each i, $f^k(b_i) = b_i$, and $f^t(b_i) \neq b_i$ for any $t < k$ and
(ii) For each $i \neq j$, and for any $t < k$, $f^t(b_i) \neq b_j$. □

The existence of 2:1 structures with arbitrary Σ^0_1 characters follows from the existence of injection structures with arbitrary Σ^0_1 characters.

Theorem 2.2. *For any Σ^0_1 character K, there is a computable 2:1 structure with character K.*

Proof. By results of [4], there is an injection structure $\mathcal{B} = (\omega, g)$ which has character K. Define a computable function $h : \omega \backslash \{0\} \to \omega$ by having $h(2n+1) = h(2n+2) = n$ for all n. Let $\mathcal{A}$ have universe $A = \omega \times \omega$ and define the two-to-one function f so that $f(b, 0) = (g(b), 0)$ for all b and $f(b, i) = (b, h(i))$ for all $i > 0$. Then $\omega \times \{0\}$ will provide a copy of $\mathcal{B}$ in $\mathcal{A}$ and, for each b, $\{b\} \times \omega$ will be a full binary tree with root $(b, 0)$ where the map f takes any node to its predecessor. □

Theorem 2.3. *A computable 2:1 structure $\mathcal{A} = (A, f)$ is computably categorical if and only if $\mathcal{A}$ has only finitely many $\mathbb{Z}$-chains.*

Proof. Suppose that $\mathcal{A} = (A, f)$ is a computable 2:1 structure with only finitely many $\mathbb{Z}$-chains, and $\mathcal{B} = (B, g)$ is a computable 2:1 structure which is isomorphic to $\mathcal{A}$. Let $Fin(\mathcal{A})$ be the union of all orbits which are k-cycles for some $k \geq 1$ in $\mathcal{A}$. Suppose that $\mathcal{A}$ has t $\mathbb{Z}$-chains and $x_1 < \cdots < x_t$ are representatives from these $\mathbb{Z}$-chains in $\mathcal{A}$. Similarly, let $Fin(\mathcal{B})$ be the union of all orbits which are k-cycles for some $k \geq 1$ in $\mathcal{B}$, and let $y_1 < \cdots < y_t$ be representatives from the t $\mathbb{Z}$-chains in $\mathcal{B}$. Note that since $Fin(\mathcal{A})$ is c.e. and $A - Fin(\mathcal{A}) = \bigcup_{i=1}^{n} O_{\mathcal{A}}(x_i)$ is c.e., it follows that both $Fin(\mathcal{A})$ and $A - Fin(\mathcal{A})$ are computable. Similarly, both $Fin(\mathcal{B})$ and $B - Fin(\mathcal{B})$ are computable.

It is always the case that if $\mathcal{A} = (A, f)$ and $\mathcal{B} = (B, g)$ are computable 2:1 structures and $Fin(\mathcal{A})$ and $Fin(\mathcal{B})$ are isomorphic, then $Fin(\mathcal{A})$ and $Fin(\mathcal{B})$ are computably isomorphic. That is, let $a_0, a_1, a_2, \ldots$ be an enumeration of $Fin(\mathcal{A})$ and $b_0, b_1, b_2, \ldots$ be an enumeration of $Fin(\mathcal{B})$. We can then construct an isomorphism $h : Fin(\mathcal{A}) \rightarrow Fin(\mathcal{B})$ in stages by a standard back-and-forth argument.

The key is to observe that for any a_i, we can compute

$$a_i = f^0(a_i), f(a_i), f^2(a_i), \ldots$$

until we find the least n_i and k_i such that $f^{n_i+k_i}(a_i) = f^{n_i}(a_i)$. Then let $C_i = (f^{n_i}(a_i), \ldots, f^{n_i+k_i-1}(a_i)$. We shall cyclicly rearrange $C_i = (c_0^i, \ldots, c_{k_i-1}^i)$ so that c_0^i is the smallest element of C_i. We shall call C_i the **cycle of $\mathcal{A}$ associated with a_i**. Thus $O_{\mathcal{A}}(a_i)$ will be a k_i-cycle. Then we can search $a_1, a_2, \ldots$ until we find $u_0^i, \ldots, u_{k_i-1}^i$ which are not in C_i such that $f(u_j^i) = c_j^i$ for $j = 0, \ldots, k_i - 1$. It then follows that $Tree_{\mathcal{A}}(u_j^i)$ is isomorphic to the complete binary tree B for $j = 0, \ldots, k_i - 1$. We shall call $\overline{C}_i = \langle (c_1^i, \ldots, c_{k_i-1}^i), (u_1^i, \ldots, u_{k_i-1}^i) \rangle$ the **extended cycle of $\mathcal{A}$ associated with a_i**.

Similarly, for any b_i, we can compute $b_i = g^0(b_i), g(b_i), g^2(b_i), \ldots$ until we find the least m_i and ℓ_i such that $g^{m_i+\ell_i}(b_i) = g^{m_i}(b_i)$. Then let $D_i = (g^{m_i}(b_i), \ldots, g^{m_i+\ell_i-1}(b_i))$. We shall cyclicly rearrange $D_i = (d_0^i, \ldots, d_{\ell_i-1}^i)$ so that d_0^i is the smallest element of D_i. We shall call D_i the **cycle of $\mathcal{B}$ associated with b_i**. Thus the orbit of b_i will be an ℓ_i-cycle. Then we can search $b_1, b_2, \ldots$ until we find $v_0^i, \ldots, v_{\ell_i-1}^i$ which are not in D_i such that $g(v_j^i) = d_j^i$ for $j = 0, \ldots, \ell_i - 1$. It then follows that $Tree_{\mathcal{B}}(v_j^i)$ is isomorphic to the complete binary tree B. We shall call $\overline{D}_i = \langle (d_0^i, \ldots, d_{\ell_i-1}^i), (v_0^i, \ldots, v_{\ell_i-1}^i) \rangle$ the **extended cycle of $\mathcal{B}$ associated with b_i**.

If $a \in A$ and $b \in B$ and both $Tree_{\mathcal{A}}(a)$ and $Tree_{\mathcal{B}}(b)$ are isomorphic to the complete binary tree B, then for all $n \geq 0$, we can define a map what we will call the canonical map $\Theta_{a,b,n} : tree_{\mathcal{A}}(a, n) \rightarrow tree_{\mathcal{B}}(b, n)$ inductively as follows. For $n = 0$, $\Theta_{a,b,0}(a) = b$. Having defined $\Theta_{a,b,n}$, we then extend it to $\Theta_{a,b,n+1}$ so that for each leaf $\ell \in Tree_{\mathcal{A}}(a, n)$, we find the two elements $\ell_1 < \ell_2$ in A such that $f(\ell_1) = f(\ell_2) = \ell$ and we find the two elements $p_1 < p_2$ in $\mathcal{B}$ such that $g(p_1) = g(p_2) = \Theta_{a,b,n}(\ell)$, and then we define $\Theta_{a,b,n+1}(\ell_1) = p_1$ and $\Theta_{a,b,n+1}(\ell_2) = p_2$. We then let $\Theta_{a,b} = \bigcup_{n \geq 0} \Theta_{a,b,n}$ and call this the canonical map from $tree_{\mathcal{A}}(a)$ onto $tree_{\mathcal{B}}(b)$.

Stage 0. First compute the extended cycle $\overline{C}_0 = \langle(c^0_0, \ldots, c^0_{k_0-1}), (u^0_0, \ldots, u^0_{k_0-1})\rangle$ of $\mathcal{A}$ associated with a_0. Then let q_0 be the least j such that the cycle D_j associated with b_j in $\mathcal{B}$ has size k_0, and let $\overline{D}_{q_0} = \langle(d^{q_0}_0, \ldots, d^{q_0}_{k_0-1}), (v^{q_0}_0, \ldots, v^{q_0}_{k_0-1})\rangle$ be the extended cycle of $\mathcal{B}$ associated with b_{q_0}. Then we define h so that $h(c^0_j) = d^{q_0}_j$ and $h(u^0_j) = v^{q_0}_j$ for $j = 0, \ldots, k_0 - 1$. This ensures that h is the canonical bijection from $Tree(u^0_j, 0) = Tree(v^{q_0}_j, 0)$ for $j = 0, \ldots, k_0 - 1$.

If $D_0 = D_{q_0}$, then let $S_0 = \{0\}$ and $T_0 = \{0\}$ and define $\phi_0 : S_0 \to T_0$ by $\phi_0(0) = 0$ and go onto stage 1. Otherwise, compute the extended cycle $\overline{D}^1_0 = \langle(d^0_0, \ldots, d^0_{\ell_0-1}), (v^0_0, \ldots, v^0_{\ell_0-1})\rangle$ of $\mathcal{B}$ associated with b_0. Then let p_0 be the least $j > 0$ such that the cycle C_j associated with a_j in $\mathcal{A}$ has size ℓ_1. Let $\overline{C}^{p_1}_1 = \langle(c^{p_0}_0, \ldots, c^{p_0}_{\ell_0-1}), (u^{p_0}_0, \ldots, u^{p_0}_{\ell_0-1})\rangle$ be the extended cycle of $\mathcal{A}$ associated with a_{p_0}. Then define h so that $h(c^{p_0}_j) = d^0_j$ and $h(u^{p_0}_j) = v^0_j$. This ensures that h is the canonical bijection from $Tree(u^{p_0}_j, 0) = Tree(v^0_j, 0)$ for $j = 0, \ldots, \ell_0 - 1$. Then let $S_0 = \{0, p_0\}$, $T_0 = \{0, q_0\}$ and define $\phi_0 : S_0 \to T_0$ by $\phi_0(0) = q_0$ and $\phi_0(p_0) = 0$.

Stage s+1.
Assume that we have defined sets S_s and T_s and bijection $\phi_s : S_s \to T_s$ and a partial function $h : A \to B$ such that

1. for all $i \leq s$, the cycle C_i associated with a_i is equal to one of the cycles C_j for some $j \in S_s$,
2. for all $i \leq s$, the cycle D_i associated with b_i is equal to one of the cycles D_j for some $j \in T_s$,
3. for all $i, j \in S_s$, the cycles C_i and C_j are distinct if $i \neq j$,
4. for all $i, j \in T_s$, the cycles D_i and D_j are distinct if $i \neq j$,
5. for all $i \in S_s$, C_i and $D_{\phi_s(i)}$ have the same size, and
6. for all $i \in S_s$, if $\phi_s(i) = j$, then h is defined so that if $\overline{C}_i = \langle(c^i_0, \ldots, c^i_{k_i-1}), (u^i_0, \ldots, u^i_{k_i-1})\rangle$ is the extended cycle of $\mathcal{A}$ associated with a_i and $\overline{D}_j = \langle(d^j_0, \ldots, d^j_{k_i-1}), (u^j_0, \ldots, u^j_{k_i-1})\rangle$ is the extended cycle of $\mathcal{B}$ associated with b_j, then $h(c^i_r) = d^j_r$, $h(u^i_r) = v^j_r$, and h restricted to $tree_{\mathcal{A}}(u^i_r, s)$ is equal to the canonical map $\Theta_{u^i_r, v^j_r, s}$ for $r = 0, \ldots, k_i - 1$.

First we extend h so that for all $i \in S_s$, if $\phi_s(i) = j$, then h is defined so that if $\overline{C}_i = \langle(c^i_1, \ldots, c^i_{k_i}), (u^i_1, \ldots, u^i_{k_i})\rangle$ is the extended cycle of $\mathcal{A}$ associated with a_i and $\overline{D}_j = \langle(d^j_0, \ldots, d^j_{k_i-1}), (v^j_0, \ldots, v^j_{k_i-1})\rangle$ is the extended cycle of $\mathcal{B}$ associated with b_j, then $h(c^i_r) = d^j_r$, $h(u^i_r) = v^j_r$, and h restricted to $tree_{\mathcal{A}}(u^i_r, s+1)$ is equal to the canonical map $\Theta_{u^i_r, v^j_r, s+1}$ for $r = 0, \ldots, k_i - 1$.

We then have 4 cases.

Case 1. C_{s+1} is equal to one of the cycles C_i for $i \in S_s$, and D_{s+1} is equal to one of the cycles D_j for $j \in T_s$.

Then let $S_{s+1} = S_s$, $T_{s+1} = T_s$, and $\phi_{s+1} = \phi_s$.

Case 2. C_{s+1} is not equal to one of the cycles C_i for $i \in S_s$, but D_{s+1} is equal to one of the cycles D_j for $j \in T_s$.
Then let $\overline{C}_{s+1} = \langle (c_0^{s+1}, \ldots, c_{k_{s+1}-1}^{s+1}), (u_0^{s+1}, \ldots, u_{k_{s+1}-1}^{s+1}) \rangle$ be the extended cycle of $\mathcal{A}$ associated with a_{s+1}. Then let q_{s+1} be the least q such that D_q is not equal to one of the cycles D_j for $j \in T_s$ and D_q has size k_{s+1}. Let $\overline{D}_{q_{s+1}} = \langle (d_0^{q_{s+1}}, \ldots, d_{k_{s+1}-1}^{q_{s+1}}), (v_0^{q_{s+1}}, \ldots, v_{k_{s+1}-1}^{q_{s+1}}) \rangle$ be the extended cycle of $\mathcal{B}$ associated with $b_{q_{s+1}}$. Then extend h so that $h(c_r^{s+1}) = d_r^{q_{s+1}}$, $h(u_r^{s+1}) = v_r^{q_{s+1}}$, and h restricted to $tree_{\mathcal{A}}(u_r^{s+1}, s+1)$ is equal to the canonical map $\Theta_{u_r^{s+1}, v_r^{q_{s+1}}, s+1}$ for $r = 0, \ldots, k_{s+1} - 1$.

Then let $S_{s+1} = S_s \cup \{s+1\}$, $T_{s+1} = T_s \cup \{q_{s+1}\}$, and extend ϕ_s to ϕ_{s+1} by letting $\phi_{s+1}(s+1) = q_{s+1}$.

Case 3. C_{s+1} is equal to one of the cycles C_i for $i \in S_s$, but D_{s+1} is not equal to one of the cycles D_j for $j \in T_s$.
Then let $\overline{D}_{s+1} = \langle (d_0^{s+1}, \ldots, d_{\ell_{s+1}-1}^{s+1}), (v_0^{s+1}, \ldots, v_{\ell_{s+1}-1}^{s+1}) \rangle$ be the extended cycle of $\mathcal{B}$ associated with b_{s+1}. Then let p_{s+1} be the least p such that C_p is not equal to one of the cycles C_i for $i \in S_s$ and C_p has size ℓ_{s+1}. Let $\overline{C}_{p_{s+1}} = \langle (c_0^{p_{s+1}}, \ldots, c_{\ell_{s+1}-1}^{p_{s+1}}), (u_0^{p_{s+1}}, \ldots, u_{\ell_{s+1}-1}^{p_{s+1}}) \rangle$ be the extended cycle of $\mathcal{A}$ associated with $a_{p_{s+1}}$. Then extend h so that $h(c_r^{p_{s+1}}) = d_r^{s+1}$, $h(u_r^{p_{s+1}}) = v_r^{s+1}$, and h restricted to $tree_{\mathcal{A}}(u_r^{p_{s+1}}, s+1)$ is equal to the canonical map $\Theta_{u_r^{p_{s+1}}, v_r^{s+1}, s+1}$ for $r = 0, \ldots, \ell_{s+1} - 1$.

Then let $S_{s+1} = S_s \cup \{p_{s+1}\}$, $T_{s+1} = T_s \cup \{s+1\}$, and extend ϕ_s to ϕ_{s+1} by letting $\phi_{s+1}(p_{s+1}) = s+1$.

