DEGREE SPECTRA OF RELATIONS ON COMPUTABLE STRUCTURES

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Denis Roman Hirschfeldt

August 1999

DEGREE SPECTRA OF RELATIONS ON COMPUTABLE STRUCTURES

Denis Roman Hirschfeldt, Ph.D.

Cornell University 1999

The study of additional relations on computable structures began with the work of Ash and Nerode [2]. The concept of degree spectra of relations was later introduced by Harizanov [18]. In this dissertation, several new examples of possible degree spectra of relations on computable structures are given. In particular, it is shown that, for every c.e. degree $\mathbf{a}$, the set $\{\mathbf{0}, \mathbf{a}\}$ can be realized as the degree spectrum of an intrinsically c.e. relation on a structure of computable dimension two, thus answering a question of Goncharov and Khoussainov [16]. Some extensions of this result are given, and the methods used in proving it are employed to construct a computably categorical structure whose expansion by a single constant has computable dimension ω. Degree spectra of relations on computable models of particular algebraic theories are also investigated. For example, it is shown that, for every $n > 0$, there is a computable integral domain with a subring whose degree spectrum consists of exactly n c.e. degrees, including $\mathbf{0}$. In contrast to this result, it is shown, for instance, that the degree spectrum of a computable relation on a computable linear ordering is either a singleton or infinite. In both cases, sufficient criteria for similar results to hold of a given class of structures are provided.

BIOGRAPHICAL SKETCH

Denis Roman Hirschfeldt was born in Rio de Janeiro, Brazil, where he attended the Escola Americana do Rio de Janeiro, graduating in 1989. He received a B.A. in mathematics from the University of Pennsylvania in 1993 and an M.S. in Computer Science from Cornell University in 1998. In 1999–2000, he will be a Postdoctoral Research Fellow at Victoria University of Wellington, New Zealand. After that, he will be an L.E. Dickson Instructor at the University of Chicago.

To my parents, Ruth and Jury.

ACKNOWLEDGEMENTS

Throughout my education, I have benefited from the craft of many dedicated and inspiring teachers, in and out of the classroom. They have not only helped shape my intellectual development, but have provided me with a model and a goal. Even if I only have the space to mention a few of them below, they all have my deep gratitude.

My parents, Ruth and Jury, to whom this dissertation is dedicated, were my first teachers and, through their love and friendship, continue to teach me always about the most important things in life. Even if my education and career keep me physically distant from them, I would like them to know that they are always with me, a constant influence, and that they will always have my deepest love, gratitude, and respect. To them and to my brother Robert I also owe a great deal for the lively and stimulating intellectual atmosphere they provided throughout my childhood and adolescence.

I could not have hoped for a better advisor than Richard Shore. The chance to interact with a world-class mathematician who also happens to be a master teacher ready to give more than generously of his time, knowledge, and experience to his students was crucial to my mathematical development. I came into Cornell a student interested in mathematics; if I can now claim to be a mathematician, Richard is the person most responsible for this. He is the sort of advisor whose students get together surreptitiously to speak well of him.

I also thank the other members of my committee, Anil Nerode and Dexter Kozen, for their help and advice.

Bakh Khoussainov has been a valuable friend and collaborator, whose enthusiasm and intellectual fearlessness have been an inspiration.

I thank Rod Downey for inviting me to Victoria University of Wellington in October–November, 1998, and for all that he taught me while I was there.

I thank Jim Lipton for steering me toward logic and Cornell.

I thank Arkadii Slinko for very useful comments regarding the material in chapter 5.

Finally, I gratefully acknowledge the support of the Alfred P. Sloan Foundation through a Doctoral Dissertation Fellowship for the academic year 1998–1999.

I have met some wonderful people in my six years in Ithaca, and have made what I am sure will be lasting friendships. Rather than risk leaving anyone out, I will thank all my friends here collectively, and hope that they will understand that I left the writing of these acknowledgements for the last minute and want to get this over and done with. Besides, the thing with true friends is, they always know who they are.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

There has been increasing interest over the last few decades in the study of the effective content of Mathematics. One field whose effective content has been the subject of a large body of work, dating back at least to the early 1960's, is model theory. (A valuable reference is the handbook [7]. In particular, the introduction and the articles by Ershov and Goncharov and by Harizanov give useful overviews, while the articles by Ash and by Goncharov cover material related to the topic of this dissertation.)

Several different notions of effectiveness of model-theoretic structures have been investigated. This dissertation is concerned with *computable* structures, that is, structures with computable domains whose constants, functions, and relations are uniformly computable.

In model theory, we identify isomorphic structures. From the point of view of computable model theory, however, two isomorphic structures might be very different. For example, it is not hard to build two isomorphic computable groups, only one of which has a computable center. We do not wish to say that these two presentations are the same. Thus, for our purposes, studying structures up to isomorphism is not enough. Instead, we study structures up to *computable* isomorphism. This leads naturally to the idea of a *computable presentation* of a structure, which is, roughly speaking, a computable copy of this structure. (Formal definitions of this and other concepts will be given below.)

One way in which we may attempt to understand the differences between non-computably isomorphic computable presentations of a structure $\mathcal{M}$ is to compare (from a computability-theoretic point of view) the images in these presentations of a particular relation on the domain of $\mathcal{M}$. (Of course, this is only interesting if this relation is not the interpretation in $\mathcal{M}$ of a relation in the language of $\mathcal{M}$.) The study of additional relations on computable structures began with the work of Ash and Nerode [2] and has been continued in a large number of papers. (References can be found in the aforementioned articles in [7].)

One approach to the study of relations on computable structures, which began with the work of Harizanov [18], is to look at the collection of (Turing) degrees of the images of a relation in different computable presentations of a structure, which is known as the *degree spectrum* of the relation. This dissertation is mainly concerned with the question of which sets of degrees can be realized as degree spectra of relations on computable structures, both in the general case and with certain restrictions imposed on the relation or the structure. The latter case will bring us to the intersection of computable model theory and computable algebra.

In the remaining parts of this introductory chapter, we first give the fundamental definitions and notations that will be used throughout the rest of this dissertation, and then summarize the results contained therein.

1.1 Basic Definitions and Notation

For basic notions of computability theory and model theory, the reader is referred to [34] and [23], respectively. By *degree*, we will mean Turing degree unless otherwise specified. In order to avoid confusion with our symbol for *or*, we will denote the join of degrees $\mathbf{a}$ and $\mathbf{b}$ by $\mathbf{a} \curlyvee \mathbf{b}$.

Whenever we mention a computably enumerable (c.e.) set X, we assume we have fixed some computable enumeration of X and let $X[s]$ denote the part of X enumerated after $s+1$ many steps. Similarly, if we mention a Δ^0_2 set Y then we assume we have fixed some computable approximation of Y and let $Y[s]$ denote the result of performing $s+1$ many steps of this approximation.

For any set X, let $X \upharpoonright m = X \cap \{0, \ldots, m-1\}$. For any function f, let $f \upharpoonright m$ be the function obtained by restricting $\mathrm{dom}(f)$ to $\mathrm{dom}(f) \upharpoonright m$.

The e^{th} Turing functional with oracle X is denoted by $\Phi_e(X)$, and its value at x by $\Phi_e(X; x)$. Let $\Phi_e(X)[s]$ be the evaluation of $\Phi_e(X[s])$ at stage s and let $\Phi_e(X, x)[s]$ be the value of this evaluation at x. The use functions of $\Phi_e(X; x)$ and $\Phi_e(X; x)[s]$ are denoted by $\varphi_e(X; x)$ and $\varphi_e(X; x)[s]$, respectively.

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 $a \in \omega$ and $b \in \omega$. We will write

$\langle a, b, c \rangle$ instead of $\langle a, \langle b, c \rangle \rangle$, and similarly for longer sequences of natural numbers. For $x \in \omega$ and $i = 0, 1$, $\pi_i(x)$ will denote 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 $\pi_1(x) = b$.

If $\vec{x} = (x_0, \ldots, x_m)$ and $\vec{y} = (y_0, \ldots, y_n)$ are sequences then $\vec{x}(i) = x_i$ and $\vec{x}^\frown \vec{y}$ is the sequence $(x_0, \ldots, x_m, y_0, \ldots, y_n)$. We will write $\vec{x}^\frown z$ instead of $\vec{x}^\frown (z)$, where (z) is the sequence consisting of the single element z.

One of the central notions of computable model theory is that of a *computable structure*. We will always assume that we are working with computable languages.

1.1.1. Definition. A structure $\mathcal{A}$ is *computable* if both its domain $|\mathcal{A}|$ and the atomic diagram of $\langle \mathcal{A}, a \rangle_{a \in |\mathcal{A}|}$ are computable.

If, in addition, the n-quantifier diagram of $\langle \mathcal{A}, a \rangle_{a \in |\mathcal{A}|}$ is computable then $\mathcal{A}$ is *n-decidable*, while if the full first-order diagram of $\langle \mathcal{A}, a \rangle_{a \in |\mathcal{A}|}$ is computable then $\mathcal{A}$ is *decidable*.

It will be more convenient to treat equality in computable structures as actual equality rather than as a relation. This is not an important distinction, however, since every computable structure is computably isomorphic to some computable structure $\mathcal{B}$ such that if $a \neq b \in |\mathcal{B}|$ then $\mathcal{B} \vDash a \neq b$.

In Chapters 3 and 4, we will have to consider partial computable directed graphs, which we define as follows.

1.1.2. Definition. 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 ($\Psi[s]$).

As we have discussed above, the following definition is a natural one to make in the context of computable model theory.

1.1.3. Definition. 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.)

If $\mathcal{M}$ has a computable presentation then it is *computably presentable.*

An important notion in computable model theory is the number of computable presentations of a computably presentable structure.

1.1.4. Definition. The *computable dimension* of a computably presentable 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.*

We will also have occasion to consider structures that, while not computably categorical, have relatively simple isomorphisms between their various computable presentations.

1.1.5. Definition. A computably presentable structure is Δ^0_2-*categorical* if any two of its presentations are isomorphic via a Δ^0_2 map.

In Chapter 3, we will also consider c.e. presentations. We will take the more general of two possible definitions of c.e. structure, in which equality is c.e. rather than computable. It will be clear that the results involving c.e. structures in Chapter 3 also hold for the less general definition.

1.1.6. Definition. A structure $\mathcal{A}$ is *c.e.* if its domain $|\mathcal{A}|$ is computable and the atomic diagram of $\langle \mathcal{A}, a\rangle_{a\in|\mathcal{A}|}$ is c.e..

An isomorphism from a structure $\mathcal{M}$ to a c.e. structure is called a *c.e. presentation* of $\mathcal{M}$. (As in the computable case, we often refer to the image of a c.e. presentation as a c.e. presentation.)

If $\mathcal{M}$ has a c.e. presentation then it is *c.e. presentable.*

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

It is convenient to assume that the domain of a c.e. structure is computable rather than c.e., but this makes no real difference, since any structure with c.e. domain is computably isomorphic to a structure with computable domain.

As we have mentioned above, the study of additional relations on computable structures began with the work of Ash and Nerode [2], who were concerned with relations that maintain some degree of effectiveness in different computable presentations of a structure.

1.1.7. Definition. Let U be a relation on the domain of a computable structure $\mathcal{A}$ and let $\mathfrak{C}$ be a class of relations. U is *intrinsically* $\mathfrak{C}$ on $\mathcal{A}$ if the image of U in any computable presentation of $\mathcal{A}$ is in $\mathfrak{C}$.

In [2], conditions that guarantee that a relation is intrinsically computable or intrinsically c.e. were given. More recent work has led to a number of other conditions guaranteeing that a relation is intrinsically $\mathfrak{C}$ for various classes $\mathfrak{C}$ (see [3], for example).

Invariant structures will often be important in this dissertation.

1.1.8. Definition. A relation U on a structure $\mathcal{M}$ is *invariant* if, for every automorphism $f : \mathcal{M} \cong \mathcal{M}$, $f(U) = U$.

The following definition, which, as mentioned above, is the main topic of study of this dissertation, is due to Harizanov [18].

1.1.9. Definition. Let U be a relation on the domain of a computable structure $\mathcal{A}$. The *degree spectrum* of U on $\mathcal{A}$, $\mathrm{DgSp}_{\mathcal{A}}(U)$, is the set of degrees of the images of U in all computable presentations of $\mathcal{A}$.

It is also interesting to consider degree spectra of relations with respect to other reducibilities.

1.1.10. Definition. Let r be a reducibility, such as many-one reducibility (m-reducibility) or truth-table reducibility. Let U be a relation on the domain of a computable structure $\mathcal{A}$. The *r-degree spectrum* of U on $\mathcal{A}$, $\mathrm{DgSp}^{\mathrm{r}}_{\mathcal{A}}(U)$, is the set of r-degrees of the images of U in all computable presentations of $\mathcal{A}$.

1.2 Summary of Results

In Chapter 2, we give some examples of sets of degrees that can be realized as degree spectra of relations on computable structures. We show, for instance, that for every c.e. degree $\mathbf{a}$ and every computable ordinal α, the set of α-c.e. degrees less than or equal to $\mathbf{a}$ can be realized as the degree spectrum of an intrinsically α-c.e. relation on a computable structure. (For a definition of α-c.e. sets and degrees, see Section 2.3.)

In Chapter 3, we consider finite degree spectra of relations. These were first studied by Harizanov [20], who showed that there exists a Δ_2^0 degree $\mathbf{a}$ such that $\{\mathbf{0}, \mathbf{a}\}$ can be realized as the degree spectrum of a relation on a computable structure of computable dimension 2. Khoussainov and Shore and Goncharov [25],[16] later improved this result by showing the existence of a c.e. degree $\mathbf{a}$ such that $\{\mathbf{0}, \mathbf{a}\}$ can be realized as the degree spectrum of an intrinsically computable relation on a computable structure of computable dimension 2.

In Section 3.2, we show that, in fact, for every c.e. degree $\mathbf{a}$, $\{\mathbf{0}, \mathbf{a}\}$ can be realized as the degree spectrum of an intrinsically c.e. relation on a computable structure. In Section 3.3, we show that this result remains true if we also require that the structure in question have computable dimension 2. This last result, which answers a question of Goncharov and Khoussainov [16], has been independently obtained by Khoussainov and Shore [24].

