A Computably Categorical Structure Whose Expansion by a Constant Has Infinite Computable Dimension

Denis R. Hirschfeldt*

Department of Mathematics, University of Chicago,

Chicago IL 60637, U.S.A.

Bakhadyr Khoussainov

Department of Computer Science, University of Auckland,

Private Bag 92019, Auckland, New Zealand

Richard A. Shore**

Department of Mathematics, Cornell University,

Ithaca NY 14853, U.S.A.

July 22, 2002

Abstract

Cholak, Goncharov, Khoussainov, and Shore [J. Symbolic Logic 64 (1999) 13–37] showed that for each $k > 0$ there is a computably categorical structure whose expansion by a constant has computable dimension k. We show that the same is true with k replaced by ω. Our proof uses a version of Goncharov's method of left and right operations.

*Partially supported by an Alfred P. Sloan Doctoral Dissertation Fellowship and NSF Grant DMS-0200465.

**Partially supported by NSF Grants DMS-0100035 and INT-9602579. Thanks also to the Mathematics Departments at Harvard and MIT for their hospitality.

1 Introduction

The properties of a structure studied by classical model theory are preserved under isomorphism. Thus, in that context, it makes sense to consider isomorphic structures to be identical. This is not the case, however, in computable model theory, since two isomorphic computable structures might have quite different computability-theoretic properties. (See [10] for examples of this phenomenon, as well as background relevant to this paper.) This leads us to study computable structures up to *computable* isomorphism, a point of view reflected in the following definition.

1.1 Definition. A structure $\mathcal{A}$ in a computable language is *computable* if both its domain $|\mathcal{A}|$ and its atomic diagram are computable. If, in addition, its existential diagram is computable, then $\mathcal{A}$ is 1-*decidable.*

An isomorphism from a structure $\mathcal{M}$ to a computable structure is called a *computable presentation* of $\mathcal{M}$. (We often abuse terminology and refer to the image of a computable presentation as a computable presentation.)

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

A structure of computable dimension 1 is said to be *computably categorical.*

The study of the relationships between different computable presentations of a structure is an important theme in computable model theory. It is roughly analogous to the classical study of the relationships between different models of a theory, but the issues and results in these two settings are often quite different. For example, it follows from the Ryll-Nardzweski Theorem that a countably categorical structure remains countably categorical when expanded by finitely many constants. It is natural to ask whether the same is true in the analogous situation in computable model theory. That is, does every computably categorical structure remain computably categorical when expanded by finitely many constants?

Millar [11] showed that, with a relatively small additional amount of decidability, computable categoricity is preserved under expansion by finitely many constants.

1.2 Theorem (Millar). *If $\mathcal{A}$ is computably categorical and* 1*-decidable then any expansion of $\mathcal{A}$ by finitely many constants remains computably categorical.*

However, preservation of categoricity does not hold in general, as was shown by Cholak, Goncharov, Khoussainov, and Shore [1].

1.3 Theorem (Cholak, Goncharov, Khoussainov, and Shore). *For each $k > 0$, there exists a computably categorical structure $\mathcal{A}$ and an $a \in |\mathcal{A}|$ such that $(\mathcal{A}, a)$ has computable dimension k.*

This result raises the following question, left open in [1], as well as in [9], where an easier proof of Theorem 1.3 was given: Does there exist a computably categorical structure whose expansion by some set of finitely many constants has computable dimension ω? In this paper we give the following positive answer to this question.

1.4 Theorem. *There exists a computably categorical structure $\mathcal{A}$ and an $a \in |\mathcal{A}|$ such that $(\mathcal{A}, a)$ has computable dimension ω.*

The proof of this theorem, which will be given in Section 3, uses techniques from [7], which in turn builds on [1, 6, 9]. The original source for the method common to all these papers is the work of Goncharov [3, 4]. In Section 2, we will introduce some of the fundamental ideas of the proof in a less complicated setting. We assume basic familiarity with computable model theory and the tree method of organizing priority constructions. References include [2] (especially the paper [5]) and [12], respectively.

The structure built in Section 3 will be a directed graph. Thus, by the results of [8], for each of the following theories, Theorem 1.4 remains true if we also require that the structure mentioned in it be a model of the given theory: symmetric, irreflexive graphs; partial orderings; lattices; rings (with zero-divisors); integral domains of arbitrary characteristic; commutative semigroups; and 2-step nilpotent groups.

Our result is still not quite the final word on what can happen to the computable dimension of a structure under expansion by constants. The computable dimension of a computable structure $\mathcal{A}$ is said to be *effectively infinite* if there is an effective procedure that, given a uniformly computable set $\mathfrak{P}$ of computable presentations of $\mathcal{A}$, produces a computable presentation of $\mathcal{A}$ that is not computably isomorphic to any of the structures in $\mathfrak{P}$. It will be easy to see that no expansion of the structure built in Section 3 by finitely many constants has effectively infinite computable dimension.

1.5 Question. Is there a computably categorical structure $\mathcal{A}$ whose expansion by some set of finitely many constants has effectively infinite computable dimension?

1.1 Notation

We denote the e^{th} partial computable function by Φ_e, and the result of running the machine computing this function for s steps on input x by $\Phi_e(x)[s]$. We use $\uparrow$ and $\downarrow$ for convergence and divergence, respectively.

For a set X, let $X \upharpoonright m = X \cap \{0, \ldots, m-1\}$. For a function f, let $f \upharpoonright m$ be the restriction of f to $\operatorname{dom} f \upharpoonright m$. For a structure $\mathcal{A}$ in a relational language and a subset S of $|\mathcal{A}|$, let $\mathcal{A} \upharpoonright S$ be the structure obtained by restricting $\mathcal{A}$ to S.

For sequences σ and τ, let $\sigma^\frown\tau$ be their concatenation. We write $\sigma^\frown x$ instead of $\sigma^\frown(x)$, where (x) is the sequence consisting of the single element x. If $\sigma = (x_0, \ldots, x_n)$ then $\sigma(i) = x_i$ and $\sigma \upharpoonright i = (x_0, \ldots, x_{i-1})$. We denote the length of σ by $|\sigma|$.

Fix a one-to-one function from $\omega \times \omega$ onto ω and let $\langle a, b \rangle$ denote the image under this function of the ordered pair consisting of a and b. We write $\langle a, b, c \rangle$ instead of $\langle a, \langle b, c \rangle \rangle$. For $x \in \omega$ and $i = 0, 1$, we write $\pi_i(x)$ for the i^{th} coordinate of the ordered pair coded by x. That is, if $x = \langle a, b \rangle$ then $\pi_0(x) = a$ and $\pi_1(x) = b$.

2 The Method of Left and Right Operations

In this section, we sketch a proof of the existence of a structure of computable dimension two. The first example of such a structure is due to Goncharov [4], as is the general method we will employ. Our purpose here is to introduce some of the basic ideas behind the proof of our main result in a less complicated setting. Thus we will be informal and leave details for the next section.

2.1 The Basic Diagonalization Strategy

First suppose we just want to build computable structures $\mathcal{A}^0$ and $\mathcal{A}^1$ that are isomorphic but not computably isomorphic. We think of this task as building $\mathcal{A}^0$ and $\mathcal{A}^1$ to be isomorphic while satisfying the following requirements:

$$\mathcal{R}_e : \Phi_e \text{ total} \Rightarrow \Phi_e \text{ is not an isomorphism from } \mathcal{A}^0 \text{ to } \mathcal{A}^1.$$

One strategy for satisfying these requirements is to dedicate some portion of $\mathcal{A}^0$ to the satisfaction of $\mathcal{R}_e$ and wait until Φ_e converges on that part of $\mathcal{A}^0$ and looks like a partial isomorphism. We can then change $\mathcal{A}^0$ and $\mathcal{A}^1$ in a way that kills that specific potential isomorphism, while still keeping the structures isomorphic.

To give this "bait and switch" idea a concrete form, let us introduce the basic building blocks of the constructions in this paper.

2.1 Definition. Let $n \in \omega$. The directed graph $[n]$ consists of $n+3$ many nodes $x_0, x_1, \ldots, x_{n+2}$ with an edge from x_0 to itself, an edge from x_{n+2} to x_1, and an edge from x_i to x_{i+1} for each $i \leqslant n+1$. We call x_0 the *top* of $[n]$ and x_{n+2} the *coding location* of $[n]$. (We keep the terminology "coding location" used in [6, 7, 9], although in this paper we will be diagonalizing rather than coding.)

A *cycle* is a copy of $[k]$ for some $k \in \omega$.

Let $S \subset \omega$. The directed graph $[S]$ consists of one copy of $[s]$ for each $s \in S$, with all the tops identified.

Figure 2.1 shows [2] and $[\{2,3\}]$ as examples.

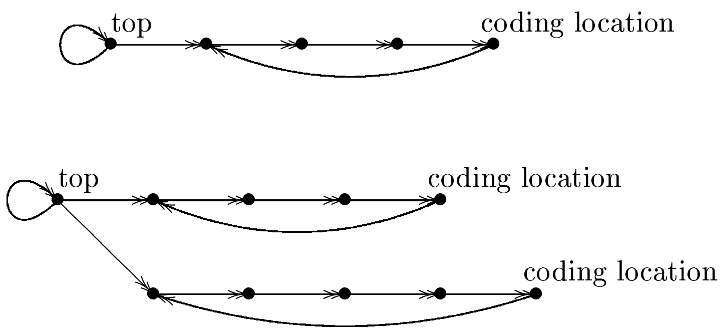


Figure 2.1: [2] and $[\{2,3\}]$

We build $\mathcal{A}^0$ and $\mathcal{A}^1$ in stages. We begin by letting $\mathcal{A}^0_0$ and $\mathcal{A}^1_0$ be computable structures with co-infinite domains, each consisting of one copy of $[k]$ for each $k \in \omega$. The following definitions will help us describe the kind of changes we make to our structures to satisfy the requirements.

2.2 Definition. Let $\mathcal{G}$ be a computable structure in the language of directed graphs whose domain is co-infinite. $\mathcal{G}$ consists of the disjoint union of a number of connected components, which from now on we will just call the *components* of $\mathcal{G}$.

Suppose that $\mathcal{G}$ has components K and L isomorphic to $[B]$ and $[C]$, respectively, where $B, C \subset \omega$ are finite. We define the operation $K \cdot L$, which takes $\mathcal{G}$ to a new computable structure extending $\mathcal{G}$, as follows. Extend K to be a copy of $[B \cup C]$ using numbers not in the domain of $\mathcal{G}$. Leave every other component of $\mathcal{G}$ (including L) unchanged.

We will also use the notation $K \cdot L$ to denote the graph $[B \cup C]$. It should always be clear which meaning of $K \cdot L$ is being used.

Given a finite sequence of operations, each of which can be performed on $\mathcal{G}$, so that no two operations in the sequence affect the same component of $\mathcal{G}$, we can perform all the operations in the sequence simultaneously on $\mathcal{G}$ to get a structure extending $\mathcal{G}$. In this case we will say that we have performed the sequence of operations on $\mathcal{G}$.

2.3 Definition. Let $\mathcal{G}$ be a computable structure in the language of directed graphs whose domain is co-infinite and let $X_0, \dots, X_n$ be components of $\mathcal{G}$ such that, for each $i \leqslant n$, X_i is isomorphic to $[S_i]$ for some finite $S_i \subset \omega$. We define two operations, each of which takes $\mathcal{G}$ to a new computable structure extending $\mathcal{G}$.

- The **L**-operation $\mathbf{L}(X_0, \dots, X_n)$ consists of performing the sequence of operations $X_0 \cdot X_1,\ X_1 \cdot X_2,\ \dots,\ X_n \cdot X_0$ on $\mathcal{G}$.
- The **R**-operation $\mathbf{R}(X_0, \dots, X_n)$ consists of performing the sequence of operations $X_0 \cdot X_n,\ X_1 \cdot X_0,\ \dots,\ X_n \cdot X_{n-1}$ on $\mathcal{G}$.

Note that if $\mathcal{H}$ is the structure obtained by performing $\mathbf{L}(X_0, \dots, X_n)$ on $\mathcal{G}$ and $\mathcal{H}'$ is the structure obtained by performing $\mathbf{R}(X_0, \dots, X_n)$ on $\mathcal{G}$ then $\mathcal{H} \cong \mathcal{H}'$.

We can now proceed as follows. For each e, we choose components X_e^0, Y_e^0, and Z_e^0 of $\mathcal{A}_0^0$ and let X_e^1, Y_e^1, and Z_e^1 be the corresponding components of $\mathcal{A}_0^1$. (This choice should be made so that no component is chosen for more than one e.) Let x_e^i be the coding location of X_e^i.

At stage s, let e be the least number such that we have not yet satisfied $\mathcal{R}_e$ and $\Phi_e(x_e^0)[s]\downarrow = x_e^1$. (If there is no such e, we do nothing at this stage.) We perform $\mathbf{L}(Y_e^0, X_e^0, Z_e^0)$ on $\mathcal{A}_s^0$ to get $\mathcal{A}_{s+1}^0$, perform $\mathbf{R}(Y_e^1, X_e^1, Z_e^1)$ on $\mathcal{A}_s^1$ to get $\mathcal{A}_{s+1}^1$, and declare $\mathcal{R}_e$ to be satisfied.

Now let $\mathcal{A}^0 = \bigcup_{s\in\omega} \mathcal{A}_s^0$ and $\mathcal{A}^1 = \bigcup_{s\in\omega} \mathcal{A}_s^1$. It is easy to show, by induction using the definition of the **L**- and **R**-operations, that $\mathcal{A}_s^0 \cong \mathcal{A}_s^1$ for each s. It is also true that whenever a component of $\mathcal{A}_s^i$ participates in an operation at stage $s+1$, so does the isomorphic component of $\mathcal{A}_s^{1-i}$. Since $\mathcal{A}^0$ and $\mathcal{A}^1$ have no infinite components, it follows that $\mathcal{A}^0 \cong \mathcal{A}^1$.

It is also easy to argue that each requirement is eventually satisfied. If $\Phi_e(x_e^0)\uparrow$ then $\mathcal{R}_e$ is vacuously satisfied. If $\Phi_e(x_e^0)\downarrow \neq x_e^1$ then X_e^0 and X_e^1 are never involved in operations, and hence any isomorphism from $\mathcal{A}^0$ to $\mathcal{A}^1$ must take x_e^0 to x_1^e. If $\Phi_e(x_e^0)\downarrow = x_e^1$ then, by induction, we eventually act to satisfy $\mathcal{R}_e$ by making sure that the components containing x_e^0 and x_e^1 are not isomorphic, so that no isomorphism from $\mathcal{A}^0$ to $\mathcal{A}^1$ can take x_e^0 to x_1^e. In any case, Φ_e is not an isomorphism from $\mathcal{A}^0$ to $\mathcal{A}^1$.

2.2 Special Components

The above construction ensures that the computable dimension of $\mathcal{A}^0$ is at least two. To have it be exactly two, we have to modify the construction to guarantee that

(2.1) if $\mathcal{G} \cong \mathcal{A}^0$ is a computable structure then $\mathcal{G}$ is computably isomorphic to either $\mathcal{A}^0$ or $\mathcal{A}^1$.

Let us first discuss one way to satisfy this property for a single fixed $\mathcal{G}$.

Let $\mathcal{G}[s]$ denote the stage s approximation to $\mathcal{G}$. Since we only care about $\mathcal{G}$ if it is isomorphic to $\mathcal{A}^0$, we can assume that $\mathcal{G}[s]$ is embeddable in $\mathcal{A}^0_s$ for all $s \in \omega$. We will also ensure throughout this construction that, for all $s \in \omega$, no component of $\mathcal{A}^i_s$ is embeddable in another component of $\mathcal{A}^i_s$.

The following is a naive procedure for trying to build a computable isomorphism from $\mathcal{A}^i$ to $\mathcal{G}$. Given a node x in $\mathcal{A}^i$, wait for a stage s such that the component X^i of $\mathcal{A}^i_s$ containing x has an isomorphic copy X in $\mathcal{G}$, then map X^i to X. The problem is that X^i may later participate in an operation, at which point there will be two copies of X^i in $\mathcal{A}^i$ and two in $\mathcal{G}$, and the component of $\mathcal{A}^i$ containing X^i may not be isomorphic to the component of $\mathcal{G}$ containing X. (For example, X^i and X may both be of the form [1]. After an operation involving components of the form [1], [2], and [3], X^i may become part of a component of the form $[1] \cdot [2]$ while X may become part of a component of the form $[1] \cdot [3]$.) If this happens, our original map will no longer be extendable to an isomorphism.

Our strategy for solving this problem is based on the following observation.

Suppose that, at some stage s, $\mathcal{A}^0_s$ has components X^0, Y^0, Z^0, and S^0; $\mathcal{A}^1_s$ has isomorphic components X^1, Y^1, Z^1, and S^1, respectively; and $\mathcal{G}[s]$ has isomorphic components X, Y, Z, and S, respectively. Now suppose we perform $\mathbf{L}(Y^0, X^0, Z^0, S^0)$ on $\mathcal{A}^0_s$ to get $\mathcal{A}^0_{s+1}$ and perform $\mathbf{R}(Y^1, X^1, Z^1, S^1)$ on $\mathcal{A}^1_s$ to get $\mathcal{A}^1_{s+1}$. Then $\mathcal{A}^0_{s+1}$ has components isomorphic to $S^0 \cdot Y^0$, $Y^0 \cdot X^0$, $X^0 \cdot Z^0$, and $Z^0 \cdot S^0$, and these are the only components of $\mathcal{A}^0_{s+1}$ that contain copies of X^0, Y^0, Z^0, or S^0. So if X, Y, Z, and S do not grow into isomorphic copies of the aforementioned components of $\mathcal{A}^0_{s+1}$ then we can win immediately by not involving these components in any further operations, thus guaranteeing that $\mathcal{G} \ncong \mathcal{A}^0$.

So if $\mathcal{G} \cong \mathcal{A}^0$ then there are only two possibilities. The first is that S grows into a copy of $S \cdot Y$, Y grows into a copy of $Y \cdot X$, X grows into a copy of $X \cdot Z$, and Z grows into a copy of $Z \cdot S$. In this case we will say that $\mathcal{G}$ "goes to the left". The other possibility is that Y grows into a copy of $S \cdot Y$, S grows into a copy of $Z \cdot S$, Z grows into a copy of $X \cdot Z$, and X grows into a copy of $Y \cdot X$. In this case we will say that $\mathcal{G}$ "goes to the right".

If $\mathcal{G}$ always goes to the left then isomorphic components of $\mathcal{A}^0_s$ and $\mathcal{G}_s$ always grow into isomorphic components of $\mathcal{A}^0$ and $\mathcal{G}$, and hence the naive procedure described above succeeds in producing an isomorphism from $\mathcal{A}^0$ to $\mathcal{G}$. Indeed, it is enough that $\mathcal{G}$ almost always go to the left (that is, cofinitely often), since we can always wait a finite amount of time before beginning to build our isomorphism. Similarly, if $\mathcal{G}$ almost always goes to the right then we can build a computable isomorphism between $\mathcal{A}^1$ and $\mathcal{G}$.

To ensure that $\mathcal{G}$ either almost always goes to the left or almost always goes to the right, we adopt the strategy of always including copies of a certain fixed component of $\mathcal{G}$, which we will call the *special component* of $\mathcal{G}$, in our operations.

That is, we first pick some component of $\mathcal{G}$ to be its special component. More precisely, we fix some n not otherwise used in the construction and wait until a copy K of $[n]$ appears in $\mathcal{G}$. We declare the component of $\mathcal{G}$ that extends K to be the special component of $\mathcal{G}$. Similarly, we call the component of $\mathcal{G}[s]$ that extends K the special component of $\mathcal{G}[s]$.

At stage 0, we define $\mathcal{A}^i_0$ as before. We also define r_0 to be 0. The value of r_s will code whether $\mathcal{G}$ goes to the left or to the right at stage s.

At stage $s+1$, we choose e as before and let X^i_e, Y^i_e, and Z^i_e be as above. Let S^i_s be the isomorphic copy in $\mathcal{A}^i_s$ of the special component S_s of $\mathcal{G}[s]$. We wait until copies X_e, Y_e, and Z_e of X^i_e, Y^i_e, and Z^i_e, respectively, are enumerated into $\mathcal{G}[s]$ and then perform the same operations as before. We wait until copies of $S_s \cdot Y_e$, $Y_e \cdot X_e$, $X_e \cdot Z_e$, and $Z_e \cdot S_s$ are enumerated into $\mathcal{G}$. Either the copy of $S_s \cdot Y_e$ or that of $Z_e \cdot S_s$ will extend S_s. Whichever one it is now becomes S_{s+1}. If $S_{s+1} \cong S_s \cdot Y_e$ then $r_{s+1} = 0$; otherwise $r_{s+1} = 1$.

The above construction ensures that if $\mathcal{G} \cong \mathcal{A}^0$ then the special component of $\mathcal{G}$ is infinite. On the other hand, it also guarantees that if $\mathcal{G}$ changes direction infinitely often (that is, if r_s does not have a limit) then no component of $\mathcal{A}^0$ is infinite, so that $\mathcal{G} \ncong \mathcal{A}^0$. This latter assertion follows from the fact that, for each $s \in \omega$, the copy of the special component of $\mathcal{G}[s+1]$ in $\mathcal{A}^{1-r_{s+1}}_{s+1}$ is a component that participates in an operation for the first time at stage $s+1$. Figure 2.2 illustrates the case $r_{s+1} = 0$. In this figure, the special components of $\mathcal{G}[s]$ and $\mathcal{G}[s+1]$ and their images are shown in boxes.

$$\text{In } \mathcal{G}: \quad \begin{tikzcd} Y_e \arrow[d] & X_e \arrow[d] & Z_e \arrow[d] & \boxed{S_s} \arrow[d] \\ Y_e \cdot X_e & X_e \cdot Z_e & Z_e \cdot S_s & \boxed{S_s \cdot Y_e} \end{tikzcd}$$

$$\text{In } \mathcal{A}^0: \quad \begin{tikzcd} Y_e^0 \arrow[d] & X_e^0 \arrow[d] & Z_e^0 \arrow[d] & \boxed{S_s^0} \arrow[d] \\ Y_e^0 \cdot X_e^0 & X_e^0 \cdot Z_e^0 & Z_e^0 \cdot S_s^0 & \boxed{S_s^0 \cdot Y_e^0} \end{tikzcd}$$

$$\text{In } \mathcal{A}^1: \quad \begin{tikzcd} Y_e^1 \arrow[d] & X_e^1 \arrow[d] & Z_e^1 \arrow[d] & \boxed{S_s^1} \arrow[d] \\ \boxed{Y_e^1 \cdot S_s^1} & X_e^1 \cdot Y_e^1 & Z_e^1 \cdot X_e^1 & S_s^1 \cdot Z_e^1 \end{tikzcd}$$

Figure 2.2: The images of the special component

2.3 Catch-up and Recovery

However, there are two problems with the above construction. First, by the same reasoning as in the last paragraph, if $\mathcal{G}$ almost always goes to the left then no component of $\mathcal{A}^1$ is infinite, while if $\mathcal{G}$ almost always goes to the right then no component of $\mathcal{A}^0$ is infinite. In either case, $\mathcal{A}^0$ and $\mathcal{A}^1$ are no longer isomorphic. Second, in general we cannot know in advance whether a given computable structure $\mathcal{G}$ is isomorphic to $\mathcal{A}^0$, nor can we effectively list the computable structures isomorphic to $\mathcal{A}^0$, so in the full construction it will not be possible to wait at each stage until the appropriate components are enumerated into $\mathcal{G}$.