Case 4. C_{s+1} is not equal to one of the cycles C_i for $i \in S_s$, and D_{s+1} is not equal to one of the cycles D_j for $j \in T_s$.
Then let $\overline{C}_{s+1} = \langle (c_0^{s+1}, \ldots, c_{k_{s+1}-1}^{s+1}), (u_0^{s+1}, \ldots, u_{k_{s+1}-1}^{s+1}) \rangle$ be the extended cycle of $\mathcal{A}$ associated with a_{s+1}. Then let q_{s+1} be the least q such that D_q is not equal to one of the cycles D_j for $j \in T_s$ and D_q has size k_{s+1}. Let $\overline{D}_{q_{s+1}} = \langle (d_0^{q_{s+1}}, \ldots, d_{k_{s+1}-1}^{q_{s+1}}), (v_0^{q_{s+1}}, \ldots, v_{k_{s+1}-1}^{q_{s+1}}) \rangle$ be the extended cycle of $\mathcal{B}$ associated with $b_{q_{s+1}}$. Then extend h so that $h(c_r^{s+1}) = d_r^{q_{s+1}}$, $h(u_r^{s+1}) = v_r^{q_{s+1}}$, and h restricted to $tree_{\mathcal{A}}(u_r^{s+1}, s+1)$ is equal to the canonical map $\Theta_{u_r^{s+1}, v_r^{q_{s+1}}, s+1}$ for $r = 0, \ldots, k_{s+1} - 1$.

If D_{s+1} is equal to $D_{q_{s+1}}$, then let $S_{s+1} = S_s \cup \{s+1\}$, $T_{s+1} = T_s \cup \{q_{s+1}\}$, and extend ϕ_s to ϕ_{s+1} by letting $\phi_{s+1}(s+1) = q_{s+1}$. Otherwise, let $\overline{D}_{s+1} = \langle (d_0^{s+1}, \ldots, d_{\ell_{s+1}-1}^{s+1}), (v_0^{s+1}, \ldots, v_{\ell_{s+1}-1}^{s+1}) \rangle$ be the extended cycle of $\mathcal{B}$ associated with b_{s+1}. Then let p_{s+1} be the least p such that C_p is not equal to one of the cycles C_i for $i \in S_s$ and is not equal to C_{s+1} and C_p has size ℓ_{s+1}. Let $\overline{C}_{p_{s+1}} = \langle (c_0^{p_{s+1}}, \ldots, c_{\ell_{s+1}-1}^{p_{s+1}}), (u_0^{p_{s+1}}, \ldots, u_{\ell_{s+1}-1}^{p_{s+1}}) \rangle$ be the extended cycle of $\mathcal{A}$ associated with $a_{p_{s+1}}$. Then extend h so that $h(c_r^{p_{s+1}}) = d_r^{s+1}$, $h(u_r^{p_{s+1}}) = v_r^{s+1}$, and h restricted to $tree_{\mathcal{A}}(u_r^{p_{s+1}}, s+1)$ is equal to the canonical map $\Theta_{u_r^{p_{s+1}}, v_r^{s+1}, s+1}$ for $r = 0, \ldots, \ell_{s+1} - 1$.

Then let $S_{s+1} = S_s \cup \{s+1, p_{s+1}\}$, $T_{s+1} = T_s \cup \{s+1, q_{s+1}\}$, and extend ϕ_s

to ϕ_{s+1} by letting $\phi_{s+1}(s+1) = q_{s+1}$ and $\phi_{s+1}(p_{s+1}) = s+1$.

It is then easy to see that h will be an isomorphism from $Fin(\mathcal{A})$ onto $Fin(\mathcal{B})$.

Next we computably map $O_{\mathcal{A}}(x_i)$ onto $O_{\mathcal{A}}(y_i)$ as follows. Let $x_{i,0} = x_i$ and $x_{i,n} = f^n(x_i)$ for $n \geq 1$. Then define $x_{i,-n}$ for $n \geq 1$ inductively as follows: $x_{i,-1}$ is the least element z such that $f(z) = x_i$. There are only two elements which map to x_i under f, and we can enumerate A until we find these two elements and then pick the least of these two elements to be $x_{i,-1}$. Then inductively for $n > 1$, we define $x_{i,-n}$ to be the least element z that such that $f(z) = x_{i,-(n-1)}$. Let $X_i = \{x_{i,n} : n \in \mathbb{Z}\}$. Similarly, we let $y_{i,0} = y_i$ and $y_{i,n} = g^n(y_i)$ for $n \geq 1$. We let $y_{i,-1}$ be the least element z such that $g(z) = y_i$ and, inductively, define $y_{i,-n}$ for $n > 1$ to the least element z such that $g(z) = y_{i,-(n-1)}$. Let $Y_i = \{y_{i,n} : n \in \mathbb{Z}\}$. Next for all $n \in \mathbb{Z}$, let $u_{i,n}$ be the element which is not in X_i such that $f(u_{i,n}) = x_{i,n}$ and let $v_{i,n}$ be the element which is not in Y_i such that $g(v_{i,n}) = y_{i,n}$. Then we define h so that for all $1 \leq i \leq t$ and $n \in \mathbb{Z}$, $h(x_{i,n}) = y_{i,n}$, $h(u_{i,n}) = v_{i,n}$, and h restricted to $tree(u_{i,n})$ equal to the canonical map $\Theta_{u_{i,n},v_{i,n}}$.

It follows that h will be a computable isomorphism from $\mathcal{A}$ onto $\mathcal{B}$ so that $\mathcal{A}$ is computably categorical.

Next suppose that $\mathcal{A} = (A, f)$ is a computable 2:1 structure such that $\mathcal{A}$ has infinitely many orbits which are $\mathbb{Z}$-chains. If $Fin(\mathcal{A})$ is not a computable set, then partition $\mathbb{N}$ into two infinite computable sets B and C. Let $a_1, a_2, \ldots,$ be an effective enumeration of $Fin(\mathcal{A})$ and let $B = \{b_1 < b_2 < \cdots\}$. Then we define the function $g : B \to B$ so that $g(b_i) = b_j$ if and only if $f(a_i) = a_j$, so that the map $f : Fin(\mathcal{A}) \to B$ defined by $f(a_i) = b_i$ will be an isomorphism from $(Fin(\mathcal{A}), f)$ onto (B, g). We can then extend g to C by effectively partitioning C into a uniform sequence of pairwise disjoint computable sets $C_0, C_1, \ldots$ and define g so that each C_i is a g-orbit which is a $\mathbb{Z}$-chain of our 2:1 structure. It will follow that $(\mathbb{N}, g)$ is a computable structure which is isomorphic to (A, f). Note that $(\mathbb{N}, g)$ has the property that the predicate $SameOrbit_{(\mathbb{N},g)}(a, b)$, which holds if and only if a and b lie in the same orbit of $(\mathbb{N}, g)$, is computable.

Instead of directly constructing a computable 2:1 structure which is not isomorphic to $(\mathbb{N}, g)$, we will modify our construction so that given any c.e. set E which is both infinite and co-infinite, we will construct a computable 2:1 structure $(\mathbb{N}, g_E)$ such that the predicate $SameOrbit_{(\mathbb{N},g)}$ is Turing equivalent to E. The idea is to slightly modify our construction of $(\mathbb{N}, g)$. That is, let $B, C, C_0, C_1, \ldots$ be as above. Then we let $g_E = g$ on B so that $Fin(\mathcal{A})$ is computably isomorphic to (B, g_E). Let $c_0, c_1, \ldots$ be the least elements of $C_0, C_1, \ldots$, respectively. Fix some effective enumeration of E and let E^s be the finite set of elements enumerated in E at stage s. Assume that $E^0 = \{0\}$ and that $card(E^s - E^{s-1}) = 1$ for all $s \geq 1$.

We construct g_E in stages. The basic idea is that at any stage s, we will be defining g_E so that the elements of C_i for $i \geq 1$ will form an orbit which will be a $\mathbb{Z}$-chain. That is, we let $c_0^i = c_i$. At any given stage s, as long as $i \notin E$, we construct what we call a partial $\mathbb{Z}$-chain of length $2s+1$. That is, we will define a sequence $c^i_{-s}, c^i_{-(s-1)}, \ldots, c^i_{-1}, c^i_0, c^i_1, \ldots, c^i_s$ and a sequence

$d^i_{-s}, d^i_{-s-1}, \ldots, d^i_{-1}, d^i_0, d^i_1, \ldots, d^i_s$ such that $g_E(c^i_k) = c^i_{k+1}$ for $-s \leq k \leq s-1$ and $g_E(d^i_k) = c^i_k$ for $-s \leq k \leq s$. Moreover, for each k, we assume that g_E is defined on initial segment I^i_s of C_i so that in the graph of g_E restricted to I^i_s, d^i_k is the root of a complete binary tree of height s. See Figure 7 for a picture of a partial $\mathbb{Z}$-chain of length 5. Thus I^i_s will be a set of size $(2s+1)2^{s+1}$. At the next stage, we first use the next 4 elements of C^i to define $c^i_{-(s+1)}, c^i_{s+1}, d^i_{(s+1)}, d^i_{s+1}$, then we will use then next $2(2^{s+2}-1)$ elements to construct the binary trees of height $s+1$ which have roots $d^i_{(s+1)}$ and d^i_{s+1}, and finally use the next $(2s+1)2^s$ to extend the binary trees with roots $d^i_{-s}, \ldots, d^i_s$ so that they have height $s+1$.

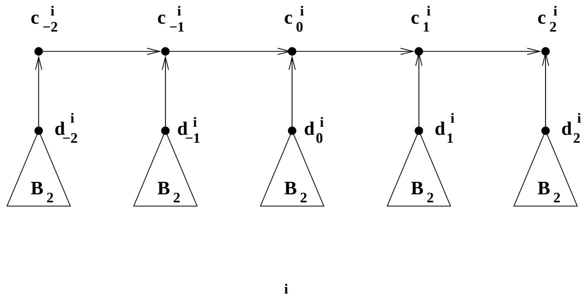

Figure 7: A partial $\mathbb{Z}$-chain of length 5.

For $i = 0$, we perform a similar construction except that the partial $\mathbb{Z}$-chain will be of length $2k_s + 1$ for some integer k_s, which will be an initial segment of $\bigcup_{i \in E^s} C_i$. That is, we will define a sequence

$$c^0_{-k_s}, c^0_{-(k_s-1)}, \ldots, c^0_{-1}, c^0_0, c^0_1, \ldots, c^0_{k_s}$$

and a sequence

$$d^0_{-k_s}, d^0_{-(k_s-1)}, \ldots, d^0_{-1}, d^0_0, d^0_1, \ldots, d^0_{k_s}$$

such that $g_E(c^0_k) = c^0_{k+1}$ for $-k_s \leq k \leq k_s - 1$ and $g_E(d^0_j) = c^0_j$ for $-k_s \leq j \leq k_s$. Moreover, for each j, we assume that g_E is defined on the initial segment I^0_s of $\bigcup_{\in E^s} C_i$ so that in the graph of g_E restricted to I^0_s, d^i_j is the root of a complete binary tree of height k_s. Then if $j \in E^{s+1} - E^s$, we will simply define $g_E(c^0_{k_s}) = c^j_{-s}$, which will have the effect of grafting the partial $\mathbb{Z}$-chain for C_j at stage s onto the front of the partial $\mathbb{Z}$-chain for 0 at stage s. We then simply have to add appropriate elements at the end of the partial $\mathbb{Z}$-chain for 0 and the corresponding binary trees at stage $s+1$, so that we have a $\mathbb{Z}$-chain of length $k_{s+1} = k_s + 2s + 2$ for 0.

It is then easy to see that this will construct a computable 2:1 structure $\mathcal{B}_E = (\mathbb{N}, g_E)$ which is isomorphic to $\mathcal{A}$. Next consider the question of the degree of predicate $SameOrbit(a, b)$ for $\mathcal{B}_E$. Note that $Fin(\mathcal{B}_E)$ is a computable set so that given $a, b \in \mathbb{N}$, we first ask if both $a, b \in Fin(\mathcal{B}_E)$. If so, then we can iterate g_E on a and b until we find the cycles Cy_a and Cy_b to which a and b are attached, respectively. Then a is in the same orbit as b if and only if $Cy_a = Cy_b$.

If both a and b are not in $Fin(\mathcal{B}_E)$, then we can find i and j such that $a \in C_i$ and $b \in C_j$. If $i = j$, then a and b are in the same orbit and if $i \neq j$, then a and b are in the same orbit if and only if $i, j \in E$. Finally if it is not the case that either both a and b are in $Fin(\mathcal{B}_E)$ or both a and b are not in $Fin(\mathcal{B}_E)$, then a and b are not in the same orbit. This shows that $SameOrbit \leq_T E$. On the other hand, $c_0, c_1 \in E$ if and only if c_0 and c_1 are in the same orbit so that $E \leq_T SameOrbit(,)$.

Clearly if E is a c.e. non-computable set, then $(\mathbb{N}, g)$ is not computably isomorphic to $(\mathbb{N}, g_E)$. Thus if $\mathcal{A}$ has infinitely many $\mathbb{Z}$-chains, then $\mathcal{A}$ is not computably categorical. □

We have the following corollaries of Theorem 2.3.

Corollary 2.4. *Suppose that $\mathcal{A} = (A, f)$ and $\mathcal{B} = (B, g)$ are computable 2:1 structures such that $Fin(\mathcal{A})$ and $Fin(\mathcal{B})$ are computable and the predicate SameOrbit is computable in both $\mathcal{A}$ and $\mathcal{B}$. Then $\mathcal{A}$ is isomorphic to $\mathcal{B}$ if and only if $\mathcal{A}$ is computably isomorphic to $\mathcal{B}$.*

Proof. Suppose that $\mathcal{A}$ is isomorphic to $\mathcal{B}$. Then by our argument in the proof of Theorem 2.3, we know that $(Fin(\mathcal{A}), f)$ is computably isomorphic to $Fin(\mathcal{B}), g)$. Then let $A - Fin(\mathcal{A}) = \{a_0 < a_1 < \cdots\}$ and $A - Fin(\mathcal{B}) = \{b_0 < b_1 < \cdots\}$. Because $SameOrbit$ is a computable predicate for $\mathcal{A}$, we can effectively determine if a_i is the smallest element in its orbit. That is, a_i is the smallest element in its orbit if and only if $\neg SameOrbit(a_j, a_i)$ hold for all $j < i$. Thus we can effectively list as $a_0 = a_{i_0} < a_{i_1} < \cdots$ all the elements of $A - Fin(\mathcal{A})$ such that a_{i_j} is the least element in its orbit. Similarly, we can effectively list as $b_0 = b_{i_0} < b_{i_1} < \ldots$ all the elements of $B - Fin(\mathcal{B})$ such that b_{i_j} is the least element in its orbit. Then we can use the procedure described in Theorem 2.3 to computably map the $\mathbb{Z}$-chain $O_{\mathcal{A}}(a_{i_j})$ onto the $\mathbb{Z}$-chain $O_{\mathcal{B}}(b_{i_j})$. Thus $\mathcal{A}$ is computably isomorphic to $\mathcal{B}$.