There are at least two natural directions in which these results can be extended. In Section 3.4, we show that, for every uniformly c.e. collection of sets S, the set of degrees of elements of S can be realized as the degree spectrum of an intrinsically computable relation on a computable structure. Then, in Section 3.5, we show that if $\alpha \in \omega \cup \{\omega\}$ then, for every α-c.e. degree $\mathbf{a}$, $\{\mathbf{0}, \mathbf{a}\}$ can be realized as the degree spectrum of an intrinsically α-c.e. relation on a computable structure. As we will see, all of these results also hold for m-degree spectra of relations, and hence for r-degree spectra of relations for any reducibility r weaker than m-reducibility.

The methods of Chapter 3 can be used to answer a question about what can happen to the computable dimension of a computably categorical structure when it is expanded by finitely many constants. Cholak, Goncharov, Khoussainov, and Shore [4] showed that if $k > 0$ then there exists a computably categorical structure $\mathcal{A}$ and an $a \in |\mathcal{A}|$ such that $\langle \mathcal{A}, a \rangle$ has computable dimension k. This left open the question of whether there exists a computably categorical structure whose expansion by some finite set of constants has computable dimension ω. In Chapter 4, which reports on joint work with Bakhadyr Khoussainov and Richard Shore, we answer this question in the affirmative.

The structures built in Chapters 3 and 4 are all directed graphs. It is interesting to ask for which well-known algebraic theories do the results of those chapters remain true if we also require that the structures in question be models of the given theory. We address this question in Chapter 5, in which we give a sufficient condition for the results of Chapters 3 and 4 (as well as other related results) to remain true if we also require that the structures in question be models of a given theory, and apply it to the cases of undirected graphs, integral domains, and commutative semigroups.

Chapter 6 contains results that are in contrast to those of Chapter 5, namely, conditions that guarantee that the degree spectrum of a relation is either a singleton or infinite. These conditions are used to show, for example, that a computable relation on a computable linear ordering is either intrinsically computable or has infinite degree spectrum.

Chapter 2

Examples of Degree Spectra of Relations

2.1 Introduction

In this chapter, we give a few examples of sets of degrees that can be realized as degree spectra of relations. Before we proceed, we make three brief observations.

The first one is rather simple: For any class of relations $\mathfrak{C}$, if U is an intrinsically $\mathfrak{C}$ k-ary relation on a computable structure $\mathcal{A}$ then $V = (|\mathcal{A}|)^k - U$ is intrinsically co-$\mathfrak{C}$ and $\mathrm{DgSp}_{\mathcal{A}}(V) = \mathrm{DgSp}_{\mathcal{A}}(U)$.

Now let U and V be k-ary relations on the domains of computable graphs $\mathcal{A} = \langle |\mathcal{A}|, E \rangle$ and $\mathcal{B} = \langle |\mathcal{B}|, F \rangle$, respectively. Let $\mathcal{C} = \langle |\mathcal{C}|, R, Q \rangle$ be the computable structure in the language with one binary and one unary relation defined by

$$|\mathcal{C}| = \{2x \mid x \in |\mathcal{A}|\} \cup \{2x+1 \mid x \in |\mathcal{B}|\},$$

$$R = \{(2x, 2y) \mid E(x, y)\} \cup \{(2x + 1, 2y + 1) \mid F(x, y)\},$$

and

$$Q = \{2x \mid x \in |\mathcal{A}|\}.$$

Let

$$W = \{(2x_0, \ldots, 2x_{k-1}) \mid (x_0, \ldots, x_{k-1}) \in U\} \cup \{(2x_0 + 1, \ldots, 2x_{k-1} + 1) \mid (x_0, \ldots, x_{k-1}) \in V\}.$$

It is easy to check that

$$\mathrm{DgSp}_{\mathcal{C}}(W) = \{\mathbf{c} \mid \exists \mathbf{a}, \mathbf{b}(\mathbf{a} \in \mathrm{DgSp}_{\mathcal{A}}(U) \wedge \mathbf{b} \in \mathrm{DgSp}_{\mathcal{B}}(V) \wedge \mathbf{c} = \mathbf{a} \curlyvee \mathbf{b})\}.$$

Furthermore, if $\mathcal{A}$ and $\mathcal{B}$ are Δ^0_2-categorical then so is $\mathcal{C}$, and, for any class of relations $\mathfrak{C}$ closed under m-equivalence and finite disjoint unions, if U and V are intrinsically $\mathfrak{C}$ then so is W.

It is not hard to modify this construction to handle the case in which U and V do not necessarily have the same arity and $\mathcal{A}$ and $\mathcal{B}$ are arbitrary computable structures, and hence establish the following result.

2.1.1. Proposition. *Let A and B be sets of degrees and let $C = \{\mathbf{c} \mid \exists \mathbf{a}, \mathbf{b}(\mathbf{a} \in A \wedge \mathbf{b} \in B \wedge \mathbf{c} = \mathbf{a} \curlyvee \mathbf{b})\}$. Let $\mathfrak{C}$ be a class of relations closed under m-equivalence and finite disjoint unions. If both A and B can be realized as degree spectra of (intrinsically $\mathfrak{C}$) relations on the domains of (Δ^0_2-categorical) computable structures then so can C.*

Our final observation is about degree spectra of relations with respect to different reducibilities. Let r and s be reducibilities such that r is stronger than s, and let U

be a relation on a computable structure $\mathcal{A}$. Then $\mathrm{DgSp}^{\mathrm{s}}_{\mathcal{A}}(U)$ is equal to the set of s-degrees that contain at least one r-degree in $\mathrm{DgSp}^{\mathrm{r}}_{\mathcal{A}}(U)$. Thus, for example, when we construct in the next section a relation U on a computable structure $\mathcal{A}$ such that $\mathrm{DgSp}^{\mathrm{m}}_{\mathcal{A}}(U)$ consists of all Σ^0_n m-degrees (other than the m-degrees of $\emptyset$ and ω), it will be the case that, for any reducibility r weaker than m-reducibility, $\mathrm{DgSp}^{\mathrm{r}}_{\mathcal{A}}(U)$ consists of all Σ^0_n r-degrees.

2.2 Σ^0_n, Π^0_n, and Δ^0_n Degrees

Let C_0 be the directed graph consisting of a single node and no edges and let C_1 be the directed graph consisting of two nodes x and y with an edge from x to y. Consider the directed graph $\mathcal{G} = \langle |\mathcal{G}|, E\rangle$ that is the disjoint union of infinitely many copies of each of C_0 and C_1. Let U be the unary relation on the domain of $\mathcal{G}$ that holds of x if and only if there is a y such that $E(x, y)$. Since U is defined by an existential formula in the language of directed graphs, U is intrinsically c.e.. We claim that $\mathrm{DgSp}_{\mathcal{G}}(U)$ consists of all c.e. degrees. (In fact, $\mathrm{DgSp}^{\mathrm{m}}_{\mathcal{G}}(U)$ consists of all c.e. m-degrees other than the m-degrees of $\emptyset$ and ω.)

Indeed, let A be an infinite and coinfinite c.e. set and let $a_0, a_1, \ldots$ be a computable enumeration of A. Define a directed graph G with edge relation F as follows. Let $|G| = \omega$ and, for $x, y \in \omega$, let $F(x, y)$ hold if and only if, for some $k \in \omega$, $x = 2a_k$ and $y = 2k + 1$. It is easy to check that G is a computable presentation of $\mathcal{G}$. Furthermore, $U^G(x) \Leftrightarrow x = 2a \wedge a \in A$, and hence $U^G \equiv_{\mathrm{m}} A$.

By modifying this example, it is possible to realize, for any $n \in \omega$, all n-c.e.

degrees as the degree spectrum of an intrinsically n-c.e. relation on a computable structure. We will not do this here, since we will show in the next section that, for any c.e. degree $\mathbf{a}$ and any computable ordinal α, we can realize the α-c.e. degrees below $\mathbf{a}$ as the degree spectrum of an intrinsically α-c.e. relation on a computable structure. Instead, we will explore another natural generalization of the above example, given in the following theorem.

2.2.1. Theorem. *Let $n > 0$. There exists an intrinsically Σ^0_n relation U on a computably presentable structure $\mathcal{M}$ such that $\mathrm{DgSp}_{\mathcal{M}}(U)$ consists of all Σ^0_n degrees.*

Proof. The structure $\mathcal{M}$ will be a directed graph; more specifically, it will be a tree. We begin by defining trees $\mathcal{N}_k$ and $\mathcal{P}_k$, $k \leqslant n$. When we say that a copy of a tree $\mathcal{T}$ is attached to a node r, we mean that there is a node s such that there is an edge from r to s and s together with its successors form a copy of $\mathcal{T}$.

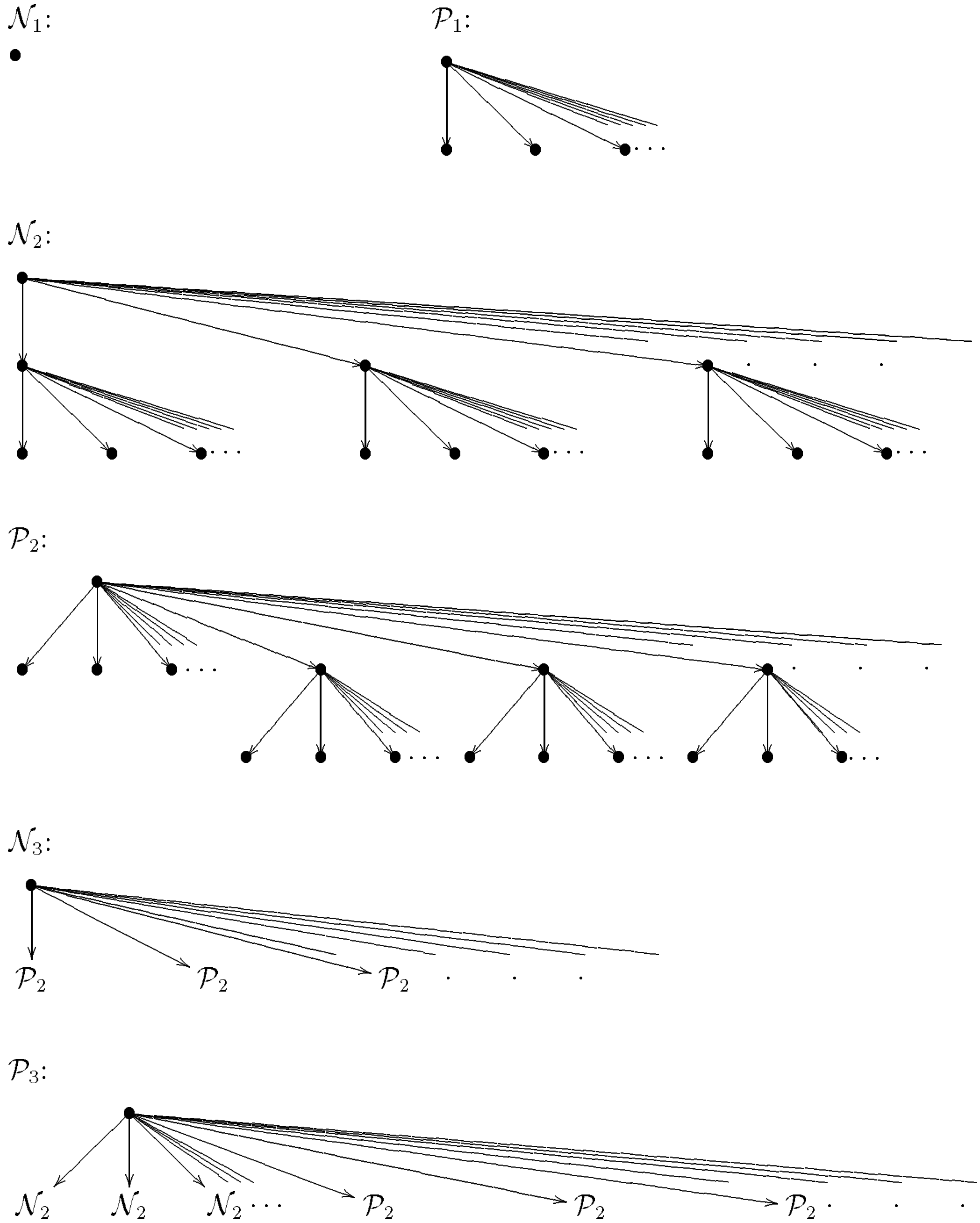
1. $\mathcal{N}_0$ consists of a single node. $\mathcal{P}_0 = \emptyset$.

2. $\mathcal{N}_1$ consists of a single node. $\mathcal{P}_1$ consists of a root node to which are attached infinitely many copies of $\mathcal{N}_0$.

3. $\mathcal{N}_{k+2}$ consists of a root node to which are attached infinitely many copies of $\mathcal{P}_{k+1}$. $\mathcal{P}_{k+1}$ consists of a root node to which are attached infinitely many copies of each of $\mathcal{N}_{k+1}$ and $\mathcal{P}_{k+1}$.

(See Figure 2.1.)

Now let $\mathcal{M} = \langle |\mathcal{M}|, E \rangle$ be the tree consisting of a root node x and infinitely many disjoint copies of each of $\mathcal{N}_n$ and $\mathcal{P}_n$ attached to x. Let U be the set of nodes of $\mathcal{M}$ of height 1 that are root nodes of copies of $\mathcal{P}_n$.



$\vdots$

Figure 2.1: The $\mathcal{N}_k$ and $\mathcal{P}_k$

We begin by showing that U is intrinsically Σ^0_n.

2.2.2. Lemma. *Let M be a computable presentation of $\mathcal{M}$. For $0 < m \leqslant n$, let S_m be the set of nodes of height $n+1-m$ in M that are root nodes of copies of $\mathcal{P}_m$. Then S_m is Σ^0_m.*

Proof. We proceed by induction. First notice that we can computably determine the height of a node in M. Since S_1 consists of all nodes x of height n such that $\exists y(E^M(x,y))$, S_1 is Σ^0_1. Now let $1 < m \leqslant n$ and assume that the lemma holds for $m-1$. Since $x \in S_m \Leftrightarrow \exists y(E^M(x,y) \wedge y \notin S_{m-1})$, the assumption that S_{m-1} is Σ^0_{m-1} implies that S_m is Σ^0_m. $\square$

Since, in the notation of the previous lemma, $U^M = S_n$, we have the following result.

2.2.3. Corollary. *For any computable presentation M of $\mathcal{M}$, U^M is Σ^0_n.*

We now show that $\mathrm{DgSp}_{\mathcal{M}}(U)$ consists of all Σ^0_n degrees.