The first problem can be fixed by performing a *catch-up operation* following each standard operation.

Remark. For readers familiar with papers such as [6, 7, 9], we note that these catch-up operations will take the place of the isomorphism recovery procedure used in those papers. This change will be needed in the proof of our main result, because there each special component will have infinitely many images. The disadvantage of using catch-up operations is that it creates nontrivial automorphisms, but this is not an issue in our main result, since the structure required in that result cannot be rigid anyhow.

We first make a small modification to the definition of $\mathcal{A}^i_0$. Instead of starting with a copy of $[n]$ for each $n \in \omega$, we reserve an infinite and co-infinite set $R \subset \omega$ and let each $\mathcal{A}^i_0$ consist of one copy of $[n]$ for each $n \in \omega - R$.

At each stage $s+1$, we define a subgraph T_{s+1} of the special component of $\mathcal{G}$ as follows. If $r_{s+1} \neq r_s$ then let T_s be the entire special component of $\mathcal{G}[s]$, and otherwise let $T_{s+1} = T_s$.

After each stage s operation as described above, let $L^i_0, \ldots, L^i_n$ be the components of $\mathcal{A}^i_{s+1}$ containing copies of T_{s+1} (listed so that $L^0_j \cong L^1_j$). Let P_j be such that $L^i_j = [P_j]$ and let $P = \bigcup_{j \leqslant n} P_j$. Let $k_0, \ldots, k_n \in R$ be distinct numbers such that $\mathcal{A}^i_{s+1}$ does not contain a k_j-cycle for any $j \leqslant n$. We extend each L^i_j to a copy of $[P \cup \{k_j\}]$.

This catch-up procedure ensures that if r_s comes to a limit by stage t then the components of $\mathcal{A}^0$ and $\mathcal{A}^1$ that contain a copy of T_t are all infinite and isomorphic. At the same time, it maintains the essential property that, for each s, no component of $\mathcal{A}^i_s$ is embeddable in another component of $\mathcal{A}^i_s$. (This property was needed in the argument given above that $\mathcal{G}$ has only two options following each operation.)

We deal with the problem of not knowing whether $\mathcal{G} \cong \mathcal{A}^0$ by using the idea of *recovery*. Instead of having a single strategy for each diagonalization requirement $\mathcal{R}_e$, we have two, $R_{0,e}$ and $R_{1,e}$, each working with its own set of components $Y^i_{0,e}$, $X^i_{0,e}$, $Z^i_{0,e}$ and $Y^i_{1,e}$, $X^i_{1,e}$, $Z^i_{1,e}$, respectively. Roughly speaking (we will be more precise below), the strategy $R_{0,e}$ works under the assumption that $\mathcal{G} \cong \mathcal{A}^0$ and $R_{1,e}$ works under the assumption that $\mathcal{G} \ncong \mathcal{A}^0$.

The strategy $R_{1,e}$ is free to perform an operation involving its set of components whenever it wants to, without caring about $\mathcal{G}$ and without involving copies of the special component of $\mathcal{G}$ in the operation. The strategy $R_{0,e}$, on the other hand, must wait for $\mathcal{G}$ to recover sufficiently often, as we now explain.

Suppose that we perform an operation at stage $s+1$ involving copies of the special component of $\mathcal{G}[s]$. To be more specific, suppose the components of $\mathcal{G}[s]$ whose copies participate in this operation are $Y_{0,k}$, $X_{0,k}$, $Z_{0,k}$, and S_s. Where we would have waited for $Y_{0,k}$, $X_{0,k}$, $Z_{0,k}$, and S_s to grow into copies of $Y_{0,k} \cdot X_{0,k}$, $X_{0,k} \cdot Z_{0,k}$, $Z_{0,k} \cdot S_s$, and $S_s \cdot Y_{0,k}$, we now just declare that we are waiting for these copies to appear in $\mathcal{G}$, and proceed with the construction.

A recovery stage is then a stage $t+1$ such that

1. $\mathcal{G}[t]$ contains copies of all the components for which we are currently waiting and

2. for each e less than or equal to the number of recovery stages before stage $t+1$, if $\mathcal{R}_e$ has not yet been satisfied then $\mathcal{G}[t]$ contains components isomorphic to $Y_{0,e}$, $X_{0,e}$, and $Z_{0,e}$.

Whenever $\mathcal{G}$ recovers, all strategies $R_{1,e}$ are *initialized*, in the sense that they have to pick new components $Y_{1,e}$, $X_{1,e}$, and $Z_{1,e}$. (This is done in such a way that the old components are never again picked by any strategy, and for each component we can find a stage after which it can never again be picked by any strategy.) If a strategy $R_{1,e}$ ever gets a chance to act to satisfy $\mathcal{R}_e$ with its currently selected components, it does so. A strategy $R_{0,e}$, on the other hand, can only act at a recovery stage and after there have been at least e many recovery stages, and must involve copies of the special component of $\mathcal{G}$ in the operation it performs.

By following the above strategy, we ensure two crucial things. One is that each $\mathcal{R}_e$ is eventually satisfied, whether or not $\mathcal{G}$ recovers infinitely often. The other is that if $\mathcal{G} \cong \mathcal{A}^0$ (in which case $\mathcal{G}$ must recover infinitely often) then we can still build a computable isomorphism between one of the $\mathcal{A}^i$ and $\mathcal{G}$. As we now show, the argument that this second fact holds is only slightly more complicated than before.

We still have that $\mathcal{G}$ must almost always go to the left or almost always go to the right. Let us assume the first case (the other being symmetric). Then we can split the components of $\mathcal{A}^0$ into those that participate in operations for the sake of 0-strategies (i.e., strategies $R_{0,e}$ which assume that $\mathcal{G}$ recovers infinitely often), and those that participate in operations for the sake of 1-strategies. We can build our isomorphism on the former class of components as before, using the fact that $\mathcal{G}$ almost always behaves like $\mathcal{A}^0$. For each component K in the latter class of components, we can find a stage after which K is guaranteed not to participate in an operation. At this stage, we can simply wait until an isomorphic component L appears in $\mathcal{G}$ and map K to L in the obvious way. This isomorphism will be correct, since we are now guaranteed that neither K nor L will ever grow. (To see that L will never grow, suppose otherwise. Then, since we are assuming that $\mathcal{G} \cong \mathcal{A}^0$, there is some s such that $\mathcal{A}^0_s$ contains both K and another component strictly containing a copy of K. But we have ensured that this will never be the case.)

2.4 The tree of strategies

We have been discussing satisfying property (2.1) for a single $\mathcal{G}$, but in the full construction we need to consider all computable graphs. Let $\mathcal{G}_0, \mathcal{G}_1, \ldots$ be a standard enumeration of all partial computable directed graphs.

Remark. We need to consider partial computable graphs because there is no effective listing of all computable directed graphs. For the sake of definiteness, we make the following definition, although we will make no explicit use of it. A *partial computable directed graph* $\mathcal{G}$ consists of two $0,1$-valued partial computable functions Φ and Ψ, the former unary and the latter binary, such that if $\Phi(x)[s]\downarrow = \Phi(y)[s]\downarrow = 1$ then $\Psi(x,y)[s]\downarrow$. The graph $\mathcal{G}$ (resp. $\mathcal{G}[s]$) is the graph whose domain has characteristic function Φ ($\Phi[s]$) and whose edge relation has characteristic function Ψ ($\Psi[s]$).

The above discussion of recovery suggests how to organize our construction using a tree of strategies. For each finite binary string σ, we have a strategy for satisfying (2.1) for $\mathcal{G}_{|\sigma|}$. The string σ represents a guess as to which $\mathcal{G}_m$, $m < |\sigma|$, recover infinitely often, with $\sigma(m) = 0$ representing a guess that $\mathcal{G}_m$ recovers infinitely often and $\sigma(m) = 1$ representing a guess that it does not.

More precisely, if $\tau = \sigma \upharpoonright m$ then $\sigma(m) = 0$ represents a guess that $\mathcal{G}_m$ *τ-recovers* infinitely often. We wait until the next section to give a more detailed discussion of τ-recovery. The basic idea is the following. When an operation performed for the sake of a strategy corresponding to a superstring of τ extends certain components of the $\mathcal{A}^i$ (as in the simpler example above), we then have to wait for $\mathcal{G}_{|\tau|}$ to provide copies of these extended components before we can allow strategies corresponding to superstrings of τ to act again. Roughly speaking, we have τ-recovery when these components are provided.

We choose a *σ-special component* of $\mathcal{G}_{|\sigma|}$ for each $\sigma \in 2^{<\omega}$. There are also 2^e many strategies for satisfying $\mathcal{R}_e$, one strategy $R_{\sigma,e}$ for each σ of length e. Each $R_{\sigma,e}$ acts according to the information encoded in σ. That is, when $R_{\sigma,e}$ wants to act, it has to wait for each $\mathcal{G}_{|\tau|}$ with $\tau^\frown 0 \subseteq \sigma$ to τ-recover, and must then involve the τ-special component of each such $\mathcal{G}_{|\tau|}$ in the diagonalization operation it performs.

We run the construction in the usual way. At each stage, we choose which path to follow based on which σ are currently recovered. The strategies on this path then get to act, and the strategies to the right of this path are initialized. This initialization process includes choosing a new σ-special component of $\mathcal{G}_{|\sigma|}$ for each σ to the right of the current path. The *true path* of the construction is the leftmost path visited infinitely often. If σ is on the true path then $R_{\sigma,|\sigma|}$ succeeds in satisfying $\mathcal{R}_{|\sigma|}$.

Furthermore, if σ is on the true path and $\mathcal{G}_{|\sigma|} \cong \mathcal{A}^0$ then the σ-special component of $\mathcal{G}_{|\sigma|}$ is involved in every operation performed for the sake of strategies corresponding to superstrings of σ. Arguing as above, for some $i = 0, 1$ we can effectively build an isomorphism between the components of $\mathcal{A}^i$ that participate in such operations and the corresponding components of $\mathcal{G}_{|\sigma|}$. This isomorphism can be extended to all of $\mathcal{A}^i$ much as before. More specifically, let K be a component of $\mathcal{A}^i$ that does not belong to a strategy corresponding to a superstring of σ. If K belongs to a strategy to the right of the true path then it is eventually guaranteed never again to participate in an operation, and hence we can extend our isomorphism to K as above. Otherwise, K belongs to a strategy above or to the left of σ. Since there are only finitely many substrings of σ, and strategies to the left of σ are active only finitely often, we need only finitely much information to extend our isomorphism to all components belonging to strategies above or to the left of σ.

There are several other details that would need to be added to the above sketch to complete the proof. In particular, we have not discussed exactly how operations involving multiple special components are performed. These compound operations complicate the analysis of the recovery process and the proof that any computable graph isomorphic to $\mathcal{A}^0$ must either almost always go to the left or almost always go to the right. Since such aspects of the method of operations depend on the particular result being proved, we leave them to the following section, where we will discuss them in the setting of our main result.

3 Proof of Theorem 1.4

We will build a computable structure $\mathcal{A}$ and a computable set $\{a_i\}_{i\in\mathbb{Z}}$ of elements of $|\mathcal{A}|$ so that the following properties hold.

(3.1) For every $i \in \mathbb{Z}$, $(\mathcal{A}, a_i) \cong (\mathcal{A}, a_0)$.

(3.2) For every $i, j \in \mathbb{Z}$, if $i \neq j$ then $(\mathcal{A}, a_i)$ is not computably isomorphic to $(\mathcal{A}, a_j)$.

(3.3) If $\mathcal{G}$ is a computable structure, $g \in |\mathcal{G}|$, and $(\mathcal{G}, g) \cong (\mathcal{A}, a_0)$, then $(\mathcal{G}, g)$ is computably isomorphic to $(\mathcal{A}, a_i)$ for some $i \in \mathbb{Z}$.

The structure $\mathcal{A}$ will include infinitely many isomorphic subgraphs, built using a version of the method of left and right operations described above. These subgraphs will be arranged in levels attached to a "backbone", as indicated in the following definition.

3.1 Definition. The *backbone graph* is the directed graph, shown in Figure 3.1, consisting of the following nodes and edges.

1. A *root node* x.
2. For each $i \in \mathbb{Z}$, an *i-master node* x_i, with an edge from x to x_i.
3. For each $i \in \mathbb{Z}$, an edge from x_i to x_{i+1}.

We will say that a directed graph $\mathcal{G}$ is *leveled* if $|\mathcal{G}|$ can be split into two disjoint sets H and I so that the following conditions are satisfied.

1. $\mathcal{G} \upharpoonright H$ is isomorphic to the backbone graph.
2. $\mathcal{G} \upharpoonright I$ consists of cycles and edges between the tops of some of these cycles.
3. The only edges in $\mathcal{G}$ between elements of H and elements of I are edges from i-master nodes, $i \in \mathbb{Z}$, to tops of cycles.
4. Let $i, j \in \mathbb{Z}$, $i \neq j$. If there is an edge from the i-master node of $\mathcal{G} \upharpoonright H$ to an element y of I then there is no edge from the j-master node of $\mathcal{G} \upharpoonright H$ to y.

We call the connected components of $\mathcal{G} \upharpoonright I$ the *components* of $\mathcal{G}$. A component isomorphic to $[n]$ for some n is called a *singleton component*. Let C be a cycle in $\mathcal{G} \upharpoonright I$ and let $i \in \mathbb{Z}$. If there is a node from the i-master node to the top of C then we say that C has *level* i. Let K be a component of $\mathcal{G}$, in the above sense. If all the cycles in K have the same level i then we say that K has level i, and define $level(K) = i$. If none of the cycles in K have levels then we say that K *has no level*. If there are two cycles in K with different levels then we say that K *has multiple levels*.

For $i \in \mathbb{Z}$, we denote by $\mathcal{G}^i$ the subgraph of $\mathcal{G}$ consisting of all level-i components of $\mathcal{G}$. We denote by $\mathcal{G}^*$ the subgraph of $\mathcal{G}$ consisting of those components of $\mathcal{G}$ that either have no level or have multiple levels.

Let $n, r \in \omega$. Suppose that $\mathcal{G}$ is such that every component M of $\mathcal{G}^*$ that has multiple levels consists of a cycle K with no level whose top is connected to the tops of infinitely many cycles $L_0, L_1, \ldots$ such that L_i has a level $g(i)$ for each $i \in \omega$. For each component M of $\mathcal{G}^*$ as above, let $\widehat{M}_{n,r}$ be the graph obtained by restricting the domain of M to the union of $|K|$ and $|L_i|$ for every $i \in \omega$ such that $|g(i) - r| \leqslant n$. We denote by $(\mathcal{G}^*)^{n,r}$ the union of all $\widehat{M}_{n,r}$ such that M is a component of $\mathcal{G}^*$. In case $r = 0$, we write simply $(\mathcal{G}^*)^n$.

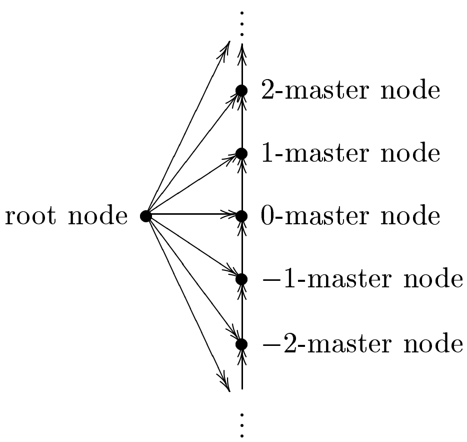


Figure 3.1: The backbone graph

The structure $\mathcal{A}$ will be a leveled graph. The components of $\mathcal{A}^*$ that have multiple levels will be of the form given in the previous paragraph, so it will make sense to talk about $(\mathcal{A}^*)^{n,r}$ and $(\mathcal{A}^*)^n$. The a_i mentioned in (3.1)–(3.3) will be the i-master nodes of $\mathcal{A}$.

Before describing the construction of $\mathcal{A}$, we note that we can restrict the class of graphs that must be considered in satisfying property (3.3). Fix a computable presentation B of the backbone graph with co-infinite domain. Every computable leveled graph is computably isomorphic to a computable graph containing B as a subgraph, so it is enough to consider such graphs. It will also be the case that every cycle in $\mathcal{A}$ will have a level except for cycles of the form $[10k]$, $k \in \omega$, which will not have levels, so it is enough to consider graphs satisfying this property.

Thus, in this section, we will only consider partial computable graphs $\mathcal{G}$ satisfying the following conditions for each $s \in \omega$.

1. $\mathcal{G}[s] \upharpoonright (|\mathcal{G}[s]| \cap |B|) \cong B \upharpoonright (|\mathcal{G}[s]| \cap |B|)$.

2. If $x \in |\mathcal{G}[s]|$ then x is contained in a cycle in $\mathcal{G}[s]$.

3. Let K be a cycle not of the form $[10k]$, $k \in \omega$, in $\mathcal{G}[s]$. There is a unique node $x \in |\mathcal{G}[s]| \cap |B|$ with an edge in $\mathcal{G}[s]$ from x to the top of K. This node x is an i-master node for some $i \in \mathbb{Z}$.

4. Let K be a cycle of the form $[10k]$, $k \in \omega$, in $\mathcal{G}[s]$. For every $x \in |B|$, there is no

edge in $\mathcal{G}[s]$ from x to the top of K.

Clearly, there exists a computable list $\mathcal{G}_0, \mathcal{G}_1, \ldots$ of all partial computable graphs that satisfy the above conditions, and this list contains all the computable graphs that must be considered in satisfying property (3.3). We will assume that, for each $n, s \in \omega$, there is an embedding of $\mathcal{G}_n[s]$ into $\mathcal{A}_s$.

The fact that $\mathcal{G}_n$ contains B means that it makes sense to speak of level i of $\mathcal{G}_n$. It also makes sense to speak of level i of $\mathcal{G}_n[s]$, with the understanding that if the i-master node of B is not in $|\mathcal{G}_n[s]|$ then level i of $\mathcal{G}_n[s]$ is empty. The reason for conditions 3 and 4 above is that they ensure that the following is true. Let K be a cycle in $\mathcal{G}_n[s]$. If K has level i in $\mathcal{G}_n[s]$ then it has level i in $\mathcal{G}_n$, while if K has no level in $\mathcal{G}_n[s]$ then it has no level in $\mathcal{G}_n$. While this fact will not be needed in our formal construction and verification, it is nevertheless useful in clarifying what we mean when we speak of the level of a component of $\mathcal{G}_n$ or $\mathcal{G}_n[s]$ in our informal discussion below.

To satisfy (3.2), we will satisfy the following requirement for each $e \in \omega$ and $i \in \mathbb{Z}$, $i \neq 0$:

$$\mathcal{R}_{\langle e,i\rangle} : \Phi_e \text{ is not an isomorphism from } (\mathcal{A}, a_0) \text{ to } (\mathcal{A}, a_i).$$

This will be enough, since for any $j, k \in \mathbb{Z}$ such that $j \neq k$, any automorphism of $\mathcal{A}$ taking a_j to a_k takes a_0 to a_i for some $i \neq 0$.

The basic idea for satisfying $\mathcal{R}_{\langle e,i\rangle}$ is simple, and makes use of the concept of left and right operations described above.

We will build $\mathcal{A}$ by beginning with a computable leveled graph $\mathcal{A}_0$ and successively applying operations. Suppose for now that $\mathcal{A}_0$ is such that all levels are isomorphic and consist of cycles, and no two components of the same level are isomorphic. (This will change later, when we consider the satisfaction of (3.3).)

We choose a singleton component E^0 of $\mathcal{A}_0^0$, let E^i be the component of $\mathcal{A}_0^i$ isomorphic to E^0, and let x and y be the coding locations of E^0 and E^i, respectively. We then wait until $\Phi_e(x)$ converges. If this never happens then we win by default. If $\Phi_e(x)\downarrow \neq y$ then we win by doing nothing, thus guaranteeing that any automorphism of $\mathcal{A}$ that takes a_0 to a_i must take x to y, which implies that Φ_e cannot be such an automorphism.

If $\Phi_e(x)\downarrow = y$ then we act to ensure that no automorphism of $\mathcal{A}$ can take x to y. We do this by performing operations on E^0 and E^i that guarantee that the components of $\mathcal{A}$ that extend each of these components are not isomorphic. Specifically, we first choose components D^0, F^0, D^i, and F^i such that D^0 and F^0 have level 0, D^i and F^i have level i, $D^0 \cong D^i$, and $F^0 \cong F^i$. Then we perform an operation that guarantees that E^0 is extended by a component containing a copy of $E^0 \cdot F^0$, while E^i is extended by a component containing a copy of $D^i \cdot E^i$.

Of course, to keep all the levels of $\mathcal{A}$ isomorphic, we also need to perform similar operations on the components of $\mathcal{A}_0^j$ isomorphic to D^0, E^0, and F^0 for each $j \in \mathbb{Z}$. Without any other features to the construction, we could do this simply by performing the operations $\mathbf{L}(D^0, E^0, F^0)$ and $\mathbf{R}(D^j, E^j, F^j)$ for $j \in \mathbb{Z}$, $j \neq 0$. Proceeding in this fashion simultaneously for each $\mathcal{R}_{\langle e,i\rangle}$, we could satisfy both (3.1) and (3.2).

However, as we will see, the satisfaction of (3.3) will require us to involve more components than just the D^j, E^j, and F^j in our operations, and will make it necessary for the sequence of operations performed to satisfy a given requirement to be periodic, in the sense that there is an $n > 0$ such that, if a row of level-i components participates in a left operation then so does the isomorphic row of level-$(i+nj)$ components for each $j \in \mathbb{Z}$, and similarly for right operations.

As an illustration, Figure 3.2, which will be explained below, shows the basic diagonalization strategy in the case in which we are satisfying $R_{\langle e,i\rangle}$ for some $e \in \omega$ and $i = 3$. An arrow from K to L means that the component K is involved in the operation and becomes a copy of L. Since we want the level-0 and level-3 components involved in the operation to go in opposite directions, the period of this operation is 4. As above, we have components D^0 and E^0, but we now need multiple components F_0^0, F_1^0, and F_2^0 in place of F^0, for reasons that should become clear after examining the figure. For each $i \in \mathbb{Z}$, D^i, E^i, F_0^i, F_1^i, and F_2^i are the level-i components isomorphic to D^0, E^0, F_0^0, F_1^0, and F_2^0, respectively. There is also a component X which acts as the link between different rows of components participating in the operation.

To understand Figure 3.2, we need to define two new kinds of basic operations.

3.2 Definition. Let $\mathcal{G}$ be a computable leveled graph whose domain is co-infinite. Let $L, K_0, K_1, \ldots$ be components of $\mathcal{G}$ isomorphic to $[x], [y_0], [y_1], \ldots$, respectively, where $x, y_0, y_1, \ldots \in \omega$, such that $K_0, K_1, \ldots$ have levels and L has no level. Let $\mathcal{S} = \{K_i \mid i \in \omega\}$. We define two operations, each of which takes $\mathcal{G}$ to a new co-infinite computable structure extending $\mathcal{G}$.

- The operation $\mathcal{S}{\cdot}L$ consists of performing the following steps, and otherwise leaving $\mathcal{G}$ unchanged. Create a new copy of $[x]$ using numbers not in the domain of $\mathcal{G}$. For each $i \in \omega$, add an edge from the top of this new copy of $[x]$ to the top of K_i.