Clearly, if $\mathcal{A}$ is computably isomorphic to $\mathcal{B}$, then $\mathcal{A}$ is isomorphic to $\mathcal{B}$. □

Corollary 2.5. *Every computable 2:1 structure (A, f) is Δ^0_2-categorical.*

Proof. Note that $Fin(\mathcal{A})$ is c.e. and hence Δ^0_2, and $SameOrbit(a, b)$ if and only if $O_{\mathcal{A}}(a) \cap C_{\mathcal{A}}(b) \neq \emptyset$, which is also a Δ^0_2 predicate. Thus the corollary follows from a relativized version of Corollary 2.4. □

3 Computably Categorical (2,0):1 Structures

Suppose that we are given a (2,0):1 structure (A, f). If an orbit $O_{\mathcal{A}}(a)$ is a k-cycle, then its graph must consist of an extended cycle

$$\overline{C} = \langle (c_0, \ldots, c_{k-1}), (d_0, \ldots, d_{k-1}) \rangle$$

together with binary trees $T_0, \ldots, T_{k-1}$ where $T_i = Tree(d_i)$ for $i = 0, \ldots, k-1$. In such a situation, if c_0 is the least element of $\{c_0, \ldots, c_{k-1}\}$, then we shall say that $O_{\mathcal{A}}(a)$ is of type $\langle (c_0, \ldots, c_{k-1}), (T_0, \ldots, T_{k-1}) \rangle$. Specifying the type of $\mathbb{Z}$-chains and ω-chains is more problematic. That is, one way to specify the graph of a $\mathbb{Z}$-chain is to give two sequences

$$\begin{aligned} \vec{c} &= (c_0, c_1, c_{-1}, c_2, c_{-2}, \ldots) \text{ and} \\ \vec{d} &= (d_0, d_1, d_{-1}, d_2, d_{-2}, \ldots) \end{aligned}$$

and a sequence of binary trees

$$\vec{T} = (T_0, T_1, T_{-1}, T_2, T_{-2}, \ldots)$$

such that for all $i \in \mathbb{Z}$, $f(c_i) = c_{i+1}$, $f(d_i) = c_i$, and $Tree(d_i) = T_i$. Similarly, one way to specify the graph of an ω-chain is to give two sequences

$$\begin{aligned} \vec{c} &= (c_0, c_1, c_2, c_3, \ldots) \text{ and} \\ \vec{d} &= (d_1, d_2, d_3, \ldots) \end{aligned}$$

and a sequence of binary trees

$$\vec{T} = (T_1, T_2, \ldots)$$

such that for all $i \in \omega$, $f(c_i) = c_{i+1}$, for all $i \geq 1$, $f(d_i) = c_i$, and $Tree(d_i) = T_i$.

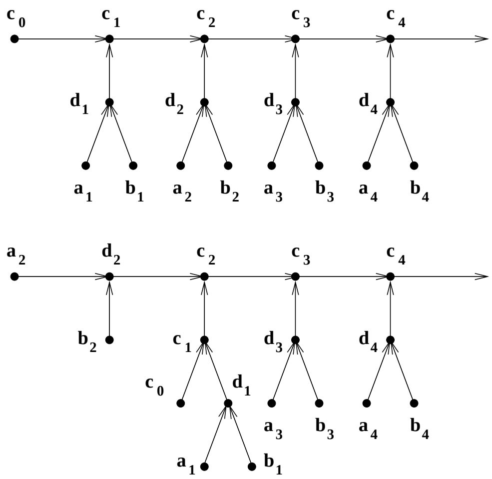

Figure 8: An ω-chain where all the attached trees are three element binary trees.

Unfortunately, the sequences of trees $\vec{T}$ depend on how we pick $\vec{c}$. For example, suppose we have an orbit which is the ω-chain pictured at the top of Figure 8. That is, the tree T_i are all three element binary trees. Then at the bottom of Figure 8, we have pictured another way to represent that ω-chain, which clearly gives rise to a different sequence of trees. Nevertheless, whenever we have two such equivalent descriptions

$$\begin{aligned}\vec{c} &= (c_0, c_1, c_2, c_3, \dots),\\ \vec{d} &= (d_1, d_2, d_3, \dots), \text{ and}\\ \vec{T} &= (T_1, T_2, \dots)\end{aligned}$$

and

$$\begin{aligned}\vec{c'} &= (c'_0, c'_1, c'_2, c'_3, \dots),\\ \vec{d'} &= (d'_1, d'_2, d'_3, \dots), \text{ and}\\ \vec{T'} &= (T'_1, T'_2, \dots),\end{aligned}$$

there will be an n large enough so that $c_i = c'_i$, $d_i = d'_i$, and $T_i = T'_i$ for all $i \geq n$ and, hence, $tree(c_n) = tree(c'_n)$.

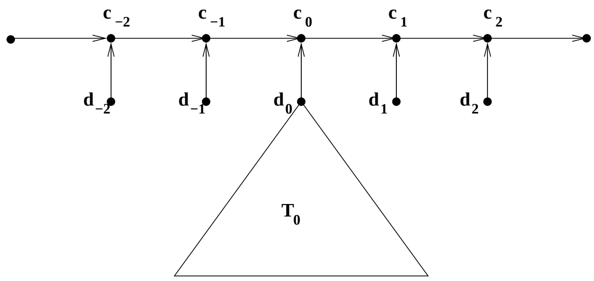

Figure 9: A $\mathbb{Z}$-chain.

A similar situation happens for $\mathbb{Z}$-chains. That is, suppose that we have the $\mathbb{Z}$-chain pictured in Figure 9. That is, the $\mathbb{Z}$-chain corresponds to the sequences

$$\begin{aligned}\vec{c} &= (c_0, c_1, c_{-1}, c_2, c_{-2}, \dots),\\ \vec{d} &= (d_0, d_1, d_{-1}, d_2, d_{-2}, \dots), \text{ and}\\ \vec{T} &= (T_0, T_1, T_{-1}, T_2, T_{-2}, \dots)\end{aligned}$$

where T_0 is isomorphic to the complete infinite binary tree B and T_i and T_{-i} are one element trees for $i > 0$. Then it is clear that, if we represent the same $\mathbb{Z}$-chain as

$$\begin{aligned}\vec{c} &= (c_0, c_1, c'_{-1}, c_2, c'_{-2}, \dots),\\ \vec{d} &= (d'_0, d_1, d'_{-1}, d_2, d'_{-2}, \dots), \text{ and}\\ \vec{T} &= (T'_0, T_1, T'_{-1}, T_2, T'_{-2}, \dots)\end{aligned}$$

where $c'_{-1} = d_0$, and $c'_{-2}, c'_{-3}, \dots$ is some infinite path through the binary tree T_0, then T'_{-i} will be isomorphic to the complete binary tree of $i \geq 1$ and T'_0 is isomorphic to $Tree_{\mathcal{A}}(c_1)$. What is worse, it is also clear that we could represent the same $\mathbb{Z}$-chain as an ω-chain starting at the d_1. Nevertheless, just as the case with ω-chains, there will be an n large enough so that $c_i = c'_i$, $d_i = d'_i$, and $T_i = T'_i$ for all $i \geq n$ and, hence, $tree(c_n) = tree(c'_n)$.

We say that a (2,0):1 structure (A, f) is **locally finite** if $tree_{\mathcal{A}}(a)$ is finite for all $a \in A$. Locally finite (2,0):1 structures are much simpler than general (2,0):1 structures. That is, in locally finite (2,0):1 structures, all orbits which are k-cycles are finite and there are no $\mathbb{Z}$-chains. We say that a computable (2,0):1 structure $(\mathcal{A}, f)$ is **highly computable** if the range of f, $ran(f)$, is computable. It is easy to see that in a locally finite computable (2,0):1 structure $(\mathcal{A}, f)$, one can effectively find the finite set $tree_{\mathcal{A}}(a)$ for any $a \in A$.

Theorem 3.1. *Suppose that $\mathcal{A} = (A, f)$ and $\mathcal{B} = (B, g)$ are isomorphic highly computable locally finite (2,0):1 structures which have only finitely many ω-chains, then $\mathcal{A}$ is computably isomorphic to $\mathcal{B}$.*

Proof. In such a case, we know that $Fin(\mathcal{A})$ and $Fin(\mathcal{B})$ are computable. It is easy to construct an isomorphism h_0 from $(Fin(\mathcal{A}), f)$ to $Fin(\mathcal{B}), g)$ by a standard back-and-forth argument. The key is that, since $\mathcal{A}$ is highly computable and locally finite, it follows that given any $a \in Fin(\mathcal{A})$, we can effectively compute the entire orbit of a. That is, as in Theorem 2, we can effectively find the extended cycle $\langle (c_0, \ldots, c_{k-1}), (d_0, \ldots, d_{k-1}) \rangle$ in $O_{\mathcal{A}}(a)$. Then we can effectively find $(T_0, \ldots, T_{k-1})$ such that $Tree_{\mathcal{A}}(d_i) = T_i$ for $i = 0, \ldots, k-1$. Given such an orbit $O_{\mathcal{A}}(a)$, we can then search through the elements of B until we find a b whose orbit is isomorphic to $O_{\mathcal{A}}(a)$. That is, we can find a b whose extended cycle is $\langle (c'_0, \ldots, c'_{k-1}), (d'_0, \ldots, d'_{k-1}) \rangle$ and binary trees $(T'_0, \ldots, T'_{k-1})$ such that $Tree_{\mathcal{B}}(d'_i) = T'_i$ for $i = 0, \ldots, k-1$ such that there is an s with $0 \leq s \leq k-1$ where T_j is isomorphic to $T_{s+j \mod k}$ for $j = 0, \ldots, k-1$. Then we can easily construct an isomorphism from $O_{\mathcal{A}}(a)$ to $O_{\mathcal{B}}(b)$.

Moreover, there must exist representatives $a_1, \ldots, a_r$ of the ω-chains in $\mathcal{A}$ and representatives $b_1, \ldots, b_r$ of the ω-chains in $\mathcal{B}$ with the following properties. Let $A_i = \{a_{i,0}, a_{i,1}, \ldots\}$ where $a_{i,0} = a_i$ and $a_{i,n} = f^n(a_i)$ for $n \geq 1$. For each $n \geq 1$, let $c_{i,n}$ be the element of A such that $c_{i,n} \in A_i$ and $f(c_{i,n}) = a_{i,n}$ and let $T_{i,n} = Tree_{\mathcal{A}}(c_{i,n})$. Similarly, let $B_i = \{b_{i,0}, b_{i,1}, \ldots\}$ where $b_{i,0} = b_i$ and $b_{i,n} = g^n(b_i)$ for $n \geq 1$. For each $n \geq 1$, let $d_{i,n}$ be the element of A such that $d_{i,n} \in A_i$ and $g(d_{i,n}) = b_{i,n}$, and let $S_{i,n} = Tree_{\mathcal{B}}(b_{i,n})$. Then we assume that for $1 \leq i \leq r$, $Tree_{\mathcal{A}}(a_i)$ is isomorphic to $Tree_{\mathcal{B}}(b_i)$ and $T_{i,n}$ is isomorphic to $S_{i,n}$ for all $n \geq 1$.

Finally, note that, for any $a \in A$ and $b \in B$ such that $Tree_{\mathcal{A}}(a)$ is isomorphic to $Tree_{\mathcal{B}}(b)$, we can construct a canonical isomorphism $\phi_{a,b}$ from $Tree_{\mathcal{A}}(a)$ onto $Tree_{\mathcal{B}}(b)$ as follows.

Stage 0. Set $\phi(a) = b$.

Stage s+1. Assume that we have defined ϕ on $tree_{\mathcal{A}}(a, s)$ so that ϕ is an isomorphism from $Tree_{\mathcal{A}}(a, s)$ onto $Tree_{\mathcal{B}}(b, s)$, and for all $x \in tree_{\mathcal{A}}(a, s)$, $Tree_{\mathcal{A}}(x)$ is isomorphic to $Tree_{\mathcal{B}}(\phi(x))$. Then extend ϕ to an isomorphism from $Tree_{\mathcal{A}}(a, s+1)$ onto $Tree_{\mathcal{B}}(b, s+1)$ as follows. For each $x \in tree_{\mathcal{A}}(a, s)$

which is in the range of f, find $x_0 < x_1$ in A such that $f(x_0) = f(x_1) = x$ and find $y_0 < y_1$ in B such that $g(y_0) = g(y_1) = \phi(x)$. By assumption, $Tree_{\mathcal{A}}(x)$ is isomorphic to $Tree_{\mathcal{B}}(\phi(x))$. Then if $Tree_{\mathcal{A}}(x_0)$ is isomorphic to $Tree_{\mathcal{A}}(x_1)$, we know that $Tree_{\mathcal{B}}(y_i)$ is isomorphic to $Tree_{\mathcal{A}}(x_0)$ for $i = 0, 1$ so that we let $\phi(x_0) = y_0$ and $\phi(x_1) = y_1$. If $Tree_{\mathcal{A}}(x_0)$ is not isomorphic to $Tree_{\mathcal{A}}(x_1)$, then there is some $s \in \{0, 1\}$ such that $Tree_{\mathcal{A}}(x_0)$ is isomorphic to $Tree_{\mathcal{B}}(y_s)$ and $Tree_{\mathcal{A}}(x_1)$ is isomorphic to $Tree_{\mathcal{B}}(y_{1-s})$. In that case, we let $\phi(x_0) = y_s$ and $\phi(x_1) = y_{1-s}$.

It follows that for each $1 \leq i \leq r$, we can define a computable isomorphism $h_i : O_{\mathcal{A}}(a_i) \to O_{\mathcal{B}}(b_i)$ by setting $h_i(a_{i,n}) = b_{i,n}$ for $n \geq 0$, $h_i(c_{i,n}) = d_{i,n}$ for $n \geq 1$, and ensuring that h_i restricted to $tree_{\mathcal{A}}(a_i)$ is the canonical isomorphism from $Tree_{\mathcal{A}}(a_i)$ onto $Tree_{\mathcal{B}}(b_i)$, and h_i restricted to $tree_{\mathcal{A}}(c_{i,n})$ is the canonical isomorphism from $Tree_{\mathcal{A}}(c_{i,n})$ onto $Tree_{\mathcal{B}}(d_{i,n})$ for $n \geq 1$.

Thus $\bigcup_{i=0}^{r} h_i$ is a computable isomorphism from $\mathcal{A}$ onto $\mathcal{B}$. □

Now suppose that $\mathcal{A} = (A, f)$ is a highly computable locally finite (2,0):1 structure such that $Fin(\mathcal{A}) = A$. In this case, the type of k-cycle is of the form $\langle (c_0, \ldots, c_{k-1}), (T_0, \ldots, T_{k-1}) \rangle$ where each T_i is a finite binary tree. There is a natural order on the set of finite binary trees determined by embeddability. That is, if T and S are finite binary trees with roots s and t, respectively, then we can think of T and S as directed graphs with all edges directed toward the root. Then we write $T \sqsubseteq S$ if there is map ϕ from the nodes of T into the nodes of S such that $\phi(r) = s$ and for any nodes x and y in T, (x, y) is a directed edge in T if and only if $(\phi(x), \phi(y))$ is a directed edge in S. Alternatively, $T \sqsubseteq S$ if and only if the directed graph S can be constructed by taking a directed graph T and replacing each leaf $\ell \in T$ with a binary tree T_ℓ with all edges directed toward the root. For example, the complete binary tree T_k of height k is embeddable in the complete binary tree of T_r of height r for all $r \geq k$. We also say that every binary tree T is embeddable in the complete binary tree B. We can then extend $\sqsubseteq$ to orbits in $\mathcal{A}$ by saying that $\mathcal{O}_{\mathcal{A}}(a) \sqsubseteq \mathcal{O}_{\mathcal{A}}(b)$ if and only if there is some $k \geq 1$ such that the type of $\mathcal{O}_{\mathcal{A}}(a)$ is $\langle (c_0, \ldots, c_{k-1}), (T_0, \ldots, T_{k-1}) \rangle$, the type of $\mathcal{O}_{\mathcal{A}}(b)$ is $\langle (d_0, \ldots, d_{k-1}), (S_0, \ldots, S_{k-1}) \rangle$, and there is some $0 \leq p \leq k-1$ such that $T_i \sqsubseteq S_{p+i \mod k}$ for $i = 0, \ldots, k-1$.