2.2.4. Lemma. *Let $R(x_0,\dots,x_n)$ be a computable relation on ω^{n+1} and let $m \leqslant n$. If m is even then let $Q = \forall$; otherwise, let $Q = \exists$. There exist uniformly computable structures $T^R_{a_m,a_{m+1},\dots,a_n} = \left\langle \left|T^R_{a_m,a_{m+1},\dots,a_n}\right|, E^R_{a_m,a_{m+1},\dots,a_n}\right\rangle$, $a_m, a_{m+1}, \dots, a_n \in \omega$, such that if*

$$\exists x_{m-1} \forall x_{m-2} \exists x_{m-3} \forall x_{m-4} \cdots Q x_0 (R(x_0,\dots,x_{m-1},a_m,\dots,a_n))$$

holds then $T^R_{a_m,a_{m+1},\dots,a_n}$ is isomorphic to $\mathcal{P}_m$, while otherwise $T^R_{a_m,a_{m+1},\dots,a_n}$ is isomorphic to $\mathcal{N}_m$, and in either case the root node of $T^R_{a_m,a_{m+1},\dots,a_n}$ is 0.

Proof. We proceed by induction. Since R is computable, the lemma clearly holds for $m = 0$. Assume the lemma holds for $m - 1$. If m is even then let $Q = \forall$ and $\overline{Q} = \exists$; otherwise, let $Q = \exists$ and $\overline{Q} = \forall$. Fix a computable presentation $\langle |P|, F\rangle$ of $\mathcal{P}_{m-1}$ with root node 0. Let $\overline{R} = \omega^{n+1} - R$. Given $a_m, a_{m+1}, \ldots, a_n \in \omega$, let

$$\left|T^R_{a_m,a_{m+1},\ldots,a_n}\right| = \{0\} \cup \{\langle 1, a, k\rangle \mid a \in \omega,\ k \in |P|\} \cup \left\{\langle c, a, k\rangle \mid c \geqslant 2,\ a \in \omega, k \in \left|T^{\overline{R}}_{a,a_m,a_{m+1},\ldots,a_n}\right|\right\}$$

and

$$E^R_{a_m,a_{m+1},\ldots,a_n} = \left\{(0, \langle c, a, 0\rangle) \mid c \geqslant 1,\ a \in \omega,\ \langle c, a, 0\rangle \in \left|T^R_{a_m,a_{m+1},\ldots,a_n}\right|\right\} \cup \{(\langle 1, a, k\rangle, \langle 1, a, l\rangle) \mid a \in \omega,\ F(k, l)\} \cup \left\{(\langle c, a, k\rangle, \langle c, a, l\rangle) \mid c \geqslant 2,\ a \in \omega,\ E^{\overline{R}}_{a,a_m,a_{m+1},\ldots,a_n}(k, l)\right\}.$$

By the induction hypothesis, there is a computable procedure to decide, given $x, a_m, a_{m+1}, \ldots, a_n \in \omega$, whether $x \in \left|T^R_{a_m,a_{m+1},\ldots,a_n}\right|$, as well as a computable procedure to decide, given $x, y, a_m, a_{m+1}, \ldots, a_n \in \omega$, whether $E^R_{a_m,a_{m+1},\ldots,a_n}(x, y)$.

Furthermore, $T^R_{a_m,a_{m+1},\ldots,a_n}$ contains infinitely many copies of $\mathcal{P}_{m-1}$ attached to 0 (if $m > 1$), and it contains infinitely many copies of $\mathcal{N}_{m-1}$ attached to 0 if and only if it contains one copy of $\mathcal{N}_{m-1}$ attached to 0 if and only if there exists an $a \in \omega$ such $T^{\overline{R}}_{a,a_m,a_{m+1},\ldots,a_n}$ is isomorphic to $\mathcal{N}_{m-1}$ if and only if there exists an $a \in \omega$ such that

$$\neg\exists x_{m-2}\forall x_{m-3}\exists x_{m-4}\forall x_{m-5}\cdots\overline{Q}x_0\left(\overline{R}(x_0, \ldots, x_{m-1}, a_m, \ldots, a_n)\right)$$

if and only if there exists an $a \in \omega$ such that

$$\forall x_{m-2}\exists x_{m-3}\forall x_{m-4}\exists x_{m-5}\cdots Qx_0(R(x_0, \ldots, x_{m-2}, a, a_m, \ldots, a_n))$$

if and only if

$$\exists x_{m-1} \forall x_{m-2} \exists x_{m-3} \forall x_{m-4} \cdots Q x_0 (R(x_0, \ldots, x_{m-1}, a_m, \ldots, a_n)).$$

$\square$

2.2.5. Corollary. *Let A be an infinite and coinfinite Σ_n^0 set. There exists a computable presentation M of $\mathcal{M}$ such that $U^M \equiv_m A$.*

Proof. If n is even then let $Q = \forall$; otherwise, let $Q = \exists$. Let R be a computable relation such that

$$\exists x_{n-1} \forall x_{n-2} \exists x_{n-3} \forall x_{n-4} \cdots Q x_0 (R(x_0, \ldots, x_{n-1}, x)) \Leftrightarrow x \in A.$$

Let T_x^R and E_x^R, $x \in \omega$, be as in Lemma 2.2.4 and define

$$|M| = \{0\} \cup \left\{ \langle x, k \rangle + 1 \mid x \in \omega,\ k \in \left| T_x^R \right| \right\}$$

and

$$E^M = \{(0, \langle x, 0 \rangle + 1) \mid x \in \omega\} \cup \left\{ (\langle x, k \rangle + 1, \langle x, l \rangle + 1) \mid x \in \omega,\ E_x^R(k, l) \right\}.$$

Since the T_x are uniformly computable, M is computable, and since A is infinite and coinfinite, there are infinitely many copies of each of $\mathcal{N}_n$ and $\mathcal{P}_n$ attached to 0 in M, and hence M is isomorphic to $\mathcal{M}$. So M is a computable presentation of $\mathcal{M}$.

Furthermore, $U^M(y)$ if and only if y has height 1 and is the root node of a copy of $\mathcal{P}_n$ if and only if y is the root node of a copy of some T_x^R such that

$$\exists x_{n-1} \forall x_{n-2} \exists x_{n-3} \forall x_{n-4} \cdots Q x_0 (R(x_0, \ldots, x_{n-1}, x))$$

if and only if, for some $x \in \omega$, $y = \langle x, 0 \rangle + 1$ and $x \in A$. $\square$

The theorem follows from Corollaries 2.2.3 and 2.2.5. ■

Remark. Notice that, in the above proof, we have in fact shown that $\mathrm{DgSp}^{\mathrm{m}}_{\mathcal{M}}(U)$ consists of all Σ^0_n m-degrees other than the m-degrees of $\emptyset$ and ω.

By the first observation in the previous section, we can replace Σ^0_n by Π^0_n in the statement of Theorem 2.2.1. The next result shows that we can also replace Σ^0_n by Δ^0_n.

2.2.6. Theorem. *Let $n > 0$. There exists an intrinsically Δ^0_n relation V on a computably presentable structure $\mathcal{N}$ such that $\mathrm{DgSp}_{\mathcal{N}}(V)$ consists of all Δ^0_n degrees.*

Proof. Let $\mathcal{M}$ and U be as in the proof of Theorem 2.2.1. Let $\mathcal{M}_0 = \langle |\mathcal{M}_0|, E_0\rangle$ and $\mathcal{M}_1 = \langle |\mathcal{M}_1|, E_1\rangle$ be copies of $\mathcal{M}$. Let U_i be the copy of U in $\mathcal{M}_i$. Let S_i be the set of all nodes of $\mathcal{M}_i$ of height 1 and let $\widehat{U}_i = S_i - U_i$. Note that U_i is intrinsically Σ^0_n and $\widehat{U}_i$ is intrinsically Π^0_n. Let $a_{i,0}, a_{i,1}, \ldots$ be the elements of U_i and let $b_{i,0}, b_{i,1}, \ldots$ be the elements of $\widehat{U}_i$.

Define $\mathcal{N} = \langle |\mathcal{N}|, E, C_0, C_1\rangle$ by

$$|\mathcal{N}| = |\mathcal{M}_0| \cup |\mathcal{M}_1| \cup \{c_0, c_1, \ldots\} \cup \{d_0, d_1, \ldots\},$$

$$\begin{aligned} E(x,y) \Leftrightarrow (x, y \in |\mathcal{M}_i| \wedge E_i(x,y)) \vee (x = c_k \wedge (y = a_{0,k} \vee y = b_{1,k})) \vee \\ (x = d_k \wedge (y = b_{0,k} \vee y = a_{1,k})), \end{aligned}$$

and

$$C_i(x) \Leftrightarrow x \in |\mathcal{M}_i|.$$

Let $V = \{c_0, c_1, \ldots\}$.

Since V can be defined both as

$$\{x \in |\mathcal{N}| \mid \neg C_0(x) \wedge \neg C_1(x) \wedge \exists y \in U_0(E(x,y))\}$$

and as

$$\{x \in |\mathcal{N}| \mid \neg C_0(x) \wedge \neg C_1(x) \wedge \forall y(E(x,y) \rightarrow y \in \widehat{U}_1)\},$$

V is intrinsically Δ^0_n.

Given an infinite and coinfinite Δ^0_n set A, we can use the construction in the proof of Theorem 2.2.1 to build computable presentations M_0 and M_1 of $\mathcal{M}$ satisfying the following conditions.

1. $|M_0| \cap |M_1| = \emptyset$.
2. $|M_0| \cup |M_1|$ is coinfinite.
3. For some computable listing $y_{0,0}, y_{0,1}, \ldots$ of the nodes of M_0 of height 1, $U^{M_0}(y_{0,k}) \Leftrightarrow k \in A$.
4. For some computable listing $y_{1,0}, y_{1,1} \ldots$ of the nodes of M_1 of height 1, $U^{M_1}(y_{1,k}) \Leftrightarrow k \notin A$.

Now build a computable presentation N of $\mathcal{N}$ as follows. Let $x_0 < x_1 < \cdots$ be the elements of $\omega - (|M_0| \cup |M_1|)$. Let $|N| = \omega$, let $C_i^N = |M_i|$, and let E^N be the union of the edge relations of M_0 and M_1 and the set $\{(x_k, y_{i,k}) \mid i < 1,\ k \in \omega\}$.

It is easy to check that N is in fact a computable presentation of $\mathcal{N}$. Furthermore, if $x \notin \{x_0, x_1, \ldots\}$ then $\neg V^N(x)$, while $V^N(x_k) \Leftrightarrow U_0^N(y_{0,k}) \Leftrightarrow k \in A$, and hence $V^N \equiv_{\mathrm{m}} A$. $\square$

It is worth noting that the results of this section stand in contrast to the following theorem, proved independently by Ash, Cholak, and Knight [1] and Harizanov [21], thus illustrating the potential differences between the general case, in which we are trying to realize certain sets of degrees as degree spectra of relations on computable structures with no additional restrictions, and cases in which we impose extra conditions on some aspect of this realization. (See Chapter 6 for more on this theme.)

2.2.7. Theorem (Ash, Cholak, and Knight; Harizanov). *Let U be a relation on the domain of a computable structure $\mathcal{A}$. Suppose that for each Δ^0_3 set C there is an isomorphism f from $\mathcal{A}$ to a computable structure $\mathcal{B}$ such that $f \leqslant_T C$ and $C \leqslant_T f(U)$. Then for any set C there is an isomorphism f from $\mathcal{A}$ to a computable model $\mathcal{B}$ such that $f \leqslant_T C$ and $C \leqslant_T f(U)$. In particular,* $\mathrm{DgSp}_{\mathcal{A}}(U)$ *contains every degree.*

2.3 Degrees Below a Given C.E. Degree

As mentioned above, it is not hard to modify the example given in the previous section of an intrinsically c.e. relation on a computable structure whose degree spectrum consists of all c.e. degrees to get, for each $n > 0$, an intrinsically n-c.e. relation on a computable structure whose degree spectrum consists of all n-c.e. degrees. A little more work can get us a similar result with α-c.e. in place of n-c.e. for any computable ordinal α. (See Definition 2.3.9 for a definition of α-c.e. sets and degrees.)

In this section, we show that, in fact, for any c.e. degree $\mathbf{a}$ and any computable ordinal α, there is an intrinsically α-c.e. relation on a computable structure whose degree spectrum consists of all α-c.e. degrees less than or equal to $\mathbf{a}$. We begin with a theorem that has a similar but simpler proof.

2.3.1. Theorem. *Let* $\mathbf{a}$ *be a c.e. degree. There exists a relation* U *on the domain of a computably presentable structure* $\mathcal{M}$ *such that* $\mathrm{DgSp}_{\mathcal{M}}(U)$ *consists of all degrees less than or equal to* $\mathbf{a}$*.*

Proof. Let A be a c.e. set in $\mathbf{a}$. Let $\sigma_0, \sigma_1, \ldots$ be a computable list of all finite binary strings.

The structure $\mathcal{M}$ will be a directed graph. We begin by defining our basic building blocks.

2.3.2. 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_0 and an edge from x_i to x_{i+1} for each $i \leqslant n+1$. We call x_0 the *top* of $[n]$.

Figure 2.2 shows $[2]$ as an example.

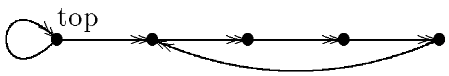


Figure 2.2: $[2]$

2.3.3. Definition. Let $\vec{m} = (m_0, m_1, \ldots, m_k) \in \omega^{k+1}$ and $S \subseteq \{0, 1, \ldots, k\}$. The directed graph $[\vec{m}, S]$ consists of the following nodes and edges.

1. $k+1$ many nodes $x_0, x_1, x_2, \ldots, x_k$ with an edge from x_i to x_{i+1} for each $i < k$.

2. For each $i \in S$, a copy of $[2m_i + 1]$ with x_i as its top.

3. For each $i \in \{0, 1, \ldots, k\} - S$, a copy of $[2m_i]$ with x_i as its top.

We call x_0 the *principal node* of $[\vec{m}, S]$. The *height* of $[\vec{m}, S]$ is defined to be $|\vec{m}|$ and its *length* is defined to be $\max\{|\sigma_m| \mid m \in \vec{m}\}$.

Figure 2.3 shows $[(1, 2, 3), \{2\}]$ as an example.