- The operation $L{\cdot}\mathcal{S}$ consists of performing the following steps, and otherwise leaving $\mathcal{G}$ unchanged. For each $i \in \omega$, create a new copy of $[y_i]$ using numbers not in the

$$\begin{array}{cc}
\begin{tikzcd} D^0 \arrow[d] \\ D^0 \cdot E^0 \end{tikzcd} \quad \begin{tikzcd} E^0 \arrow[d] \\ E^0 \odot (F_0^0, F_1^0, F_2^1) \end{tikzcd} & \begin{tikzcd} E^1 \arrow[d] \\ E^1 \cdot D^1 \end{tikzcd} \quad \begin{tikzcd} F_0^1 \arrow[d] \\ F_0^1 \odot (E^1, F_1^1, F_2^1) \end{tikzcd}
\end{array}$$

similarly for all levels $\equiv 0 \bmod 4$ similarly for all levels $\equiv 1 \bmod 4$

$$\begin{array}{cc}
\begin{tikzcd} E^2 \arrow[d] \\ E^2 \cdot D^2 \end{tikzcd} \quad \begin{tikzcd} F_1^2 \arrow[d] \\ F_1^2 \odot (E^2, F_0^2, F_2^2) \end{tikzcd} & \begin{tikzcd} E^3 \arrow[d] \\ E^3 \cdot D^3 \end{tikzcd} \quad \begin{tikzcd} F_2^3 \arrow[d] \\ F_2^3 \odot (E^3, F_0^3, F_1^3) \end{tikzcd}
\end{array}$$

similarly for all levels $\equiv 2 \bmod 4$ similarly for all levels $\equiv 3 \bmod 4$

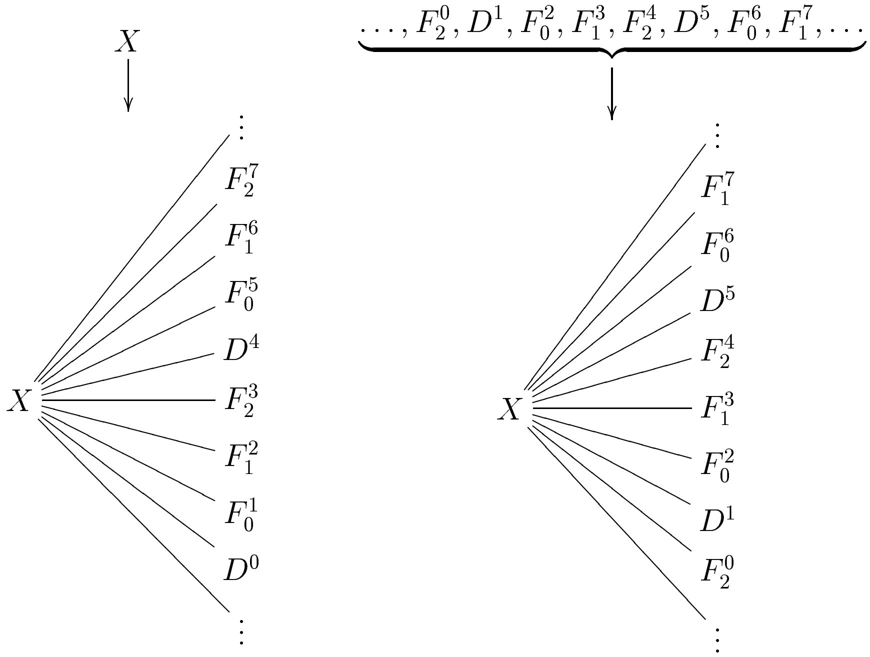
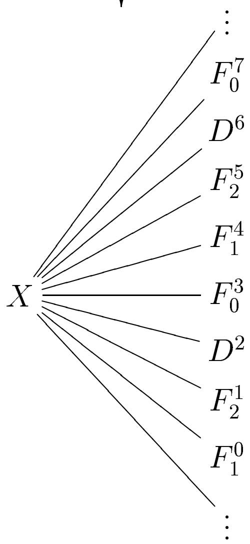


Figure 3.2: The basic diagonalization strategy

Figure 3.2 (Continued)

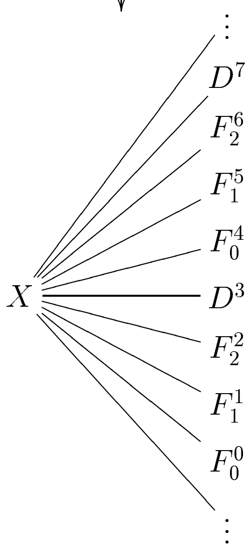
$$\underbrace{\ldots, F_1^0, F_2^1, D^2, F_0^3, F_1^4, F_2^5, D^6, F_0^7, \ldots}$$

$$\downarrow$$



$$\underbrace{\ldots, F_0^0, F_1^1, F_2^2, D^3, F_0^4, F_1^5, F_2^6, D^7, \ldots}$$

$$\downarrow$$



domain of $\mathcal{G}$. For each $i \in \omega$, add an edge from the top of L to the top of the new copy of $[y_i]$ and add an edge from the $level(K_i)$-master node to the top of the new copy of $[y_i]$.

3.3 Definition. Let $\mathcal{G}$ be a computable leveled graph whose domain is co-infinite. Let L and $K_0, K_1, \ldots, K_n$ be components of $\mathcal{G}$ isomorphic to $[x]$ and $[y_0], [y_1], \ldots, [y_n]$, respectively, where $x, y_0, y_1, \ldots, y_n \in \omega$, such that $K_0, K_1, \ldots, K_n$ have levels. (It does not matter for this definition whether L has a level, although this will always be the case when we apply it.)

The operation $L \odot (K_0, K_1, \ldots, K_n)$, taking $\mathcal{G}$ to a new computable structure extending $\mathcal{G}$, consists of performing the following steps, and otherwise leaving $\mathcal{G}$ unchanged. For each $i \leqslant n$, create a new copy of $[y_i]$ using numbers not in the domain of $\mathcal{G}$. For each $i \leqslant n$, add an edge from the top of L to the top of the new copy of $[y_i]$, an edge from the top of the new copy of $[y_i]$ to the top of L, and an edge from the $level(K_i)$-master node to the top of the new copy of $[y_i]$. For each $i, j \leqslant n$, $i \neq j$, add an edge from the top of the new copy of $[y_i]$ to the top of the new copy of $[y_j]$.

As an example of the operation in Definition 3.3, suppose that K_0, K_1, and K_2 are copies of [2], [3], and [4], respectively. Let i, j, k be such that $\{i, j, k\} = \{0, 1, 2\}$. The operation $K_i \odot (K_j, K_k)$ consists of extending K_i to a copy of the graph shown in Figure 3.3, adding an edge from the $level(K_j)$-master node to the new copy of K_j, and adding an edge from the $level(K_k)$-master node to the new copy of K_k.

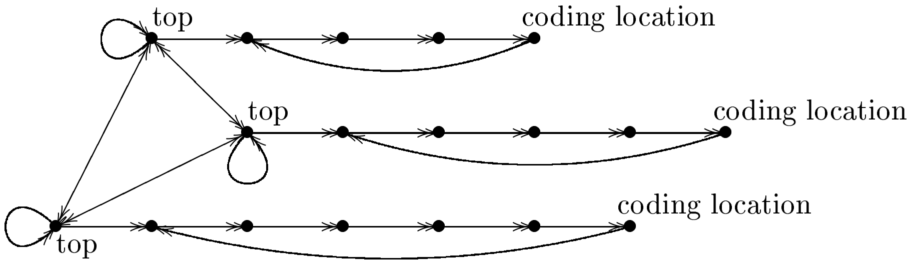


Figure 3.3: The result of any of $[2] \odot ([3], [4])$, $[3] \odot ([2], [4])$, or $[4] \odot ([2], [3])$

In Figure 3.2, the result of either of the operations $L \cdot \mathcal{S}$ or $\mathcal{S} \cdot L$ is represented by L with a line to each element of $\mathcal{S}$, while the result of the operation $L \odot (K_0, K_1, \ldots, K_n)$ is represented simply by $L \odot (K_0, K_1, \ldots, K_n)$.

As we have just seen, we will be performing infinite operations in our construction. Thus, at a stage $s+1$, we might add infinitely many new nodes and edges to $\mathcal{A}_s$ to obtain $\mathcal{A}_{s+1}$. We will do this in such a way that the only edges in $\mathcal{A} = \bigcup_{t \in \omega} \mathcal{A}_t$ between nodes of $\mathcal{A}_{s+1}$ are those already present in $\mathcal{A}_{s+1}$.

As in Section 2, we will use special components to satisfy (3.3). The idea is similar to what we outlined above. For each finite binary string σ, there will be a strategy for satisfying (3.3) for $\mathcal{G}_{|\sigma|}$. As before, the string σ will represent a guess as to which $\mathcal{G}_m$, with $m < |\sigma|$, $(\sigma \upharpoonright m)$-recover infinitely often, with $\sigma(m) = 0$ representing a guess that $\mathcal{G}_m$ $(\sigma \upharpoonright m)$-recovers infinitely often and $\sigma(m) = 1$ representing a guess that it does not. The concept of σ-recovery will be similar to what was discussed above, although we will of course eventually give a formal definition in this case. We will not allow σ-recovery unless there is τ-recovery for all τ such that $\tau^\frown 0 \subseteq \sigma$.

For each $\sigma \in 2^{<\omega}$, $\mathcal{G}_{|\sigma|}$ will have a σ-special component, which will change each time σ is initialized. At each stage s in the construction, we will have a guess $r_{\sigma,s}$ as to which level of $\mathcal{A}$ behaves like level 0 of $\mathcal{G}_{|\sigma|}$, in the same sense that, in Section 2, r_s was a guess as to which copy of the structure constructed in that section behaved like $\mathcal{G}$.

Suppose that copies of the σ-special component of $\mathcal{G}_{|\sigma|}$ participate in an operation at a stage $s+1$ (we say that σ is *active* at stage $s+1$) and $t+1$ is the next σ-recovery stage after stage $s+1$. If the component of $\mathcal{A}_t$ that is isomorphic to the special component of $\mathcal{G}_{|\sigma|}[t]$ extends the component of $\mathcal{A}_s$ that is isomorphic to the special component of $\mathcal{G}_{|\sigma|}[s]$ then $r_{\sigma,t+1} = r_{\sigma,s+1}$; otherwise, $r_{\sigma,t+1} \neq r_{\sigma,s+1}$.

By performing operations involving the images in $\mathcal{A}$ of the σ-special component of $\mathcal{G}_{|\sigma|}$ (one image for each level of $\mathcal{A}$), we will ensure that, for σ on the true path of the construction, if $\mathcal{G}_{|\sigma|} \cong \mathcal{A}$ then the σ-special component of $\mathcal{G}_{|\sigma|}$ is infinite. We will also ensure that either for some $i \in \mathbb{Z}$ there is a level-i component that, from some point in the construction on, always goes in the same direction as the special component of $\mathcal{G}_{|\sigma|}$, or there is no component in $\mathcal{A}$ isomorphic to the special component of $\mathcal{G}_{|\sigma|}$. As we will see, this will mean that if $\mathcal{G}_{|\sigma|} \cong \mathcal{A}$ then, for each $j \in \mathbb{Z}$, from some stage s_j in the construction on, the j^{th} level of $\mathcal{G}_{|\sigma|}$ will go in the same direction as $(j+i)^{\text{th}}$ level of $\mathcal{A}$ at all stages at which σ is active.

The reason that s_j will depend on j is that the operations in this construction will involve infinitely many components at a time. Thus, we cannot make it a requirement for σ-recovery that $\mathcal{G}_{|\sigma|}$ provide all components that will be used in the next operation to be performed at a stage at which σ is active. Instead, we will only require that $\mathcal{G}_{|\sigma|}$ provide the necessary components for a finite number of levels; each time σ recovers, the number of levels that must be provided for the next recovery will increase.

To illustrate the recovery process, consider Figure 3.4, which presents an operation that might be performed at some stage $s+1$ of our construction, ignoring for now all components indexed by τ, whose role will be explained later.

Our construction will be such that $\mathcal{A}_s$ will have the following properties. For each $i, j \in \mathbb{Z}$, $\mathcal{A}_s^i \cong \mathcal{A}_s^j$. For each $i \in \mathbb{Z}$, no component K of $\mathcal{A}_s^i$ is embeddable in another component L of $\mathcal{A}_s$ unless, for some $j \in \mathbb{Z}$, L is the (unique) component of $\mathcal{A}_s^j$ isomorphic to K. No singleton component of $\mathcal{A}_s^*$ is embeddable in another component of $\mathcal{A}_s$.

In Figure 3.4, we are assuming that each of Z_σ^0, B_σ^0, C_σ^0, $Y_{\sigma,0}^0$, $Y_{\sigma,1}^0$, $Y_{\sigma,2}^0$, D^0, E^0, F_0^0, F_1^0, and F_2^0 are singleton components of $\mathcal{A}_s^0$, X is a singleton component of $\mathcal{A}_s$ that has no level, and S_σ^0 is the copy of the σ-special component of $\mathcal{G}_{|\sigma|}[s]$ in $\mathcal{A}_s^0$. For each $i \in \mathbb{Z}$, Z_σ^i, B_σ^i, S_σ^i, C_σ^i, $Y_{\sigma,0}^i$, $Y_{\sigma,1}^i$, $Y_{\sigma,2}^i$, D^i, E^i, F_0^i, F_1^i, and F_2^i are the components of $\mathcal{A}_s^i$ isomorphic to Z_σ^0, B_σ^0, S_σ^0, C_σ^0, $Y_{\sigma,0}^0$, $Y_{\sigma,1}^0$, $Y_{\sigma,2}^0$, D^0, E^0, F_0^0, F_1^0, and F_2^0, respectively.

Suppose that $s+1$ is a σ-recovery stage such that $r_{\sigma,s+1} = 0$ and we perform the operation pictured in Figure 3.4 on $\mathcal{A}_s$ to obtain $\mathcal{A}_{s+1}$, and then wait for $\mathcal{G}_{|\sigma|}$ to σ-recover at some stage $t+1 > s+1$. Notice that this operation preserves the relevant automorphisms of $\mathcal{A}$. That is, if $(\mathcal{A}_s, a_0) \cong (\mathcal{A}_s, a_i)$ then $(\mathcal{A}_{s+1}, a_0) \cong (\mathcal{A}_{s+1}, a_i)$.

The definition of σ-recovery will be such that $\mathcal{G}_{|\sigma|}[s]$ contains a component $\widehat{X}$ isomorphic to X and, for some $k \in \omega$ and all $i \in \mathbb{Z}$ with $|i| \leqslant k$, $\mathcal{G}_{|\sigma|}[s]$ contains level-i components $\widehat{Z}^i$, $\widehat{B}^i$, $\widehat{S}^i$, $\widehat{C}^i$, $\widehat{Y}_0^i$, $\widehat{Y}_1^i$, $\widehat{Y}_2^i$, $\widehat{D}^i$, $\widehat{E}^i$, $\widehat{F}_0^i$, $\widehat{F}_1^i$, and $\widehat{F}_2^i$ isomorphic to Z_σ^i, B_σ^i, S_σ^i, C_σ^i, $Y_{\sigma,0}^i$, $Y_{\sigma,1}^i$, $Y_{\sigma,2}^i$, D^i, E^i, F_0^i, F_1^i, and F_2^i, respectively.

Let $\overline{\mathcal{A}}_t$ be the union of $(\mathcal{A}_t^*)^k$ and $\mathcal{A}_t^i$ for each $i \in \mathbb{Z}$ such that $|i| \leqslant k$. Let $\overline{\mathcal{G}}_{|\sigma|}[t]$ be the union of $(\mathcal{G}_{|\sigma|}^*[t])^k$ and $\mathcal{G}_{|\sigma|}^i[t]$ for each $i \in \mathbb{Z}$ such that $|i| \leqslant k$.

For $i \in \mathbb{Z}$, $|i| \leqslant k$, let $\widetilde{X}$, $\widetilde{Z}^i$, $\widetilde{B}^i$, $\widetilde{S}^i$, $\widetilde{C}^i$, $\widetilde{Y}_0^i$, $\widetilde{Y}_1^i$, $\widetilde{Y}_2^i$, $\widetilde{D}^i$, $\widetilde{E}^i$, $\widetilde{F}_0^i$, $\widetilde{F}_1^i$, and $\widetilde{F}_2^i$ be the intersection of the components of $\mathcal{G}_{|\sigma|}[t]$ that extend $\widehat{X}$, $\widehat{Z}^i$, $\widehat{B}^i$, $\widehat{S}^i$, $\widehat{C}^i$, $\widehat{Y}_0^i$, $\widehat{Y}_1^i$, $\widehat{Y}_2^i$, $\widehat{D}^i$, $\widehat{E}^i$, $\widehat{F}_0^i$, $\widehat{F}_1^i$, and $\widehat{F}_2^i$, respectively, with $\overline{\mathcal{G}}_{|\sigma|}[t]$.

The fact that $\mathcal{G}_{|\sigma|}$ σ-recovers at stage $t+1$ will mean that $\widetilde{X}$, $\widetilde{Z}^i$, $\widetilde{B}^i$, $\widetilde{S}^i$, $\widetilde{C}^i$, $\widetilde{Y}_0^i$, $\widetilde{Y}_1^i$, $\widetilde{Y}_2^i$, $\widetilde{D}^i$, $\widetilde{E}^i$, $\widetilde{F}_0^i$, $\widetilde{F}_1^i$, and $\widetilde{F}_2^i$ for $|i| \leqslant k$ must all be isomorphic to components of $\overline{\mathcal{A}}_t$.

Thus, since σ is not active in the interval $(s+1, t+1)$, there are two possibilities. Either the σ-special component $\widetilde{S}^0$ of $\mathcal{G}_{|\sigma|}[t]$ is isomorphic to $S_\sigma^0 \cdot B_\sigma^0$ or it is isomorphic to $S_\sigma^0 \cdot C_\sigma^0$.

In the first case, $r_{\sigma,t+1} \neq 0$, and the copy of the σ-special component of $\mathcal{G}_{|\sigma|}[t]$ in $\mathcal{A}_t$ extends a singleton component of $\mathcal{A}_s$. In fact, every time we have an action at stage $u+1$ involving copies of the σ-special component of $\mathcal{G}_{|\sigma|}[u]$ and, for the next σ-recovery stage $v+1$, we have $r_{\sigma,v+1} \neq r_{\sigma,u+1}$, then the copy of the σ-special component

$$\begin{array}{ccc} D^0 & & E^0 \\ \downarrow & & \downarrow \\ D^0 \cdot E^0 & & E^0 \odot (F_0^0, F_1^0, F_2^1) \end{array}$$

similarly for all levels $\equiv 0 \bmod 4$

$$\begin{array}{ccc} E^1 & & F_0^1 \\ \downarrow & & \downarrow \\ E^1 \cdot D^1 & & F_0^1 \odot (E^1, F_1^1, F_2^1) \end{array}$$

similarly for all levels $\equiv 1 \bmod 4$

$$\begin{array}{ccc} E^2 & & F_1^2 \\ \downarrow & & \downarrow \\ E^2 \cdot D^2 & & F_1^2 \odot (E^2, F_0^2, F_2^2) \end{array}$$

similarly for all levels $\equiv 2 \bmod 4$

$$\begin{array}{ccc} E^3 & & F_2^3 \\ \downarrow & & \downarrow \\ E^3 \cdot D^3 & & F_2^3 \odot (E^3, F_0^3, F_1^3) \end{array}$$

similarly for all levels $\equiv 3 \bmod 4$

Figure 3.4: A $3, (0, 1)$-operation

Figure 3.4 (Continued)

$$
\begin{array}{cccc}
Z^0_\sigma & B^0_\sigma & S^0_\sigma & C^0_\sigma \\
\downarrow & \downarrow & \downarrow & \downarrow \\
Z^0_\sigma \cdot B^0_\sigma & B^0_\sigma \cdot S^0_\sigma & S^0_\sigma \cdot C^0_\sigma & C^0_\sigma \odot (Y^0_{\sigma,0}, Y^0_{\sigma,1}, Y^0_{\sigma,2})
\end{array}
$$

similarly for all levels $\equiv 0 \bmod 4$

$$
\begin{array}{cccc}
B^1_\sigma & S^1_\sigma & C^1_\sigma & Y^1_{\sigma,0} \\
\downarrow & \downarrow & \downarrow & \downarrow \\
B^1_\sigma \cdot Z^1_\sigma & S^1_\sigma \cdot B^1_\sigma & C^1_\sigma \cdot S^1_\sigma & Y^1_{\sigma,0} \odot (C^1_\sigma, Y^1_{\sigma,1}, Y^1_{\sigma,2})
\end{array}
$$

similarly for all levels $\equiv 1 \bmod 4$

$$
\begin{array}{cccc}
B^2_\sigma & S^2_\sigma & C^2_\sigma & Y^2_{\sigma,1} \\
\downarrow & \downarrow & \downarrow & \downarrow \\
B^2_\sigma \cdot Z^2_\sigma & S^2_\sigma \cdot B^2_\sigma & C^2_\sigma \cdot S^2_\sigma & Y^2_{\sigma,1} \odot (C^2_\sigma, Y^2_{\sigma,0}, Y^2_{\sigma,2})
\end{array}
$$

similarly for all levels $\equiv 2 \bmod 4$

$$
\begin{array}{cccc}
B^3_\sigma & S^3_\sigma & C^3_\sigma & Y^3_{\sigma,2} \\
\downarrow & \downarrow & \downarrow & \downarrow \\
B^3_\sigma \cdot Z^3_\sigma & S^3_\sigma \cdot B^3_\sigma & C^3_\sigma \cdot S^3_\sigma & Y^3_{\sigma,2} \odot (C^3_\sigma, Y^3_{\sigma,0}, Y^3_{\sigma,1})
\end{array}
$$

similarly for all levels $\equiv 3 \bmod 4$

Figure 3.4 (Continued)

$$\begin{tikzcd}
B_\tau^0 \arrow[d] & S_\tau^0 \arrow[d] & C_\tau^0 \arrow[d] & Y_{\tau,2}^0 \arrow[d] \\
B_\tau^0 \cdot Z_\tau^0 & S_\tau^0 \cdot B_\tau^0 & C_\tau^0 \cdot S_\tau^0 & Y_{\tau,2}^0 \odot (C_\tau^0, Y_{\tau,0}^0, Y_{\tau,1}^0)
\end{tikzcd}$$

similarly for all levels $\equiv 0 \bmod 4$

$$\begin{tikzcd}
Z_\tau^1 \arrow[d] & B_\tau^1 \arrow[d] & S_\tau^1 \arrow[d] & C_\tau^1 \arrow[d] \\
Z_\tau^1 \cdot B_\tau^1 & B_\tau^1 \cdot S_\tau^1 & S_\tau^1 \cdot C_\tau^1 & C_\tau^1 \odot (Y_{\tau,0}^1, Y_{\tau,1}^1, Y_{\tau,2}^1)
\end{tikzcd}$$

similarly for all levels $\equiv 1 \bmod 4$

$$\begin{tikzcd}
B_\tau^2 \arrow[d] & S_\tau^2 \arrow[d] & C_\tau^2 \arrow[d] & Y_{\tau,1}^2 \arrow[d] \\
B_\tau^2 \cdot Z_\tau^2 & S_\tau^2 \cdot B_\tau^2 & C_\tau^2 \cdot S_\tau^2 & Y_{\tau,0}^2 \odot (C_\tau^2, Y_{\tau,1}^2, Y_{\tau,2}^2)
\end{tikzcd}$$

similarly for all levels $\equiv 2 \bmod 4$

$$\begin{tikzcd}
B_\tau^3 \arrow[d] & S_\tau^3 \arrow[d] & C_\tau^3 \arrow[d] & Y_{\tau,1}^3 \arrow[d] \\
B_\tau^3 \cdot Z_\tau^3 & S_\tau^3 \cdot B_\tau^3 & C_\tau^3 \cdot S_\tau^3 & Y_{\tau,1}^3 \odot (C_\tau^3, Y_{\tau,0}^3, Y_{\tau,2}^3)
\end{tikzcd}$$

similarly for all levels $\equiv 3 \bmod 4$

Figure 3.4 (Continued)