We say that a computable (2,0):1 structure $\mathcal{A} = (A, f)$ has an **explicitly computable cycle structure** if $\mathcal{A}$ is locally finite, $Fin(\mathcal{A}) = A$, and there is a computable function h such that for all $k \geq 1$, $h(k)$ is equal to the code of a list $((D_1, d_1), \ldots, (D_{\ell_k}, d_{\ell_k})$ where any orbit $O_{\mathcal{A}}(a)$ which is a k-cycle is isomorphic to one of $D_1, \ldots, D_{\ell_k}$ and there are exactly d_i k-cycles in $\mathcal{A}$ which are isomorphic to D_i for $i = 1, \ldots, \ell_r$. In addition, we assume that the poset $\mathcal{P}_k = (\{D_1, \ldots, D_{\ell_k}\}, \sqsubseteq)$, where $\sqsubseteq$ is the embeddability relation has the property that d_i is finite if D_i is not a minimal element in $\mathcal{P}_k$ and $d_i \in \mathbb{N} \cup \{\omega\}$ if D_i is a minimal element of $\mathcal{P}_k$.

We claim that if $\mathcal{A} = (A, f)$ has an **explicitly computable cycle structure**, then $\mathcal{A}$ is highly computable. Clearly, it is enough to show that we can effectively compute the $\mathcal{A}$-orbit of a for any $a \in A$. Given an element $a \in A$, we can first compute $a, f(a), f^2(a), \ldots$ until we find the k such that $O_{\mathcal{A}}(a)$ is a k-cycle. At that point, we start enumerating A and computing f until we find the required number of copies of D_i for all non-minimal elements of $\mathcal{P}_k$. If $O_{\mathcal{A}}(a)$ is one of those k-cycles, then we have explicitly computed the $O_{\mathcal{A}}(a)$. If not, $O_{\mathcal{A}}(a)$ is isomorphic to a minimal element of $\mathcal{P}_k$. We know that none of the minimal elements of $\mathcal{P}_k$ are embeddable in each other, which means that we can compute long enough until we see enough of the partial structure of $O_{\mathcal{A}}(a)$ to distinguish it from the other minimal elements of $\mathcal{P}_k$. At that point, we will know the isomorphism type of $O_{\mathcal{A}}(a)$ so that we can continue to enumerate A and compute f until we have found all the elements of $O_{\mathcal{A}}(a)$.

Thus we have the following corollary of Theorem 3.1.

Corollary 3.2. *If $\mathcal{A} = (A, f)$ is a computable (2,0):1 structure which has an explicitly computable cycle structure, then $\mathcal{A}$ is computably categorical. Furthermore, the argument above relativizes to show that the indicated structures are in fact relatively computably categorical.*

Next we consider the special case of locally finite structures $\mathcal{A} = (\omega, f)$ such that for some fixed k, $\mathcal{A}$ consists exactly of an infinite number of orbits each containing a k-cycle. Following the notation of Lempp, McCoy, R. Miller and Solomon [10], we say that $\mathcal{A}$ is **strongly finite** if there exists a finite set $\{D_1, \ldots, D_\ell\}$ such that every orbit is isomorphic to D_i for some $i \leq \ell$ and, furthermore, there do not exist $D_i \neq D_j$ such that there are infinitely many orbits of type D_i and infinitely many orbits of type j such that D_i is embeddable into D_j. Then we have the following corollary of Theorem 3.1

Proposition 3.3. *Suppose that, for a fixed finite k, $\mathcal{A} = (A, f)$ consists of an infinite number of orbits, each containing a k-cycle, and is strongly finite. Then $\mathcal{A}$ is relatively computably categorical.*

Proof. We prove this by describing the Scott formulas. First we observe that the relation "$\mathcal{O}(x) = \mathcal{O}(y)$" is c.e., since x and y are in the same orbit if and only if, for some natural numbers m and n, $f^m(x) = f^n(y)$. Furthermore, if we have a bound M on the size of the orbits, as we do here, then this relation is in fact Δ^0_1, since we can bound m and n by M.

For each type D_j which occurs only finitely often, choose a member of each orbit of type D_j as a parameter. Then $\mathcal{O}(x)$ has type D_j for such a j if and only if it is the same orbit as one of the parameters. If x is in one of the remaining orbits, let $\mathcal{O}^t(x) = \{y : (\exists m < t)(\exists n < t)(f^m(x) = f^n(y))\}$. Then $\mathcal{O}(x)$ is of type D_j if and only if for some t, $\mathcal{O}^t(x)$ is of type D_i. That is, once the orbit of x looks like D_i and is known not be one of the orbits of type D_j where D_i is embeddable into D_j, then $\mathcal{O}^t(x) = \mathcal{O}(x)$ since it cannot grow into anything else. Then the condition that $\mathcal{O}(x)$ has type D_i is a c. e.formula consisting of a disjunction over natural numbers t of a c. e.formula which describes the condition that $\{y : (\exists m < t)(\exists n < t)(f^m(x) = f^n(y))\}$ is isomorphic to D_i. Now for every i, we have a canonical copy of D_i and we can find a particular subset $S(x)$ of D_i and specify that $d \in S$ if and only if there is an isomorphism taking $\mathcal{O}(x)$ to D_i which maps x to d. Then the c.e. Scott formula of x first states the orbit type of $\mathcal{O}(x)$ and then indicates the set $S(x)$.

For a tuple $(x_1, \ldots, x_m)$ of elements, the Scott formula consists of the individual Scott formulas for $x_1, \ldots, x_m$ together with, for each pair x_i and x_j, either the statement that $\mathcal{O}(x_i) = \mathcal{O}(x_j)$, or the statement that $\mathcal{O}(x_i) \neq \mathcal{O}(x_j)$, and finally a statement that specifies for each tuple $y_1, \ldots, y_n$ taken from $x_1, \ldots, x_m$ which belong to the same orbit of type D_i, which tuples $d_1, \ldots, d_n$ could be the images of $y_1, \ldots, y_n$ under an isomorphism of $\mathcal{O}(x_i)$ with D_i. □

Unlike the case of computable 2:1 structures, we cannot characterize the computably categorical, locally finite, highly computable (2,0):1 structures as those which have only finitely many ω-chains. We can construct a computably categorical, locally finite, highly computable (2,0):1 structure $\mathcal{A} = (A, f)$ with infinitely many ω-chains as follows. First we assume that there is a fixed $r \geq 0$ such that any k-cycle in $\mathcal{A}$ has type $\langle (c_0, \ldots, c_{k-1}), (T_0, \ldots, T_{k-1}) \rangle$ where $T_0, T_1, \ldots, T_{k-1}$ are all binary trees of height $\leq r$. Next assume that for all $t > r$, there is a unique ω-chain C_t in $\mathcal{A}$

$$\langle (a_0, a_1, a_2, \ldots), (T_1, T_2, \ldots) \rangle$$

such that T_i is a complete binary tree of height t. The key thing to observe about the ω-chain C_t is that the only ways to represent it as an ω-chain

$$\langle (a'_0, a'_1, a'_2, \ldots), (T'_1, T'_2, \ldots) \rangle,$$

which is different from $\langle (a_0, a_1, a_2 \ldots), (T_1, T_2, \ldots) \rangle$ is to have a'_0 correspond to a leaf in one of the trees T_n. In such a situation, T'_i will be a complete binary tree of height i for $i = 1, \ldots, t$, $T'_{t+1} = Tree_{\mathcal{A}}(a_{i,n-1})$, and T'_i is the complete binary tree of size t for $i \geq r_2$. It follows that we can recognize the type of any element x which is in ω-chain in $\mathcal{A}$ by simply starting at a and computing $x_1, x_2, \ldots$ and $y_1, y_2, \ldots$ where $x_i = f^i(x)$, y_i is an element which is not equal to x_{i-1} such that $f(y_i) = x_i$ and $S_i = Tree_{\mathcal{A}}(y_i)$ until we see a j such that S_j and S_{j+1} are both complete binary trees of size t. Then we know that a belongs to an ω-chain of the form

$$\langle (a_0, a_1, a_2, \ldots), (T_1, T_2, \ldots) \rangle,$$

where T_i is a complete binary tree of height $t > r$. Moreover, we can find the corresponding a_0 in the tree S_{t+1}.

It follows that we can effectively determine whether an element in $a \in A$ is in $Fin(\mathcal{A})$, since its orbit will not have any elements c such that $Tree_{\mathcal{A}}(c)$ is a complete binary tree of size $t > r$ if $c \notin Fin(\mathcal{A})$, in which case we can effectively find a^t_0 such that $O_{\mathcal{A}}(c)$ is of type

$$\langle (a^t_0, a^t_1, a^t_2, \ldots), (T^t_1, T^t_2, \ldots) \rangle,$$

where T^t_i is a complete binary tree of height $t > r$. Thus if $\mathcal{B} = (B, g)$ is a highly computable locally finite (2,0):1 structure which is isomorphic to $\mathcal{A}$, then $Fin(\mathcal{B})$ is computable and hence, by our argument in Theorem 3.1, we can construct a computable isomorphism h_0 mapping $(Fin(\mathcal{A}), f)$ onto $(Fin(\mathcal{B}), g)$. Moreover, for any $t > r$, we can effectively find b_0^t such that the orbit of b_0^t is an ω-chain

$$\langle (b_0^t, b_1^t, b_2^t, \ldots), (T_1^t, T_2^t, \ldots) \rangle$$

where $b_i^t = g^i(b_0^t)$ and T_i^t is a complete binary tree of height $t > r$ for all $i \geq 1$. We can then define a bijection h_t from the orbit of a_0^t in $\mathcal{A}$ to the orbit of b_0^t in $\mathcal{B}$ by finding $c_i^t \in A$ and d_i^t in B for $i \geq 1$ such that $c_i^t \neq a_{i-1}^t$ and $f(c_i^t) = a_i^t$ and $d_i^t \neq b_{i-1}^t$ and $g(d_i^t) = b_i^t$ for $i \geq 1$ and defining h_t so that $h(a_i^t) = b_i^t$ for $i \geq 0$, $h(c_i^t) = d_i^t$ for $i \geq 1$, and ensuring that h_t restricted to $tree_{\mathcal{A}}(c_i^t)$ is the canonical isomorphism from $Tree_{\mathcal{A}}(c_i^t)$ onto $Tree_{\mathcal{B}}(d_i^t)$. It follows that $h = \bigcup_{t \geq 0} h_t$ is a computable isomorphism from $\mathcal{A}$ onto $\mathcal{B}$ so that $\mathcal{A}$ is computably categorical relative to the highly computable locally finite (2,0):1 structures.

It should be clear that we can construct infinitely many such examples by picking r, any computable set $S \subseteq \{r+1, r+2, \ldots\}$ and constructing a highly computable locally finite (2,0):1 structure $\mathcal{A} = (A, f)$ such that:

1. the only k-chains of $\mathcal{A}$ are of type $\langle (c_0, \ldots, c_{k-1}), (T_0, \ldots, T_{k-1}) \rangle$ where the height of T_i is $\leq r$,

2. the only ω-chains of $\mathcal{A}$ are of the form

$$\langle (a_0, a_1, a_2, \ldots), (T_1, T_2, \ldots) \rangle$$

where T_i is a complete binary tree of height $t \in S$, and

3. for each $t \in S$, $\mathcal{A}$ has s_r ω-chains of type

$$\langle (a_0, a_1, a_2, \ldots), (T_1, T_2, \ldots) \rangle$$

where T_i is a complete binary tree of height t such that $s_r \in (\mathbb{N} - \{0\}) \cup \{\omega\}$.

4 Non-Computably Categorical (2,0):1 Structures

In this section, we shall show that if we drop the hypothesis that a locally finite (2,0):1 structure $\mathcal{A} = (A, f)$ is highly computable or has explicitly computable cycle structure, then there are many examples of computable (2,0):1 structures which are not computably categorical structures even in the case where $Fin(\mathcal{A}) = A$. For example, we have the following theorem.

Theorem 4.1. *Suppose that $\mathcal{A} = (A, f)$ is a computable locally finite (2,0):1 structure such that $Fin(\mathcal{A}) = A$ and there exist two distinct types of orbits which are k-cycles,*
$D_1 = \langle (d_0, \ldots, d_{k-1}), (T_0, \ldots, T_{k-1}) \rangle$ and $D_2 = \langle (e_0, \ldots, e_{k-1}), (S_0, \ldots, S_{k-1}) \rangle$, such that $\mathcal{A}$ has infinitely many k-cycles which are isomorphic to D_i for $i = 1, 2$ and D_1 is embeddable into D_2. Then there exists a computable (2,0):1 structure $\mathcal{B} = (\mathbb{N}, g)$ which is isomorphic to $\mathcal{A}$ but is not computably isomorphic to $\mathcal{A}$.

Proof. First let ϕ be a 1:1 computable function which maps A onto the set of odd numbers O in $\mathbb{N}$. Define h on 0 so that ϕ is an isomorphism. Next on the even numbers E define h so that we create infinitely many copies

$$\langle (c_0^m, \ldots, c_{k-1}^m), (T_0^m, \ldots, T_{k-1}^m) \rangle_{m \geq 0}$$

of C_1 such that $c_0^0 < c_0^1 < c_0^2 < \cdots$ is a computable sequence. If $\mathcal{C} = (\mathbb{N}, h)$ is not computably isomorphic to $\mathcal{A}$, then we are done. Otherwise, we construct, in stages, a computable (2,0):1 structure $\mathcal{B} = (\mathbb{N}, g)$ which is isomorphic to $\mathcal{A}$ but not computably isomorphic to $\mathcal{C}$.

Let ϕ_e denote the partial computable function computed by the e-th Turing machine M_e and let $\phi_{e,s}(x)$ denote the result, if any, of carrying out the computation of M_e on input x for s steps. If this computation has not returned a value, then we write $\phi_{e,s}(x) \uparrow$ and if it has returned a value, then we write $\phi_{e,s}(x) \downarrow$.

Note that for any $a \in \mathbb{N}$, we can compute the sequence $a, h(a), h^2(a), \ldots$ long enough until we find the cycle $C_a = (z_0^a, \ldots, z_{k_a-1}^a)$ corresponding to the orbit of a where z_0^a is the smallest element of $\{z_0^a, \ldots, z_{k_a-1}^a\}$. It follows that we can compute the sequence $y_0 < y_1 < \cdots$ such that $Y = \{y_i : i \geq 0\} = \{z_0^a : a \in \mathbb{N}\}$. It follows that $c_0^0 < c_0^1 < \cdots$ is a computable subsequence of $y_0, y_1, \ldots$. That is, there is a computable increasing function q such that $y_{q(i)} = c_0^i$ for all $i \geq 0$.

For any $j \notin ran(q)$, we let $O_{\mathcal{C},s}(y_j)$ denote the set of $x \leq s$ such that either x is in the cycle $C(y_j) = (y_j = y_{j,0}, \ldots, y_{j,k_j-1})$ of h determined by y_j or $h^k(x) \in \{y_{j,0}, \ldots y_{j,k_j-1}\}$. For any $y_j \in ran(q)$, we let $O_{\mathcal{C},s}(y_j)$ denote $O_{\mathcal{C}}(y_j)$. Note that, by the construction, we can compute $O_{\mathcal{C}}(y_j)$ if $j \in ran(q)$. In either case, we shall call $O_{\mathcal{C},s}(y_j)$ the partial orbit of y_j at stage s.