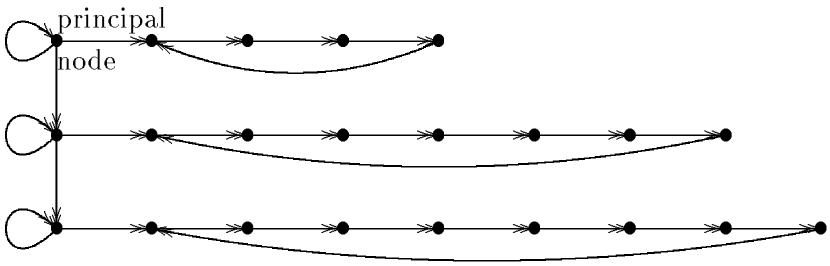


Figure 2.3: $[(1, 2, 3), \{2\}]$

The idea behind Definition 2.3.3 is that the $[\vec{m}, S]$ can be used to represent computations in which we are computably approximating a Δ^0_2 oracle, with the strings σ_m, $m \in \vec{m}$, representing initial segments of approximations of the oracle and the elements of S representing stages at which the computation changes its mind about its output at a particular input. Of course, we are only interested in the case in which the oracle is A. This leads to the following definition.

2.3.4. Definition. We say that $[\vec{m}, S]$, $\vec{m} = (m_0, m_1, \ldots, m_k)$, $S \subseteq \{0, 1, \ldots, k\}$, is *$A$-acceptable* if it satisfies both of the following conditions.

1. $A \upharpoonright |\sigma_{m_k}| = \sigma_{m_k}$.

2. If $i < k$ then $\sigma_{m_i} \neq A \upharpoonright |\sigma_{m_i}|$ and, for every $j < |\sigma_{m_i}|$, $\sigma_{m_i}(j) = 1 \Rightarrow A(j) = 1$.

We define $A[s]$-acceptability analogously.

Note that, if we think of an A-acceptable $[\vec{m}, S]$ as an approximation of some computation relative to A in the manner described above then condition 2 in Definition 2.3.4 makes sense because A is c.e.. This condition is important for two reasons. As we will see, together with condition 1 it ensures that, given a copy of an A-acceptable $[\vec{m}, S]$ in some computable graph, we can A-computably determine $\vec{m}$ and S. Furthermore, it guarantees that if $[\vec{m}, S]$ is $A[s]$-acceptable, $\vec{n}$ is a proper initial segment of $\vec{m}$, and $T \subseteq \{0, \dots, |\vec{n}| - 1\}$, then $[\vec{n}, T]$ is not $A[t]$-acceptable for any $t \geqslant s$, and hence is not A-acceptable.

We now define $\mathcal{M}$ and U.

2.3.5. Definition. Let $\mathcal{M}'$ consist of the disjoint union of infinitely many copies of each A-acceptable $[\vec{m}, S]$, $\vec{m} \in \omega^{k+1}$, $S \subseteq \{0, 1, \dots, k\}$, $k \in \omega$. Let T be the set of principal nodes of these copies.

Let $\mathcal{M}$ consist of $\mathcal{M}'$ and one additional *root node* x, with an edge from x to each element of T. We call the connected components of $\mathcal{M}'$ the *components* of $\mathcal{M}$. For any computable presentation M of $\mathcal{M}$, we call the image of x in M the root node of M.

Let U be the set of all elements of T that are principal nodes of components of $\mathcal{M}$ of the form $[\vec{m}, S]$, $|S|$ odd.

Let $[\vec{m}, S]$ have length l. The following facts follow easily from the definitions.

1. For any $s \in \omega$ such that $A[s] \upharpoonright l = A \upharpoonright l$, $[\vec{m}, S]$ is A-acceptable if and only if it is $A[s]$-acceptable.

2. If $[\vec{m}, S]$ is $A[s]$-acceptable, $A[s+1] \upharpoonright l \neq A[s] \upharpoonright l$, and m is such that $\sigma_m = A[s+1] \upharpoonright l$, then $[\vec{m}^\frown m, S]$ is $A[s+1]$-acceptable.

3. For any $m \in \omega$ and $T = S$ or $T = S \cup \{|\vec{m}|\}$, $[\vec{m}, S]$ can be extended to $[\vec{m}^\frown m, T]$ by adding new nodes and edges.

We now need to show that U and $\mathcal{M}$ have the desired properties. We begin by showing that every degree in $\mathrm{DgSp}_{\mathcal{M}}(U)$ is less than or equal to $\mathbf{a}$.

2.3.6. Lemma. *If M is a computable presentation of $\mathcal{M}$ then $U^M \leqslant_T A$.*

Proof. Let T be set of all nodes y of M such that there is an edge from y to itself. Let $y \in T$. Then y is the top of a copy of $[k]$ for some $k \in \omega$. Let m be such that $k = 2m$ or $k = 2m + 1$. Define $\sigma(y) = \sigma_m$ and $c(y) = k - 2m$. Note that T is computable, and so are the maps taking $y \in T$ to $\sigma(y)$ and $c(y)$.

To A-computably determine whether $x \in U^M$, we can proceed as follows. First, check whether there is an edge from the root node of M to x. If not then $x \notin U^M$. Otherwise, x is the principal node of a copy of some $[\vec{m}, S]$.

By the definition of $\mathcal{M}$, there is a unique list $x_0, \dots, x_n$ of elements of T such that $x = x_0$, for all $i < n$ there is an edge from x_i to x_{i+1}, and $\sigma(x_n) = A \upharpoonright |\sigma(x_n)|$. Clearly, we can A-computably find $x_0, \dots, x_n$, and hence A-computably determine $c = \sum_{i=0}^{n} c(x_i)$. By the definition of U, $x \in U^M$ if and only if c is odd. $\square$

Now, given a set $B = \Phi_e(A)$, we need to build a computable presentation M of $\mathcal{M}$ such that $U^M \equiv_{\mathrm{T}} B$. (In fact, we will build M so that $U^M \equiv_{\mathrm{m}} B$.) We take advantage of the fact that $\mathcal{M}$ contains infinitely many copies of each of its components and proceed as follows. We first construct a computable presentation N of $\mathcal{M}$ such that U^N is computable. We then add to this presentation A-acceptable components C_n, $n \in \omega$, such that the principal node of C_n is in U^M if and only if $n \in B$.

At each stage $s+1$ in the construction of M, we will have approximations $C_n[s+1]$ for each n such that $\Phi_e(A;n)[t]\downarrow$ for some $t \leqslant s$. Each such $C_n[s+1]$ will be a copy of some $[\vec{m}, S]$ such that, for the last element m of $\vec{m}$ and the largest $t \leqslant s$ such that $\Phi_e(A;n)[t]\downarrow$, $\sigma_m = A[t] \restriction \varphi_e(A;n)[t]$ and $|S| \equiv \Phi_e(A;n)[t] \bmod 2$.

Every time the computation $\Phi_e(A;n)$ changes, we change the approximation of C_n to reflect this. Since $\Phi_e(A;n)$ is total, this will guarantee that $C_n = \lim_s C_n[s]$ is A-acceptable and is a copy of some $[\vec{m}, S]$ such that $|S| \equiv \Phi_e(A;n) \bmod 2$.

2.3.7. Lemma. *There exists a computable presentation N of $\mathcal{M}$ such that U^N is computable.*

Proof. We build N in stages. By the beginning of each stage $s+1$, we will have built components $C_0[s], \ldots, C_{k_s-1}[s]$ for some $k_s \in \omega$, where each $C_i[s]$ will be a copy of some $A[s]$-acceptable $\left[\vec{m}_i[s], S_i\right]$. For each i, $\vec{m}_i[s]$ will have a limit $\vec{m}_i$, and thus $C_i[s]$ will have a limit C_i.

stage 0. Choose 0 as the root node of N. Let $k_0 = 0$.

stage $s+1$. We break the stage up into two phases.

1. Define k_{s+1}, $\vec{m}_i[s+1]$, and S_i, $k_s \leqslant i < k_{s+1}$, so that the set $\{[\vec{m}_i[s+1], S_i] \mid k_s \leqslant i < k_{s+1}\}$ contains every $A[s+1]$-acceptable $[\vec{m}, S]$ whose height and length are less than or equal to s. For each $k_s \leqslant i < k_{s+1}$, build a new copy $C_i[s+1]$ of $[\vec{m}_i[s+1], S_i]$ using fresh large numbers and add an edge from 0 to the principal node of $C_i[s+1]$.

2. For each $C_i[s]$, $i < k_s$, if $C_i[s]$ is not $A[s+1]$-acceptable then proceed as follows. Let m be such that $\sigma_m = A[s+1] \upharpoonright l$, where l is the length of $C_i[s]$. Let $\vec{m}_i[s+1] = \vec{m}_i[s]^\frown m$. Extend $C_i[s]$ to a copy $C_i[s+1]$ of $[\vec{m}_i[s+1], S_i]$ using fresh large numbers. Note that, since $C_i[s]$ is $A[s]$-acceptable but not $C_i[s+1]$-acceptable, $C_i[s+1]$ is $A[s+1]$-acceptable.

   On the other hand, if $C_i[s]$ is $A[s+1]$-acceptable then let $\vec{m}_i[s+1] = \vec{m}_i[s]$ and $C_i[s+1] = C_i[s]$.

Let $i \in \omega$ and let s be such that $C_i[s]$ is defined. It is easy to check that, for any $t \geqslant s$, $C_i[t]$ is $A[t]$ acceptable and has the same length as $C_i[s]$. Thus there exists a $t \geqslant s$ such that, for all $u \geqslant t$, $C_i[u] = C_i[t]$, and hence $C_i = \lim_u C_i[u]$ is well-defined and A-acceptable. So every C_i is a copy of some component of $\mathcal{M}$.

Now suppose that $[\vec{m}, S]$ is A-acceptable and has length l and let $s \geqslant l$ be such that $A[s+1] \upharpoonright l = A \upharpoonright l$. Then for every $t \geqslant s$ there exists $k_{t-1} \leqslant i < k_t$ such that $C_i[t]$ is a copy of $[\vec{m}, S]$ and, by the choice of s, $C_i = C_i[t]$. Thus each component of $\mathcal{M}$ has infinitely many copies in N. Together with the result of the previous paragraph, this shows that N is a computable presentation of $\mathcal{M}$.

To determine whether $x \in U^N$, all we need to do is to look for a stage s in the construction during which numbers greater than x are used. Then $x \in U^N$ if and only if it is the principal node of some $\left[\vec{m}_i[s], S_i\right]$, $i < k_s$, with $|S_i|$ odd. $\square$

2.3.8. Lemma. *Let $B \leqslant_T A$. There is a computable presentation $M = \langle |M|, E\rangle$ of $\mathcal{M}$ such that $U^M \equiv_m B$.*

Proof. Let e be such that $\Phi_e(A) = B$. By Lemma 2.3.7, there is a computable presentation N of $\mathcal{M}$ such that U^N is computable. We can assume that $D = \omega - |N|$ is infinite.

We extend N to another computable presentation M of $\mathcal{M}$ in stages. When we make use of fresh numbers in the construction, we take them from D in order. We adopt the conventions that $n \leqslant s \Rightarrow \Phi_e(A; n)[s]\uparrow$ and $A[s+1] \restriction \varphi_e(A; n)[s] \neq A[s] \restriction \varphi_e(A; n)[s] \Rightarrow \Phi_e(A; n)[s+1]\uparrow$.

At the beginning of stage $s+1$, we have copies $C_n[s]$ of graphs $\left[\vec{m}_n[s], S_n[s]\right]$ for each $n < s$ such that $\Phi_e(A; n)[t]\downarrow$ for some $t < s$. For each $n \leqslant s$, we proceed as follows.

If $\Phi_e(A; n)[s]\downarrow$ and $\Phi_e(A; n)[t]\uparrow$ for all $t < s$ then let m be such that $\sigma_m = A[s] \restriction \varphi_e(A; n)[s]$ and let $\vec{m}_n[s+1] = (m)$. If $\Phi_e(A; n)[s] = 0$ then let $S_n[s+1] = \emptyset$; otherwise, let $S_n[s+1] = \{0\}$. Let $C_n[s+1]$ be a new copy of $\left[\vec{m}_n[s+1], S_n[s+1]\right]$, formed using fresh numbers in D, and add an edge from the root node of N to the principal node of $C_n[s+1]$.

If $C_n[s]$ is defined, $\Phi_e(A; n)[s]\downarrow$, and $\Phi_e(A; n)[s-1]\uparrow$, then let m be such that $\sigma_m = A[s] \restriction \varphi_e(A; n)[s]$ and let $\vec{m}_n[s+1] = \vec{m}_n[s]^\frown m$. If $\Phi_e(A; n)[s] \equiv |S_n[s]| \bmod 2$ then let $S_n[s+1] = S_n[s]$; otherwise, let $S_n[s+1] = S_n[s] \cup \{|\vec{m}_n[s]|\}$. Extend $C_n[s]$

to a copy $C_n[s+1]$ of $\left[\vec{m}_n[s+1], S_n[s+1]\right]$, using fresh numbers in D.

If neither of the previous two cases holds then let $\vec{m}_n[s+1] = \vec{m}_n[s]$, $S_n[s+1] = S_n[s]$, and $C_n[s+1] = C_n[s]$.

It is easy to check that M is a computable presentation of $\mathcal{M}$. In particular, the following facts hold.

1. Whenever $C_n[s]$ changes, it is only to reflect the fact that a number has entered A below the use of the computation $\Phi_e(A; n)$.

2. $C_n[s]$ will necessarily change to reflect the last change in this use.

Thus each $C_n[s]$ comes to a limit C_n, and it is then a copy of an A-acceptable $[\vec{m}_n, S_n]$. Let x_n be the principal node of C_n.

We wish to show that $U^M \equiv_{\mathrm{m}} B$. By our choice of N, $U^M \cap |N|$ is computable, so it suffices to show that $n \in B \Leftrightarrow x_n \in U^M$, that is, that $\forall n \in \omega(B(n) \equiv |S_n| \bmod 2)$.

Fix n and let s be the least number such that $\Phi_e(A; n)[t] \downarrow = \Phi_e(A; n)$ for all $t \geqslant s$. By the minimality of s, $\Phi_e(A; n)[s-1] \uparrow$, and hence one of the first two cases in the description of the stage $s+1$ action of the construction of M holds for n, so that $B(n) = \Phi_e(A; n) = \Phi_e(A; n)[s] \equiv |S_n[s+1]| \bmod 2$. Furthermore, neither of these cases ever holds after stage $s+1$, so that $S_n = S_n[s+1]$. Thus $B(n) \equiv |S_n| \bmod 2$. $\square$

The theorem follows from Lemmas 2.3.6 and 2.3.8. $\blacksquare$

Remark. It is easy to give an example of a relation on a computable structure whose degree spectrum contains all degrees, and for any degree **a**, it is equally easy to give an example of a relation on a computable structure whose degree spectrum is $\{\mathbf{a}\}$. Thus, realizing all degrees above a given (not necessarily c.e.) degree as the degree spectrum of a relation on a computable structure is an easy application of Proposition 2.1.1.