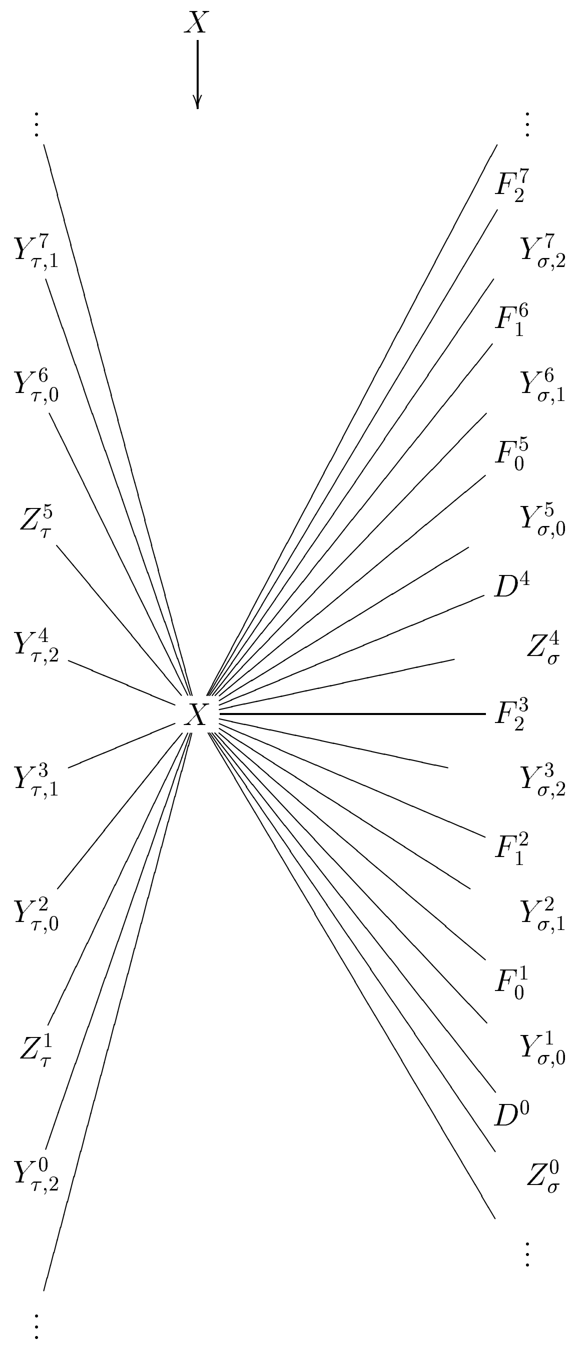


Figure 3.4 (Continued)

$$\underbrace{\begin{array}{c} \ldots, F_2^0, D^1, F_0^2, F_1^3, F_2^4, D^5, F_0^6, F_1^7, \ldots \\ \ldots, Y_{\sigma,2}^0, Z_\sigma^1, Y_{\sigma,0}^2, Y_{\sigma,1}^3, Y_{\sigma,2}^4, Z_\sigma^5, Y_{\sigma,0}^6, Y_{\sigma,1}^7, \ldots \\ \ldots, Y_{\tau,1}^0, Y_{\tau,2}^1, Z_\tau^2, Y_{\tau,0}^3, Y_{\tau,1}^4, Y_{\tau,2}^5, Z_\tau^6, Y_{\tau,0}^7, \ldots \end{array}}$$

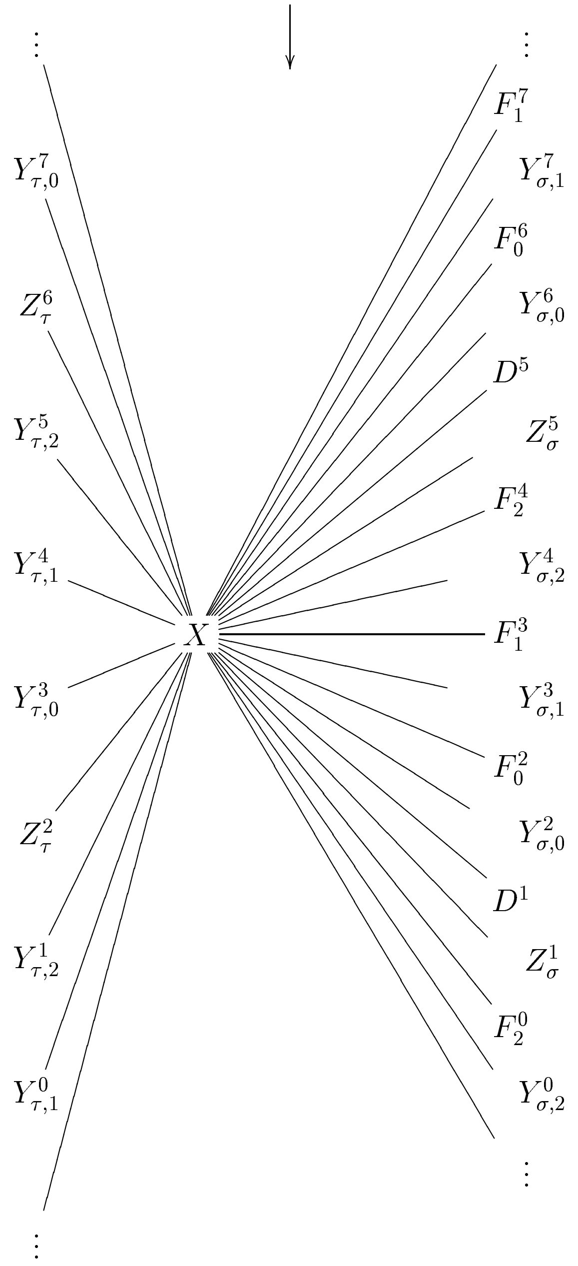


Figure 3.4 (Continued)

$$\ldots, F_1^0, F_2^1, D^2, F_0^3, F_1^4, F_2^5, D^6, F_0^7, \ldots$$
$$\ldots, Y_{\sigma,1}^0, Y_{\sigma,2}^1, Z_\sigma^2, Y_{\sigma,0}^3, Y_{\sigma,1}^4, Y_{\sigma,2}^5, Z_\sigma^6, Y_{\sigma,0}^7, \ldots$$
$$\ldots, Y_{\tau,0}^0, Y_{\tau,1}^1, Y_{\tau,2}^2, Z_\tau^3, Y_{\tau,0}^4, Y_{\tau,1}^5, Y_{\tau,2}^6, Z_\tau^7, \ldots$$

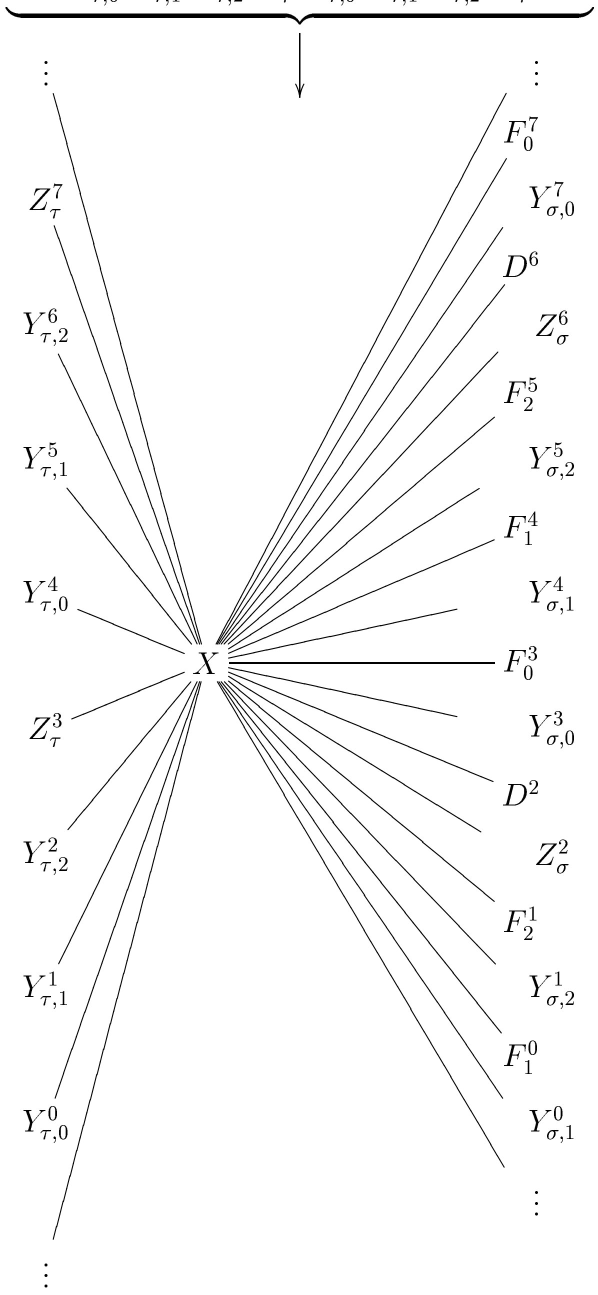
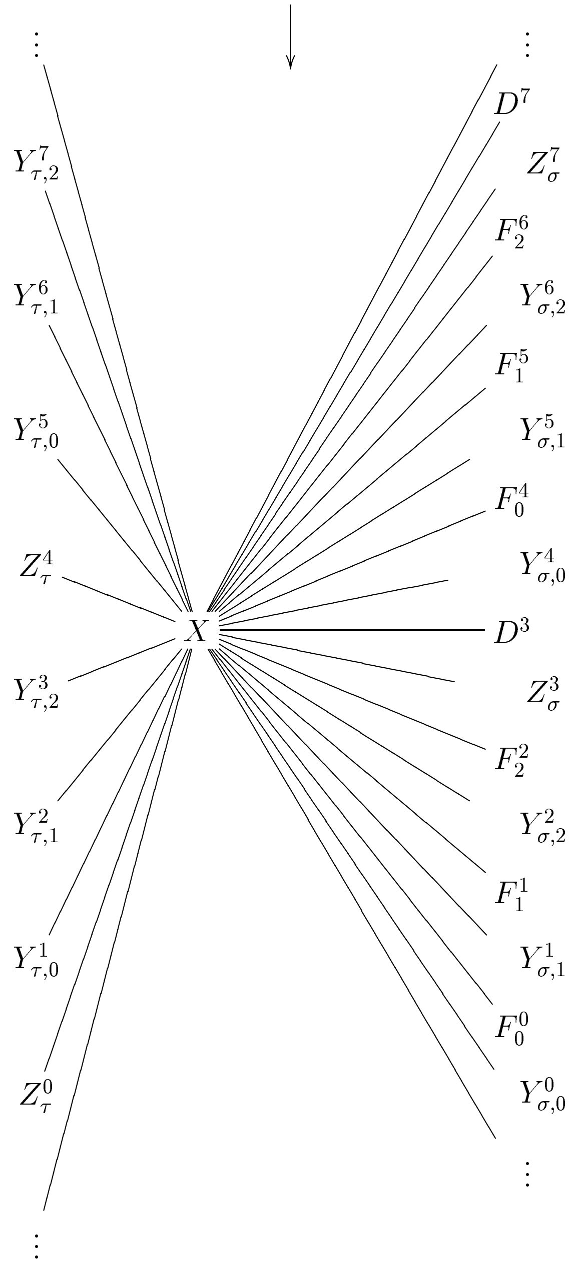


Figure 3.4 (Continued)

$$\underbrace{\begin{array}{c} \ldots, F_0^0, F_1^1, F_2^2, D^3, F_0^4, F_1^5, F_2^6, D^7, \ldots \\ \ldots, Y_{\sigma,0}^0, Y_{\sigma,1}^1, Y_{\sigma,2}^2, Z_\sigma^3, Y_{\sigma,0}^4, Y_{\sigma,1}^5, Y_{\sigma,2}^6, Z_\sigma^7, \ldots \\ \ldots, Z_\tau^0, Y_{\tau,0}^1, Y_{\tau,1}^2, Y_{\tau,2}^3, Z_\tau^4, Y_{\tau,0}^5, Y_{\tau,1}^6, Y_{\tau,2}^7, \ldots \end{array}}$$



of $\mathcal{G}_{|\sigma|}[v]$ in $\mathcal{A}_v$ will extend a singleton component of $\mathcal{A}_u$. We will make sure that if $r_{\sigma,u}$ has no limit then, for each $i \in \mathbb{Z}$, there are infinitely many stages u such that $r_{\sigma,u} = i$. This will guarantee that if σ is on the true path of the construction, $\mathcal{G}_{|\sigma|} \cong \mathcal{A}$, and $r_{\sigma,u}$ has no limit then there is no component of $\mathcal{A}$ isomorphic to the σ-special component of $\mathcal{G}_{|\sigma|}[t]$.

In the second case, we can check that, for each $i \in \mathbb{Z}$ such that $|i| \leqslant k$, level i of $\mathcal{G}_{|\sigma|}$ must have gone in the same direction as level i of $\mathcal{A}$ with respect to those components involved in the operation under consideration.

That is, let $\check{X}$, $\check{Z}^i$, $\check{B}^i$, $\check{S}^i$, $\check{C}^i$, $\check{Y}^i_0$, $\check{Y}^i_1$, $\check{Y}^i_2$, $\check{D}^i$, $\check{E}^i$, $\check{F}^i_0$, $\check{F}^i_1$, and $\check{F}^i_2$ be the intersection of the components of $\mathcal{A}_t$ that extend X, Z^i_σ, B^i_σ, S^i_σ, C^i_σ, $Y^i_{\sigma,0}$, $Y^i_{\sigma,1}$, $Y^i_{\sigma,2}$, D^i, E^i, F^i_0, F^i_1, and F^i_2, respectively, with $\overline{\mathcal{A}}_t$. All the components of $\mathcal{A}_t$ that contain copies of C^0_σ are isomorphic to either $\check{S}^0$ or $\check{C}^0$. Since $\widetilde{S}^0$ and $\widetilde{C}^0$ have the same level, it cannot be the case that $\widetilde{S}^0 \cong \widetilde{C}^0$. Thus, since we are assuming that $\widetilde{S}^0 \cong \check{S}^0$, it follows that $\widetilde{C}^0 \cong \check{C}^0$.

Continuing to argue in this way, we see that $\widetilde{Y}^0_l \cong \check{Y}^0_l$ for each $l < 3$, which implies that $\widetilde{X} \cong \check{X}$, which implies that $\widetilde{Z}^0 \cong \check{Z}^0$, which implies that $\widetilde{B}^0 \cong \check{B}^0$.

Now, using the fact that $\widetilde{X} \cong \check{X}$, we can check that, for $i \equiv 0 \bmod 4$ such that $|i| \leqslant k$, we have $\widetilde{Z}^i \cong \check{Z}^i$, which implies that $\widetilde{B}^i \cong \check{B}^i$, which implies that $\widetilde{S}^i \cong \check{S}^i$, which implies that $\widetilde{C}^i \cong \check{C}^i$, which implies that $\widetilde{Y}^i_l \cong \check{Y}^i_l$ for each $l < 3$.

If $i \equiv l+1 \bmod 4$ with $l < 3$ and $|i| \leqslant k$ then, again using the fact that $\widetilde{X} \cong \check{X}$, we can check that $\widetilde{Y}^i_l \cong \check{Y}^i_l$, which implies that $\widetilde{Y}^i_m \cong \check{Y}^i_m$ for each $m < 3$, and also that $\widetilde{C}^i \cong \check{C}^i$. This in turn implies that $\widetilde{S}^i \cong \check{S}^i$, which implies that $\widetilde{B}^i \cong \check{B}^i$, which implies that $\widetilde{Z}^i \cong \check{Z}^i$.

Similar arguments show that, for all $i \in \mathbb{Z}$ such that $|i| \leqslant k$, we have $\widetilde{D}^i \cong \check{D}^i$, $\widetilde{E}^i \cong \check{E}^i$, and $\widetilde{F}^i_l \cong \check{F}^i_l$ for each $l < 3$.

Thus we see that, for each $i \in \mathbb{Z}$ such that $|i| \leqslant k$, level i of $\mathcal{G}_{|\sigma|}$ goes in the same direction as level i of $\mathcal{A}$ with respect to those components involved in the operation under consideration. (Since k increases with each σ-recovery, the fact that the above argument only works for the level-i components such that $|i| \leqslant k$ will not be a problem.)

In the previous argument, the fact that $r_{\sigma,s+1} = 0$ was crucial. Indeed, suppose that $r_{\sigma,s+1} = 1$, we perform the operation described above at stage $s+1$, σ then recovers at stage $t+1$, and $r_{\sigma,t+1} = 1$. We could not then argue as above, because from the fact that $\widetilde{S}^1 \cong \check{S}^1$ it does not follow that $\widetilde{X} \cong \check{X}$. Thus, to argue that, for each $i \in \mathbb{Z}$, there is a stage after which level i of $\mathcal{G}_{|\sigma|}$ always goes in the same direction as level $i + \lim_s r_{\sigma,s}$ of $\mathcal{A}$ at stages at which σ is active, we need to make sure that, whenever we involve copies of the σ-special component of $\mathcal{G}_{|\sigma|}$ in an operation at stage $s+1$, the row of level-$(r_{\sigma,s+1})$

components of $\mathcal{A}$ that contains a copy of the σ-special component of $\mathcal{G}_{|\sigma|}$ goes to the left.

This is illustrated in Figure 3.4. Here we are assuming that the operation pictured is happening at a stage $s+1$ such that $r_{\sigma,s+1} = 0$ and $r_{\tau,s+1} = 1$ (that is why we call this a $3,(0,1)$-operation). Now if σ recovers at a stage $t+1 > s+1$ and $r_{\sigma,t+1} = 0$ then we can argue as above that if $|i|$ is sufficiently small then level i of $\mathcal{G}_{|\sigma|}$ goes in the same direction as level i of $\mathcal{A}$ as far as the components involved in this operation are concerned. But also, if τ recovers at a stage $t+1 > s+1$ and $r_{\tau,t+1} = 1$ then we can argue in much the same way that if $|i|$ is sufficiently small then level i of $\mathcal{G}_{|\tau|}$ goes in the same direction as level $i+1$ of $\mathcal{A}$ as far as the components involved in this operation are concerned.

In general, whenever we perform an operation at a stage $s+1$ at which the strings $\sigma_0, \ldots, \sigma_{k-1}$ are active, that operation will be an $n,(r_{\sigma_0,s+1}, \ldots, r_{\sigma_{k-1},s+1})$-operation for some $n > 0$, as defined below.

3.4 Definition. Let $\mathcal{G}$ be a computable leveled graph whose domain is co-infinite.

Let $n > 0$, $k \geqslant 0$, and $d_0, \ldots, d_{k-1} \in \mathbb{Z}$. Suppose that, for each $i \in \mathbb{Z}$, $j < k$, and $m < n$, we have defined components $Y^i_{j,m}$, X, Z^i_j, B^i_j, S^i_j, C^i_j, D^i, E^i, and F^i_m, of which all but S^i_j are singleton components, all but X have levels, and X has no level.

The $n,(d_0, \ldots, d_{k-1})$-*operation*

$$\mathbf{O}_{n,(d_0,\ldots,d_{k-1})}\left(\left\{Y^i_{j,m}\right\}, X, \left\{Z^i_j\right\}, \left\{B^i_j\right\}, \left\{S^i_j\right\}, \left\{C^i_j\right\}, \left\{D^i\right\}, \left\{E^i\right\}, \left\{F^i_m\right\}\right)$$

consists of applying the following sequences of operations to $\mathcal{G}$.

- $X \cdot \mathcal{S}_0, \mathcal{S}_1 \cdot X, \mathcal{S}_2 \cdot X, \ldots, \mathcal{S}_n \cdot X$, where

$$\begin{aligned}\mathcal{S}_m = \left\{Z_j^{d_j+m+p(n+1)} \mid j < k,\ p \in \mathbb{Z}\right\} \cup \left\{D^{m+p(n+1)} \mid p \in \mathbb{Z}\right\} \cup \\ \left\{Y_{j,q}^{d_j+m+q+1+p(n+1)} \mid j < k,\ p \in \mathbb{Z},\ q < n\right\} \cup \\ \left\{F_q^{m+q+1+p(n+1)} \mid p \in \mathbb{Z},\ q < n\right\}.\end{aligned}$$

- For each $i \equiv 0 \bmod n+1$ in $\mathbb{Z}$:

$$D^i \cdot E^i,\ E^i \odot \left(F^i_0, \ldots, F^i_{n-1}\right).$$

- For each $j < k$ and $i \equiv d_j \bmod n+1$ in $\mathbb{Z}$:

$$Z^i_j \cdot B^i_j,\ B^i_j \cdot S^i_j,\ S^i_j \cdot C^i_j,\ C^i_j \odot \left(Y^i_{j,0}, \ldots, Y^i_{j,n-1}\right).$$

- For each $i \in \mathbb{Z}$ such that $i \equiv l + 1 \bmod n + 1$ with $l < n$:

$$E^i \cdot D^i,\ F^i_l \odot \left(E^i, F^i_0, \ldots, F^i_{l-1}, F^i_{l+1}, \ldots, F^i_{n-1}\right).$$

- For each $j < k$ and $i \in \mathbb{Z}$ such that $i \equiv l + d_j + 1 \bmod n + 1$ with $l < n$:

$$B^i_j \cdot Z^i_j,\ S^i_j \cdot B^i_j,\ C^i_j \cdot S^i_j,\ Y^i_{j,l} \odot \left(C^i_j, Y^i_{j,0}, \ldots, Y^i_{j,l-1}, Y^i_{j,l+1}, \ldots, Y^i_{j,n-1}\right).$$

Note that this definition allows for the case $k = 0$, in which the only components involved in the operation are X and the D^i, E^i, and F^i_m.

We now discuss the version of the catch-up procedure that will be used in our construction. For each stage s, we define a subgraph $T_{\sigma,s}$ of the σ-special component of $\mathcal{G}_{|\sigma|}$. Whenever $r_{\sigma,s+1} \neq r_{\sigma,s}$, we define $T_{\sigma,s+1}$ to be the entire σ-special component of $\mathcal{G}_{|\sigma|}[s]$. Otherwise, we define $T_{\sigma,s+1} = T_{\sigma,s}$. Whenever copies of the σ-special component of $\mathcal{G}_{|\sigma|}$ are involved in an operation and σ later recovers at a stage $t + 1$, we perform a $T_{\sigma,t+1}$-*catch-up operation* on $\mathcal{A}_s$, as defined below. (Copies of $[10m + 9]$ will not be used for any other purpose in the construction.)

3.5 Definition. Let $\mathcal{G}$ be a computable leveled graph with co-infinite domain that contains copies of $[10m + 9]$ for only finitely many $m \in \omega$.