We will use a finite injury priority argument to define a Δ_2^0 function $\psi : \mathbb{N} \to \mathbb{N}$ which is the limit of computable functions $\psi^{(s)}$ and the computable function g on $\mathbb{N}$ in stages so that at any stage s, if $j \leq s$ and $j \notin ran(q)$, then $\psi^{(s)}$ maps the partial orbit of y_j at stage s, $O_{\mathcal{C},s}(y_j)$, onto a partial orbit of g which is isomorphic to the orbit $O_{\mathcal{C},s}(y_j)$. On the elements of the form $y_{q(i)}$ where $q(i) \leq s$, we will define $\psi^{(s)}$ and g so that $\psi^{(s)}$ maps the orbit $O_{\mathcal{C},s}(y_{q(i)})$ into an orbit which is either isomorphic to D_1 or D_2. This way we will ensure that $\mathcal{B} = (\mathbb{N}, g)$ is isomorphic to $\mathcal{A}$. At each stage s, we will place Γ_j markers on the partial g-orbits which are isomorphic to the partial orbits $O_{\mathcal{C},s}(y_j)$ under $\phi^{(s)}$ for $j \leq s$ such that $j \notin ran(q)$.

We will have two sets of requirements that we must meet.

N_e: $lim_s \psi^{(s)}(x)$ exists for all $x \in O_{\mathcal{C}}(y_e)$ and ψ maps $O_{\mathcal{C}}(y_e)$ onto a $\mathcal{B}$-orbit which is isomorphic to $O_{\mathcal{C}}(y_e)$ if $e \notin ran(q)$ or is not a $\mathcal{B}$-orbit which is isomorphic to either D_1 and D_2 if $e \in ran(q)$.

P_e: Either

1. ϕ_e is not 1:1 on its domain,
2. there exists i such that ϕ_e is not defined on $O_{\mathcal{C}}(y_{q(i)})$, or

3. there exists i such that ϕ_e is defined on $O_{\mathcal{C}}(y_{q(i)})$, but $O_{\mathcal{C}}(y_{q(i)})$ is not isomorphic to $O_{\mathcal{B}}(\phi_e(y_{q(i)}))$.

Our basic strategy for meeting a requirement P_e is to simply compute $\phi_{e,s}(y_{q(0)}), \ldots, \phi_{e,s}(y_{q(s)})$ until we find an i such that the partial orbit of $\phi_{e,s}(y_{q(s)})$ under g as defined at stage s is not in the union of the partial orbits that are used to meet the requirements N_a for $a \leq e$ or P_b for $b < e$. That is, none of the elements of the partial orbit of $\phi_{e,s}(y_{q(s)})$ under g as defined at stage s have either Γ_j markers on them for $j \leq e$ or Δ_j markers on them for $j < e$. At this point, if the partial orbit of $\phi_{e,s}(y_{q(s)})$ is consistent with being isomorphic to D_1, then we extend g by using new elements of $\mathbb{N}$ so that the g-orbit $\phi_{e,s}(y_{q(s)})$ is isomorphic to D_2. We then put Δ_e markers on the elements of this orbit. If the g-orbit of $\phi_{e,s}(y_{q(s)})$ was being used to ensure that ψ is an isomorphism to some orbit y_j where $j > e$, then we simply use new elements to create a partial g-orbit which is isomorphic to the partial orbit $O_{\mathcal{C},s}(y_j)$.

Stage 0. Find the cycle $C(y_0) = (y_0 = y_0^0, y_1^0, \ldots, y_{k_0-1}^0)$. Then define g so that $g(0) = 1, g(1) = 2, \ldots, g(k_0 - 2) = k_0 - 1, g(k_0 - 1) = 0$ and define $\psi^{(0)}$ so that $\psi^{(0)}(y_{j,0}) = j$ for $0 \leq j \leq k_0 - 1$. Put Γ_0 markers on $0, \ldots, k_0 - 1$.

Stage s+1. Assume we have defined $\psi^{(s)}$ on the union of the partial orbits at stage s of all y_j for $j \leq s$ and g is defined on a finite subset I_s of $\mathbb{N}$ so that:

1. $\psi^{(s)}$ is a 1:1 function onto I_s,
2. for all y_j with $j \leq s$ and y_j not in the range of q, $\psi^{(s)}(O_{\mathcal{C},s}(y_j))$ is a g-orbit in (I_s, g) and $(\psi^{(s)}(O_{\mathcal{C},s}(y_j)), g)$ is isomorphic to $(O_{\mathcal{C},s}(y_j), h)$ and there are Γ_j markers on the elements of $\psi^{(s)}(O_{\mathcal{C},s}(y_j))$,
3. for all y_j with $j \leq s$ and y_j in the range of q, $\psi^{(s)}(O_{\mathcal{C}}(y_j))$ is contained in a g-orbit which is isomorphic to either D_1 or D_2.

First look for an $e \leq s+1$ such that $\phi_{e,s}$ is 1:1 on its domain, there currently are no elements with Δ_e markers, and there is a $j \leq s$ such that $j \in ran(q)$ and either $\phi_{e,s}(y_j)$ maps to an element outside of I_s or to an element of I_s which does not have a Γ_i marker on it or a Δ_i marker on it for any $i < j$. If no such e exists, then use elements from an initial segment of elements of $\mathbb{N} - I_s$ and define g on those elements to create a g-orbit which is isomorphic to $O_{\mathcal{C},s+1}(y_{s+1})$. Then define $\psi^{(s+1)}$ on $O_{\mathcal{C},s+1}(y_{s+1})$ so that it is an isomorphism which sends y_{s+1} to the least element of the cycle of the orbit and the map from any tree that feed into the cycle of y_j is the canonical map to the corresponding tree in the cycle of $\psi^{(s+1)}(y_{s+1})$. Put Γ_{s+1} markers on the elements of this new g-orbit if $s+1 \notin ran(q)$. Then let $\psi^{(s+1)} = \psi^{(s)}$ on $\bigcup_{j \leq s} O_{\mathcal{C},s}(y_j)$. Finally, for all $j \leq s$, $j \notin ran(q)$, use elements from an initial segment of $\mathbb{N} - (I_s \cup \psi^{(s+1)}(O_{\mathcal{C},s+1}(y_{s+1})))$ and define g on those elements so that the g-orbit of $\psi^{(s)}(y_j)$ is isomorphic to $O_{\mathcal{C},s+1}(y_j)$. Put Γ_j markers on the new elements in image of $\psi^{(s+1)}(O_{\mathcal{C},s+1}(y_j))$ and define $\psi^{(s+1)}$ on $O_{\mathcal{C},s+1}(y_j) - O_{\mathcal{C},s}(y_j)$ so that $\psi^{(s+1)}$ restricted to $O_{\mathcal{C},s+1}(y_j)$ is an isomorphism to the g-orbit of $\psi^{(s)}(y_j)$.

If such an e exists, then let e_{s+1} be the least such e. Then we have two cases.

Case 1. $\phi_{e,s}(y_{e_{s+1}}) \notin I_s$.
Then use an initial segment of elements in $\mathbb{N} - I_s - \{\phi_{e,s}(y_{e_{s+1}})\}$ and define g on those elements and $\phi_{e,s}(y_{e_{s+1}})$ to create a g-orbit which is isomorphic to D_2 where $\phi_{e,s}(y_{e_{s+1}})$ plays the role of the least element in the cycle of D_2. Define $\psi^{(s+1)}$ on $O_{\mathcal{C}}(y_{e_{s+1}})$ so that it is an isomorphism which sends $y_{e_{s+1}}$ to the least element of the cycle of the orbit, and the map from any tree that feeds into the cycle of y_j is the canonical map to the corresponding tree in the cycle of $\psi^{(s+1)}(y_{e_{s+1}})$. Put $\Delta_{e_{s+1}}$ markers on the g-orbit of $\psi^{(s+1)}(y_{e_{s+1}})$. In addition, create a g-orbit which is isomorphic to $O_{\mathcal{C},s+1}(y_{s+1})$. Then define $\psi^{(s+1)}$ on $O_{\mathcal{C},s+1}(y_{s+1})$ so that it is an isomorphism which sends the y_{s+1} to the least element of the cycle of the orbit, and the map from any tree that feeds into the cycle of y_j is the canonical map to the corresponding tree in the cycle of $\psi^{(s+1)}(y_{s+1})$. Put Γ_{s+1} markers on the elements of this new g-orbit if y_{s+1} is not in the range of q. Then let $\psi^{(s+1)} = \psi^{(s)}$ on $\bigcup_{j \leq s, y_j} O_{\mathcal{C},s}(y_j)$. Finally, for all $j \notin ran(q)$ for $j \leq s$, take elements from an initial segment of the elements of $\mathbb{N}$ which have not been used in the construction up to this point and define g on those elements so that the g-orbit of $\psi^{(s)}(y_j)$ is isomorphic to $O_{\mathcal{C},s+1}(y_j)$. Put Γ_j markers on the new elements in the image $\psi^{(s+1)}(O_{\mathcal{C},s+1}(y_j))$ and define $\psi^{(s+1)}$ on $O_{\mathcal{C},s+1}(y_j) - O_{\mathcal{C},s}(y_j)$ so that $\psi^{(s+1)}$ restricted to $O_{\mathcal{C},s+1}(y_j)$ is an isomorphism to the g-orbit of $\psi^{(s)}(y_j)$.

Case 2. $\phi_{e,s}(y_{e_{s+1}}) \in I_s$.
Consider the current g-orbit O of $\phi_{e,s}(y_{e_{s+1}})$. If $\psi^{(s)}$ induces an embedding of O into D_1, then use elements from an initial segment of $\mathbb{N} - I_s$ and define g on those elements so that we extend O to an orbit which is isomorphic to D_2. Put $\Delta_{e_{s+1}}$ markers on all the elements of this new D_2-orbit. Now suppose that the elements of O had Γ_r markers on them for some r, where $e_{s+1} < r \leq s$. Then we remove all those Γ_r markers and take an initial segment of the elements of $\mathbb{N}$ that have not been used up to this point and define g to create a new copy of $O_{\mathcal{C},s+1}(y_j)$. Then define $\psi^{(s+1)}$ on $O_{\mathcal{C},s+1}(y_j)$ so that it is an isomorphism which sends the y_j to the least element of the cycle of the new g-orbit, and the the map from any tree that feeds into the cycle of y_j is the canonical map to the corresponding tree in the cycle of $\psi^{(s+1)}(y_j)$. Similarly, define $\psi^{(s+1)}$ on $O_{\mathcal{C}}(y_{e_{s+1}})$ so that it is an isomorphism which sends $y_{e_{s+1}}$ to the least element of the cycle of the new g-orbit which is isomorphic to D_2, and the map from any tree that feeds into the cycle of y_j is the canonical map to the corresponding tree in the cycle of $\psi^{(s+1)}(y_{e_{s+1}})$. In addition, create a g-orbit which is isomorphic to $O_{\mathcal{C},s+1}(y_{s+1})$, and define $\psi^{(s+1)}$ on $O_{\mathcal{C},s+1}(y_{s+1})$ so that it is an isomorphism which sends y_{s+1} to the least element of the cycle of the orbit and the the map from any tree that feeds into the cycle of y_j is the canonical map to the corresponding tree in the cycle of $\psi^{(s+1)}(y_{s+1})$. Put Γ_{s+1} markers on the elements of this new g-orbit if y_{s+1} is not in the range of q. Then let $\psi^{(s+1)} = \psi^{(s)}$ on $\bigcup_{j \in \{0,\ldots,s\}-\{r\}} O_{\mathcal{C},s}(y_j)$. Finally, for all $j \notin ran(q)$ for $j \leq s$,

take elements from an initial segment of elements that have not currently been used in the construction and define g on those elements so that the g-orbit of $\psi^{(s)}(y_j)$ is isomorphic to $O_{\mathcal{C},s+1}(y_j)$. Put Γ_j markers on the new elements in the image $\psi^{(s+1)}(O_{\mathcal{C},s+1}(y_j))$ and define $\psi^{(s+1)}$ on $O_{\mathcal{C},s+1}(y_j) - O_{\mathcal{C},s}(y_j)$ so that $\psi^{(s+1}$ restricted to $O_{\mathcal{C},s+1}(y_j)$ is an isomorphism to the g-orbit of $\psi^{(s)}(y_j)$.

If $\psi^{(s)}$ does not induce an embedding of O into D_1, then O is inconsistent with having its pre-image under $\psi^{(s)}$ isomorphic to D_1. In this case, put $\Delta_{e_{s+1}}$ markers on all the elements of O. Then use elements from an initial segment of elements of $\mathbb{N} - I_s$ and define g on those elements to create a g-orbit which is isomorphic to $O_{\mathcal{C},s+1}(y_{s+1})$, and define $\psi^{(s+1)}$ on $O_{\mathcal{C},s+1}(y_{s+1})$ so that it is an isomorphism which sends y_{s+1} to the least element of the cycle of the orbit, and the map from any tree that feeds into the cycle of y_j is the canonical map to the corresponding tree in the cycle of $\psi^{(s+1)}(y_{s+1})$. Put Γ_{s+1} markers on the elements of this new g-orbit if y_{s+1} is not in the range of q. Then let $\psi^{(s+1)} = \psi^{(s)}$ on $\bigcup_{j\leq s} O_{\mathcal{C},s}(y_j)$. Finally, for all $j \notin ran(q)$ for $j \leq s$, use elements from an initial segment of $\mathbb{N} - (I_s \cup \psi^{(s+1)}(O_{\mathcal{C},s+1}(y_{s+1})))$, and define g on those elements so that the g-orbit of $\psi^{(s)}(y_j)$ is isomorphic to $O_{\mathcal{C},s+1}(y_j)$. Put Γ_j markers on the new elements in the image $\psi^{(s+1)}(O_{\mathcal{C},s+1}(y_j))$ and define $\psi^{(s+1)}$ on $O_{\mathcal{C},s+1}(y_j) - O_{\mathcal{C},s}(y_j)$ so that $\psi^{(s+1)}$ restricted to $O_{\mathcal{C},s+1}(y_j)$ is an isomorphism to the g-orbit of $\psi^{(s)}(y_j)$.

This completes the construction. It is easy to see that each step is effective and, hence, g is computable since we never change the value of $g(x)$ for any x.

Next observe that if e_{s+1} is defined, then there is a $e_{s+1} \in ran(q)$ and our action ensures that $\phi_{e_{s+1}}(y_{e_{s+1}})$ has $\mathcal{B}$-orbit which is not isomorphic to D_1. Thus ϕ_e can not be an isomorphism from $\mathcal{C}$ onto $\mathcal{B}$. Moreover, we will never remove the $\Delta_{e_{s+1}}$ markers that we placed at stage $s+1$ which means that we will never take an action to meet the requirement $P_{e_{s+1}}$ after stage $s+1$.