Similarly, combining Theorem 2.3.1 with Proposition 2.1.1, we see that if $\mathbf{a} < \mathbf{b}$ are degrees and $\mathbf{b}$ is c.e. then there exists a relation U on the domain of a computably presentable structure $\mathcal{M}$ such that $\mathrm{DgSp}_{\mathcal{M}}(U)$ consists of all degrees in the interval $[\mathbf{a}, \mathbf{b}]$.

In general, the various examples of possible degree spectra of relations given in this dissertation can often be combined to yield further examples by using Proposition 2.1.1, and we will make no explicit mention of this fact below.

We now show how to modify the proof of Theorem 2.3.1 in order to realize all the α-c.e. degrees below a given c.e. degree as the degree spectrum of a relation on a computable structure.

The definition of α-c.e. sets and degrees depends on the choice of ordinal notation system; see [6] for details. When we talk about α-c.e. sets and degrees, where α is a computable ordinal, we assume that we have fixed a univalent, computably related ordinal notation system with a notation for α (and hence for all ordinals less than α) and let $\ulcorner\beta\urcorner$ denote the unique notation for $\beta \leqslant \alpha$ in this system.

It is slightly cumbersome to give a definition of α-c.e. sets that works for both $\alpha < \omega$ (where we want to agree with the definition of n-c.e. sets, $n \in \omega$, given by the difference hierarchy) and $\alpha \geqslant \omega$. The following (slightly nonstandard) definition works well for our purposes, and is easily seen to be equivalent to standard definitions of n-c.e. and α-c.e. sets and degrees (as in [6]).

2.3.9. Definition. Let α be a computable ordinal and assume we have fixed a univalent, computably related ordinal notation system with a notation for α. Let $\ulcorner\beta\urcorner$ denote the unique notation for $\beta \leqslant \alpha$ in this system.

A set A is *α-c.e.* if there exists a partial computable binary function Ψ satisfying the following conditions for all $x \in \omega$. (We will say that Ψ witnesses the fact that A is α-c.e..)

1. $\Psi(\ulcorner\alpha\urcorner, x)\downarrow = 0$.

2. If $\alpha \geqslant \omega$ then there exists a $\beta < \alpha$ such that $\Psi(\ulcorner\beta\urcorner, x)\downarrow$.

3. For the least $\beta \leqslant \alpha$ such that $\Psi(\ulcorner\beta\urcorner, x)\downarrow$, $\Psi(\ulcorner\beta\urcorner, x) = A(x)$.

A degree is α-c.e. if it contains an α-c.e. set.

2.3.10. Theorem. *Let α be a computable ordinal and let* $\mathbf{a}$ *be a c.e. degree. There exists an intrinsically α-c.e. relation U on the domain of a computably presentable structure $\mathcal{M}$ such that* $\mathrm{DgSp}_{\mathcal{M}}(U)$ *consists of all α-c.e. degrees less than or equal to* $\mathbf{a}$.

Proof. This proof is similar to that of Theorem 2.3.1; we give the necessary changes. Unless otherwise noted, we use the same notation and conventions as in that proof.

Let A be a c.e. set in $\mathbf{a}$. In the proof of Theorem 2.3.1, we had graphs $[\vec{m}, S]$ that could be used to represent computations in which we computably approximate the oracle A. In this proof, we also want to be able to represent the nonincreasing sequences of ordinals less than or equal to α that can be associated to computable approximations of α-c.e. sets. This leads to the following definitions.

2.3.11. Definition. Let $\vec{m} = (m_0, m_1, \dots, m_k) \in \omega^{k+1}$, let $S \subseteq \{0, 1, \dots, k\}$, and let $\vec{\gamma} = (\gamma_0, \gamma_1, \dots, \gamma_k) \in (\alpha+1)^{k+1}$ be nonincreasing. The directed graph $[\vec{m}, S, \vec{\gamma}]$ consists of the following nodes and edges.

1. $k+1$ many nodes $x_0, x_1, x_2, \dots, x_k$ with an edge from x_i to x_{i+1} for each $i < k$.

2. For each $i \in S$, a copy of $[\langle m_i, \ulcorner\gamma_i\urcorner, 1\rangle]$ with x_i as its top.

3. For each $i \in \{0, 1, \dots, k\} - S$, a copy of $[\langle m_i, \ulcorner\gamma_i\urcorner, 0\rangle]$ with x_i as its top.

As before, we call x_0 the principal node of $[\vec{m}, S, \vec{\gamma}]$. The *height* of $[\vec{m}, S, \vec{\gamma}]$ is defined to be $|\vec{m}|$, its *length* is defined to be $\max\{|\sigma_m| \mid m \in \vec{m}\}$, and its *range* is defined to be $\max\{\ulcorner\gamma_i\urcorner \mid i \leqslant k\}$.

2.3.12. Definition. Let $\vec{m} = (m_0, m_1, \dots, m_k) \in \omega^{k+1}$, let $S \subseteq \{0, 1, \dots, k\}$, and let $\vec{\gamma} = (\gamma_0, \gamma_1, \dots, \gamma_k) \in (\alpha+1)^{k+1}$ be nonincreasing. We say that $[\vec{m}, S, \vec{\gamma}]$ is *A-acceptable* if it satisfies all of the following conditions.

1. $A \upharpoonright |\sigma_{m_k}| = \sigma_{m_k}$.

2. For $i < k$, $\sigma_{m_i} \neq A \upharpoonright |\sigma_{m_i}|$ and, for every $j < |\sigma_{m_i}|$, $\sigma_{m_i}(j) = 1 \Rightarrow A(j) = 1$.

3. For $i > 0$, if $i \in S$ then $\gamma_i \neq \gamma_{i-1}$.

4. If $\alpha \geqslant \omega$ or $0 \in S$ then $\gamma_0 < \alpha$.

We define $A[s]$-acceptability analogously.

We define $\mathcal{M}$ and U much as before.

2.3.13. Definition. Let $\mathcal{M}'$ consist of the disjoint union of infinitely many copies of each A-acceptable $[\vec{m}, S, \vec{\gamma}]$, where, for some $k \in \omega$, $\vec{m} \in \omega^{k+1}$, $S \subseteq \{0, 1, \ldots, k\}$, and $\vec{\gamma} = (\gamma_0, \gamma_1, \ldots, \gamma_k) \in (\alpha+1)^{k+1}$ is nonincreasing. Let T be the set of principal nodes of these copies.

Let $\mathcal{M}$ consist of $\mathcal{M}'$ and one additional root node x, with an edge from x to each element of T.

Let U be the set of all elements of T that are principal nodes of connected components of $\mathcal{M}'$ of the form $[\vec{m}, S, \vec{\gamma}]$, $|S|$ odd.

The following lemma has essentially the same proof as Lemma 2.3.6.

2.3.14. Lemma. *If M is a computable presentation of $\mathcal{M}$ then $U^M \leqslant_T A$.*

We also need to check that U is intrinsically α-c.e..

2.3.15. Lemma. *If M is a computable presentation of $\mathcal{M}$ then U^M is α-c.e..*

Proof. Let S be the set of all nodes y of M such that there is an edge from the root node of M to y. Let T be set of all nodes y of M such that there is an edge from y to itself. Let $y \in T$. Then y is the top of a copy of $[k]$ for some $k \in \omega$. Let m, $\beta \leqslant \alpha$, and $i \leqslant 1$ be such that $k = \langle m, \ulcorner \beta \urcorner, i \rangle$. Define $\sigma(y) = \sigma_m$, $\beta(y) = \beta$, and $c(y) = i$. Note that T is computable, and so are the maps taking $y \in T$ to $\sigma(y)$, $\beta(y)$, and $c(y)$.

Define the partial computable binary function Ψ as follows.

stage 0. For all $x \in \omega$, $\Psi(\ulcorner\alpha\urcorner, x) = 0$. If $x \notin S$ then $\Psi(\ulcorner\beta\urcorner, x) = 0$ for all $\beta < \alpha$. If $x \in S$ then $\Psi(\ulcorner\beta(x)\urcorner, x) = c(x)$.

stage $s+1$. For all $x \in S$, proceed as follows. Let $x_0, \ldots, x_n$ be the longest chain of elements of $T \upharpoonright s \cup \{x\}$ such that $x = x_0$ and for all $i < n$ there is an edge from x_i to x_{i+1}. Let $c = |\{i < n \mid c(x_i) \neq c(x_{i+1})\}|$. If $\Psi(\ulcorner\beta(x_n)\urcorner, x)$ has not yet been defined then $\Psi(\ulcorner\beta(x_n)\urcorner, x) = c$.

It is not hard to check that Ψ witnesses that U^M is α-c.e. in the sense of Definition 2.3.9. $\square$

2.3.16. Lemma. *There exists a computable presentation N of $\mathcal{M}$ such that U^N is computable.*

Proof. We build N in stages in much the same way as before.

stage 0. Choose 0 as the root node of N. Let $k_0 = 0$.

stage $s+1$. We break the stage up into two phases.

1. Define k_{s+1}, $\vec{m}_i[s+1]$, S_i, and $\vec{\gamma}_i[s+1]$, $k_s \leqslant i < k_{s+1}$, so that the set $\{[\vec{m}_i[s+1], S_i, \vec{\gamma}_i[s+1]] \mid k_s \leqslant i < k_{s+1}\}$ contains every $A[s+1]$-acceptable $[\vec{m}, S, \vec{\gamma}]$ of height, length, and range less than or equal to s. For each $k_s \leqslant i < k_{s+1}$, build a new copy $C_i[s+1]$ of $[\vec{m}_i[s+1], S_i, \vec{\gamma}_i[s+1]]$ using fresh large numbers and add an edge from 0 to the principal node of $C_i[s+1]$.

2. For each $C_i[s]$, $i < k_s$, if $C_i[s]$ is not $A[s+1]$-acceptable then proceed as follows. Let m be such that $\sigma_m = A[s+1] \upharpoonright l$, where l is the length of $C_i[s]$. Let k be

the height of $C_i[s]$. Let $\vec{m}_i[s+1] = \vec{m}_i[s]^\frown m$ and $\vec{\gamma}_i[s+1] = \vec{\gamma}_i[s]^\frown\vec{\gamma}_i(k-1)$. Extend $C_i[s]$ to a copy $C_i[s+1]$ of $\left[\vec{m}_i[s+1], S_i, \vec{\gamma}_i[s+1]\right]$ using fresh large numbers.

On the other hand, if $C_i[s]$ is $A[s+1]$-acceptable then let $\vec{m}_i[s+1] = \vec{m}_i[s]$, $\vec{\gamma}_i[s+1] = \vec{\gamma}_i[s]$, and $C_i[s+1] = C_i[s]$.

It is easy to check, as before, that, for every $i \in \omega$, $C_i = \lim_s C_i[s]$ is well-defined and A-acceptable. So every C_i is a copy of some component of $\mathcal{M}$. Moreover, by the same argument as in the proof of Lemma 2.3.7, each component of $\mathcal{M}$ has infinitely many copies in N. Thus N is a computable presentation of $\mathcal{M}$.

As before, to determine whether $x \in U^N$, all we need to do is to look for a stage s in the construction during which numbers greater than x are used. Then $x \in U^N$ if and only if it is the principal node of some $\left[\vec{m}_i[s], S_i, \vec{\gamma}_i[s]\right]$, $i < k_s$, with $|S_i|$ odd. $\square$

2.3.17. Lemma. *Let $B \leqslant_T A$ be α-c.e.. There exists a computable presentation $M = \langle |M|, E\rangle$ of $\mathcal{M}$ such that $U^M \equiv_m B$.*

Proof. This proof is much the same as the proof of Lemma 2.3.8; we give the necessary changes.

Let Ψ be a partial computable binary function witnessing the fact that B is α-c.e.. It is not hard to see that there exists an $e \in \omega$ with the following properties.

1. $\Phi_e(A) = B$.

2. If $\Phi_e(A;n)[s]\downarrow$ then, for the least $\beta \leqslant \alpha$ such that $\Psi(\ulcorner\beta\urcorner, n)[s]\downarrow$, $\Phi_e(A;n)[s] = \Psi(\ulcorner\beta\urcorner, n)$.

3. For the least number s such that $\Phi_e(A;n)[s]\downarrow$, if either $\alpha \geqslant \omega$ or $\Phi_e(A;n)[s] = 1$ then $\Psi(\ulcorner\beta\urcorner, n)[s]\downarrow$ for some $\beta < \alpha$.

By Lemma 2.3.16, there is a computable presentation N of $\mathcal{M}$ such that U^N is computable. We can assume that $D = \omega - |N|$ is infinite.

We extend N to another computable presentation M of $\mathcal{M}$ in stages. When we make use of fresh numbers in the construction, we take them from D in order.

At the beginning of stage $s+1$, we have copies $C_n[s]$ of graphs $\left[\vec{m}_n[s], S_n[s], \vec{\gamma}_n[s]\right]$ for each $n < s$ such that $\Phi_e(A;n)[t]\downarrow$ for some $t < s$. For each $n \leqslant s$, we proceed as follows.

If $\Phi_e(A;n)[s]\downarrow$ and $\Phi_e(A;n)[t]\uparrow$ for all $t < s$ then let m be such that $\sigma_m = A[s] \upharpoonright \varphi_e(A;n)[s]$ and let $\vec{m}_n[s+1] = (m)$. If $\Phi_e(A;n)[s] = 0$ then let $S_n[s+1] = \emptyset$; otherwise, let $S_n[s+1] = \{0\}$. Let $\beta \leqslant \alpha$ be the least ordinal such that $\Psi(\ulcorner\beta\urcorner, n)[s]\downarrow$ and let $\vec{\gamma}_n[s+1] = (\beta)$. Let $C_n[s+1]$ be a new copy of $\left[\vec{m}_n[s+1], S_n[s+1], \vec{\gamma}_n[s+1]\right]$, formed using fresh numbers in D, and add an edge from the root node of N to the principal node of $C_n[s+1]$.