Let T be a subgraph of $\mathcal{G}$. Suppose that there are finitely many level-0 components $L_{0,0}, \ldots, L_{0,n}$ of $\mathcal{G}$ that contain a copy of T and that each $L_{0,m}$, $m \leqslant n$, is a copy of $[P_m]$ for some finite $P_m \subset \omega$. Let $P = \bigcup_{m \leqslant n} P_m$.

For $i \in \mathbb{Z}$, let $L_{i,0}, \ldots, L_{i,n}$ be the components of $\mathcal{G}^i$ isomorphic to $L_{0,0}, \ldots, L_{0,n}$, respectively.

Let $l_0, \ldots, l_n$ be the $n + 1$ least numbers of the form $10m + 9$, $m \in \omega$, such that $\mathcal{G}$ does not contain copies of any of $[l_0], \ldots, [l_n]$.

The T-catch-up operation taking $\mathcal{G}$ to a new computable structure extending $\mathcal{G}$ consists of extending each $L_{i,m}$, $i \in \mathbb{Z}$, $m \leqslant n$, to a copy of $[P \cup \{l_m\}]$, using numbers not in the domain of $\mathcal{G}$.

Performing a $T_{\sigma,t+1}$-catch-up operation on $\mathcal{A}_s$ will count as σ being active, which means that we must then wait for σ-recovery before allowing σ to be active again.

If σ is on the true path of the construction and $r_{\sigma,s}$ comes to a limit then $T_{\sigma,s}$ comes to a limit T. It is not hard to see that, in this case, by performing catch-up operations as described, we guarantee that every component of $\mathcal{A}$ that contains a copy of T is infinite, and that all such components are isomorphic. This will be enough to ensure that (3.2) is satisfied, while at the same time helping us to construct the computable isomorphisms needed to satisfy (3.3), because it will mean that if $\mathcal{G}_{|\sigma|} \cong \mathcal{A}$ then any embedding of a copy of T in $\mathcal{A}$ into $\mathcal{G}_{|\sigma|}$ can be extended to an isomorphism from $\mathcal{A}$ to $\mathcal{G}_{|\sigma|}$.

We will separate the stages at which we attempt to satisfy the $\mathcal{R}$-requirements from the stages at which we perform catch-up operations, reserving the even stages for the former purpose and the odd ones for the latter. To guarantee that every σ on the true path is active at infinitely many even stages and infinitely many odd stages, we will call recovery at odd stages *phase*-1 *recovery* and recovery at even stages *phase*-2 *recovery*, and will require that phase-1 σ-recovery stages and phase-2 σ-recovery stages alternate.

There will be multiple strategies for satisfying each requirement $\mathcal{R}_{\langle e,i\rangle}$, one strategy R_σ for each $\sigma \in 2^{<\omega}$ such that $|\sigma| = \langle e,i\rangle$. Each of these strategies will work with a different set of components, which will be subject to initialization. If σ is accessible at some stage $2s+2$ in the construction and no requirements of stronger priority require attention then R_σ will have a chance to act as described above. If it does then copies of the τ-special components of $G_{|\tau|}[2s+1]$ will be involved in the operation performed at stage $2s+2$ if and only if $\tau^\frown 0 \subseteq \sigma$, that is, if and only if $2s+2$ is a τ-recovery stage.

We now give a few more definitions and conventions which will be used below.

Fix a computable one-to-one function from $2^{<\omega}$ onto ω and let $\ulcorner\sigma\urcorner$ denote the image under this function of the string σ. Fix a computable function ξ from ω onto $\mathbb{Z}$ such that, for each $i \in \mathbb{Z}$, there are infinitely many $n \in \omega$ for which $\xi(n) = i$.

3.6 Definition. Let $\mathcal{G}$ be a directed graph. We denote by $(\mathcal{G})_\sigma$ the subgraph of $\mathcal{G}$ consisting of those components C of $\mathcal{G}$ that satisfy both of the following conditions.

1. C is not isomorphic to $[x]$ for any $x \in \omega$.
2. C contains a copy of $[10\langle\ulcorner\sigma\urcorner, j\rangle + l]$, $j \in \omega$, $l \in \{2,3,4,5,6,7\}$, or a copy of $[10\langle\ulcorner\sigma\urcorner, j, k\rangle + l]$, $j,k \in \omega$, $l \in \{1,8\}$.

Define $(\mathcal{G})_{\supseteq\sigma} = \bigcup_{\tau\supseteq\sigma}(\mathcal{G})_\tau$.

In the particular case of $\mathcal{G}^*$, we will wish to define $(\mathcal{G}^*)_\sigma$ somewhat differently.

3.7 Definition. Let $\mathcal{G}$ be a leveled graph. We denote by $(\mathcal{G}^*)_\sigma$ the subgraph of $\mathcal{G}$ consisting of the non-singleton components of $\mathcal{G}^*$ that contain a copy of $[10\langle\ulcorner\sigma\urcorner, j\rangle]$, $j \in \omega$. Let $n, r \in \omega$. We denote by $(\mathcal{G}^*)^{n,r}_\sigma$ the subgraph of $\mathcal{G}$ consisting of the non-singleton components of $(\mathcal{G}^*)^{n,r}$ that contain a copy of $[10\langle\ulcorner\sigma\urcorner, j\rangle]$, $j \in \omega$. If $r = 0$ then we write simply $(\mathcal{G}^*)^n_\sigma$.

Define $(\mathcal{G}^*)_{\supseteq\sigma} = \bigcup_{\tau\supseteq\sigma}(\mathcal{G}^*)_\tau$ and $(\mathcal{G}^*)^{n,r}_{\supseteq\sigma} = \bigcup_{\tau\supseteq\sigma}(\mathcal{G}^*)^{n,r}_\tau$. If $r = 0$ then we write simply $(\mathcal{G}^*)^n_{\supseteq\sigma}$.

Let k be the number of times σ has been initialized before stage t. Suppose there is a least stage $s \leqslant t$ such that $\mathcal{G}_{|\sigma|}[s]$ has a level-0 component K isomorphic to $[10\langle\ulcorner\sigma\urcorner, k\rangle+3]$. We call the component of $\mathcal{G}_{|\sigma|}[t]$ that extends K the *σ-special component of $\mathcal{G}_{|\sigma|}[t]$*. If σ is initialized only finitely often, say k many times, and there is a least stage s such that $\mathcal{G}_{|\sigma|}[s]$ has a level-0 component K isomorphic to $[10\langle\ulcorner\sigma\urcorner, k\rangle + 3]$ then we call the component of $\mathcal{G}_{|\sigma|}$ that extends K the *σ-special component of $\mathcal{G}_{|\sigma|}$*.

3.1 The Construction

We now proceed with the construction of $\mathcal{A}$. We will start with a computable structure $\mathcal{A}^0$ with co-infinite domain. To ensure that we can carry out the construction, we require that, when we add elements to the domain of $\mathcal{A}_s$ at stage $s+1$ to get $\mathcal{A}_{s+1}$, we do this in such a way that $\mathcal{A}_{s+1}$ remains co-infinite. To ensure that $\mathcal{A}$ is computable, we require that the collection of sets $(|\mathcal{A}_{s+1}| - |\mathcal{A}_s|)_{s\in\omega}$ be uniformly computable.

stage 0. Let $\mathcal{A}_0$ be a computable leveled graph with co-infinite domain consisting of the following nodes and edges in addition to the ones required by Definition 3.1.

1. For each $i \in \mathbb{Z}$, $k \in \omega$, and $0 < l < 9$, a copy of $[10k + l]$ with an edge from the i-master node to its top.
2. For each $k \in \omega$, a copy of $[10k]$.

For each $\sigma \in 2^{<\omega}$, let $r_{\sigma,0} = 0$ and $T_{\sigma,0} = \emptyset$.

stage $2s + 1$. For $\sigma \in 2^{<\omega}$, let $recov(\sigma, 2s)$ be the number of σ-recovery stages before stage $2s + 1$. Define the string $\sigma[2s + 1] \in 2^{[0,s]}$ by recursion as follows, beginning with $n = 0$. Let $\sigma = \sigma[2s + 1] \upharpoonright n$. Say that $2s + 1$ is a *phase*-1 *σ-recovery stage* and that σ is *semi-recovered* if all of the following conditions hold.

1. σ is not currently semi-recovered.
2. Every τ such that $\tau^\frown 0 \subseteq \sigma$ has fully recovered (defined below) at least $|\sigma| + 1$ many times.
3. $\mathcal{G}_n[2s]$ has a σ-special component isomorphic to some component of $\mathcal{A}^0_{2s}$.

4. For each $i \in \mathbb{Z}$ such that $|i| \leqslant recov(\sigma, 2s)$, $(\mathcal{G}_n^i[2s])_\sigma \cong (\mathcal{A}_{2s}^0)_\sigma$.

5. For each $i \in \mathbb{Z}$ such that $|i| \leqslant recov(\sigma, 2s)$, $(\mathcal{G}_n^i[2s])_{\supseteq\sigma^\frown 0} \cong (\mathcal{A}_{2s}^0)_{\supseteq\sigma^\frown 0}$.

6. $(\mathcal{G}_n^*[2s])_{\supseteq\sigma^\frown 0}^{recov(\sigma,2s)} \cong (\mathcal{A}_{2s}^*)_{\supseteq\sigma^\frown 0}^{recov(\sigma,2s)}$.

If $2s+1$ is a σ-recovery stage then let $\sigma[2s+1](n) = 0$. Otherwise, let $\sigma[2s+1](n) = 1$.

For each σ such that $2s+1$ is a σ-recovery stage, proceed as follows. Let $i = r_{\sigma,2s}$. Let $S_{\sigma,2s}$ be the component of $\mathcal{A}_{2s}^i$ that is isomorphic to the σ-special component of $\mathcal{G}_{|\sigma|}[2s]$. If $2s+1$ is either the first σ-recovery stage ever or the first σ-recovery stage since the last time σ was initialized then let $r_{\sigma,2s+1} = 0$ and $T_{\sigma,2s+1} = S_{\sigma,2s}$. Otherwise, proceed as follows. Let $2t+2$ be the last σ-recovery stage before stage $2s+1$. If $S_{\sigma,2s}$ extends $S_{\sigma,2t+1}$ then let $r_{\sigma,2s+1} = i$ and $T_{\sigma,2s+1} = T_{\sigma,2s}$; otherwise, declare $2s+1$ to be a *σ-change stage*, let n be the number of σ-change stages before stage $2s+1$, let $r_{\sigma,2s+1} = \xi(n)$, and let $T_{\sigma,2s+1} = S_{\sigma,2s}$.

For each $\sigma \in 2^{<\omega}$ such that $2s+1$ is not a σ-recovery stage, let $r_{\sigma,2s+1} = r_{\sigma,2s}$ and $T_{\sigma,2s+1} = T_{\sigma,2s}$.

Declare each σ to the right of $\sigma[2s+1]$ to have been *initialized*. This includes declaring σ to be neither semi-recovered nor fully recovered.

Proceed as follows to obtain $\mathcal{A}_{2s+1}$ from $\mathcal{A}_{2s}$. For each $\sigma \in 2^{<\omega}$ such that $2s+1$ is a σ-recovery stage, perform the $T_{\sigma,2s+1}$-catch-up operation and say that σ is *active* at stage $2s+1$.

stage $2s+2$. For $\sigma \in 2^{<\omega}$, let $recov(\sigma, 2s+1)$ be the number of σ-recovery stages before stage $2s+2$, let $init(\sigma, 2s+1)$ be the number of times σ has been initialized before stage $2s+2$, and let $c(\sigma, 2s+1) = \max(recov(\sigma, 2s+1), init(\sigma, 2s+1))$.

Define the string $\sigma[2s+2] \in 2^{[0,s]}$ by recursion as follows, beginning with $n = 0$. Say that $2s+2$ is a *phase-2 σ-recovery stage* and that σ is *fully recovered* (and hence not semi-recovered) if all of the following conditions hold.

1. σ is currently semi-recovered.

2. The σ-special component of $\mathcal{G}_n[2s+1]$ is isomorphic to some component of $\mathcal{A}_{2s+1}^0$.

3. For each $i \in \mathbb{Z}$ such that $|i| \leqslant recov(\sigma, 2s+1)$, $(\mathcal{G}_n^i[2s+1])_\sigma \cong (\mathcal{A}_{2s+1}^0)_\sigma$.

4. For each $i \in \mathbb{Z}$ such that $|i| \leqslant recov(\sigma, 2s+1)$, $(\mathcal{G}_n^i[2s+1])_{\supseteq\sigma^\frown 0} \cong (\mathcal{A}_{2s+1}^0)_{\supseteq\sigma^\frown 0}$.

5. If $\tau \supseteq \sigma^\frown 0$ has not yet fully recovered since the last time it was initialized and $|\tau| \leqslant recov(\sigma, 2s{+}1)$ then, for each $i \in \mathbb{Z}$ such that $|i| \leqslant recov(\sigma, 2s{+}1)$, $\mathcal{G}^i_n[2s{+}1]$ has a component isomorphic to $[10\langle\ulcorner\tau\urcorner, init(\tau, 2s+1)\rangle + 3]$.

6. Let τ be such that either $\tau = \sigma$ or both $\tau \supseteq \sigma^\frown 0$ and $|\tau| \leqslant recov(\sigma, 2s+1)$. Let $i \in \mathbb{Z}$ be such that $|i| \leqslant recov(\sigma, 2s+1)$. For each $m < c(\sigma, 2s+1)$, there is a component of $\mathcal{G}^i_n[2s+1]$ isomorphic to $[10\langle\ulcorner\tau\urcorner, c(\tau, 2s+1), m\rangle + 1]$. For each $l \in \{2, 4, 5\}$, there is a component of $\mathcal{G}^i_n[2s{+}1]$ isomorphic to $[10\langle\ulcorner\tau\urcorner, c(\tau, 2s{+}1)\rangle + l]$.

7. Let τ be such that $\mathcal{R}_{|\tau|}$ has not yet been satisfied (defined below), $\tau \supseteq \sigma^\frown 0$, and $|\tau| \leqslant recov(\sigma, 2s+1)$. Let $i \in \mathbb{Z}$ be such that $|i| \leqslant recov(\sigma, 2s+1)$. There is a component of $\mathcal{G}_n[s]$ isomorphic to $[10\langle\ulcorner\tau\urcorner, init(\tau, 2s{+}1)\rangle]$. For each $l \in \{6, 7\}$, there is a component of $\mathcal{G}^i_n[s]$ isomorphic to $[10\langle\ulcorner\tau\urcorner, init(\tau, 2s+1)\rangle + l]$. For each $m < \pi_1(\ulcorner\tau\urcorner)$, there is a component of $\mathcal{G}^i_n[s]$ isomorphic to $[10\langle\ulcorner\tau\urcorner, init(\tau, 2s{+}1), m\rangle + 8]$.

If $2s+2$ is a σ-recovery stage then let $\sigma[2s+2](n) = 0$. Otherwise, let $\sigma[2s+2](n) = 1$.

For each σ such that $2s+2$ is a σ-recovery stage, proceed as follows. Let $i = r_{\sigma,2s+1}$. Let $S_{\sigma,2s+1}$ be the component of $\mathcal{A}^i_{2s+1}$ that is isomorphic to the σ-special component of $\mathcal{G}_{|\sigma|}[2s{+}1]$. Let $2t{+}1$ be the last σ-recovery stage before stage $2s{+}2$. If $S_{\sigma,2s+1}$ extends $S_{\sigma,2t}$ then let $r_{\sigma,2s+2} = i$ and $T_{\sigma,2s+2} = T_{\sigma,2s+1}$; otherwise, declare $2s{+}2$ to be a *σ-change stage*, let n be the number of σ-change stages before stage $2s+2$, let $r_{\sigma,2s+2} = \xi(n)$, and let $T_{\sigma,2s+2} = S_{\sigma,2s+1}$.

For each $\sigma \in 2^{<\omega}$ such that $2s+2$ is not a σ-recovery stage, let $r_{\sigma,2s+2} = r_{\sigma,2s+1}$ and $T_{\sigma,2s+2} = T_{\sigma,2s+1}$.

Declare each σ to the right of $\sigma[2s+2]$ to have been *initialized.* This includes declaring σ to be neither semi-recovered nor fully recovered.

Say that R_σ, $\sigma \subseteq \sigma[2s+2]$, *requires attention* if $\mathcal{R}_{|\sigma|}$ has not yet been satisfied, $\pi_1(|\sigma|) \leqslant c(\tau, 2s+1)$ for all τ such that $\tau^\frown 0 \subseteq \sigma$, and, for the coding locations x and y of the (unique) copies of $[10\langle\ulcorner\sigma\urcorner, init(\sigma, 2s+1)\rangle + 7]$ in $\mathcal{A}^0_{2s+1}$ and $\mathcal{A}^{\pi_1(|\sigma|)}_{2s+1}$, respectively, $\Phi_{\pi_0(|\sigma|)}(x)[s]\downarrow = y$.

Let e be the least number less than s such that $R_{\sigma[2s+2](e)}$ requires attention. (If no such e exists then end the stage.) If $\sigma^\frown 0 \subseteq \sigma[2s+2]$ then say that σ is *active* at stage $2s+2$.

Let X_{2s+1} be the component of $\mathcal{A}_{2s+1}$ isomorphic to $[10\langle\ulcorner\sigma[2s+2](e)\urcorner, init(\sigma[2s+2](e), 2s+1)\rangle]$. For each $i \in \mathbb{Z}$ and $m < \pi_1(e)$, let D^i_{2s+1}, E^i_{2s+1}, and $F^i_{m,2s+1}$ be the components of $\mathcal{A}^i_{2s+1}$ isomorphic to $[10\langle\ulcorner\sigma[2s+2](e)\urcorner, init(\sigma[2s+2](e), 2s+1)\rangle + 6]$,

$[10\langle\ulcorner\sigma[2s{+}2](e)\urcorner, init(\sigma[2s{+}2](e), 2s{+}1)\rangle{+}7]$, and $[10\langle\ulcorner\sigma[2s{+}2](e)\urcorner, init(\sigma[2s{+}2](e), 2s{+}1), m\rangle + 8]$, respectively.

Let $\sigma_0, \ldots, \sigma_{k-1}$ be all the strings that are active at stage $2s+2$. For each $j < k$, $i \in \mathbb{Z}$, and $m < \pi_1(e)$, let $Y^i_{\sigma_j,m,2s+1}$, $Z^i_{\sigma_j,2s+1}$, $B^i_{\sigma_j,2s+1}$, and $C^i_{\sigma_j,2s+1}$ be the level-i components of $\mathcal{A}_{2s+1}$ isomorphic to $[10\langle\ulcorner\sigma_j\urcorner, c(\sigma_j, 2s+1), m\rangle + 1]$, $[10\langle\ulcorner\sigma_j\urcorner, c(\sigma_j, 2s+1)\rangle + 2]$, $[10\langle\ulcorner\sigma_j\urcorner, c(\sigma_j, 2s+1)\rangle + 4]$, and $[10\langle\ulcorner\sigma_j\urcorner, c(\sigma_j, 2s+1)\rangle + 5]$, respectively. For each $j < k$ and $i \in \mathbb{Z}$, let $S^i_{\sigma_j,2s+1}$ be the level-i component of $\mathcal{A}_{2s+1}$ isomorphic to $S_{\sigma_j,2s+1}$.

For $j < k$, let $d_j = r_{\sigma_j,2s+2}$. Perform

$$\mathbf{O}_{\pi_1(e),(d_0,\ldots,d_{k-1})}\big(\{Y^i_{\sigma_j,m,2s+1}\}, X_{2s+1}, \{Z^i_{\sigma_j,2s+1}\}, \{B^i_{\sigma_j,2s+1}\}, \{S^i_{\sigma_j,2s+1}\}, \\ \{C^i_{\sigma_j,2s+1}\}, \{D^i_{2s+1}\}, \{E^i_{2s+1}\}, \{F^i_{m,2s+1}\}\big)$$

on $\mathcal{A}_{2s+1}$ to get $\mathcal{A}_{2s+2}$. Declare $\mathcal{R}_e$ to be satisfied.

This completes the construction. Let $\mathcal{A} = \bigcup_{s\in\omega} \mathcal{A}_s$. Since the collection of sets $(|\mathcal{A}_{s+1}| - |\mathcal{A}_s|)_{s\in\omega}$ is uniformly computable, $\mathcal{A}$ is computable. We now wish to argue that properties (3.1)–(3.3) are satisfied. Theorem 1.4 will then follow immediately.

3.2 Verification

Define the *true path* TP of the construction to be the leftmost path of 2^ω such that there are infinitely many stages s with $\sigma[s] \in TP$. For each $i \in \mathbb{Z}$, let a_i be the i-master node of $\mathcal{A}$.

We begin by showing that property (3.2) is satisfied.

3.8 Lemma. *If $\sigma \in TP$ then R_σ requires attention only finitely often.*

Proof. Assume by induction that there is a stage s such that, for all $\tau \subsetneq \sigma$, R_τ does not require attention after stage s. Let $2t+2 > s$ be such that R_σ requires attention at stage $2t+2$. Then, by definition, $\sigma \subseteq \sigma[2t+2]$, and hence $\mathcal{R}_{|\sigma|}$ is satisfied at stage $2t+2$, which implies that R_σ will never again require attention. □

3.9 Lemma. *Let $e = \langle j, i\rangle$. If $\mathcal{R}_e$ is ever satisfied then Φ_j is not an isomorphism from $(\mathcal{A}, a_0)$ to $(\mathcal{A}, a_i)$.*

Proof. Suppose that $\mathcal{R}_e$ is satisfied at stage $2s+2$. Let $\sigma = \sigma[2s+2](e)$. Let K and L be the components of $\mathcal{A}^0_{2s+1}$ and $\mathcal{A}^i_{2s+1}$, respectively, that are isomorphic to

$[10\langle\ulcorner\sigma\urcorner, init(\sigma, 2s+1)\rangle + 7]$, and let x and y be the coding locations of K and L, respectively. Since R_σ requires attention at stage $2s+2$, $\Phi_j(x)\downarrow = y$.

Let K' and L' be the components of $\mathcal{A}^0$ and $\mathcal{A}^i$ that extend K and L, respectively. The operation performed at stage $2s+2$ guarantees that $K' \ncong L'$, so no isomorphism from $(\mathcal{A}, a_0)$ to $(\mathcal{A}, a_i)$ can take x to y. $\square$

3.10 Lemma. *For every $i, j \in \mathbb{Z}$ such that $i \neq j$, $(\mathcal{A}, a_i)$ is not computably isomorphic to $(\mathcal{A}, a_j)$.*

Proof. Since, given $k, l \in \mathbb{Z}$ such that $k \neq l$, any automorphism of $\mathcal{A}$ taking a_k to a_l takes a_0 to a_i for some $i \neq 0$, it is enough to show that, for each $j \in \omega$ and $i \in \mathbb{Z}$, $i \neq 0$, Φ_j is not an isomorphism from $(\mathcal{A}, a_0)$ to $(\mathcal{A}, a_i)$.

Fix $j \in \omega$ and let $e = \langle j, i\rangle$. Let $\sigma = TP(e)$, let s be a stage after which no R_τ, $\tau \subsetneq \sigma$, requires attention and such that σ is not initialized after stage s, and let k be the total number of times σ is initialized. If $\mathcal{R}_e$ is ever satisfied then, by Lemma 3.9, we are done. So suppose that $\mathcal{R}_e$ is never satisfied.