It is a straightforward induction to show that for each $j \notin ran(q)$, the $\lim_{s\to\infty} \psi^{(s)}(x) = \psi(x)$ exists for $x \in O_{\mathcal{C}}(y_j)$ and that ψ restricted to $(O_{\mathcal{C}}(y_j), h)$ is an isomorphism onto $(O_{\mathcal{B}}(\phi(y_j)), g)$. That is, we can only be forced to have $\psi^{(s)}(x) \neq \psi^{(s+1)}(x)$ for any $x \in O_{\mathcal{C},s}(y_j)$ for an $s \geq j$ if we are taking an action to meet a requirement P_e for $e \leq j$. Since we can only take an action for P_e once, it follows that there will be a t large enough so that $O_{\mathcal{C},t}(y_j) = O_{\mathcal{C}}(y_j)$ and $\psi^{(t)}(x) = \psi^{(s)}(x)$ for all $s \geq t$ and $x \in O_{\mathcal{C}}(y_j)$. By the construction, at each stage $s \geq j$, $\psi^{(s)}$ is an isomorphism from $(O_{\mathcal{C},s}(y_j), h)$ to $(\psi^{(s)}(O_{\mathcal{C},s}(y_j)), g)$. Thus ψ is an isomorphism from $(O_{\mathcal{C}}(y_j), h)$ onto $\psi(O_{\mathcal{C}}(y_j)), g)$. A similar argument will show that for each $j \notin ran(q)$, the $\lim_{s\to\infty} \psi^{(s)}(x) = \psi(x)$ exists for $x \in O_{\mathcal{C}}(y_j)$ and that ψ restricted to $(O_{\mathcal{C}}(y_j), h)$ either maps it into a g-orbit which is isomorphic to either D_1 or D_2. It then follows that $\mathcal{B} = (\mathbb{N}, g)$ is isomorphic to $\mathcal{C}$.

Thus the only thing that we have to do to show that $\mathcal{B}$ is not computably isomorphic to $\mathcal{C}$ is to show that we satisfy all the requirements P_e. Suppose for a contradiction, that ϕ_e is an isomorphism from $\mathcal{B}$ into $\mathcal{C}$. Then there will be a stage t large enough so that:

(i) we never take any action for a requirement P_i with $i < e$ after stage t,

(ii) $O_{\mathcal{C},t}(y_j) = O_{\mathcal{C}}(y_j)$ for all $j \leq e$ such that $j \notin ran(q)$,

(iii) for all $j < e$, $\psi^{(s)}(x) = \psi^{(t)}(x)$ for all $x \in O_{\mathcal{C}}(y_j)$, and

(iv) $\phi_{e,t}(y_r)$ is defined for all $r \leq 1 + \sum_{j \leq e} card(O_{\mathcal{C}}(y_j))$.

Since we are assuming that ϕ_e is an isomorphism from $\mathcal{C}$ to $\mathcal{B}$, there must be y_j in the range of q such that $\phi_e(y_j)$ maps to an element which does not have a Γ_r marker on it for any $r < e$. But then y_j could be used to satisfy the requirement P_e at stage $t+1$. Thus either $e_{t+1} = e$ in which case we take an action at stage $s+1$ to ensure that $O_{\mathcal{B}}(\phi_e(y_j))$ is not isomorphic to D_1 or there is an $s \leq t$ such that $e_s = e$. In either case, our construction ensures that $O_{\mathcal{B}}(\phi_e(y_j))$ is isomorphic to D_1. Thus there can be no such e and, hence, $\mathcal{B}$ is not computably isomorphic to $\mathcal{C}$. □

Another simple condition which ensures that a computable locally finite (2,0):1 structure $\mathcal{A} = (A, f)$ is not computably categorical is that there is an computable increasing chain of orbits which are k-cycles. That is, we say that $\mathcal{A} = (A, f)$ has a **highly computable ascending chain of k-cycles** if there is a computable sequence of elements $a_0^0, a_0^1, \ldots$ and a computable function z such that for each $i \geq 0$:

1. $O_{\mathcal{A}}(a_0^i)$ is a k-cycle $D_i = \langle (a_0^i, \ldots, a_{k-1}^i), (T_0^i, \ldots, T_{k-1}^i) \rangle$,
2. $z(i)$ is the canonical index of $O_{\mathcal{A}}(a_0^i)$, and
3. D_i is embeddable into D_{i+1}.

Then we have the following theorem.

Theorem 4.2. *Suppose that $\mathcal{A} = (A, f)$ is a computable (2,0):1 structure and $\mathcal{A}$ has a highly computable ascending chain of k-cycles for some k. Then there is a computable (2,0):1 structure $\mathcal{B} = (\mathbb{N}, g)$ such that $\mathcal{B}$ is isomorphic but not computably isomorphic to $\mathcal{A}$.*

Proof. Our proof is a slight modification of the proof of Theorem 4.1. That is, if $A \neq \mathbb{N}$, then let $A = \{a_0 < a_1 < \cdots\}$. Then let $\theta(a_i) = i$ and define g on $\mathbb{N}$ so that θ is an isomorphism from $\mathcal{A}$ onto $\mathcal{C} = (\mathbb{N}, g)$. Then, as in the proof of Theorem 4.1, we let $y_0 < y_1 < \cdots$ be the set of the least elements that appear in the cycles of $\mathcal{C}$. Because $\mathcal{A}$ has a highly computable ascending chain of k-cycles, there is an increasing computable function q such that $y_{q(0)} < y_{q(1)} < \cdots$ and $O_{y_{q(i)}}$ is a k-cycle $D_i = \langle (y_{q(i)} = y_0^{q(i)}, \ldots y_{k-1}^{q(i)}), (T_0^{q(i)}, \ldots, T_{k-1}^{q(i)}) \rangle$ such that D_i is embeddable in D_{i+1} and we can uniformly compute a canonical index of $O_{\mathcal{C}}(y_{q(i)})$.

For any $j \notin ran(q)$, we let $O_{\mathcal{C},s}(y_j)$ denote the set of $x \leq s$ such that either x is in the cycle $C(y_j) = (y_j = y_{j,0}, \ldots, y_{j,k_j-1})$ of h determined by y_j or

$h^k(x) \in \{y_{j,0}, \dots, y_{j,k_j-1}\}$. For any $j \in ran(q)$, we let $O_{\mathcal{C},s}(y_j)$ denote $O_{\mathcal{C}}(y_j)$. Note that, by the construction, we can compute $O_{\mathcal{C}}(y_j)$ if $j \in ran(q)$. In either case, we shall call $O_{\mathcal{C},s}(y_j)$ the partial orbit of y_j at stage s.

We will use a finite injury priority argument to define a Δ^0_2 function $\psi : \mathbb{N} \to \mathbb{N}$ which is the limit of a computable sequence of functions $\psi^{(s)}$, and a computable function g on $\mathbb{N}$ in stages so that at any stage s, if $j \leq s$, then $\psi^{(s)}$ maps the partial orbit of y_j at stage s, $O_{\mathcal{C},s}(y_j)$, onto a partial orbit of g which is isomorphic to the orbit $O_{\mathcal{C},s}(y_j)$. At each stage s, we will place Γ_j markers on on the partial g-orbits which are isomorphic to the partial orbits $O_{\mathcal{C},s}(y_j)$ under $\phi^{(s)}$.

We will have two sets of requirements that we must meet.

N_e: $lim_s \psi^{(s)}(x)$ exists for all $x \in O_{\mathcal{C}}(y_e)$ and ψ maps $O_{\mathcal{C}}(y_e)$ onto a $\mathcal{B}$-orbit which is isomorphic to $O_{\mathcal{C}}(y_e)$.

P_e: Either

1. ϕ_e is not 1:1 on its domain,
2. there exists i such that ϕ_e is not defined on $O_{\mathcal{C}}(y_{q(i)})$, or
3. there exists i such that ϕ_e is defined on $O_{\mathcal{C}}(y_{q(i)})$, but $O_{\mathcal{C}}(y_{q(i)})$ is not isomorphic to $O_{\mathcal{B}}(\phi_e(y_{q(i)}))$.

Our basic strategy for meeting a requirement P_e is to simply compute $\phi_{e,s}(y_{q(0)}), \dots, \phi_{e,s}(y_{q(s)})$ until we find an i such that the partial orbit of $\phi_{e,s}(y_{q(s)})$ under g as defined at stage s is not in the union of the partial orbits that are used to meet the requirements N_a for $a \leq e$ or P_b for $b < e$. That is, none of the elements of the partial orbit of $\phi_{e,s}(y_{q(s)})$ under g as defined at stage s have either Γ_j markers on them for $j \leq e$ or Δ_j markers on them for $j < e$. At this point, if the partial orbit of $\phi_{e,s}(y_{q(s)})$ is consistent with being isomorphic to D_i, then we extend g by using new elements of $\mathbb{N}$ so that the g-orbit $\phi_{e,s}(y_{q(s)})$ is isomorphic to D_j for some $j > q(s)$. We then put Δ_e markers on the elements of this orbit. If the g-orbit $\phi_{e,s}(y_{q(s)})$ was being used to ensure that ψ is an isomorphism to some orbit y_j where $j > e$, then we simply use new elements to create a partial g-orbit which is isomorphic to the partial orbit $O_{\mathcal{C},s}(y_j)$.

Stage 0. Find the cycle $C(y_0) = (y_0 = y^0_0, y^0_1, \dots, y^0_{k_0-1})$. Then define g so that $g(0) = 1, g(1) = 2, \dots, g(k_0 - 2) = k_0 - 1, g(k_0 - 1) = 0$ and define $\phi^{(0)}$ so that $\phi^{(0)}(y_{j,0}) = j$ for $0 \leq j \leq k_0 - 1$. Put Γ_0 markers on $0, \dots, k_0 - 1$. Let $\ell_0 = 0$.

Stage s+1. Assume we have defined $\psi^{(s)}$ on the union of the partial orbits at stage s of all y_j for $j \leq \ell_s$ where $\ell_s \geq s$ and g is defined on a finite subset I_s of $\mathbb{N}$ so that $\psi^{(s)}$ is 1:1 function onto I_s and for all $j \leq \ell_s$, $(\mathcal{O}_{C,s}(y_j), h)$ is isomorphic to $(\psi^{(s)}(\mathcal{O}_{C,s}(y_j)), g)$.

First look for an $e \leq s+1$ such that $\phi_{e,s}$ is 1:1 on its domain, there currently are no elements with Δ_e markers, and there is a $j \leq s$ such that $j \in ran(q)$

and either $\phi_{e,s}(y_j)$ maps to an element outside of I_s or to an element of I_s which does not have a Γ_i marker on it or a Δ_i marker on it for some $i < j$. If no such e exists, then set $\ell_{s+1} = 1 + \ell_s$ and use elements from an initial segment of elements of $\mathbb{N} - I_s$ and define g on those elements to create a g-orbit which is isomorphic to $O_{\mathcal{C},s+1}(y_{\ell_s+1})$. Define $\psi^{(s+1)}$ on $O_{\mathcal{C},s+1}(y_{\ell_s+1})$ so that it is an isomorphism which sends y_{ℓ_s+1} to the least element of the cycle of that orbit, and the map from any trees that feed into the cycle of y_{ℓ_s+1} is the canonical map to the corresponding tree in the cycle of $\psi^{(s+1)}(y_{\ell_s+1})$. Put Γ_{ℓ_s+1} markers on the elements of this new g-orbit. Then let $\psi^{(s+1)} = \psi^{(s)}$ on $\bigcup_{j\leq s} O_{\mathcal{C},s}(y_j)$. Finally, for all $j \leq \ell_s$, use elements from an initial segment of $\mathbb{N}-(I_s \cup \psi^{(s+1)}(O_{\mathcal{C},s+1}(y_{s+1})))$ and define g on those elements so that the g-orbit of $\psi^{(s)}(y_j)$ is isomorphic to $O_{\mathcal{C},s+1}(y_j)$. Put Γ_j markers on these new elements in the image $\psi^{(s+1)}(O_{\mathcal{C},s+1}(y_j))$ and define $\psi^{(s+1)}$ on $O_{\mathcal{C},s+1}(y_j) - O_{\mathcal{C},s}(y_j)$ so that $\psi^{(s+1)}$ restricted to $O_{\mathcal{C},s+1}(y_j)$ is an isomorphism to the g-orbit of $\psi^{(s)}(y_j)$.

If such an e exists, then let e_{s+1} be the least such e. Then we have two cases.

Case 1. $\phi_{e,s}(y_{e_{s+1}}) \notin I_s$.
In this case, we use an initial segment of elements in $\mathbb{N} - I_s - \{\phi_{e,s}(y_{e_{s+1}})\}$ and define g on those elements and $\phi_{e,s}(y_{e_{s+1}})$ to create an g-orbit which is isomorphic to $D_{q(\ell_s+1)}$ where $\phi_{e,s}(y_{e_{s+1}})$ plays the role of the least element in the cycle of $D_{q(\ell_s+1)}$. Put Δ_{ℓ_s+1} markers on this orbit. Let $\ell_{s+1} = 1 + q(\ell_s + 1)$. Similarly, for each j where $\ell_s < j < \ell_{s+1}$, we use new elements from an initial segment of elements which have not been used up to this point and define g on those elements to create g-orbits which are isomorphic to $O_{\mathcal{C},s+1}(y_j)$ and put Γ_j markers on these new elements. Define $\psi^{(s+1)}$ on $O_{\mathcal{C}}(y_{\ell_{s+1}})$ so that it is an isomorphism which sends $y_{\ell_{s+1}}$ to $\phi_{e,s}(y_{e_{s+1}})$, and the map from any tree that feeds into the cycle of y_j is the canonical map to the corresponding tree in the cycle of $\psi^{(s+1)}(y_{e_{s+1}})$. Put $\Delta_{e_{s+1}}$ markers on the elements of the g-orbit of $\psi^{(s+1)}(y_{e_{s+1}})$. Similarly, for $\ell_s < j < \ell_{s+1}$, define $\phi^{(s+1)}(y_j)$ to be the least element in the cycle of the new g-orbit we created to be isomorphic to $O_{\mathcal{C},s+1}(y_j)$, and the map from any tree that feeds into the cycle of y_j is the canonical map to the corresponding tree in the cycle of the g-orbit that we created to be isomorphic to $O_{\mathcal{C},s+1}(y_j)$. Then let $\psi^{(s+1)} = \psi^{(s)}$ on $\bigcup_{j\leq s} O_{\mathcal{C},s}(y_j)$. Finally, for all $j \leq \ell_s$, take elements from an initial segment of $\mathbb{N}$ which have not been used up to this point and define g on those elements so that the g-orbit of $\psi^{(s)}(y_j)$ is isomorphic to $O_{\mathcal{C},s+1}(y_j)$. Put Γ_j markers on the new elements in the image $\psi^{(s+1)}(O_{\mathcal{C},s+1}(y_j))$ and define $\psi^{(s+1)}$ on $O_{\mathcal{C},s+1}(y_j) - O_{\mathcal{C},s}(y_j)$ so that $\psi^{(s+1)}$ restricted to $O_{\mathcal{C},s+1}(y_j)$ is an isomorphism to the g-orbit of $\psi^{(s)}(y_j)$.