If $C_n[s]$ is defined, $\Phi_e(A;n)[s]\downarrow$, and $\Phi_e(A;n)[s-1]\uparrow$, then let m be such that $\sigma_m = A[s] \upharpoonright \varphi_e(A;n)[s]$ and let $\vec{m}_n[s+1] = \vec{m}_n[s]^\frown m$. If $\Phi_e(A;n)[s] \equiv |S_n[s]| \bmod 2$ then let $S_n[s+1] = S_n[s]$; otherwise, let $S_n[s+1] = S_n[s] \cup \{|\vec{m}_n[s]|\}$. Let $\beta \leqslant \alpha$ be the least ordinal such that $\Psi(\ulcorner\beta\urcorner, n)[s]\downarrow$ and let $\vec{\gamma}_n[s+1] = \vec{\gamma}_n[s]^\frown\beta$. Extend $C_n[s]$ to a copy $C_n[s+1]$ of $\left[\vec{m}_n[s+1], S_n[s+1], \vec{\gamma}_n[s+1]\right]$, using fresh numbers in D.

If neither of the previous two cases holds then let $\vec{m}_n[s+1] = \vec{m}_n[s]$, $S_n[s+1] = S_n[s]$, $\vec{\gamma}_n[s+1] = \vec{\gamma}_n[s]$, and $C_n[s+1] = C_n[s]$.

It is easy to check that M is a computable presentation of $\mathcal{M}$, and the proof that $U^M \equiv_{\mathrm{m}} B$ is the same as before. □

The theorem follows from Lemmas 2.3.14, 2.3.15 and 2.3.17. ■

Suppose that we add to Definition 2.3.4 the condition that $|\sigma_{m_0}| = |\sigma_{m_1}| = \cdots = |\sigma_{m_k}|$ and define $\mathcal{M}$ and U as in Definition 2.3.5. Then, given a computable presentation M of $\mathcal{M}$, the use of the A-computable procedure given in the proof of Lemma 2.3.6 for determining whether $x \in U^M$ is a computable function of x, and hence $U^M \leqslant_{\mathrm{wtt}} A$. Furthermore, Lemma 2.3.7 can be proved as before, as can Lemma 2.3.8 for $B \leqslant_{\mathrm{wtt}} A$. (In the proof, we need to pick e so that there is a computable bound f on the use of $\Phi_e(A)$ and then adopt the convention that $\varphi_e(A;n)[s] = f(n)$ for all $n, s \in \omega$.) Similar changes can be made to the proof of Theorem 2.3.10. Thus we have the following result.

2.3.18. Theorem. *Let* $\mathbf{a}$ *be a c.e. wtt-degree. There exists a relation* U *on the domain of a computably presentable structure* $\mathcal{M}$ *such that* $\mathrm{DgSp}_{\mathcal{M}}^{wtt}(U)$ *consists of all wtt-degrees less than or equal to* $\mathbf{a}$.

Let $n \in \omega$. *There exists an intrinsically* n*-c.e. relation* U *on the domain of a computably presentable structure* $\mathcal{M}$ *such that* $\mathrm{DgSp}_{\mathcal{M}}^{wtt}(U)$ *consists of all* n*-c.e. wtt-degrees less than or equal to* $\mathbf{a}$.

Remark. The reason we restrict ourselves to n-c.e. wtt-degrees, $n \in \omega$, in the second part of the Theorem 2.3.18 is that every wtt-degree less than or equal to a c.e. wtt-degree is ω-c.e..

2.4 Easy Finite Degree Spectra

As we will see in Chapter 3, realizing sets of degrees of finite cardinality greater than 1 as degree spectra of relations on computable structures normally requires fairly complicated constructions. It is possible, however, to obtain finite degree spectra of relations as easy corollaries to two results, one in computable model theory, and the other in classical computability theory.

First of all, for any $n > 1$, the existence of a relation with a two-element degree spectrum that includes $\mathbf{0}$ follows from the existence of a rigid structure of computable dimension 2, which was shown by Goncharov [11]. (This has been noted by Harizanov (see [19]).)

Indeed, suppose that $\mathcal{A}$ is such a computable structure, assume without loss of generality that $|\mathcal{A}| = \omega$, and let R be the binary relation that holds of $x, y \in |\mathcal{A}|$ if and only if $y = x + 1$. Clearly, if $\mathcal{B}$ is a computable structure and $f : \mathcal{A} \cong \mathcal{B}$ then $\deg(f(R)) = \deg(f)$. So the fact that $\mathcal{A}$ is rigid and has computable dimension 2 implies that $\mathrm{DgSp}_{\mathcal{A}}(R) = \{\mathbf{0}, \mathbf{a}\}$ for some nonzero degree $\mathbf{a}$. It should be noted that, by a result of Goncharov [13] (see Chapter 6), $\mathbf{a}$ cannot be Δ^0_2.

It is also easy to give an example of an intrinsically d.c.e. relation on a Δ^0_2-categorical structure with a two-element degree spectrum, but one that does not include $\mathbf{0}$. As we have seen in the previous section, there exists an intrinsically d.c.e.

unary relation U on the domain of a computable directed graph $\mathcal{A}$ whose degree spectrum is the set of d.c.e. degrees.

Now let $\mathbf{d}$ be a maximal incomplete d.c.e. degree, as constructed in [5]. (That is, $\mathbf{d} \neq \mathbf{0}'$ is d.c.e. and there are no d.c.e. degrees in $(\mathbf{d}, \mathbf{0}')$.) It is easy to define a d.c.e. relation V on the domain of a computably categorical directed graph $\mathcal{B}$ whose degree spectrum is the singleton $\{\mathbf{d}\}$.

By Proposition 2.1.1, there exists an intrinsically d.c.e. relation W on the domain of a Δ^0_2-categorical computable structure $\mathcal{C}$ whose degree spectrum is

$$\{\mathbf{c} \mid \exists \mathbf{a}, \mathbf{b}(\mathbf{a} \in \mathrm{DgSp}_{\mathcal{A}}(U) \wedge \mathbf{b} \in \mathrm{DgSp}_{\mathcal{B}}(V) \wedge \mathbf{c} = \mathbf{a} \curlyvee \mathbf{b})\} =$$
$$\{\mathbf{c} \mid \mathbf{d} \leqslant \mathbf{c} \text{ and } \mathbf{c} \text{ is d.c.e.}\} = \{\mathbf{d}, \mathbf{0}'\}.$$

It is not hard to see that W can be chosen to be invariant.

The fact that $\mathcal{C}$ is Δ^0_2-categorical and W is invariant is interesting in light of the results of Chapter 6, where it is shown that no finite set of degrees containing $\mathbf{0}$ can be the degree spectrum of an invariant relation on a Δ^0_2-categorical computable structure.

Chapter 3

Finite Degree Spectra of Relations

3.1 Introduction

The Ash-Nerode type conditions mentioned in Section 1.1 usually imply that the degree spectrum of a relation is either a singleton or infinite. Indeed, for various classes of degrees, conditions have been formulated that guarantee that the degree spectrum of a relation consists of all the degrees in the given class (see [1], for example). Motivated by these considerations, as well as by Goncharov's examples [11] of structures of finite computable dimension, Harizanov and Millar suggested the study of relations with finite degree spectra.

Harizanov [20] was the first to give an example of an intrinsically Δ_2^0 relation with a two-element degree spectrum that includes $\mathbf{0}$.

3.1.1. Theorem (Harizanov). *For some Δ_2^0 but not c.e. degree $\mathbf{a}$, there is a relation U on the domain of a computable structure $\mathcal{A}$ of computable dimension 2 such* *that* $DgSp_{\mathcal{A}}(U) = \{\mathbf{0}, \mathbf{a}\}$.

Khoussainov and Shore and Goncharov [16],[25] showed the existence of an intrinsically c.e. relation with a two-element degree spectrum.

3.1.2. Theorem (Khoussainov and Shore, Goncharov). *For some c.e. degree* $\mathbf{a}$, *there is an intrinsically c.e. relation* U *on the domain of a computable structure* $\mathcal{A}$ *of computable dimension* 2 *such that* $DgSp_{\mathcal{A}}(U) = \{\mathbf{0}, \mathbf{a}\}$.

This left open the question, asked explicitly in [16], of which (c.e.) degrees can be the nonzero element of a two-element degree spectrum. In this chapter we show that every c.e. degree belongs to some two-element degree spectrum whose other element is $\mathbf{0}$. We begin by temporarily ignoring the issue of computable dimension and establishing the following result.

3.1.3. Theorem. *Let* $\mathbf{a} > \mathbf{0}$ *be a c.e. degree. There exists an intrinsically c.e. relation* U *on the domain of a computable structure* $\mathcal{A}$ *such that* $DgSp_{\mathcal{A}}(U) = \{\mathbf{0}, \mathbf{a}\}$.

The proof of this theorem, which will be given in Section 3.2, uses techniques from [25], which in turn builds on work of Goncharov [10],[11] and Cholak, Goncharov, Khoussainov, and Shore [4].

In Section 3.3, we show how to modify the proof of Theorem 3.1.3 to obtain the following result, which is also due independently to Khoussainov and Shore [24], whose proof uses a complicated modification of their proof of Theorem 3.1.2.

3.1.4. Theorem. *Let* $\mathbf{a} > \mathbf{0}$ *be a c.e. degree. There exists an intrinsically c.e. relation* U *on the domain of a computable structure* $\mathcal{A}$ *of computable dimension*

2 *such that* $DgSp_{\mathcal{A}}(U) = \{\mathbf{0}, \mathbf{a}\}$. *In addition,* $\mathcal{A}$ *can be picked so that every c.e. presentation of* $\mathcal{A}$ *is computable, which implies that* $\mathcal{A}$ *has c.e. dimension* 2.

In [25], Khoussainov and Shore also proved the following theorem.

3.1.5. Theorem (Khoussainov and Shore). *For each computable poset* $\mathcal{P}$ *there exists an intrinsically c.e. relation* U *on the domain of a computable structure* $\mathcal{A}$ *such that* $\langle DgSp_{\mathcal{A}}(U), \leqslant_T \rangle \cong \mathcal{P}$. *If* $\mathcal{P}$ *has a least element then we can pick* U *and* $\mathcal{A}$ *so that* $\mathbf{0} \in DgSp_{\mathcal{A}}(U)$.

In Section 3.4, we show how to modify the proof of Theorem 3.1.3 to establish the following extension of Theorem 3.1.5.

3.1.6. Theorem. *Let* $\{A_i\}_{i \in \omega}$ *be a uniformly c.e. (u.c.e.) collection of sets. There exists an intrinsically c.e. relation* U *on the domain of a computable structure* $\mathcal{A}$ *such that* $DgSp_{\mathcal{A}}(U) = \{\deg(A_i) \mid i \in \omega\}$.

Another way in which we can extend Theorem 3.1.3 is by broadening our focus from the c.e. degrees to larger classes of degrees. In Section 3.5, we establish the following result.

3.1.7. Theorem. *Let* $\alpha \in \omega \cup \{\omega\}$ *and let* $\mathbf{b} > \mathbf{0}$ *be an* α*-c.e. degree. There exists an intrinsically* α*-c.e. relation* V *on the domain of a computable structure* $\mathcal{B}$ *of computable dimension* 2 *such that* $DgSp_{\mathcal{B}}(V) = \{\mathbf{0}, \mathbf{b}\}$.

Remark. Since we will only be dealing with $\alpha \leqslant \omega$ in this chapter, we will not need to worry about the problems with the definition of α-c.e. mentioned in Section 2.3. See Section 3.5 for details.

The structure $\mathcal{B}$ will be an extension of the structure $\mathcal{A}$ constructed in the proof of Theorem 3.1.4 for an appropriate c.e. degree $\mathbf{a}$.

Remark. One interesting consequence of Theorem 3.1.7 is that there exists a minimal degree $\mathbf{b}$ such that $\{\mathbf{0}, \mathbf{b}\}$ is realized as the degree spectrum of a relation on a computable structure.

Theorems 3.1.6 and 3.1.7 can be conflated to produce the following results, which can be proved by combining the modifications to the proof of Theorem 3.1.3 presented in Sections 3.3, 3.4 and 3.5.

3.1.8. Theorem. *Let $\alpha \in \omega \cup \{\omega\}$ and let $\{A_i\}_{i\in\omega}$ be a uniformly α-c.e. collection of sets. There exists an intrinsically α-c.e. relation U on the domain of a computable structure $\mathcal{A}$ such that $DgSp_{\mathcal{A}}(U) = \{\deg(A_i) \mid i \in \omega\}$.*

3.1.9. Theorem. *Let $\alpha \in \omega \cup \{\omega\}$ and let $\mathbf{a}_0, \dots, \mathbf{a}_n$ be α-c.e. degrees. There exists an intrinsically α-c.e. relation U on the domain of a computable structure $\mathcal{A}$ of computable dimension $n+1$ such that $DgSp_{\mathcal{A}}(U) = \{\mathbf{a}_0, \dots, \mathbf{a}_n\}$.*

It will be clear from their proofs that Theorems 3.1.3, 3.1.4, and 3.1.7 remain true with *degree* replaced by *m-degree* and $DgSp_{\mathcal{A}}(U)$ replaced by $DgSp^m_{\mathcal{A}}(U)$. Thus, for any reducibility r weaker than m-reducibility, both theorems remain true with *degree* replaced by *r-degree* and $DgSp_{\mathcal{A}}(U)$ replaced by $DgSp^r_{\mathcal{A}}(U)$. The same holds of Theorems 3.1.6 and 3.1.8 if we require that $A_i \neq \emptyset$ and $A_i \neq \omega$ for all $i \in \omega$, and of Theorem 3.1.9 if we require that the m-degrees of $\emptyset$ and ω are not on the list $\mathbf{a}_0, \dots, \mathbf{a}_n$.

3.2 Proof of Theorem 3.1.3

In this section we prove the following theorem.

3.1.3. Theorem. *Let* $\mathbf{a} > \mathbf{0}$ *be a c.e. degree. There exists an intrinsically c.e. relation* U *on the domain of a computable structure* $\mathcal{A}$ *such that* $DgSp_{\mathcal{A}}(U) = \{\mathbf{0}, \mathbf{a}\}$.

Proof. Let A be a c.e. set that is not computable and let $a_0, a_1, \ldots$ be a computable enumeration of A. Let $A[0] = \emptyset$, $A[s+1] = \{a_0, \ldots, a_s\}$. We wish to construct computable structures $\mathcal{A}^0$ and $\mathcal{A}^1$ and unary relations U^0 and U^1 on the domains of $\mathcal{A}^0$ and $\mathcal{A}^1$, respectively, so that the following properties hold.