This means that the components K and L of $\mathcal{A}^0_0$ and $\mathcal{A}^i_0$, respectively, isomorphic to $[10\langle\ulcorner\sigma\urcorner, k\rangle + 7]$ never participate in operations. Let x and y be the coding locations of K and L, respectively. Since $\mathcal{R}_e$ is never satisfied, R_σ never requires attention after stage s. So it cannot be the case that $\Phi_j(x)\downarrow = y$. But K and L are the unique copies of $[10\langle\ulcorner\sigma\urcorner, k\rangle + 7]$ in $\mathcal{A}^0$ and $\mathcal{A}^i$, respectively, so any isomorphism from $(\mathcal{A}, a_0)$ to $(\mathcal{A}, a_i)$ must take x to y. $\square$

In showing that (3.1) and (3.3) are satisfied, we will need a few easily checked facts about the construction. We say that a component of $\mathcal{A}$ participates in an operation at stage $s+1$ if it extends a component of $\mathcal{A}_s$ that participates in an operation at stage $s+1$.

Let $\mathcal{G}$ and $\mathcal{H}$ be leveled graphs, let K and L be components of $\mathcal{G}$ and $\mathcal{H}$, respectively, and let $i \in \mathbb{Z}$. We say that K is i-isomorphic to L if there is an isomorphism $f : K \cong L$ such that, for all $x \in K$ and $j \in \mathbb{Z}$, if there is an edge from the j-master node of $\mathcal{G}$ to x then there is an edge from the $(j+i)$-master node of $\mathcal{H}$ to $f(x)$.

3.11 Lemma. *Let K and L be distinct components of $\mathcal{A}_s$. If K is not a copy of $[10k+l]$ for any $k \in \omega$ and $l \in \{1, 2, 6, 8\}$ then K and L are not extended by the same component of $\mathcal{A}$. If K and L are extended by the same component M of $\mathcal{A}$ then M is a component of $\mathcal{A}^*$.*

3.12 Lemma. *Every component of $\mathcal{A}$ has a level unless it contains a copy of $[10k]$ for some $k \in \omega$.*

3.13 Lemma. *For each $s \in \omega$ and $i, j \in \mathbb{Z}$, $(\mathcal{A}_s, a_i) \cong (\mathcal{A}_s, a_j)$ and no component K of $\mathcal{A}_s$ is embeddable in another component L of $\mathcal{A}_s$ unless K is k-isomorphic to L for some $k \in \mathbb{Z}$, $k \neq 0$. Furthermore, if a component of $\mathcal{A}_s^i$ participates in an operation at stage $s+1$ then so does the (unique) isomorphic component of $\mathcal{A}_s^j$.*

3.14 Lemma. *A component of $\mathcal{A}$ that does not contain a copy of $[10k]$ for any $k \in \omega$ is infinite if and only if it participates in operations infinitely often.*

3.15 Lemma. *Let $k, j \in \omega$ and $\sigma \in 2^{<\omega}$. Any component of $\mathcal{A}$ containing a copy of $[10k]$, $[10\langle \ulcorner\sigma\urcorner, j, k\rangle + l]$, $l \in \{1, 8\}$, or $[10\langle \ulcorner\sigma\urcorner, j\rangle + l]$, $l \in \{2, 6, 7\}$, can participate in an operation at most once. Any component of $\mathcal{A}^i$ containing a copy of $[10\langle \ulcorner\sigma\urcorner, j\rangle + l]$, $l \in \{3, 4, 5\}$, can participate in operations only at stages at which σ is active.*

Note that, since $\mathcal{A}_0$ contains only one copy of each $[10k]$, $k \in \omega$, Lemma 3.15 implies that, for each $k \in \omega$, there is at most one stage at which a copy of $[10k]$ participates in an operation.

3.16 Lemma. *Let K be an infinite component of $\mathcal{A}$ that contains a copy of $[10k]$ for some $K \in \omega$ and let $m \in \omega$. Then $K \cap (\mathcal{A}^*)^m$ is not embeddable in any component $L \neq K$ of $\mathcal{A}$ unless K and L are i-isomorphic for some $i \in \mathbb{Z}$, $i \neq 0$.*

3.17 Lemma. *If σ is initialized at stage $s+1$ then no components of $(\mathcal{A})_\sigma$ that participate in operations at stages before stage $s+1$ can participate in an operation after stage s.*

3.18 Lemma. *Suppose that $r_{\sigma,s} = i \neq r_{\sigma,s+1}$. Of all the components of $(\mathcal{A}^i)_\sigma$ that participate in operations at stages before stage $s+1$, the only one that can participate in an operation after stage s is the one that extends $S_{\sigma,s}$.*

3.19 Lemma. *Let u be a stage after which σ is never initialized. Let $s+1 > u$ be a σ-recovery stage that is not the first such stage after u. Let $t+1$ be the last σ-recovery stage before stage $s+1$. If $r_{\sigma,s} = i \neq r_{\sigma,s+1}$ then $S_{\sigma,s}$ extends either $B^i_{\sigma,2u+1}$ or $C^i_{\sigma,2u+1}$ for some $2u+1 \in [t, s)$.*

3.20 Lemma. *Let T be a subgraph of $\mathcal{A}$. If components K and L of $\mathcal{A}$, each containing a copy of T, are involved in T-catch-up operations infinitely often then K and L are infinite and $K \cong L$.*

3.21 Lemma. *Let $\sigma \in TP$ and suppose that $\lim_s r_{\sigma,s}$ exists. Then $T_{\sigma,s}$ comes to a limit T and every infinite component of $(\mathcal{A}_s)_\sigma$ that does not contain a copy $[10k]$ for any $k \in \omega$ contains a copy of T.*

We now turn to showing that property (3.1) is satisfied. This requires us to characterize the infinite components of $\mathcal{A}$.

3.22 Lemma. *Let $\sigma \in 2^{<\omega}$. If $\sigma^\frown 0$ is to the left of TP then no component of $(\mathcal{A})_\sigma$ is infinite unless it contains a copy of $[10k]$ for some $k \in \omega$.*

Proof. If $\sigma^\frown 0$ is to the left of TP then σ is active only finitely often, so the lemma follows from Lemmas 3.14 and 3.15. $\square$

3.23 Lemma. *Let $\sigma \in 2^{<\omega}$. If σ is to the right of TP then no component of $(\mathcal{A})_\sigma$ is infinite unless it contains a copy of $[10k]$ for some $k \in \omega$.*

Proof. This follows immediately from Lemmas 3.14 and 3.17. $\square$

3.24 Lemma. *Let $\sigma \in 2^{<\omega}$ be such that $\sigma^\frown 0 \in TP$. If $r_{\sigma,s}$ does not have a limit then no component of $(\mathcal{A})_\sigma$ is infinite unless it contains a copy of $[10k]$ for some $k \in \omega$.*

Proof. By Lemma 3.12, it is enough to show that, for each $i \in \mathbb{Z}$, no component of $(\mathcal{A}^i)_\sigma$ is infinite.

Let $i \in \mathbb{Z}$. If s is the $(n+1)^{\text{st}}$ σ-change stage then $r_{\sigma,s} = \xi(n)$. Thus there are infinitely many stages s such that $r_{\sigma,s} = i$.

Suppose that $r_{\sigma,s} = i \neq r_{\sigma,s+1}$ and let $t+1$ be the last σ-recovery stage before stage $s+1$. By Lemma 3.18, of all the components of $(\mathcal{A}^i)_\sigma$ that have participated in operations at stages before stage $s+1$, the only one that can participate in an operation after stage s is the component L that extends $S_{\sigma,s}$. By Lemma 3.19, L extends either $B^i_{\sigma,2u+1}$ or $C^i_{\sigma,2u+1}$ for some $2u+1 \in [t,s)$. But, for all $2u+1 \in [t,s)$, $B^i_{\sigma,2u+1}$ and $C^i_{\sigma,2u+1}$ are singleton components, and hence did not participate in an operation at any stage before stage $t+1$.

Thus, no component of $(\mathcal{A}^i)_\sigma$ that participates in an operation before stage $t+1$ can do so again after stage s. The lemma now follows from Lemma 3.14. $\square$

3.25 Lemma. *Let $k \in \omega$. There are finitely many components $K_0, \dots, K_n$ of $\mathcal{A}$ that contain a copy of $[10k]$, and these can be chosen so that K_j is i-isomorphic to K_k for all $j, k \leqslant n$ and $i \in \mathbb{Z}$ such that $i \equiv k - j \bmod n+1$.*

Proof. Let K be the copy of $[10k]$ in $\mathcal{A}_0$. If K never participates in an operation then the lemma is trivially true with $n = 0$ and $K_0 = K$. Otherwise, there is a stage $2s + 2$ such that K participates in an operation at stage $2s + 2$ and, by Lemma 3.15, for any $t \neq 2s+2$, no component of $\mathcal{A}$ that contains a copy of $[10k]$ participates in an operation at stage t. The lemma now follows easily from the definition of the operation performed at stage $2s + 2$. □

3.26 Lemma. *Let σ be such that $\sigma^\frown 0 \in TP$ and $\lim_s r_{\sigma,s}$ exists. There are infinitely many infinite components of $(\mathcal{A})_\sigma$ that do not contain a copy of $[10k]$ for any $k \in \omega$. Let $K_0, K_1, \ldots$ be all such components. Each K_j, $j \in \omega$, has a level. For each $i \in \mathbb{Z}$ there are infinitely many $j \in \omega$ such that $level(K_j) = i$. For all $j, k \in \omega$, $K_j \cong K_k$.*

Proof. Let u be a stage after which σ is never initialized and such that $r_{\sigma,t} = \lim_s r_{\sigma,s}$ for all $t > u$. Let $2s + 1$ be the first phase-1 σ-recovery stage after stage u and let $T = T_{\sigma,2s+1}$. Since the σ-special component of $\mathcal{G}_{|\sigma|}[2s]$ contains a copy of T, for each phase-2 σ-recovery stage $2t + 2$ after stage $2s + 1$, each level of $\mathcal{A}_{2t+2}$ has a component that contains a copy of T and extends a singleton component of $\mathcal{A}_{2t+1}$. Thus there are infinitely many components $K_0, K_1, \ldots$ that contain a copy of T. Furthermore, each K_j, $j \in \omega$, has a level, and for each $i \in \mathbb{Z}$ there are infinitely many $j \in \omega$ such that $level(K_j) = i$.

For each phase-1 σ-recovery stage $2t+1 > 2s+1$, we have $T_{\sigma,2t+1} = T$, so $K_0, K_1, \ldots$ are involved in T-catch-up operations infinitely often. Thus, by Lemma 3.20, $K_0, K_1, \ldots$ are infinite and $K_j \cong K_k$ for all $j, k \in \omega$.

We are left with showing that any component of $(\mathcal{A})_\sigma$ that does not contain a copy of T or a copy of $[10k]$ for any $k \in \omega$ is finite. By Lemma 3.14, it is enough to show that any component of $(\mathcal{A})_\sigma$ that does not contain a copy of T participates in operations only finitely often. But the only components of $(\mathcal{A})_\sigma$ that participate in an operation at an odd stage after stage $2s+1$ are ones that contain a copy of T, while for $2t+2 > 2s+1$, the only non-singleton components of $(\mathcal{A}_{2t+1})_\sigma$ that participate in an operation at stage $2t+2$ are the ones that are isomorphic to the special component of $\mathcal{G}_{|\sigma|}[2t + 1]$, and therefore contain a copy of T. □

3.27 Lemma. *For every $i \in \mathbb{Z}$, $(\mathcal{A}, a_i) \cong (\mathcal{A}, a_0)$.*

Proof. By Lemma 3.13, it is enough to define a 1–1 map f from the set of infinite components of $\mathcal{A}$ onto itself such that, for each infinite component K of $\mathcal{A}$, $f(K)$ is i-isomorphic to K.

Let $k \in \omega$ be such that a copy of $[10k]$ participates in an operation at some point in the construction and let $K_0, \ldots, K_n$ be as in Lemma 3.25. For $j \leqslant n$, let $k \leqslant n$ be such that $k - j \equiv i \bmod n+1$ and define $f(K_j) = K_k$.

Let σ be such that $\sigma^\frown 0 \in TP$ and $\lim_s r_{\sigma,s}$ exists. For each $j \in \mathbb{Z}$, let $K_0^j, K_1^j, \ldots$ be a list of all infinite components of $(\mathcal{A}^j)_\sigma$ that do not contain a copy of $[10k]$ for any $k \in \omega$. By Lemma 3.26, each such list is infinite. For $j \in \mathbb{Z}$ and $n \in \omega$, define $f(K_n^j) = K_n^{j+i}$.

By Lemmas 3.22, 3.23, and 3.24, f is defined on all infinite components of $\mathcal{A}$. By Lemmas 3.25 and 3.26, f is 1–1 and onto, and, for each infinite component K of $\mathcal{A}$, $f(K)$ is i-isomorphic to K. □

We are left with showing that property (3.3) is satisfied. We begin by showing that if $\sigma \in TP$ and $\mathcal{G}_{|\sigma|} \cong \mathcal{A}$ then $\lim_s r_{\sigma,s}$ is well-defined.

3.28 Lemma. *If $\sigma \in TP$ and $\mathcal{G}_{|\sigma|} \cong \mathcal{A}$ then there are infinitely many σ-recovery stages, and hence the σ-special component of $\mathcal{G}_{|\sigma|}$ is infinite.*

Proof. If $\sigma \in TP$ and $\mathcal{G}_{|\sigma|} \cong A$ then $\mathcal{G}_{|\sigma|}$ has a σ-special component. Assume for a contradiction that there are only m many σ-recovery stages. Let s_0 be the last σ-recovery stage. (If there are no σ-recovery stages then let s_0 be the first stage at which $\mathcal{G}_{|\sigma|}$ has a σ-special component.) Since σ is not active at any stage that is not a σ-recovery stage, σ is not active at any stage $t \geqslant s_0$. By the definition of TP, there is a stage $s \geqslant s_0$ by which every τ such that $\tau^\frown 0 \subseteq \sigma$ has fully recovered at least $|\sigma| + 1$ many times and such that σ is not initialized at any stage greater than or equal to s.

There are two cases.

Case 1. σ is not semi-recovered at stage s. Then the first condition in the definition of phase-1 σ-recovery stage is met at every stage greater than or equal to s.

By the choice of s, the second condition in the definition of phase-1 σ-recovery stage is met at every stage greater than or equal to s.

Consider the components of $\mathcal{A}^0$ that contain a copy of the σ-special component of $\mathcal{G}_{|\sigma|}$. By Lemma 3.15, each such component is finite. Thus, if the third condition in the definition of phase-1 σ-recovery stage is not eventually satisfied after stage s then the σ-special component of $\mathcal{G}_{|\sigma|}$ is not isomorphic to any component of $\mathcal{A}^0$.

Now consider $(\mathcal{A}^0)_\sigma$. Again by Lemma 3.15, $(\mathcal{A}^0)_\sigma$ is finite. So, since there are only finitely many $i \in \mathbb{Z}$ such that $|i| \leqslant recov(\sigma, s)$, if the fourth condition in the definition of phase-1 σ-recovery stage is not eventually satisfied after stage s then $(\mathcal{G}^i_{|\sigma|})_\sigma \ncong (\mathcal{A}^0)_\sigma$ for some $i \in \mathbb{Z}$.

Since we are assuming that there are only finitely many σ-recovery stages, $\sigma^\frown 1 \in TP$. Thus it follows from Lemmas 3.14 and 3.15 that $(\mathcal{A}^0_s)_{\supseteq\sigma^\frown 0}$ is finite. So, since there are only finitely many $i \in \mathbb{Z}$ such that $|i| \leqslant recov(\sigma, s)$, if the fifth condition in the definition of phase-1 σ-recovery stage is not eventually satisfied after stage s then $(\mathcal{G}^i_{|\sigma|})_{\supseteq\sigma^\frown 0} \ncong (\mathcal{A}^0)_{\supseteq\sigma^\frown 0}$ for some $i \in \mathbb{Z}$.

Since we are assuming that $\sigma^\frown 0$ is to the left of TP, $(\mathcal{A}^*_s)^{recov(\sigma,s)}_{\supseteq\sigma^\frown 0}$ is finite. So if the last condition in the definition of phase-1 σ-recovery stage is not eventually satisfied after stage s then $(\mathcal{G}^*_{|\sigma|})_{\supseteq\sigma^\frown 0} \ncong (\mathcal{A}^*)_{\supseteq\sigma^\frown 0}$.

Case 2. σ is semi-recovered at stage s. Then the first condition in the definition of phase-2 σ-recovery stage is met at every stage greater than or equal to s.

By the same arguments as above we have the following facts. If the second condition in the definition of phase-2 σ-recovery stage is not eventually satisfied after stage s then the σ-special component of $\mathcal{G}_{|\sigma|}$ is not isomorphic to any component of $\mathcal{A}^0$. If the third condition in the definition of phase-2 σ-recovery stage is not eventually satisfied after stage s then $(\mathcal{G}^i_{|\sigma|})_\sigma \ncong (\mathcal{A}^0)_\sigma$ for some $i \in \mathbb{Z}$. If the fourth condition in the definition of phase-2 σ-recovery stage is not eventually satisfied after stage s then $(\mathcal{G}^i_{|\sigma|})_{\supseteq\sigma^\frown 0} \ncong (\mathcal{A}^0)_{\supseteq\sigma^\frown 0}$ for some $i \in \mathbb{Z}$.

Since we are assuming that $\sigma^\frown 0$ is to the left of TP, there is a stage $t \geqslant s$ after which no τ such that $\tau \supseteq \sigma^\frown 0$ is initialized. Any such τ that has not fully recovered since the last time it was initialized never again recovers, and hence there is a component of $\mathcal{A}^0$ isomorphic to $[10\langle\ulcorner\tau\urcorner, init(\tau, t)\rangle + 3]$. Since there are only finitely many τ and $i \in \mathbb{Z}$ such that $|\tau|, |i| \leqslant recov(\sigma, s)$, if the fifth condition in the definition of phase-2 σ-recovery stage is not eventually satisfied after stage s then $\mathcal{G}^i_{|\sigma|} \ncong \mathcal{A}^0$ for some $i \in \mathbb{Z}$.

Now let τ be such that either $\tau = \sigma$ or both $\tau \supseteq \sigma^\frown 0$ and $|\tau| \leqslant recov(\sigma, s)$, and let $i \in \mathbb{Z}$ be such that $|i| \leqslant recov(\sigma, s)$. Clearly, $c(\tau, t)$ reaches a limit $c(\tau)$. It is easy to see that, for any stage $2t + 2$ at which τ is active, $c(\tau, 2t + 1) < c(\tau, s) = c(\tau)$. So, for each $l \in \{2, 4, 5\}$, there is a unique component of $\mathcal{A}^0$ that contains a copy of $[10\langle\ulcorner\tau\urcorner, c(\tau)\rangle + l]$, and it is isomorphic to $[10\langle\ulcorner\tau\urcorner, c(\tau)\rangle + l]$. Similarly, for each $m < c(\sigma, s)$, there is a unique component of $\mathcal{A}^0$ that contains a copy of $[10\langle\ulcorner\tau\urcorner, c(\tau), m\rangle + 1]$, and it is isomorphic to $[10\langle\ulcorner\tau\urcorner, c(\tau), m\rangle + 1]$. Thus, if the sixth condition in the definition of phase-2 σ-recovery stage is not eventually satisfied after stage s then, for some $i \in \mathbb{Z}$, there is a component of $\mathcal{A}^0$ that is not isomorphic to any component of $\mathcal{G}^i_{|\sigma|}$.

Finally, let $\tau \supseteq \sigma^\frown 0$ and $i \in \mathbb{Z}$ be such that $|\tau|, |i| \leqslant recov(\sigma, s)$ and consider $\mathcal{R}_{|\tau|}$. If this requirement is ever satisfied then the last condition in the definition of phase-2 σ-recovery stage will be satisfied for these particular τ and i at all sufficiently large stages. On the other hand, suppose that $\mathcal{R}_{|\tau|}$ is never satisfied. It is easy to see that $init(\tau, t)$ reaches a limit $init(\tau)$, and there is a component of $\mathcal{A}$ isomorphic to $[10\langle\ulcorner\tau\urcorner, init(\tau)\rangle]$. Similarly, for each $l \in \{6, 7\}$, there is a component of $\mathcal{A}^0$ isomorphic to $[10\langle\ulcorner\tau\urcorner, init(\tau)\rangle + l]$, and, for each $m < \pi_1(\ulcorner\tau\urcorner)$, there is a component of $\mathcal{A}^0$ isomorphic to $[10\langle\ulcorner\tau\urcorner, init(\tau), m\rangle + 8]$. Thus, if the last condition in the definition of phase-2 σ-recovery stage is not eventually satisfied after stage s then, for some $i \in \mathbb{Z}$, there is a component of $\mathcal{A}^0$ that is not isomorphic to any component of $\mathcal{G}^i_{|\sigma|}$.

In either case, $\mathcal{G}_{|\sigma|}$ cannot be isomorphic to $\mathcal{A}$, contrary to hypothesis. So there are infinitely many σ-recovery stages.

Now let v be a stage after which σ is never initialized. Given any two stages $2u+2 > 2t+2 > v$ at which σ is active, the σ-special component of $\mathcal{G}_{|\sigma|}[2u+1]$ properly extends the σ-special component of $\mathcal{G}_{|\sigma|}[2t+1]$. Thus, to establish the second part of the lemma, it is enough to show that σ is active infinitely often. But it is easy to find infinitely many $\tau \supset \sigma^\frown 0$ such that R_τ eventually requires attention. Each time such an R_τ requires attention, σ is active. $\square$

3.29 Lemma. *If $\sigma \in TP$ and $\mathcal{G}_{|\sigma|} \cong \mathcal{A}$ then $\lim_s r_{\sigma,s}$ is well-defined.*

Proof. This follows immediately from Lemmas 3.24 and 3.28. $\square$

Now fix $\sigma \in TP$ such that $\mathcal{G}_{|\sigma|} \cong \mathcal{A}$. Let $n = |\sigma|$ and let g be the 0-master node of $\mathcal{G}_n$. By Lemma 3.29, $r = \lim_s r_{\sigma,s}$ is well-defined. We wish to show that $(\mathcal{G}_n, g)$ is computably isomorphic to $(\mathcal{A}, a_r)$. Let b be the unique map from the backbone graph $\mathcal{B}$ of $\mathcal{A}$ to the backbone graph B of $\mathcal{G}_n$ that takes a_r to g. Note that b is computable.

We will define a computable isomorphism f from $\mathcal{A}$ to $\mathcal{G}_n$ extending b. Our method will be to divide $|\mathcal{A}|$ into a finite collection of (not necessarily disjoint) c.e. sets and define f independently on each of these sets. We will need to be somewhat careful here, because $\mathcal{A}$ is not rigid (i.e., $\mathcal{A}$ has nontrivial automorphisms), but Lemma 3.32 below justifies our approach.

Let $\omega^- = \{k \in \omega \mid k \not\equiv 9 \bmod 10\}$. Let $N_0 = \{(k, i) \mid k \in \omega^-,\ k \not\equiv 0 \bmod 10,\ i \in \mathbb{Z}\}$ and $N_1 = \{k \in \omega \mid k \equiv 0 \bmod 10\}$. Let $N = N_0 \cup N_1$. For $p = (k, i) \in N_0$, let $\pi(p) = k$; for $k \in N_1$, let $\pi(k) = k$.