Case 2. $\phi_{e,s}(y_{e_{s+1}}) \in I_s$.
Consider the current g-orbit O of $\phi_{e,s}(y_{e_{s+1}})$. If $\psi^{(s)}$ induces an embedding of O into $D_{e_{s+1}}$, then use an initial segment of $\mathbb{N} - I_s$ to add new elements and define g to extend O to an orbit which is isomorphic to $D_{q(\ell_s+1)}$. Put $\Delta_{e_{s+1}}$ markers on all elements of this new $D_{q(\ell_s+1)}$-orbit. Define $\psi^{(s+1)}$ on

$\mathcal{O}_{\mathcal{C}}(y_{q(\ell_s+1)})$ so that it is an isomorphism from $(\mathcal{O}_{\mathcal{C}}(y_{q(\ell_s+1)}), h)$ onto this new g-orbit of $\phi_{e,s}(y_{e_{s+1}})$. Now if the elements of O had Γ_r markers on them for some r where $e_{s+1} < r \leq \ell_s$, then remove all those Γ_r markers. Then use these now unmarked elements for initial segments of those elements that have currently not been used in the construction, and define g on those elements to create a new copy of $O_{\mathcal{C},s+1}(y_r)$. Then define $\psi^{(s+1)}$ on $O_{\mathcal{C},s+1}(y_r)$ so that it is an isomorphism which sends y_r to the least element of the cycle of the new g-orbit, and the map from any tree that feeds into the cycle of y_j is the canonical map to the corresponding tree in the cycle of $\psi^{(s+1)}(y_r)$. Set $\ell_{s+1} = q(\ell_s + 1)$. Also use an initial segment of those elements that have currently not been used in the construction up to this point and define g on those elements to create a new copy of $O_{\mathcal{C},s+1}(y_i)$ for all i such that $\ell_s < i < \ell_{s+1}$. Define $\psi^{(s+1)}$ on $O_{\mathcal{C},s+1}(y_i)$ so that it is an isomorphism which sends y_i to the least element of the cycle of the new g-orbit isomorphic to $O_{\mathcal{C},s+1}(y_i)$, such that the map from any tree that feeds into the cycle of y_i is the canonical map to the corresponding tree in the cycle of $\psi^{(s+1)}(y_i)$. Also put Γ_i markers on the new g-orbit of $\psi^{(s+1)}(y_i)$. Then let $\psi^{(s+1)} = \psi^{(s)}$ on $\bigcup_{j\in\{0,\ldots,s\}-\{r\}} O_{\mathcal{C},s}(y_j)$. Finally, for all $j \leq s$, $j \notin ran(q)$, take elements from an initial segment of elements that have not currently been used in the construction and define g on those elements so that the g-orbit of $\psi^{(s)}(y_j)$ is isomorphic to $O_{\mathcal{C},s+1}(y_j)$. Put Γ_j markers on the new elements in the image $\psi^{(s+1)}(O_{\mathcal{C},s+1}(y_j))$ and define $\psi^{(s+1)}$ on $O_{\mathcal{C},s+1}(y_j) - O_{\mathcal{C},s}(y_j)$ so that $\psi^{(s+1)}$ restricted to $O_{\mathcal{C},s+1}(y_j)$ is an isomorphism to the g-orbit of $\psi^{(s)}(y_j)$.

If $\psi^{(s)}$ does not induce an embedding of O into $D_{e_{s+1}}$, then O is inconsistent with having its pre-image under $\psi^{(s)}$ isomorphic to $D_{e_{s+1}}$. In this case, put $\Delta_{e_{s+1}}$ markers on all the elements of O. Then set $\ell_{s+1} = \ell_s + 1$. Next use elements from an initial segment of elements of $\mathbb{N} - I_s$ and define g on those elements to create a g-orbit which is isomorphic to $O_{\mathcal{C},s+1}(y_{\ell_s+1})$ and define $\psi^{(s+1)}$ on $O_{\mathcal{C},s+1}(y_{\ell_s+1})$ so that it is an isomorphism which sends y_{ℓ_s+1} to the least element of the cycle of the orbit. and the map from any tree that feeds into the cycle of y_j is the canonical map to the corresponding tree in the cycle of $\psi^{(s+1)}(y_{s+1})$. Put Γ_{ℓ_s+1} markers on the elements of this new g-orbit. Then let $\psi^{(s+1)} = \psi^{(s)}$ on $\bigcup_{j\leq s} O_{\mathcal{C},s}(y_j)$. Finally, for all $j \leq s$, $j \notin ran(q)$, use elements from an initial segment of $\mathbb{N} - (I_s \cup \psi^{(s+1)}(O_{\mathcal{C},s+1}(y_{s+1})))$ and define g on those elements so that the g-orbit of $\psi^{(s)}(y_j)$ is isomorphic to $O_{\mathcal{C},s+1}(y_j)$. Put Γ_j markers on the new elements in image of $\psi^{(s+1)}(O_{\mathcal{C},s+1}(y_j))$ and define $\psi^{(s+1)}$ on $O_{\mathcal{C},s+1}(y_j) - O_{\mathcal{C},s}(y_j)$ so that $\psi^{(s+1)}$ restricted to $O_{\mathcal{C},s+1}(y_j)$ is an isomorphism to the g-orbit of $\psi^{(s)}(y_j)$.

This completes the construction. It is easy to see that each step is effective and, hence, g is computable since we never change the value of $g(x)$ for any x.

Next observe that if e_{s+1} is defined, then $e_{s+1} \in ran(q)$ and our action ensures that $\phi_{e_{s+1}}(y_{e_{s+1}})$ has a $\mathcal{B}$-orbit which is not isomorphic to $D_{e_{s+1}}$. Thus ϕ_e can not be an isomorphism from $\mathcal{C}$ onto $\mathcal{B}$. Moreover, we will never remove the $\Delta_{e_{s+1}}$ markers that we placed at stage $s+1$, which means that we will never take an action to meet the requirement $P_{e_{s+1}}$ after stage $s+1$.

It is a straightforward induction to show that for each y_j such that $j \geq 0$, the $\lim_{s\to\infty} \psi^{(s)}(x) = \psi(x)$ exists for $x \in O_{\mathcal{C}}(y_j)$ and that ψ restricted to $(O_{\mathcal{C}}(y_j), h)$ is an isomorphism on $O_{\mathcal{B}}(\phi(y_j))$. That is, we can only be forced to have $\psi^{(s)}(x) \neq \psi^{(s+1)}(x)$ for any $x \in O_{\mathcal{C},s}(y_j)$ for an $s \geq j$ if we are taking an action to meet a requirement P_e for $e \leq j$. Since we can only take an action for P_e once, it follows that there will be a t large enough so that $O_{\mathcal{C},t}(y_j) = O_{\mathcal{C}}(y_j)$ and $\psi^{(t)}(x) = \psi^{(s)}(x)$ for all $s \geq t$ and $x \in O_{\mathcal{C}}(y_j)$. By the construction, at each stage $s \geq j$, $\psi^{(s)}$ is an isomorphism from $(O_{\mathcal{C},s}(y_j), h)$ to $(\psi^{(s)}(O_{\mathcal{C},s}(y_j)), g)$. Thus ψ is an isomorphism form $(O_{\mathcal{C}}(y_j), h)$ onto $\psi((O_{\mathcal{C}}(y_j)), g)$.

Thus the only thing that we need to do to show that $\mathcal{B}$ is not computably isomorphic to $\mathcal{C}$ is to show that we satisfy all the requirements P_e. Suppose for a contradiction, that ϕ_e is an isomorphism from $\mathcal{B}$ into $\mathcal{C}$. Then there will be a stage t large enough so that: (i) we never take any action for a requirement P_i with $i < e$ after stage t, (ii) $O_{\mathcal{C},t}(y_j) = O_{\mathcal{C}}(y_j)$ for all $j < e$, (iii) for all $j < e$, $\psi^{(s)}(x) = \psi^{(t)}(x)$ for all $x \in O_{\mathcal{C}}(y_j)$, and (iv) $\phi_{e,t}(y_r)$ is defined for all $r \leq 1 + \sum_{j \leq e} card(O_{\mathcal{C}}(y_j))$. Since we are assuming that ϕ_e is an isomorphism from $\mathcal{C}$ to $\mathcal{B}$, there must be y_j in the range of q such that $\phi_e(y_j)$ maps to an element which does not have a Γ_r marker on it for any $r < e$. But then y_j could be used to satisfy the requirement P_e at stage $t+1$. Thus either $e_{t+1} = e$, in which case we take an action at stage $s+1$ to ensure that $O_{\mathcal{B}}(\phi_e(y_j))$ is not isomorphic to $\mathcal{O}_{\mathcal{C}}(y_j)$, or there is an $s \leq t$ such that $e_s = e$. In either case, our construction ensures that $O_{\mathcal{B}}(\phi_e(y_j))$ is isomorphic to $\mathcal{O}_{\mathcal{C}}(y_j)$. Thus there can be no such e and, hence, $\mathcal{B}$ is not computably isomorphic to $\mathcal{C}$.

□

Next we give two simple examples where, even though we are given quite a bit of information about the possible isomorphism types of k-cycles in a computable (2,0):1 structure $\mathcal{A}$, there still exists a computable (2,0):1 structure which is isomorphic to $\mathcal{A}$ but is not computably isomorphic to $\mathcal{A}$.

For the first example, we construct locally finite computable (2,0):1 structures $\mathcal{A} = (\mathbb{N}, f)$ and $\mathcal{B} = (\mathbb{N}, g)$ such that: (i) $Fin(\mathcal{A}) = Fin(\mathcal{B}) = \mathbb{N}$, (ii) $\mathcal{A}$ and $\mathcal{B}$ are isomorphic but not computably isomorphic, and (iii) for any $k \geq 1$, there are only two types of k-cycles $\langle (c_0, \ldots, c_{k-1}), (T_0, \ldots, T_{k-1}) \rangle$, one, which we shall call E_k, where all the T_i are one-element binary trees and one, which we shall call F_k, where all the trees T_i are three-element binary trees. Thus, for example, E_4 and F_4 are pictured in Figure 10.

In fact, we can construct $\mathcal{A} = (\mathbb{N}, f)$ and $\mathcal{B} = (\mathbb{N}, g)$ so that for each $k \geq 1$, $\mathcal{A}$ and $\mathcal{B}$ have exactly one k-cycle isomorphic to E_k, and either 1 or 2 k-cycles which are isomorphic to F_k such that $\mathcal{A}$ and $\mathcal{B}$ are not computably isomorphic.

The construction of $\mathcal{A}$ and $\mathcal{B}$ is quite easy. That is, on the even numbers E, define f and g so that we have computable (2,0):1 structures which have exactly one copy of $E_k = \langle (c_0^k, \ldots, c_{k-1}^k), (T_0^k, \ldots, T_{k-1}^k) \rangle$ and one copy of $F_k = \langle (d_0^k, \ldots, d_{k-1}^k), (S_0^k, \ldots, S_{k-1}^k) \rangle$ for each $k \geq 1$. Thus each T_i^j is a one-element tree and each S_i^j is a three element binary tree. Then for each k, attempt to compute $\phi_k(c_0^k)$. We then have two cases.

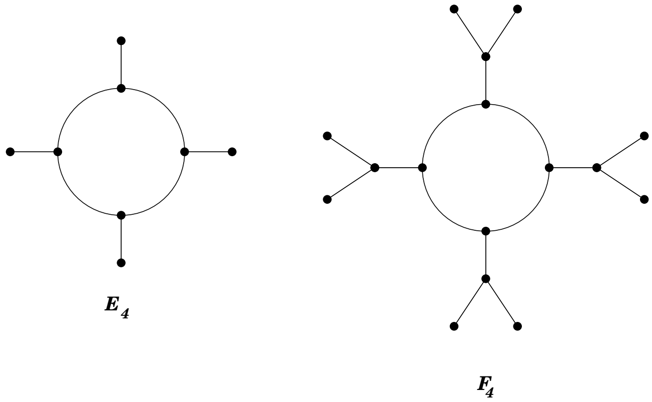

Figure 10: The cycle types E_4 and F_4.

Case 1. $\phi_k(c_0^k)\downarrow$ and $\phi_k(c_0^k) \in \{c_0^k, \ldots, c_{k-1}^k\}$.
In this case, we will use new elements from the odd numbers and define f on these odd numbers to extend $\langle(c_0^k, \ldots, c_{k-1}^k), (T_0^k, \ldots, T_{k-1}^k)\rangle$ to a cycle of type F_k. We shall then use new odd numbers and define f on these numbers to create a new cycle of type E_k in $\mathcal{A}$. We will also use new odd numbers and define g on those numbers to create a new cycle of type F_k. This will ensure that ϕ_e cannot be an isomorphism from $\mathcal{A}$ onto $\mathcal{B}$ since ϕ_k will map an element of a k-cycle of type F_k into a k-cycle of type E_k. Thus in this case, $\mathcal{A}$ and $\mathcal{B}$ will have one k-cycle of type E_k and two k-cycles of type F_k.

Case 2. $\phi_k(c_0^k)\uparrow$, or $\phi_k(c_0^k)\downarrow$ and $\phi_k(c_0^k) \notin \{c_0^k, \ldots, c_{k-1}^k\}$.
In this case, we do nothing to the k-cycles in $\mathcal{A}$ or $\mathcal{B}$. Then we know that ϕ_k cannot be an isomorphism from $\mathcal{A}$ onto $\mathcal{B}$. In this case, both $\mathcal{A}$ and $\mathcal{B}$ will have one k-cycle of type E_k and one k-cycle of type F_k.

Note that there are infinitely many k such that ϕ_k is the identity so that we will be in Case 1 infinitely often and, hence, f and g will be defined on all of $\mathbb{N}$. It is easy to see that $\mathcal{A} = (\mathbb{N}, f)$ and $\mathcal{B} = (\mathbb{N}, g)$ are computable (2,0):1 structures such that $\mathcal{A}$ and $\mathcal{B}$ are isomorphic but not computably isomorphic.

Next we construct similar examples of locally finite (2,0):1 structures $\mathcal{A} = (\mathbb{N}, f)$ and $\mathcal{B} = (\mathbb{N}, g)$ such that: (i) $Fin(\mathcal{A}) = Fin(\mathcal{B}) = \mathbb{N}$, (ii) $\mathcal{A}$ and $\mathcal{B}$ are isomorphic but not computably isomorphic, and (iii) for any $k \geq 1$, there are exactly two types of k-cycles where either the two cycle types are E_k and F_k or the cycle types are F_k and $G_k = \langle(b_0^k, \ldots, b_{k-1}^k), (R_0^k, \ldots, R_{k-1}^k)\rangle$, where each R_i^k is a complete binary tree of height 2. For example, G_4 is pictured in Figure 11.

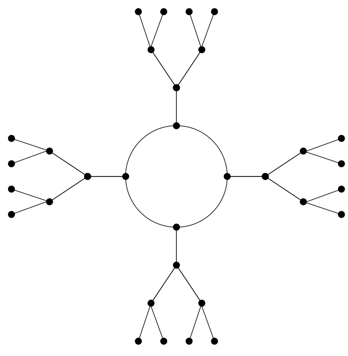

Figure 11: The cycle type G_4.

The construction of $\mathcal{A}$ and $\mathcal{B}$ is very similar. That is, on the even numbers E, define f and g so that we have computable (2,0):1 structures which have exactly one copy of $E_k = \langle (c_0^k, \ldots, c_{k-1}^k), (T_0^k, \ldots, T_{k-1}^k) \rangle$ and one copy of $F_k = \langle (d_0^k, \ldots, d_{k-1}^k), (S_0^k, \ldots, S_{k-1}^k) \rangle$ for each $k \geq 1$. Thus each T_i^j is a one-element tree and each S_i^j is al three-element binary tree. Then for each k, attempt to compute $\phi_k(c_0^k)$. We then have two cases.

Case 1. $\phi_k(c_0^k) \downarrow$ and $\phi_k(c_0^k) \in \{c_0^k, \ldots, c_{k-1}^k\}$.
In this case, we will use new elements from the odd numbers and define f on these odd numbers to extend $\langle (c_0^k, \ldots, c_{k-1}^k), (T_0^k, \ldots, T_{k-1}^k) \rangle$ to a cycle of type G_k. We will also use new odd numbers and define g on those numbers to extend the cycle type of E_k to F_k and the cycle type of F_k to G_k. This will ensure that ϕ_k cannot be an isomorphism from $\mathcal{A}$ onto $\mathcal{B}$ since ϕ_k will map an element of a k-cycle of type G_k into a k-cycle of type F_k. Thus in this case, $\mathcal{A}$ and $\mathcal{B}$ will have one k-cycle of type F_k and one k-cycle of type G_k.