(3.2.1) $\mathcal{A}^0 \cong \mathcal{A}^1$ via an isomorphism that carries U^0 to U^1.

(3.2.2) $U^0 \equiv_{\mathrm{m}} A$ and U^1 is computable.

(3.2.3) If $\mathcal{G} \cong \mathcal{A}^0$ is a computable structure then the image of U^0 in $\mathcal{G}$ is either computable or m-equivalent to A.

Our structures will be directed graphs. We begin by defining our basic building blocks.

3.2.1. Definition. Let $n \in \omega$. Recall from Section 2.3 that 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 $i \leqslant n+1$. As before, we call x_0 the *top* of $[n]$. We call x_{n+2} the *coding location* of $[n]$.

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

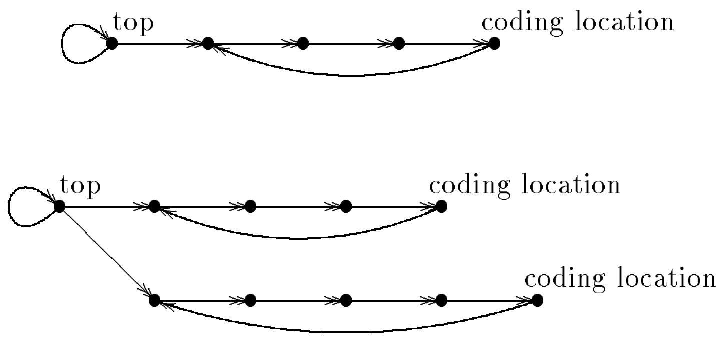


Figure 3.1: [2] and $[\{2,3\}]$

Now let us consider how we could go about satisfying (3.2.1) and (3.2.2) above. 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 $k \in \omega$. If at each stage $s+1$ we enumerate the coding location of the copy of $[3a_s]$ in $\mathcal{A}^0_0$ into U^0 then we will have ensured that $U^0 \equiv_{\mathrm{m}} A$. However, we also wish to make U^1 computable while guaranteeing that $\mathcal{A}^0 \cong \mathcal{A}^1$ via an isomorphism that carries U^0 to U^1. To describe how we can do this, we need two more definitions.

3.2.2. 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 applied to $\mathcal{G}$, so that no two operations in the sequence affect the same component of $\mathcal{G}$, we can apply all of the operations in the sequence simultaneously to $\mathcal{G}$ to get a structure extending $\mathcal{G}$. In this case we will say that we have applied the sequence of operations to $\mathcal{G}$.

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

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

We can now proceed as follows. At stage $s+1$, let X_s^i, Y_s^i, and Z_s^i be the copies in $\mathcal{A}_s^i$ of $[3a_s]$, $[3a_s+1]$, and $[3a_s+2]$, respectively. Perform $\mathbf{L}(Y_s^0, X_s^0, Z_s^0)$ on $\mathcal{A}_s^0$ to get $\mathcal{A}_{s+1}^0$ and perform $\mathbf{R}(Y_s^1, X_s^1, Z_s^1)$ on $\mathcal{A}_s^1$ to get $\mathcal{A}_{s+1}^1$. (In order to ensure that $\mathcal{A}^0$ and $\mathcal{A}^1$ are computable, the new numbers added to their domains at this stage are assumed to be greater than s.) Put the coding location of the old copy of $[3a_s]$ in

$\mathcal{A}^0_{s+1}$ (that is, the copy that was already in $\mathcal{A}^0_0$) into U^0 and put the coding location of the new copy of $[3a_s]$ in $\mathcal{A}^1_{s+1}$ into U^1.

Figure 3.2 pictures what happens on either side of the construction. For each $i = 0, 1$, the copy of $[3a_s]$ whose coding location enters U^i is underlined.

$$\begin{array}{ccc} {[3a_s+1]} & {[3a_s]} & {[3a_s+2]} \\ \downarrow & \downarrow & \downarrow \\ {[3a_s+1]\cdot[3a_s]} & \underline{[3a_s]}\cdot[3a_s+2] & {[3a_s+2]\cdot[3a_s+1]} \end{array}$$

$$\begin{array}{ccc} {[3a_s+1]} & {[3a_s]} & {[3a_s+2]} \\ \downarrow & \downarrow & \downarrow \\ {[3a_s+1]\cdot[3a_s+2]} & {[3a_s]\cdot[3a_s+1]} & {[3a_s+2]}\cdot\underline{[3a_s]} \end{array}$$

Figure 3.2: The basic coding strategy (top: $\mathcal{A}^0$ / bottom: $\mathcal{A}^1$)

Now let $\mathcal{A}^0 = \bigcup_{s\in\omega} \mathcal{A}^0_s$ and $\mathcal{A}^1 = \bigcup_{s\in\omega} \mathcal{A}^1_s$. It is easy to show, by induction using the definition of the **L**- and **R**-operations, that for each s, $\mathcal{A}^0_s \cong \mathcal{A}^1_s$ via an isomorphism that carries $U^0[s]$ to $U^1[s]$. (Here $U^i[s]$ is the set of all numbers that have entered U^i by the end of stage s.) Furthermore, whenever a component of $\mathcal{A}^i_s$ participates in an operation at stage $s+1$, so does the isomorphic component of $\mathcal{A}^{1-i}_s$. Since $\mathcal{A}^0$ and $\mathcal{A}^1$ have no infinite components, it follows that $\mathcal{A}^0 \cong \mathcal{A}^1$ via an isomorphism that carries U^0 to U^1.

Furthermore, it is still true that $U^0 \equiv_{\mathrm{m}} A$, since a number is in U^0 if and only if it is the coding location of the copy of $[3a]$ in $\mathcal{A}^0_0$ for some $a \in A$. On the other hand, any number put into U^1 at a stage $s+1$ is a new number, and is therefore greater than s. Thus U^1 is computable.

So we see that the above construction is enough to satisfy (3.2.1) and (3.2.2). We now consider how to satisfy (3.2.3). Let us begin by attempting to satisfy this property for a particular computable structure $\mathcal{G}$. That is, we want to ensure that if $\mathcal{G} \cong \mathcal{A}^0$ then the image of U^0 in $\mathcal{G}$ is either computable or m-equivalent to A. The way in which we do this is based on the following observation.

Let U be the image of U^0 in $\mathcal{G}$ and let $\mathcal{G}[s]$ denote the stage s approximation to $\mathcal{G}$. Assume that for all $s \in \omega$, $\mathcal{A}^0_s$, $\mathcal{A}^1_s$, and $\mathcal{G}[s]$ have no non-trivial automorphisms.

Suppose that at some stage s, $\mathcal{A}^0_s$ has components X^0_s, Y^0_s, Z^0_s, and S^0_s, $\mathcal{A}^1_s$ has isomorphic components X^1_s, Y^1_s, Z^1_s, and S^1_s, respectively, and $\mathcal{G}[s]$ has isomorphic components X_s, Y_s, Z_s, and S_s, respectively. Now suppose we perform $\mathbf{L}(Y^0_s, X^0_s, Z^0_s, S^0_s)$ on $\mathcal{A}^0_s$ to get $\mathcal{A}^0_{s+1}$ and perform $\mathbf{R}(Y^1_s, X^1_s, Z^1_s, S^1_s)$ on $\mathcal{A}^1_s$ to get $\mathcal{A}^1_{s+1}$. Then $\mathcal{A}^0_{s+1}$ has components isomorphic to $S^0_s \cdot Y^0_s$, $Y^0_s \cdot X^0_s$, $X^0_s \cdot Z^0_s$, and $Z^0_s \cdot S^0_s$, and these are the only components of $\mathcal{A}^0_{s+1}$ that contain copies of X^0_s, Y^0_s, Z^0_s, or S^0_s. So if X_s, Y_s, Z_s, and S_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 $\mathcal{G} \ncong A^0$.

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

Now, if the coding location of X_s^0 is put into U^0 and the coding location of the new copy of X_s^1 is put into U^1 then the coding location of the copy of X_s that is part of the component isomorphic to $X_s \cdot Z_s$ is in U. In other words, if $\mathcal{G}$ goes to the left then the coding location of X_s in $\mathcal{G}[s]$ is in U, while if $\mathcal{G}$ goes to the right then the coding location of the copy of X_s in $\mathcal{G} - \mathcal{G}[s]$ is in U. It is easy to conclude from this that if $\mathcal{G}$ goes to the left at all but finitely many stages then $U \equiv_m A$, while if $\mathcal{G}$ goes to the right at all but finitely many stages then U is computable.

So it is enough to ensure that $\mathcal{G}$ either almost always goes to the left or almost always goes to the right. This can be done by always using the same component of $\mathcal{G}$, which we will call the *special component* of $\mathcal{G}$, as S_s.

That is, we first pick some component of $\mathcal{G}$ to be its special component. Say we pick the one that extends the first copy of $[0]$ to appear in $\mathcal{G}$. (Let us assume that $0 \notin A$.) At stage 0, we define $\mathcal{A}_0^i$ as above and wait until a copy of $[0]$ is enumerated into $\mathcal{G}$. 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 let X_s^i, Y_s^i, and Z_s^i be the copies in $\mathcal{A}_s^i$ of $[3a_s]$, $[3a_s+1]$, and $[3a_s+2]$, respectively, and let S_s^i be the isomorphic copy in $\mathcal{A}_s^i$ of the special component S_s of $\mathcal{G}[s]$. We wait until copies of X_s^i, Y_s^i, and Z_s^i are enumerated into $\mathcal{G}[s]$ and then perform the same operations as before. We then wait until copies of $S_s \cdot Y_s$, $Y_s \cdot X_s$, $X_s \cdot Z_s$, and $Z_s \cdot S_s$ are enumerated into $\mathcal{G}$. Either the copy of $S_s \cdot Y_s$ or that of $Z_s \cdot S_s$ will extend S_s. Whichever one it is now becomes S_{s+1}. If $S_{s+1} \cong S_s \cdot Y_s$ then $r_{s+1} = 0$; otherwise $r_{s+1} = 1$.

The above construction ensures that if $\mathcal{G} \cong A^0$ then the special component of $\mathcal{G}$

is infinite. On the other hand, it is not hard to check that 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 $\mathcal{G} \ncong A^0$.

However, there are two problems with this construction. First of all, it is easy to check that 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, (3.2.1) no longer holds.

We solve this by re-using components in operations. The idea is roughly as follows. Instead of using four components in our operations, we use six. That is, at stage $s+1$, in addition to the components mentioned above, we pick two other components B^0_s and C^0_s of $\mathcal{A}^0_s$ and isomorphic components B^1_s and C^1_s of $\mathcal{A}^1_s$, perform $\mathbf{L}(Y^0_s, X^0_s, Z^0_s, B^0_s, S^0_s, C^0_s)$ on $\mathcal{A}^0_s$ to get $\mathcal{A}^0_{s+1}$, and perform $\mathbf{R}(Y^1_s, X^1_s, Z^1_s, B^1_s, S^1_s, C^1_s)$ on $\mathcal{A}^1_s$ to get $\mathcal{A}^1_{s+1}$. (In order to accommodate the extra components, X^i_s will be the copy of $[6a_s]$ in $\mathcal{A}^i_s$. A similar change will be made for the other components.)

As long as $\mathcal{G}$ is going in the same direction, we designate every other stage as an *isomorphism recovery stage*. At such a stage $s+1$, if $r_s = 0$ then we let C^0_s be the component of $\mathcal{A}^0_s$ that extends B^0_{s-1} and let C^1_s be the isomorphic component of $\mathcal{A}^1_s$. On the other hand, if $r_s = 1$ then we let B^1_s be the component of $\mathcal{A}^1_s$ that extends C^1_{s-1} and let B^0_s be the isomorphic component of $\mathcal{A}^0_s$. Whenever $\mathcal{G}$ changes direction, we restart this isomorphism recovery process.

It is straightforward to check that this strategy guarantees that if r_s has a limit then the copies of the special component of $\mathcal{G}$ in $\mathcal{A}^0$ and $\mathcal{A}^1$ are isomorphic, while still ensuring that if r_s does not have a limit then no component of $\mathcal{A}^0$ or $\mathcal{A}^1$ is infinite. We will give an example below to illustrate isomorphism recovery.

Another problem that we must face in the full construction is that, in general, we can not know in advance whether a given computable structure $\mathcal{G}$ is isomorphic to $\mathcal{A}^0$. So it is not possible to wait at each stage until the appropriate components are enumerated into $\mathcal{G}$. To get around this, we introduce the notion of a *recovery stage*.

At stage $s+1$, where we would have waited for $\mathcal{G}$ to provide components Y_s, X_s, Z_s, B_s, and C_s, we now simply do not involve copies of the special component of $\mathcal{G}$ in our operations unless these components are provided. (That is, if these components are not in $\mathcal{G}[s]$ then we perform $\mathbf{L}(Y_s^0, X_s^0, Z_s^0)$ on $\mathcal{A}_s^0$ to get $\mathcal{A}_{s+1}^0$ and perform $\mathbf{R}(Y_s^1, X_s^1, Z_s^1)$ on $\mathcal{A}_s^1$ to get $\mathcal{A}_{s+1}^1$.) Furthermore, where we would have waited for Y_s, X_s, Z_s, B_s, S_s, and C_s to grow into copies of $Y_s \cdot X_s$, $X_s \cdot Z_s$, $Z_s \cdot B_s$, $B_s \cdot S_s$, $S_s \cdot C_s$, and $C_s \cdot Y_s$, we now just declare that we are waiting for these copies to appear in $\mathcal{G}$.

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

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

2. for each $j \notin A[s]$ that is less than or equal to the number of recovery stages before stage $s+1$, $\mathcal{G}[s]$ contains components that can be used as Y_t, X_t, Z_t, B_t, and C_t if $a_t = j$ for some $t > s$.

(These conditions will be made more precise in the full construction, which will be presented shortly.)

Now suppose that $\mathcal{G} \cong \mathcal{A}^0$. Say that $\mathcal{G}$ is *active* at a given stage if isomorphic copies of its special component participate in the operations performed at that stage. We want there to be infinitely many recovery stages. This will happen as long as there is a bound on how often $\mathcal{G}$ can be active while waiting for recovery.