3.30 Definition. Let $p = (k, i) \in N_0$ and $s \in \omega$. We denote by (p) and $(p)_s$ the components of $\mathcal{A}$ and $\mathcal{A}_s$, respectively, that extend the unique copy of $[k]$ in $\mathcal{A}^i_0$.

Let $k \in N_1$ and $s \in \omega$. We denote by (k) and $(k)_s$ the components of $\mathcal{A}$ and $\mathcal{A}_s$, respectively, that extend the unique copy of $[k]$ in $\mathcal{A}_0$.

For $H \subseteq \omega^-$, let $\widetilde{H} = \{(k,i) \mid k \in H,\ k \not\equiv 0 \bmod 10,\ i \in \mathbb{Z}\} \cup \{k \in H \mid k \equiv 0 \bmod 10\}$. For $S \subseteq N$, let P_S be the graph obtained by restricting the domain of $\mathcal{A}$ to $|\mathcal{B}| \cup \bigcup_{p \in S} |(p)|$.

3.31 Lemma. *Let S and S' be disjoint subsets of N and let f and f' be embeddings of P_S and $P_{S'}$, respectively, into $\mathcal{G}_n$. Then f and f' agree on $P_S \cap P_{S'}$.*

Proof. By Lemma 3.11, if $p \neq q \in N$ are such that $(p) = (q)$ then (p) is a component of $\mathcal{A}^*$. This clearly implies that $f^{-1} \circ (f' \upharpoonright P_S \cap P_{S'})$ can be extended to an automorphism of $\mathcal{A}^*$. But it is also easy to check that $\mathcal{A}^*$ is rigid. Thus f and f' must agree on $P_S \cap P_{S'}$. □

3.32 Lemma. *Let $S_0, \ldots, S_m$ be pairwise disjoint computable subsets of N such that $\bigcup_{i=0}^m S_i = N$. Suppose that, for each $i \leqslant m$, there exists a computable embedding $f_i \supset b$ from P_{S_i} into $\mathcal{G}_n$ such that $\bigcup_{i=0}^m \operatorname{rng}(f_i) = |\mathcal{G}_n|$. Suppose further that, for $i, j \leqslant m$ such that $i \neq j$, if K and L are components of P_{S_i} and P_{S_j}, respectively, and $f_i(K) = f_j(L)$, then $K = L$. Then there exists a computable isomorphism $f \supset b$ from $\mathcal{A}$ to $\mathcal{G}_n$.*

Proof. Since $S_0, \ldots, S_m$ are computable, $P_{S_0}, \ldots, P_{S_m}$ are c.e.. Since $\bigcup_{i=0}^m S_i = N$, $\bigcup_{i=0}^m P_{S_i} = \mathcal{A}$. Define f as follows. Given $x \in \mathcal{A}$, wait until x is enumerated into some P_{S_i}, $i \leqslant m$, and then let $f(x) = f_i(x)$. It is easy to check that the conditions imposed on $S_0, \ldots, S_m$ and $f_0, \ldots, f_m$, together with Lemma 3.31, imply that f is an isomorphism from $\mathcal{A}$ to $\mathcal{G}_n$. □

We will partition ω^- into the pairwise disjoint computable sets $H_0, \ldots, H_5$ shown in Table 1. This will induce a partition of N into $\widetilde{H}_0, \ldots, \widetilde{H}_5$. We will further partition $\widetilde{H}_3$ into $\widetilde{H}_3^0 = \{p \in \widetilde{H}_3 \mid (p)$ is infinite and does not contain a copy of $[10k]$ for any $k \in \omega\}$ and $\widetilde{H}_3^1 = \widetilde{H}_3 - \widetilde{H}_3^0$.

The following two lemmas give us a useful tool for computing f.

3.33 Lemma. *Let $p \in N$ and suppose there is a stage s such that, for each $t \geqslant s$, $(p)_t$ does not participate in an operation at stage $t+1$. Then $(p) \cong (p)_s$.*

Proof. Clearly, if $(p)_t$ does not participate in an operation at stage $t+1$ then $(p)_{t+1} \cong (p)_t$. So, by induction, $(p)_t \cong (p)_{s+1}$ for all $t \geqslant s$. Since $(p) = \bigcup_{t \in \omega} (p)_t$, the lemma follows. □

Table 1: $H_0, \ldots, H_5$

H_0	$\{10\langle\ulcorner\tau\urcorner, k, j\rangle + 1,\ 10\langle\ulcorner\tau\urcorner, k\rangle + l \mid \tau$ to the left of σ or $\tau^\frown 1 \subseteq \sigma;$ $j, k \in \omega;\ l \in \{2, 3, 4, 5\}\}$
H_1	$\{10\langle\ulcorner\tau\urcorner, k, j\rangle + 8,\ 10\langle\ulcorner\tau\urcorner, k\rangle + l \mid \tau$ to the left of $\sigma;\ j, k \in \omega;\ l \in \{0, 6, 7\}\}$
H_2	$\{10\langle\ulcorner\tau\urcorner, k, j\rangle + d,\ 10\langle\ulcorner\tau\urcorner, k\rangle + l \mid \tau$ to the right of $\sigma^\frown 0;\ j, k \in \omega;$ $d \in \{1, 8\};\ l \in \{0, 2, 3, 4, 5, 6, 7\}\}$
H_3	$\{10\langle\ulcorner\tau\urcorner, k, j\rangle + 1,\ 10\langle\ulcorner\tau\urcorner, k\rangle + l \mid \tau^\frown 0 \subseteq \sigma;\ j, k \in \omega;\ l \in \{2, 3, 4, 5\}\}$
H_4	$\{10\langle\ulcorner\tau\urcorner, k, j\rangle + 8,\ 10\langle\ulcorner\tau\urcorner, k\rangle + l \mid \tau \subseteq \sigma;\ j, k \in \omega;\ l \in \{0, 6, 7\}\}$
H_5	$\{10\langle\ulcorner\tau\urcorner, k, j\rangle + 1,\ 10\langle\ulcorner\tau\urcorner, k\rangle + l \mid \tau = \sigma$ or $\sigma^\frown 0 \subseteq \tau;\ j, k \in \omega;\ l \in \{2, 3, 4, 5\}\} \cup$ $\{10\langle\ulcorner\tau\urcorner, k, j\rangle + 8,\ 10\langle\ulcorner\tau\urcorner, k\rangle + l \mid \sigma^\frown 0 \subseteq \tau;\ j, k \in \omega;\ l \in \{0, 6, 7\}\}$

3.34 Lemma. *Let $H \subseteq \omega^-$, $h : H \to \omega$, and $S \subseteq \widetilde{H}$ all be computable. Suppose that, for each $p \in S$ and $t \geqslant h(\pi(p))$, $(p)_t$ does not participate in an operation at stage $t + 1$. Then there is a unique embedding $f \supset b$ of P_S into $\mathcal{G}_n$, and f is computable.*

Proof. Let $x \in P_S$ and let $p \in S$ be such that $x \in (p)$. By Lemma 3.33, $(p)_{h(\pi(p))} \cong (p)$, so either (p) is finite or it contains a copy of $[10k]$ for some $k \in \omega$.

First suppose that (p) is finite. By Lemma 3.13, no finite component K of $\mathcal{A}$ is embeddable in another component L of $\mathcal{A}$ unless K and L are i-isomorphic for some $i \in \mathbb{Z}$, $i \neq 0$, so there is a unique finite set $T \subset \mathcal{G}_n$ for which there is an isomorphism $g_x : \mathcal{A} \upharpoonright (|(p)| \cup |\mathcal{B}|) \cong \mathcal{G}_n \upharpoonright (|T| \cup |B|)$ extending b, and this isomorphism is unique.

Now suppose that (p) contains a copy of $[10k]$ for some $k \in \omega$. If (p) does not participate in an operation before stage $h(\pi(p)) + 1$ then (p) is finite, so the previous case applies. Otherwise, let m be such that $x \in (p) \cap (\mathcal{A}^*)^m$. By Lemma 3.16, $(p) \cap (\mathcal{A}^*)^m$ is not embeddable in any component $L \neq K$ of $\mathcal{A}$ unless (p) and L are i-isomorphic for some $i \in \mathbb{Z}$, $i \neq 0$, so there is a unique finite set $T \subset \mathcal{G}_n$ for which there is an isomorphism $g_x : \mathcal{A} \upharpoonright (|(p) \cap (\mathcal{A}^*)^m| \cup |\mathcal{B}|) \cong \mathcal{G}_n \upharpoonright (|T| \cup |B|)$ extending b, and this isomorphism is unique.

In either case, define $f(x) = g_x(x)$. By the uniqueness of T and g_x, f is the unique embedding of P_S into $\mathcal{G}_n$ extending b. Furthermore, it is easy to see that g_x can be computably determined given $x \in P_S$, which implies that f is computable. $\square$

3.35 Lemma. *Let H_0 consist of all numbers of the form $10\langle\ulcorner\tau\urcorner, k, j\rangle + 1$ or $10\langle\ulcorner\tau\urcorner, k\rangle + l$,* *with either τ to the left of σ or $\tau^\frown 1 \subseteq \sigma$; $j, k \in \omega$; and $l \in \{2, 3, 4, 5\}$. Then there is a unique embedding $f_0 \supset b$ of $P_{\widetilde{H}_0}$ into $\mathcal{G}_n$, and f_0 is computable.*

Proof. Let T be the set of all τ which are either to the left of σ or such that $\tau^\frown 1 \subseteq \sigma$. Since $\sigma \in TP$, only finitely many elements of T ever recover, and the ones that do recover, do so only finitely often. So there exists a stage s such that if $\tau \in T$ then τ is not active after stage s. If we let $h(m) = s$ for all $m \in H_0$ then the hypotheses of Lemma 3.34 are satisfied for $S = \widetilde{H}_0$. $\square$

3.36 Lemma. *Let H_1 consist of all numbers of the form $10\langle\ulcorner\tau\urcorner, k, j\rangle + 8$ or $10\langle\ulcorner\tau\urcorner, k\rangle + l$, with τ to the left of σ; $j, k \in \omega$; and $l \in \{0, 6, 7\}$. Then there is a unique embedding $f_1 \supset b$ of $P_{\widetilde{H}_1}$ into $\mathcal{G}_n$, and f_1 is computable.*

Proof. Since $\sigma \in TP$, there exists a stage s such that if τ is to the left of σ then τ is not accessible after stage s. If we let $h(m) = s$ for all $m \in H_1$ then the hypotheses of Lemma 3.34 are satisfied for $S = \widetilde{H}_1$. $\square$

3.37 Lemma. *Let τ be to the right of $\sigma^\frown 0$. Let m be of the form $10\langle\ulcorner\tau\urcorner, k, j\rangle + d$ or $10\langle\ulcorner\tau\urcorner, k\rangle + l$, with $j, k \in \omega$; $d \in \{1, 8\}$; and $l \in \{0, 2, 3, 4, 5, 6, 7\}$. Let $s + 1$ be the stage at which τ is initialized for the $(k+1)^{st}$ time. Let $i \in \mathbb{Z}$. If $m \in N_1$ then let $p = m$; otherwise, let $p = (m, i)$. Then (p) does not participate in an operation after stage s.*

Proof. If a singleton component of $\mathcal{A}_t$ of the form $[10\langle\ulcorner\tau\urcorner, q\rangle + l]$, with $l \in \{0, 3, 6, 7\}$, or $[10\langle\ulcorner\tau\urcorner, q, j\rangle + 8]$ participates in an operation at a stage $t + 1 > s$ then $q = init(\tau, t) > k$. If a singleton component of $\mathcal{A}_t$ of the form $[10\langle\ulcorner\tau\urcorner, q\rangle + l]$, with $l \in \{2, 4, 5\}$, or $[10\langle\ulcorner\tau\urcorner, q, j\rangle + 1]$ participates in an operation at a stage $t + 1 > s$ then $q = c(\tau, t) \geqslant init(\tau, t) > k$. So if (p) does not participate in an operation before stage $s + 1$ then it does not participate in an operation after stage s.

On the other hand, if (p) participates in an operation before stage $s+1$ then the fact that it does not participate in an operation after stage s follows from Lemma 3.17. $\square$

3.38 Lemma. *Let H_2 consist of all numbers of the form $10\langle\ulcorner\tau\urcorner, k, j\rangle + d$ or $10\langle\ulcorner\tau\urcorner, k\rangle + l$, with τ to the right of $\sigma^\frown 0$; $j, k \in \omega$; $d \in \{1, 8\}$; and $l \in \{0, 2, 3, 4, 5, 6, 7\}$. Then there is a unique embedding $f_2 \supset b$ of $P_{\widetilde{H}_2}$ into $\mathcal{G}_n$, and f_2 is computable.*

Proof. If $m \in H_2$ is of the form $10\langle\ulcorner\tau\urcorner, k, j\rangle + d$ or $10\langle\ulcorner\tau\urcorner, k\rangle + l$ then define $h(m)$ to be the first stage by which τ has been initialized $k + 1$ many times (which exists, since $\sigma^\frown 0 \in TP$). Then, by Lemma 3.37, the hypotheses of Lemma 3.34 are satisfied for $S = \widetilde{H}_2$. $\square$

Let H_3 be the set of all numbers of the form $10\langle\ulcorner\tau\urcorner, k, j\rangle + 1$ or $10\langle\ulcorner\tau\urcorner, k\rangle + l$, with $\tau^\frown 0 \subseteq \sigma$; $j, k \in \omega$; and $l \in \{2, 3, 4, 5\}$. Let $\widetilde{H}_3^0$ be the set of all $p \in \widetilde{H}_3$ such that (p) is infinite and does not contain a copy of $[10k]$ for any $k \in \omega$. Let $\widetilde{H}_3^1 = \widetilde{H}_3 - \widetilde{H}_3^0$.

3.39 Lemma. *$\widetilde{H}_3^0$ is computable.*

Proof. By Lemmas 3.15, 3.22, and 3.24, every element of $\widetilde{H}_3^0$ must be of the form $(10\langle\ulcorner\tau\urcorner, k\rangle + l, i)$, where $\tau^\frown 0 \subseteq \sigma$; $\lim_s r_{\tau,s}$ exists; $k, i \in \omega$; and $l \in \{3, 4, 5\}$. Let $p \in N$ be of this form. We will describe an effective procedure for deciding whether $p \in \widetilde{H}_3^0$.

First suppose that $l = 3$. Since $\lim_s r_{\tau,s}$ exists, $T_{\tau,s}$ comes to a limit T by some stage u. Since $\tau \in TP$, $init(\tau, s)$ comes to a limit $init(\tau)$. If $k \neq init(\tau)$ then $p \notin \widetilde{H}_3^0$. Otherwise, by Lemma 3.21, $p \in \widetilde{H}_3^0$ if and only if $(p)_u$ contains a copy of T.

Now suppose that $l \in \{4, 5\}$. Now let $t \geqslant u$ be a stage by which τ has recovered $k + 1$ many times, which must exist, since $\tau^\frown 0 \in TP$. Arguing as in the proof of Lemma 3.37, we see that if (p) has not participated in an operation by stage t then it will never participate in an operation, in which case $p \notin \widetilde{H}_3^0$. On the other hand, if (p) has participated in an operation by stage t then, by Lemma 3.21, $p \in \widetilde{H}_3^0$ if and only if $(p)_t$ contains a copy of T. $\square$

For $\tau \subset \sigma$, let S_τ be the set of all $p \in \widetilde{H}_3^0$ such that (p) is a component of $(\mathcal{A})_\tau$. The following lemma is easily checked.

3.40 Lemma. *For each $\tau \subset \sigma$, P_{S_τ} is c.e..*

For $\tau \subset \sigma$, let M_τ be the union of all infinite components of $(\mathcal{G}_n)_\tau$ that do not contain a copy of $[10k]$ for any $k \in \omega$. Let $M = \bigcup_{\tau\subset\sigma} M_\tau$.

3.41 Lemma. *For each $\tau \subset \sigma$, M_τ is c.e..*

Proof. By Lemmas 3.15, 3.22, and 3.24, it is enough to show that M_τ is c.e. for each $\tau^\frown 0 \subseteq \sigma$ such that $\lim_s r_{\tau,s}$ exists.

Fix such a τ. Since $\lim_s r_{\tau,s}$ exists, $T_{\tau,s}$ comes to a limit T. By Lemma 3.21, the components of M_τ are exactly those that contain a copy of T. Since T is finite, we can effectively enumerate such components. $\square$

3.42 Lemma. *There exists a computable isomorphism $f_3^0 \supset b$ from $P_{\widetilde{H}_3^0}$ to the graph obtained by restricting the domain of $\mathcal{G}_n$ to $|B| \cup |M|$.*

Proof. Let $\tau \subset \sigma$. By Lemma 3.40, P_{S_τ} is c.e.. By Lemma 3.41, so is M_τ. Thus there exists a computable 1–1 and onto map d_τ from the tops of components of P_{S_τ} to the tops of components of M_τ such that if x is the top of a level-i component of P_{S_τ}, $i \in \mathbb{Z}$, then $d_\tau(x)$ is the top of a level-$(i-r)$ component of M_τ. By Lemma 3.26, d_τ can be extended to a computable isomorphism f_3^τ from P_{S_τ} to M_τ. Now define $f_3^0 = b \cup \bigcup_{\tau\subset\sigma} f_3^\tau$. □

3.43 Lemma. *There is a unique embedding $f_3^1 \supset b$ of $P_{\widetilde{H}_3^1}$ into $\mathcal{G}_n$, and f_3^1 is computable.*

Proof. First let $p = (m, i) \in \widetilde{H}_3^1$, where m is of the form $10\langle\ulcorner\tau\urcorner, k\rangle + 3$. Note that in this case (p) cannot contain a copy of $[10k']$ for any $k' \in \omega$. Let $init(\tau) = \lim_s init(\tau, s)$.

If $k < init(\tau)$ then let s be the least stage by which τ has been initialized $k+1$ many times. Arguing as in the proof of Lemma 3.37, we see that (p) does not participate in an operation after stage s. In this case, let $h(m) = s$.

If $k > init(\tau)$ then (p) never participates in an operation. In this case, let $h(m) = 0$.

If $k = init(\tau)$ then it must be the case that $r_{\tau,s}$ has no limit, since otherwise (p) would be infinite, which would imply that $p \in \widetilde{H}_3^0$. Thus $T_{\tau,s}$ has no limit, which means that we can find a stage s such that $(p)_s$ does not contain a copy of $T_{\tau,s}$. It is not hard to check that (p) does not participate in an operation after stage s. In this case, let $h(m) = s$.

Now let $p = (m, i) \in \widetilde{H}_3^1$, where m is of the form $10\langle\ulcorner\tau\urcorner, k, j\rangle+1$ or $10\langle\ulcorner\tau\urcorner, k\rangle+l$, with $l \in \{2, 4, 5\}$. Let $2s+2$ be the least phase-2 τ-recovery stage such that $c(\tau, 2s+1) > k$, (p) does not participate in an operation at stage $2s+2$, and $(p)_{2s+1}$ does not contain a copy of $T_{\tau,2s+1}$. Such a stage must exist, since otherwise (p) would be infinite but would not contain a copy of $[10k']$ for any $k' \in \omega$, which would imply that $p \in \widetilde{H}_3^0$. It is not hard to check that (p) does not participate in an operation after stage s. In this case, let $h(m) = s$.

Now the hypotheses of Lemma 3.34 are satisfied for $S = \widetilde{H}_3^1$. □

Let $f_3 = f_3^0 \cup f_3^1$.

3.44 Lemma. *Let H_4 consist of all numbers of the form $10\langle\ulcorner\tau\urcorner, k, j\rangle+8$ or $10\langle\ulcorner\tau\urcorner, k\rangle+l$, with $\tau \subseteq \sigma$; $j, k \in \omega$; and $l \in \{0, 6, 7\}$. Then there is a unique embedding $f_4 \supset b$ of $P_{\widetilde{H}_4}$ into $\mathcal{G}_n$, and f_4 is computable.*

Proof. Let $m \in H_4$ be of the form $10\langle\ulcorner\tau\urcorner, k, j\rangle + 8$ or $10\langle\ulcorner\tau\urcorner, k\rangle + l$. Let $i \in \mathbb{Z}$. If $m \in N_1$ then let $p = m$; otherwise, let $p = (m, i)$. If $\mathcal{R}_{|\tau|}$ is never satisfied then (p) never participates in an operation. In this case, let $h(m) = 0$. If $\mathcal{R}_{|\tau|}$ is satisfied at stage s

then (p) never participates in an operation after stage s. In this case, let $h(m) = s$. Since there are only finitely many $\tau \subseteq \sigma$, h is computable, and hence the hypotheses of Lemma 3.34 are satisfied for $S = \widetilde{H}_4$. $\square$

Let H_5' consist of all numbers of the form $10\langle\ulcorner\tau\urcorner, k, j\rangle + 1$ or $10\langle\ulcorner\tau\urcorner, k\rangle + l$, with either $\tau = \sigma$ or $\sigma^\frown 0 \subseteq \tau$; $j, k \in \omega$; and $l \in \{2, 3, 4, 5\}$. Let H_5'' consist of all numbers of the form $10\langle\ulcorner\tau\urcorner, k, j\rangle + 8$ or $10\langle\ulcorner\tau\urcorner, k\rangle + l$, with $\sigma^\frown 0 \subseteq \tau$; $j, k \in \omega$; and $l \in \{0, 6, 7\}$. Let $H_5 = H_5' \cup H_5''$.

3.45 Lemma. *Let τ be such that $\tau = \sigma$ or $\sigma^\frown 0 \subseteq \tau$. Let u be a stage after which σ is never initialized and such that $r_{\sigma,s} = r$ for all $s \geqslant u$. Let $2s + 1 > u$ be a phase-1 τ-recovery stage and let $2t + 2$ be the next phase-2 σ-recovery stage after stage $2s + 1$. Let $i \in \mathbb{Z}$ be such that $|i - r| \leqslant recov(\sigma, 2s + 1)$. Let $K_0, \ldots, K_m$ be the components of $(\mathcal{A}^i_{2s})_\tau$ that participate in an operation at stage $2s+1$. Let $K_0', \ldots, K_m'$ be the components of $\mathcal{A}_{2t+1}$ that extend $K_0, \ldots, K_m$, respectively. Then the following hold.*

1. *There exist components $\widehat{K}_0, \ldots, \widehat{K}_m$ of $\mathcal{G}^{i-r}_n[2s]$ such that $\widehat{K}_0 \cong K_0, \ldots, \widehat{K}_m \cong K_m$.*
2. *Let $\widehat{K}_0', \ldots, \widehat{K}_m'$ be the components of $\mathcal{G}_n[2t + 1]$ that extend $\widehat{K}_0, \ldots, \widehat{K}_m$, respectively. Then $\widehat{K}_0' \cong K_0', \ldots, \widehat{K}_m' \cong K_m'$.*

Proof. Since $2s+1$ is a τ-recovery stage, it is also a σ-recovery stage, so the first part of the lemma follows from the definition of phase-1 σ-recovery stage; we prove the second part.