Case 2. $\phi_k(c_0^k) \uparrow$, or $\phi_k(c_0^k) \downarrow$ and $\phi_k(c_0^k) \notin \{c_0^k, \ldots, c_{k-1}^k\}$.
In this case, we do nothing to the k-cycles in $\mathcal{A}$ or $\mathcal{B}$. Then we know that ϕ_k cannot be an isomorphism from $\mathcal{A}$ onto $\mathcal{B}$. In this case, both $\mathcal{A}$ and $\mathcal{B}$ will have one k-cycle of type E_k and one k-cycle of type F_k.

Note that there are infinitely many k such that ϕ_k is the identity so that we will be in Case 1 infinitely often and, hence, f and g will be defined on all of $\mathbb{N}$. It is easy to see that $\mathcal{A} = (\mathbb{N}, f)$ and $\mathcal{B} = (\mathbb{N}, g)$ are computable (2,0):1 structures such that $\mathcal{A}$ and $\mathcal{B}$ are isomorphic but not computably isomorphic.

Next, we will briefly consider Δ_2^0- and Δ_3^0-categoricity of $(2,0):1$ structures. We have the following corollary to the proof of Theorem 3.1.

Theorem 4.3. *Any computable locally finite (2,0):1 structure with only finitely many ω-chains is Δ_2^0-categorical.*

Proof. Observe that $ran(f)$ is a c.e. set. Thus the isomorphism constructed as in the proof of Theorem 3.1 will be computable in an $\mathbf{0}'$ oracle and is therefore Δ_2^0. □

Next we consider structures which are not Δ_2^0-categorical.

Theorem 4.4. *There is a computable locally finite (2,0):1 structure $\mathcal{A}$, consisting of infinitely many ω-chains with attached finite trees, which is not Δ_2^0-categorical.*

Proof. Let T_0 be the one-element binary tree and T_1 be the three-element binary tree. We let $\mathcal{A}_k = (\mathbb{N}, f)$ be a computable (2,0):1 structure that consists a single ω chain that starts at a_0, has $a_i = f^i(a_0)$ for $i \geq 1$ and elements $\{b_1, b_2, \ldots\}$ disjoint from $\{a_0, a_1, a_2, \ldots\}$ such that $f(b_i) = a_i$ where $Tree_{\mathcal{A}_k}(b_i)$ is isomorphic to T_1 if $1 \leq i \leq k$ and is isomorphic to T_0 if $i > k$. For example, the graphs of $\mathcal{A}_0$ and $\mathcal{A}_3$ are pictured in rows 1 and 2, respectively, in Figure 12. We let $\mathcal{A}_\infty = (\mathbb{N}, f)$ be a computable (2,0):1 structure that consists a single ω-chain that starts at a_0, has $a_i = f^i(a_0)$ for $i \geq 1$ and elements $\{b_1, b_2, \ldots\}$ disjoint from $\{a_0, a_1, a_2, \ldots\}$ such that $f(b_i) = a_i$ where $Tree_{\mathcal{A}_k}(b_i)$ is isomorphic to T_1 for all $i \geq 1$. The graph of $\mathcal{A}_\infty$ is pictured at the bottom of Figure 12.

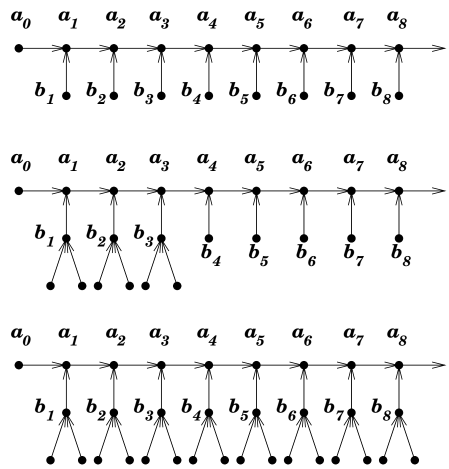

Figure 12: The ω-chains $\mathcal{A}_0$, $\mathcal{A}_3$ and $\mathcal{A}_\infty$.

The desired computable structure $\mathcal{A}$ will consist of infinitely many copies of $\mathcal{A}_n$, for each n, and infinitely many copies of $\mathcal{A}_\infty$. Clearly,+ there is a computable copy $\mathcal{A} = (E, f)$ where E is the set of even numbers such that for each n, the orbit of $4\langle n, k+1\rangle$ is of the form $\mathcal{A}_k$ if $k \geq 0$ and is of the form $\mathcal{A}_\infty$ if $k = 0$. In this case, the set of n such that $\mathcal{O}_{\mathcal{A}}(4n)$ is isomorphic to $\mathcal{A}_\infty$ is computable.

Now we can build a computable (2,0):1 structure $\mathcal{B} = (\mathbb{N}, g)$ which is isomorphic to $\mathcal{A}$ such that the representatives of the orbits of $\mathcal{B}$ are $\{4\langle n, k\rangle : n, k \geq 0\}$

but for which the set of

$$\{4\langle n,k\rangle : \mathcal{O}_{\mathcal{B}}(4\langle n,k\rangle) \text{ is isomorphic to } \mathcal{A}_\infty\}$$

is a Π^0_2-complete set, as follows. Let $Inf = \{e : W_e \text{ is infinite}\}$ be the usual Π^0_2-complete set. Let $g = f$ on the even numbers E. If $\langle n,k\rangle$ is not of the form $\langle e,1\rangle$, then $\mathcal{O}_{\mathcal{A}}(4\langle n,k\rangle) = \mathcal{O}_{\mathcal{B}}(4\langle n,k\rangle)$. We then use odd numbers to define the orbits $\mathcal{O}_{\mathcal{B}}(4\langle e,1\rangle)$. Originally, the orbit $\mathcal{O}_{\mathcal{A}}(4\langle e,1\rangle)$ looks like $\mathcal{A}_0$ so assume that $\mathcal{A}$ is defined such that the chain starts at $a^e_0 = 4\langle e,1\rangle$ and $f^i(a^e_0) = a^e_i$ and b^e_i is the element in the orbit different from a^e_{i-1} such that $f(b^e_i) = a^e_i$. Then whenever a new element appears in W_e at stage s, extend $tree(b^e_i)$ from T_0 to T_1, if necessary, for each $i < s$. If W_e is infinite, then it is clear that $\mathcal{O}_{\mathcal{B}}(4\langle e,1\rangle)$ will be isomorphic to $\mathcal{A}_\infty$. If W_e is empty, then $\mathcal{O}_{\mathcal{B}}(4\langle e,1\rangle)$ will be isomorphic to $\mathcal{A}_0$. Finally, if W_e is finite, then $\mathcal{O}_{\mathcal{B}}(4\langle e,1\rangle)$ will be isomorphic to $\mathcal{A}_s$ for some $s \geq 1$. Since there are infinitely many e such that W_e is empty, there will be infinitely many e such that $\mathcal{O}_{\mathcal{B}}(4\langle e,1\rangle)$ is isomorphic to $\mathcal{A}_0$. Moreover, $e \in Inf$ if and only if $\mathcal{O}_{\mathcal{B}}(4\langle e,1\rangle)$ is isomorphic to $\mathcal{A}_\infty$. Hence, the set of $4\langle n,k\rangle$ such that $\mathcal{O}_{\mathcal{B}}(4\langle n,k\rangle)$ is isomorphic to $\mathcal{A}_\infty$ is a Π^0_2-complete set.

We claim that $\mathcal{B}$ cannot be Δ^0_2-isomorphic to $\mathcal{A}$. That is, if ϕ is a Δ^0_2 isomorphism from $\mathcal{A}$ onto $\mathcal{B}$, then we can decide whether $\mathcal{O}_{\mathcal{B}}(4\langle n,k\rangle)$ is isomorphic to $\mathcal{A}_\infty$ by finding $x = \phi^{-1}(4\langle n,k\rangle)$ and then computing f until we find that $x \in \mathcal{O}_{\mathcal{A}}(4\langle r,s\rangle)$. It would then follow that the set of $4\langle n,k\rangle$ such that $\mathcal{O}_{\mathcal{B}}(4\langle n,k\rangle)$ is isomorphic to $\mathcal{A}_\infty$ is a Δ^0_2 set. Thus the two computable structures $\mathcal{A}$ and $\mathcal{B}$ are isomorphic, but not Δ^0_2-isomorphic. □

For our final result, we first need to consider the isomorphism problem for orbits.

Proposition 4.5. *Let $\mathcal{A}$ be a computable locally finite (2,0):1 structure. Then*

1. $\{(a,b) : \mathcal{O}(a)$ *is isomorphic to* $\mathcal{O}(b)\}$ *is* Σ^0_3, *and*
2. $\{(a,b) : \mathcal{O}(a)$ *is isomorphic to* $\mathcal{O}(b)$ *where the isomorphism maps* a *to* $b\}$ *is* Π^0_2.

Proof. First note that $\mathcal{O}(a)$ is finite if and only if $f^{m+k}(a) = f^m(a)$ for some a, so that this is a Σ^0_1 relation. Given that $\mathcal{A}$ is locally finite, we can then use $\mathbf{0}'$ as an oracle to test whether $\mathcal{O}(a)$ is finite and, if it is finite, then we can again use $\mathbf{0}'$ as an oracle to compute $\mathcal{O}(a)$. Then given two such orbits $\mathcal{O}(a)$ and $\mathcal{O}(b)$, we can simply inspect them to see whether they are isomorphic.

Given a and b such that $\mathcal{O}(a)$ and $\mathcal{O}(b)$ are infinite, we can use an oracle for $\mathbf{0}'$ to compute the sequences $tree(f^i(a))$ and $tree(f^i(b))$. Then there is an isomorphism from $\mathcal{O}(a)$ to $\mathcal{O}(b)$ mapping a to b if and only if $tree(f^i(a))$ and $tree(f^i(b))$ are isomorphic for each i. So this is a Π^0_2 question. Then $\mathcal{O}(a)$ is isomorphic to $\mathcal{O}(b)$ if and only if there exist $x \in \mathcal{O}(a)$ and $y \in \mathcal{O}(b)$ such that there is an isomorphism mapping x to y. □

Theorem 4.6. *Every computable locally finite (2,0):1 structure is Δ^0_3-categorical.*

Proof. Let $\mathcal{A}$ and $\mathcal{B}$ be two isomorphic computable locally finite (2,0):1 structures. We can use $\mathbf{0}''$ as an oracle to compute an isomorphism H from $\mathcal{A}$ onto $\mathcal{B}$ as follows. First enumerate a sequence of representatives of the orbits of $\mathcal{A}$, starting with $a_0 = 0$ and letting a_{n+1} be the least element of $\mathcal{A}$ not in the orbit of a_i for any $i \leq n$, and we can similarly compute $b_0, b_1, \dots$ so that $\mathcal{B} = \bigcup_i \mathcal{O}(b_i)$. Since we know that $\mathcal{B}$ contains an orbit isomorphic to $\mathcal{O}(a_0)$, we can compute using $\mathbf{0}''$ an element $b = H(a_0)$ such that there is an isomorphism of $\mathcal{O}(a_0)$ to $\mathcal{O}(b)$ mapping a_0 to b. Now let $A_0 = \mathcal{O}(a_0)$ and let $B_0 = \mathcal{O}(b)$. The construction of H continues by a back-and-forth argument. At stage $2s$, we will have a partial isomorphism H_s from a subset A_{2s} of $\mathcal{A}$ onto a subset B_{2s} of $\mathcal{B}$, so that for all $i < s$, $a_i \in A_{2s}$ and $b_i \in B_{2s}$. Now at stage $2s+1$, we check to see whether $a_{2s+1} \in A_{2s}$ and if not, we find the least b not in B_{2s} such that there is an isomorphism h mapping $\mathcal{O}(a_{2s+1})$ to $\mathcal{O}(b)$. Then we let $A_{2s+1} = A_{2s} \cup \mathcal{O}(a_{2s+1})$ and let $B_{2s+1} = B_{2s} \cup \mathcal{O}(b)$ and extend the mapping H_{2s} to H_{2s+1} by adding this isomorphism h to H_{2s}. Similarly, at stage $2s+1$, we check to see whether $b_{2s+1} \in B_{2s}$ and if not, we find the least a not in A_{2s} such that there is an isomorphism h mapping $\mathcal{O}(b_{2s+1})$ to $\mathcal{O}(a)$ and extend the isomorphism as above. □

For any $k \geq 3$, we define a $k:1$ structure $\mathcal{A} = (A, f)$ to consist of a function f where for all $x \in A$, $f^{-1}(x)$ is a size k and $(k,0):1$ structure $\mathcal{A} = (A, f)$ to consist of function f where for all $x \in A$, $f^{-1}(x)$ is either of size k or empty. It should be clear that we can prove analogues of all our results for $k:1$ and $(k,0):1$ structures.

References

[1] C. Ash and J. Knight, *Computable Structures and the Hyperarithmetical Hierarchy*, Elsevier, Amsterdam, 2000.

[2] W. Calvert, D. Cenzer, V. Harizanov, and A. Morozov, Effective categoricity of equivalence structures, *Annals of Pure and Applied Logic* 141 (2006), pp. 61–78.

[3] D. Cenzer, V. Harizanov and J.B. Remmel, Σ^0_1 and Π^0_1 equivalence structures, *Ann. Pure and Applied Logic* 162 (2011), pp. 490–503.

[4] D. Cenzer, V. Harizanov, and J.B. Remmel, Effective categoricity of injection structures, in *Models of Computation in Context, CIE 2011*, B. Loewe, D. Normann, I. Soskov, A. Soskova, editors, Springer Lecture Notes in Computer Science 6735 (2011), pp. 51–60

[5] D. Cenzer, V. Harizanov, and J.B. Remmel, Computability-theoretic properties of injection structures, to appear in *Algebra and Logic.*

[6] D. Cenzer, G. LaForte, and J.B. Remmel, Equivalence structures and isomorphisms in the difference hierarchy, *Journal of Symbolic Logic* 74 (2009), pp. 535–556.

[7] D. Cenzer, V. Harizanov and J.B. Remmel, Σ_1^0 and Π_1^0 equivalence structures, in *Mathematical Theory and Computational Practice, CIE 2009*, K. Ambos-Spies, B. Loewe and W. Merkle, editors, Springer Lecture Notes in Computer Science, vol. 5635 (2009), pp. 99–108.

[8] D. Cenzer and J.B. Remmel, Feasibly categorical abelian groups, *Feasible Math II, Proceedings 1992 Cornell Workshop*, eds. P. Clote and J.B. Remmel, Birkhauser (1995), pp. 91–153.

[9] N.G. Khisamiev, Constructive Abelian groups, in: Yu. L. Ershov, S.S. Goncharov, A. Nerode, and J.B. Remmel, editors, *Handbook of Recursive Mathematics*, vol. 2 (North-Holland, Amsterdam, 1998), pp. 1177–1231.

[10] S. Lempp, C. McCoy, R. Miller and R. Solomon, Computable categoricity of trees of finite height, *J. Symbolic Logic* 70 (2005), pp. 151-215.

[11] R.I. Soare, *Recursively Enumerable Sets and Degrees*, (Springer, Berlin, 1987).