Let $j \leqslant m$. Since no component of $(\mathcal{A})_\tau$ participates in an operation in the interval $(2s + 1, 2t + 2)$, the definition of the catch-up operation performed at stage $2s + 1$ guarantees that K_j' is the unique component of $\mathcal{A}^i_{2t+1}$ that contains a copy of K_j. This means that $\widehat{K}_j'$ is the unique component of $\mathcal{G}^{i-r}[2t + 1]$ that contains a copy of $\widehat{K}_j$. By the definition of phase-2 σ-recovery stage, there must exist a component of $\mathcal{G}^{i-r}[2t + 1]$ isomorphic to K_j', so it must be the case that $\widehat{K}_j' \cong K_j'$. $\square$

3.46 Lemma. *Let τ be such that $\tau = \sigma$ or $\sigma^\frown 0 \subseteq \tau$. Let u be a stage after which σ is never initialized and such that $r_{\sigma,s} = r$ for all $s \geqslant u$. Let $2s + 2 > u$ be a phase-2 τ-recovery stage and let $2t + 1$ be the next phase-1 σ-recovery stage after stage $2s + 2$. Let $\overline{\mathcal{A}}_{2t}$ be the union of $(\mathcal{A}^*_{2t})^{recov(\sigma,2s+1),r}$ (see page 14) and $\mathcal{A}^i_{2t}$ for each $i \in \mathbb{Z}$ such that $|i - r| \leqslant recov(\sigma, 2s + 1)$. Let $\overline{\mathcal{G}}_n[2t]$ be the union of $(\mathcal{G}^*_n[2t])^{recov(\sigma,2s+1)}$ and $\mathcal{G}^i_n[2t]$ for each $i \in \mathbb{Z}$ such that $|i| \leqslant recov(\sigma, 2s + 1)$.*

Let $i \in \mathbb{Z}$ be such that $|i - r| \leqslant recov(\sigma, 2s+1)$. Let e be such that $\mathcal{R}_e$ is satisfied at stage $2s+2$ and let $m = \pi_1(e)$. Let $Y_0, \dots, Y_{m-1}$, X, Z, B, S, C, D, E, and $F_0, \dots, F_{m-1}$ be $Y^i_{\tau,0,2s+1}, \dots, Y^i_{\tau,m,2s+1}$, X_{2s+1}, $Z^i_{\tau,2s+1}$, $B^i_{\tau,2s+1}$, $S^i_{\tau,2s+1}$, $C^i_{\tau,2s+1}$, D^i_{2s+1}, E^i_{2s+1}, and $F^i_{0,2s+1}, \dots, F^i_{m,2s+1}$, respectively. Let $Y'_0, \dots, Y'_{m-1}$, X', Z', B', S', C', D', E', and $F'_0, \dots, F'_{m-1}$ be the intersection of the components of $\mathcal{A}_{2t}$ that extend $Y_0, \dots, Y_{m-1}$, X, Z, B, S, C, D, E, and $F_0, \dots, F_{m-1}$, respectively, with $\overline{\mathcal{A}}_{2t}$. Then the following hold.

1. *There exists a component $\widehat{X}$ of $\mathcal{G}^{i-r}_n[2s+1]$ such that $\widehat{X} \cong X$. There exist components $\widehat{Y}_0, \dots, \widehat{Y}_{m-1}$, $\widehat{Z}$, $\widehat{B}$, $\widehat{S}$, $\widehat{C}$, $\widehat{D}$, $\widehat{E}$, and $\widehat{F}_0, \dots, \widehat{F}_{m-1}$ of $\mathcal{G}^{i-r}_n[2s+1]$ such that $\widehat{Y}_0 \cong Y_0, \dots, \widehat{Y}_{m-1} \cong Y_{m-1}$, $\widehat{Z} \cong Z$, $\widehat{B} \cong B$, $\widehat{S} \cong S$, $\widehat{C} \cong C$, $\widehat{D} \cong D$, $\widehat{E} \cong E$, and $\widehat{F}_0 \cong F_0, \dots, \widehat{F}_{m-1} \cong F_{m-1}$.*
2. *Let $\widehat{Y}'_0, \dots, \widehat{Y}'_{m-1}$, $\widehat{X}'$, $\widehat{Z}'$, $\widehat{B}'$, $\widehat{S}'$, $\widehat{C}'$, $\widehat{D}'$, $\widehat{E}'$, and $\widehat{F}'_0, \dots, \widehat{F}'_{m-1}$ be the intersection of the components of $\mathcal{G}_n[2t]$ that extend $\widehat{Y}_0, \dots, \widehat{Y}_{m-1}$, $\widehat{X}$, $\widehat{Z}$, $\widehat{B}$, $\widehat{S}$, $\widehat{C}$, $\widehat{D}$, $\widehat{E}$, and $\widehat{F}_0, \dots, \widehat{F}_{m-1}$, respectively, with $\overline{\mathcal{G}}_n[2t]$. Then $\widehat{Y}'_0 \cong Y'_0, \dots, \widehat{Y}'_{m-1} \cong Y'_{m-1}$, $\widehat{X}' \cong X'$, $\widehat{Z}' \cong Z'$, $\widehat{B}' \cong B'$, $\widehat{S}' \cong S'$, $\widehat{C}' \cong C'$, $\widehat{D}' \cong D'$, $\widehat{E}' \cong E'$, and $\widehat{F}'_0 \cong F'_0, \dots, \widehat{F}'_{m-1} \cong F'_{m-1}$.*

Proof. Since $2s+2$ is a τ-recovery stage, if $\tau \neq \sigma$ then σ must have fully recovered at least $|\tau|+1$ many times by stage $2s+1$, so the first part of the lemma follows from the definition of phase-2 recovery stage; we prove the second part. There are several cases.

We begin with the $\tau = \sigma$ and $i = r$ case. (We need to treat this case separately to show that $\widehat{X}' \cong X'$, which will be needed for the other cases.) Since $i = r_{\sigma,2s+2}$, the row of level-i components corresponding to σ in the operation performed at stage $2s+2$ goes to the left. That is, Z' is a copy of $Z \cdot B$; B' is a copy of $B \cdot S$; S' is a copy of $S \cdot C$; C' is a copy of $C \odot (Y_0, \dots, Y_{m-1})$; each Y_j, $j < m$, contains a copy of $Y_j \cdot X$; and X contains a copy of $X \cdot Z$.

By definition, $\widehat{S}$ and $\widehat{S}'$ are the σ-special components of $\mathcal{G}_n[2s+1]$ and $\mathcal{G}_n[2t]$, respectively. Thus, since $r_{\sigma,2t+1} = r_{\sigma,2t} = r$ and $2t+1$ is a σ-recovery stage, $\widehat{S}' \cong S'$.

All the components of $\mathcal{A}_{2t}$ that contain a copy of C are isomorphic to either S' or C'. Since $\widehat{S}' \cong S'$, it must be the case that $\widehat{C}' \cong C'$.

Let $j < m$. All the components of $\mathcal{A}_{2t}$ that contain a copy of Y_j are isomorphic to either C' or Y'_j. Since $\widehat{C}' \cong C'$, it must be the case that $\widehat{Y}'_j \cong Y'_j$.

The only components of $\mathcal{A}_{2t}$ that contain a copy of X are $Y'_0, \dots, Y'_{m-1}$ and X'. Since $\widehat{Y}'_j \cong Y'_j$ for each $j < m$, it must be the case that $\widehat{X}' \cong X'$.

The only components of $\mathcal{A}_{2t}$ that contain a copy of Z are X' and components isomorphic to Z'. Since $\widehat{X}' \cong X'$, it must be the case that $\widehat{Z}' \cong Z'$.

All the components of $\mathcal{A}_{2t}$ that contain a copy of B are isomorphic to either Z' or B'. Since $\widehat{Z}' \cong Z'$, it must be the case that $\widehat{B}' \cong B'$.

We now deal with the $i \equiv r_{\tau,2s+2} \bmod m+1$ case. As in the first case, the row of level-i components corresponding to τ in the operation performed at stage $2s+2$ goes to the left.

The previous case shows that $\widehat{X}' \cong X'$.

The only components of $\mathcal{A}_{2t}$ that contain a copy of Z are X' and components isomorphic to Z'. Since $\widehat{X}' \cong X'$, it must be the case that $\widehat{Z}' \cong Z'$.

All the components of $\mathcal{A}_{2t}$ that contain a copy of B are isomorphic to either Z' or B'. Since $\widehat{Z}' \cong Z'$, it must be the case that $\widehat{B}' \cong B'$.

All the components of $\mathcal{A}_{2t}$ that contain a copy of S are isomorphic to either B' or S'. Since $\widehat{B}' \cong B'$, it must be the case that $\widehat{S}' \cong S'$.

All the components of $\mathcal{A}_{2t}$ that contain a copy of C are isomorphic to either S' or C'. Since $\widehat{S}' \cong S'$, it must be the case that $\widehat{C}' \cong C'$.

Let $j < m$. All the components of $\mathcal{A}_{2t}$ that contain a copy of Y_j are isomorphic to either C' or Y_j'. Since $\widehat{C}' \cong C'$, it must be the case that $\widehat{Y}_j' \cong Y_j'$.

We now deal with the $i \not\equiv r_{\tau,2s+2} \bmod m+1$ case. Let $l < m$ be such that $i \equiv l+r_{\tau,2s+2}+1 \bmod m+1$. In this case, the row of level-i components corresponding to σ in the operation performed at stage $2s+2$ goes to the right. That is, B' is a copy of $B \cdot Z$; S' is a copy of $S \cdot B$; C' is a copy of $C \cdot S$; Y_l' is a copy of $Y_l' \odot (C, Y_0, \ldots, Y_{l-1}, Y_{l+1}, \ldots, Y_{m-1})$; each Y_j', with $j < m$ and $j \neq l$, contains a copy of $Y_j \cdot X$; X' contains a copy of $X \cdot Y_l$; and Z' contains a copy of $Z \cdot X$.

As before, the first case shows that $\widehat{X}' \cong X'$.

The only components of $\mathcal{A}_{2t}$ that contain a copy of Y_l are X' and components isomorphic to Y_l'. Since $\widehat{X}' \cong X'$, it must be the case that $\widehat{Y}_l' \cong Y_l'$.

Let $j < m$, $j \neq l$. All the components of $\mathcal{A}_{2t}$ that contain a copy of Y_j are isomorphic to either Y_l' or Y_j'. Since $\widehat{Y}_l' \cong Y_l'$, it must be the case that $\widehat{Y}_j' \cong Y_j'$.

All the components of $\mathcal{A}_{2t}$ that contain a copy of C are isomorphic to either Y_l' or C'. Since $\widehat{Y}_l' \cong Y_l'$, it must be the case that $\widehat{C}' \cong C'$.

All the components of $\mathcal{A}_{2t}$ that contain a copy of S are isomorphic to either C' or S'. Since $\widehat{C}' \cong C'$, it must be the case that $\widehat{S}' \cong S'$.

All the components of $\mathcal{A}_{2t}$ that contain a copy of B are isomorphic to either S' or B'. Since $\widehat{S}' \cong S'$, it must be the case that $\widehat{B}' \cong B'$.

All the components of $\mathcal{A}_{2t}$ that contain a copy of Z are isomorphic to either B' or Z'. Since $\widehat{B}' \cong B'$, it must be the case that $\widehat{Z}' \cong Z'$.

Finally, we deal with the case of D, E, and $F_0, \ldots, F_{m-1}$. First suppose that $i \equiv 0 \bmod m+1$. In this case, the row of components containing E in the operation performed at stage $2s+2$ goes to the left. That is, D' is a copy of $D \cdot E$; E' is a copy of $E \odot (F_0, \ldots, F_{m-1})$; each F_j', $j < m$, contains a copy of $F_j \cdot X$; and X' contains a copy of $X \cdot D$.

As before, the first case shows that $\widehat{X}' \cong X'$.

The only components of $\mathcal{A}_{2t}$ that contain a copy of D are X' and components isomorphic to D'. Since $\widehat{X}' \cong X'$, it must be the case that $\widehat{D}' \cong D'$.

All the components of $\mathcal{A}_{2t}$ that contain a copy of E are isomorphic to either D' or E'. Since $\widehat{D}' \cong D'$, it must be the case that $\widehat{E}' \cong E'$.

Let $j < m$. All the components of $\mathcal{A}_{2t}$ that contain a copy of F_j are isomorphic to either E' or F_j'. Since $\widehat{E}' \cong E'$, it must be the case that $\widehat{F}_j' \cong F_j'$.

Now suppose that $i \not\equiv 0 \bmod m+1$. Let $l < m$ be such that $i \equiv l+1 \bmod m+1$. In this case, the row of components containing E in the operation performed at stage $2s+2$ goes to the right. That is, E' is a copy of $E \cdot D$; F_l' is a copy of $F_l' \odot (E, F_0, \ldots, F_{l-1}, F_{l+1}, \ldots, Y_{m-1})$; each F_j', with $j < m$ and $j \neq l$, contains a copy of $F_j \cdot X$; X' contains a copy of $X \cdot F_l$; and D' contains a copy of $D \cdot X$.

As before, the first case shows that $\widehat{X}' \cong X'$.

The only components of $\mathcal{A}_{2t}$ that contain a copy of F_l are X' and components isomorphic to F_l'. Since $\widehat{X}' \cong X'$, it must be the case that $\widehat{F}_l' \cong F_l'$.

Let $j < m$, $j \neq l$. All the components of $\mathcal{A}_{2t}$ that contain a copy of F_j are isomorphic to either F_l' or F_j'. Since $\widehat{F}_l' \cong F_l'$, it must be the case that $\widehat{F}_j' \cong F_j'$.

All the components of $\mathcal{A}_{2t}$ that contain a copy of E are isomorphic to either F_l' or E'. Since $\widehat{F}_l' \cong F_l'$, it must be the case that $\widehat{E}' \cong E'$.

All the components of $\mathcal{A}_{2t}$ that contain a copy of D are isomorphic to either E' and D'. Since $\widehat{E}' \cong E'$, it must be the case that $\widehat{D}' \cong D'$. □

The following lemma can be easily checked.

3.47 Lemma. *Let $p \in \widetilde{H}_5$ and suppose that $(p)_{2s+1}$ participates in an operation at stage $2s+2$. Then $(p)_{2s+1}$ is one of $Y^i_{\tau,m,2s+1}$, X_{2s+1}, $Z^i_{\tau,2s+1}$, $B^i_{\tau,2s+1}$, $S^i_{\tau,2s+1}$, $C^i_{\tau,2s+1}$,*

D^i_{2s+1}, E^i_{2s+1}, *or* $F^i_{m,2s+1}$, *where either* $\tau = \sigma$ *or* $\sigma^\frown 0 \subseteq \tau$, $m \in \omega$, *and* $i \in \mathbb{Z}$. *Furthermore,* σ *is active at stage* $2s+2$.

3.48 Lemma. *Let* u *be a stage after which* σ *is never initialized and such that* $r_{\sigma,s} = r$ *for all* $s \geqslant u$. *Let* $s+1 > u$ *be a* σ*-recovery stage and let* $t+1$ *be the next* σ*-recovery stage after stage* $s+1$. *Let* $p \in \widetilde{H}_5$. *Suppose there exists a component* L *of* $\mathcal{G}_n[s]$ *that is* $(-r)$*-isomorphic to* $(p)_s$. *Then the component* L' *of* $\mathcal{G}_n[t]$ *that extends* L *is isomorphic to* $(p)_t$.

Proof. If (p) does not participate in an operation in the interval $(s,t]$ then $(p)_t \cong (p)_s$. Since $L' \supseteq L$, $(p)_t$ is not properly embeddable in any component of $\mathcal{A}_t$, and, by convention, $\mathcal{G}_n[t]$ is embeddable in $\mathcal{A}_t$, this means that $L' \cong (p)_t$.

Otherwise, the lemma follows from Lemmas 3.45, 3.46, and 3.47. □

3.49 Lemma. *Let* u *be a stage after which* σ *is never initialized and such that* $r_{\sigma,s} = r$ *for all* $s \geqslant u$. *Let* $x \in P_{\widetilde{H}_5}$. *There exists a* σ*-recovery stage* $s+1 > u$ *such that* x *is contained in* $(p)_s$ *for some* $p \in \widetilde{H}_5$ *and there exists a* $(-r)$*-isomorphic component* L *of* $\mathcal{G}_n[s]$. *For any such* s, *if we let* d *be the unique isomorphism from* $(p)_s$ *to* L *and let* L' *be the component of* $\mathcal{G}_n$ *that extends* L *then* d *can be extended to an isomorphism from* (p) *to* L'.

Proof. If x is contained in a finite component of $\mathcal{A}$ then the existence of s and L follows from the fact that $\mathcal{G}_n \cong \mathcal{A}$. Otherwise, there is a σ-recovery stage $s+1 > u$ such that x is contained in $(p)_s$, $p \in \widetilde{H}_5$, and $(p)_s$ is involved in an operation at stage $s+1$. Now it follows from Lemmas 3.45 and 3.46 that there is a component L of $\mathcal{G}_n[s]$ isomorphic to $(p)_s$.

Let $s+1 = s_0+1 < s_1+1 < \cdots$ be the σ-recovery stages greater than or equal to $s+1$. Let L_i be the component of $\mathcal{G}_n[s_i]$ that extends L and let L' be the component of $\mathcal{G}_n$ that extends L. Using Lemma 3.48 and induction, we see that, for each $i \geqslant 0$, there is a unique isomorphism $g_i : (p)_{s_i} \cong L_i$. Note that $g_0 = g$. Clearly, if $j > i$ then g_j extends g_i. Thus the limit g' of the g_i is well-defined and is an isomorphism from (p) to L'. □

Let T be the graph obtained by restricting the domain of $\mathcal{G}_n$ to the union of $|\mathcal{B}|$ with the domain of the set of all components of $\mathcal{G}_n$ that contain a copy of $[m]$ for some $m \in H_5$.

3.50 Lemma. *There exists a computable isomorphism* $f_5 \supset b$ *from* $P_{\widetilde{H}_5}$ *to* T.

Proof. We begin by defining $f_5 \restriction \mathcal{B} \equiv b$. Now let u be a stage after which σ is never initialized and such that $r_{\sigma,s} = r$ for all $s \geqslant u$. Given $x \in P_{\widetilde{H}_5}$, find the least σ-recovery stage $s+1 > u$ such that x is contained in a component $(p)_s$, $p \in \widetilde{H}_5$, of $\mathcal{A}_s$ and there exists a component L of $\mathcal{G}_n[s]$ that is $(-r)$-isomorphic to $(p)_s$. Such a stage exists by Lemma 3.49. Let d_x be the unique isomorphism from $(p)_s$ to L and define $f_5(x) = d_x(x)$.

We need to show that f_5 is computable, that it is an embedding, and that its range is all of T.

Since d_x can be computably determined given $x \in P_{\widetilde{H}_5}$, f_5 is computable.

By Lemma 3.49, all we need to do to show that f_5 is an embedding is to show that if x and y are both contained in a component (p), $p \in \widetilde{H}_5$, then $f_5(x)$ and $f_5(y)$ are contained in the same component of $\mathcal{G}_n$. But this follows from Lemma 3.48, which implies, by induction, that if the least σ-recovery stage $s+1 > u$ such that x is contained in $(p)_s$ is greater than or equal to the least σ-recovery stage $t+1 > u$ such that y is contained in $(p)_t$ then d_x extends d_y.

Finally, notice that, for any $s \in \omega$, if K is a component of $\mathcal{A}_s$ that contains a copy of $[m]$ for some $m \in H_5$ then K is $(p)_s$ for some $p \in \widetilde{H}_5$.

Let L be a component of T. If L is a singleton component then the fact that $\mathcal{G}_n \cong \mathcal{A}$ implies that, for some σ-recovery stage $s+1 > u$, there is a component K of $\mathcal{A}_s$ that is r-isomorphic to L. Since K is $(p)_s$ for some $p \in \widetilde{H}_5$, L is in the range of f_5.

If L is not a singleton component then it is in $(\mathcal{G}_n)_\tau$ or $(\mathcal{G}_n^*)_\tau$ for some τ such that $\tau = \sigma$ or $\sigma^\frown 0 \subseteq \tau$. Let $x \in L$ and let $t > u$ be a stage such that x is contained in a component of $\mathcal{G}_n[s]$ that contains a copy of $[m]$ for some $m \in H_5$. By the definition of σ-recovery stage, there is some σ-recovery stage $s+1 \geqslant t$ and components L' and K of $\mathcal{G}_n[s]$ and $\mathcal{A}_s$, respectively, such that $x \in L'$ and K is r-isomorphic to L'. Since K is $(p)_s$ for some $p \in \widetilde{H}_5$, x is in the range of f_5. Thus L is in the range of f_5. $\square$

Now $\widetilde{H}_0, \dots, \widetilde{H}_5$ are computable subsets of N such that $\bigcup_{i=0}^5 \widetilde{H}_i = N$. It is not hard to check that, for $i, j \leqslant 5$ such that $i \neq j$, if K and L are components of $P_{\widetilde{H}_i}$ and $P_{\widetilde{H}_j}$, respectively, and $f_i(K) = f_j(L)$, then K and L are components of $\mathcal{A}^*$, from which it follows that $K = L$. Furthermore, the uniqueness of f_0, f_1, f_2, f_3^1, and f_4, together with the surjectivity of f_3^0 and f_5, imply that $\bigcup_{i=0}^m \mathrm{rng}(f_i) = |\mathcal{G}_n|$. So, combining Lemmas 3.35, 3.36, 3.38, 3.42, 3.43, 3.44, and 3.50 with Lemma 3.32, we have the following result.

3.51 Lemma. *There exists a computable isomorphism from $(\mathcal{A}, a_r)$ to $(\mathcal{G}_n, g)$.*

Theorem 1.4 follows from Lemmas 3.10, 3.27, and 3.51.

References

[1] P. Cholak, S. S. Goncharov, B. Khoussainov, and R. A. Shore, Computably categorical structures and expansions by constants, J. Symbolic Logic 64 (1999) 13–37.

[2] Y. L. Ershov, S. S. Goncharov, A. Nerode, and J. B. Remmel (eds.), Handbook of Recursive Mathematics, vol. 138–139 of Studies in Logic and the Foundations of Mathematics (Elsevier Science, Amsterdam, 1998).

[3] S. S. Goncharov, Computable single-valued numerations, Algebra and Logic 19 (1980) 325–356.

[4] S. S. Goncharov, Problem of the number of non-self-equivalent constructivizations, Algebra and Logic 19 (1980) 401–414.

[5] V. S. Harizanov, Pure computable model theory, in Ershov et al. [2] 3–114.

[6] D. R. Hirschfeldt, Degree spectra of intrinsically c.e. relations, J. Symbolic Logic 66 (2001) 441–469.

[7] D. R. Hirschfeldt, Degree spectra of relations on structures of finite computable dimension, Ann. Pure Appl. Logic 115 (2002) 233–277.

[8] D. R. Hirschfeldt, B. Khoussainov, R. A. Shore, and A. M. Slinko, Degree spectra and computable dimension in algebraic structures, Ann. Pure Appl. Logic 115 (2002) 71–113.

[9] B. Khoussainov and R. A. Shore, Computable isomorphisms, degree spectra of relations, and Scott families, Ann. Pure Appl. Logic 93 (1998) 153–193.

[10] B. Khoussainov and R. A. Shore, Effective model theory: the number of models and their complexity, in S. B. Cooper and J. K. Truss (eds.), Models and Computability, vol. 259 of London Mathematical Society Lecture Note Series (Cambridge University Press, Cambridge, 1999) 193–239.

[11] T. Millar, Recursive categoricity and persistence, J. Symbolic Logic 51 (1986) 430–434.

[12] R. I. Soare, Recursively Enumerable Sets and Degrees, Perspect. Math. Logic (Springer–Verlag, Heidelberg, 1987).