Let P be the set of all $j \in \omega$ that do not enter A before the j^{th} recovery stage. Let M be the set of all coding locations of copies of $[6j]$, $j \in P$, in $\mathcal{G}$ and let N be the set of all coding locations of copies of $[6j]$, $j \notin P$, in $\mathcal{G}$. By the definition of recovery stage, $\mathcal{G}$ will be active at each stage $s+1$ such that $a_s \in P$. We make it a rule that $\mathcal{G}$ is not active at any other stage. This clearly provides the desired bound on the number of times $\mathcal{G}$ can be active while waiting for recovery.

Arguing as before, we conclude that if $\mathcal{G}$ almost always goes to the left then $U \cap M \equiv_{\mathrm{m}} A$, while if $\mathcal{G}$ almost always goes to the right then $U \cap M$ is computable. But P, N, and $U \cap N$ are computable, since if we wait until the j^{th} recovery stage then we can tell whether $j \in P$, and if $j \notin P$ then $j \in A$. So if $\mathcal{G}$ almost always goes to the left then $U \equiv_{\mathrm{m}} A$, while if $\mathcal{G}$ almost always goes to the right then U is computable. Thus (3.2.3) is satisfied for this $\mathcal{G}$.

We remark that the modification to the construction that we have just described makes the definition of isomorphism recovery stage a little more complicated, in that we will not want a stage to be an isomorphism recovery stage unless it is a *first stage*, that is, the first stage at which $\mathcal{G}$ is active after a recovery stage. We will discuss this further below.

Before proceeding, let us look at two examples. The first one illustrates what happens when $\mathcal{G}$ recovers. Suppose that $s < t < u < v$ are such that $s+1$ is a first stage, $r_{s+1} = 0$, $v+1$ is the next recovery stage after stage $s+1$, and $t+1$ and $u+1$ are the only two stages between stages $s+1$ and $v+1$ at which $\mathcal{G}$ is active.

Figure 3.3 pictures what happens on the $\mathcal{A}^0$ side of the construction. From now on, we will use the notation R^i_s in place of S^i_s, since this is the notation that we will adopt in the full construction. This change is made because R^i_w might not be isomorphic to the special component of $\mathcal{G}[w]$ if $w+1$ is not a recovery stage.

$$
\begin{tikzcd}
Y^0_s \arrow[d] & X^0_s \arrow[d] & Z^0_s \arrow[d] & B^0_s \arrow[d] & R^0_s \arrow[dd] & C^0_s \arrow[d] \\
Y^0_s \cdot X^0_s & X^0_s \cdot Z^0_s & Z^0_s \cdot B^0_s & B^0_s \cdot R^0_s & & C^0_s \cdot Y^0_s \\
Y^0_t \arrow[d] & X^0_t \arrow[d] & Z^0_t \arrow[d] & B^0_t \arrow[d] & R^0_s \cdot C^0_s \arrow[dd] & C^0_t \arrow[d] \\
Y^0_t \cdot X^0_t & X^0_t \cdot Z^0_t & Z^0_t \cdot B^0_t & B^0_t \cdot R^0_s \cdot C^0_s & & C^0_t \cdot Y^0_t \\
Y^0_u \arrow[d] & X^0_u \arrow[d] & Z^0_u \arrow[d] & B^0_u \arrow[d] & R^0_s \cdot C^0_s \cdot C^0_t \arrow[d] & C^0_u \arrow[d] \\
Y^0_u \cdot X^0_u & X^0_u \cdot Z^0_u & Z^0_u \cdot B^0_u & B^0_u \cdot R^0_s \cdot C^0_s \cdot C^0_t & R^0_s \cdot C^0_s \cdot C^0_t \cdot C^0_u & C^0_u \cdot Y^0_u
\end{tikzcd}
$$

Figure 3.3: Recovery

Note that, by the definition of recovery stage, the special component of $\mathcal{G}[s]$ is isomorphic to R^0_s and, for each $w = s, t, u$, $\mathcal{G}[s]$ has components Y_w, X_w, Z_w, B_w, and C_w isomorphic to Y^0_w, X^0_w, Z^0_w, B^0_w, and C^0_w, respectively.

Since $\mathcal{G}$ recovers at stage $v+1$, there are two possibilities. The first one is that the special component of $\mathcal{G}[v]$ is isomorphic to one of $B_s^0 \cdot R_s^0$, $B_t^0 \cdot R_s^0 \cdot C_s^0$, or $B_u^0 \cdot R_s^0 \cdot C_s^0 \cdot C_t^0$. In this case, $r_{v+1} = 1$.

The second possibility is that the special component of $\mathcal{G}[v]$ is isomorphic to $R_s^0 \cdot C_s^0 \cdot C_t^0 \cdot C_u^0$. In this case, the component of $\mathcal{G}[v]$ that extends C_u must be the one isomorphic to $C_u^0 \cdot Y_u^0$. From this it follow that the component of $\mathcal{G}[v]$ that extends Y_u must be the one isomorphic to $Y_u^0 \cdot X_u^0$. Proceeding in this fashion, we see that for each $w = s, t, u$, the component of $\mathcal{G}[v]$ that extends X_w is the one isomorphic to $X_w^0 \cdot Z_w^0$.

Notice that in the previous argument it is crucial that no component of $\mathcal{A}^0$ other than the one that extends R_s^0 participates in operations more than once in the interval $(s, v]$. This is the reason for requiring that isomorphism recovery happen only at first stages.

Our second example illustrates isomorphism recovery. Suppose that $s < t < u < v < w$ are such that $s+1$ and $v+1$ are first stages, $t+1$ and $u+1$ are the only stages between $s+1$ and $v+1$ at which $\mathcal{G}$ is active, and $w+1$ is the first stage after stage $v+1$ at which $\mathcal{G}$ is active. Suppose further that $r_{s+1} = r_{t+1} = r_{u+1} = r_{v+1} = r_{w+1} = 0$.

Figure 3.4 pictures what happens on either side of the construction. The key point to notice here is that if $R_t^0 \cong R_t^1$ then R_w^0 extends R_t^0, R_w^1 extends R_t^1, and $R_w^0 \cong R_w^1$. This pattern would allow us to prove by induction that if r_s has a limit then each $\mathcal{A}^i$ has a unique infinite component S^i and $S^0 \cong S^1$.

In the full construction, we will of course need to satisfy (3.2.3) for every computable directed graph. Let $\mathcal{G}_0, \mathcal{G}_1, \ldots$ be a standard enumeration of all partial computable directed graphs. In our construction, we will define the concepts of

$$\begin{tikzcd}
Y_t^0 \arrow[d] & X_t^0 \arrow[d] & Z_t^0 \arrow[d] & B_t^0 \arrow[d] & R_t^0 \arrow[d] & C_t^0 \arrow[d] \\
Y_t^0 \cdot X_t^0 & X_t^0 \cdot Z_t^0 & Z_t^0 \cdot B_t^0 & B_t^0 \cdot R_t^0 \arrow[dddrr, equal] & R_t^0 \cdot C_t^0 \arrow[d, equal] & C_t^0 \cdot Y_t^0 \\
Y_u^0 \arrow[d] & X_u^0 \arrow[d] & Z_u^0 \arrow[d] & B_u^0 \arrow[d] & R_u^0 \arrow[d] & C_u^0 \arrow[d] \\
Y_u^0 \cdot X_u^0 & X_u^0 \cdot Z_u^0 & Z_u^0 \cdot B_u^0 & B_u^0 \cdot R_t^0 \cdot C_t^0 & R_t^0 \cdot C_t^0 \cdot C_u^0 \arrow[d, equal] & C_u^0 \cdot Y_u^0 \\
Y_v^0 \arrow[d] & X_v^0 \arrow[d] & Z_v^0 \arrow[d] & B_v^0 \arrow[d] & R_v^0 \arrow[d] & C_v^0 \arrow[d] \\
Y_v^0 \cdot X_v^0 & X_v^0 \cdot Z_v^0 & Z_v^0 \cdot B_v^0 & B_v^0 \cdot R_t^0 \cdot C_t^0 \cdot C_u^0 & R_t^0 \cdot C_t^0 \cdot C_u^0 \cdot B_t^0 \arrow[d, equal] & B_t^0 \cdot R_t^0 \cdot Y_v^0 \\
 & & & & R_w^0 &
\end{tikzcd}$$

$$\begin{tikzcd}
Y_t^1 \arrow[d] & X_t^1 \arrow[d] & Z_t^1 \arrow[d] & B_t^1 \arrow[d] & R_t^1 \arrow[d] & C_t^1 \arrow[d] \\
Y_t^1 \cdot C_t^1 & X_t^1 \cdot Y_t^1 & Z_t^1 \cdot X_t^1 & B_t^1 \cdot Z_t^1 & R_t^1 \cdot B_t^1 \arrow[dddr, equal] & C_t^1 \cdot R_t^1 \\
Y_u^1 \arrow[d] & X_u^1 \arrow[d] & Z_u^1 \arrow[d] & B_u^1 \arrow[d] & R_u^1 \arrow[d] \arrow[ur, equal] & C_u^1 \arrow[d] \\
Y_u^1 \cdot C_u^1 & X_u^1 \cdot Y_u^1 & Z_u^1 \cdot X_u^1 & B_u^1 \cdot Z_u^1 & C_t^1 \cdot R_t^1 \cdot B_u^1 & C_u^1 \cdot R_t^1 \cdot C_t^1 \\
Y_v^1 \arrow[d] & X_v^1 \arrow[d] & Z_v^1 \arrow[d] & B_v^1 \arrow[d] & R_v^1 \arrow[d] \arrow[ur, equal] & C_v^1 \arrow[d] \\
Y_v^1 \cdot R_t^1 \cdot B_t^1 & X_v^1 \cdot Y_v^1 & Z_v^1 \cdot X_v^1 & B_v^1 \cdot Z_v^1 & C_u^1 \cdot R_t^1 \cdot C_t^1 \cdot B_v^1 & R_t^1 \cdot B_t^1 \cdot C_t^1 \cdot C_u^1 \\
 & & & & R_w^1 \arrow[ur, equal] &
\end{tikzcd}$$

Figure 3.4: Isomorphism recovery (top: $\mathcal{A}^0$ / bottom: $\mathcal{A}^1$)

n-recovery stage, n-isomorphism recovery stage, and so forth.

We will be able to satisfy (3.2.3) for each $\mathcal{G}_n$ independently. We first need some notation to allow us to distinguish the components that are used to satisfy (3.2.3) for a particular $\mathcal{G}_n$.

3.2.4. Definition. Let $\mathcal{G}$ be a directed graph. We denote by $(\mathcal{G})_n$ 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 either a copy of $[6n + 3]$ or a copy of $[6\langle n, j\rangle + l]$ for some $j \in \omega$, $l \in \{1, 2, 4, 5\}$.

The idea is that the components of $(\mathcal{A}^i)_n$ are the ones used in the construction to satisfy (3.2.3) for $\mathcal{G}_n$, and that $(\mathcal{A}^i_s)_n$ is the subgraph of $\mathcal{A}^i_s$ consisting of all such components that have participated in operations before stage $s + 1$.

We also need new **L**- and **R**-operations in order to involve components of $(\mathcal{A}^i)_n$ for different n's in operations at the same stage.

3.2.5. Definition. Let $\mathcal{G}$ be a computable structure in the language of directed graphs whose domain is co-infinite. Let $K_0, K_1, \ldots, K_n$ and L be components of $\mathcal{G}$ isomorphic to $[y_0], [y_1], \ldots, [y_n]$ and $[x]$, respectively, where $y_0, y_1, \ldots, y_n, x \in \omega$. We define two operations, each of which takes $\mathcal{G}$ to a new computable structure extending $\mathcal{G}$.

- The operation $(K_0, K_1, \ldots, K_n) \cdot L$ 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 $i \leqslant n$, add an edge from the top of this new copy of $[x]$ to the top of K_i.

- The operation $L \cdot (K_0, K_1, \ldots, K_n)$ 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 $[y_i]$.

For example, suppose that L, K_0, and K_1 are copies of [2], [3], and [4], respectively. Then the operation $(K_0, K_1) \cdot L$ consists of extending $K_0 \cup K_1$ to a copy of the graph shown in Figure 3.5, while the operation $L \cdot (K_0, K_1)$ consists of extending L to a copy of that same graph.

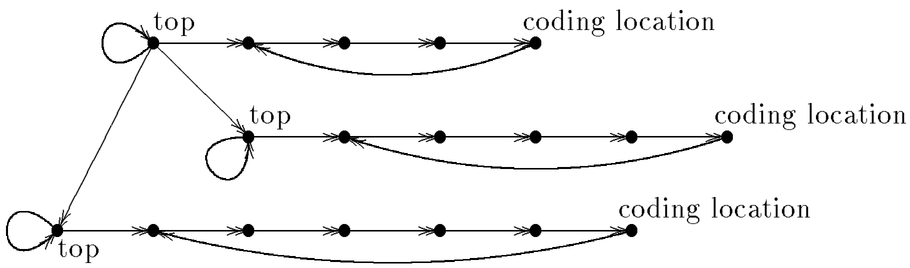


Figure 3.5: The result of either of the operations $([3], [4]) \cdot [2]$ or $[2] \cdot ([3], [4])$

3.2.6. Definition. Let $\mathcal{G}$ be a computable structure in the language of directed graphs whose domain is co-infinite. We say that a component C of $\mathcal{G}$ is a *set component* if it is isomorphic to $[T]$ for some finite $T \subset \omega$. If T is a singleton then we say that C is a *singleton component.*

Let $Y_0, \ldots, Y_n$, X, $Z_0, \ldots, Z_n$, $B_0, \ldots, B_n$, $S_0, \ldots, S_n$, and $C_0, \ldots, C_n$ be components of $\mathcal{G}$ such that for each $i \leqslant n$, X, Y_i, and Z_i are singleton components and B_i, S_i, and C_i are set components. We define two operations, each of which takes $\mathcal{G}$ to a new computable structure extending $\mathcal{G}$.

- The $\mathbf{L}$-*operation*

  $$\mathbf{L}(Y_0, \ldots, Y_n; X; Z_0, \ldots, Z_n; B_0, S_0, C_0; \ldots; B_n, S_n, C_n)$$

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

  $$(Y_0, \ldots, Y_n) \cdot X,\ X \cdot (Z_0, \ldots, Z_n),\ Z_0 \cdot B_0,\ \ldots,\ Z_n \cdot B_n,$$
  $$B_0 \cdot S_0,\ \ldots,\ B_n \cdot S_n,\ S_0 \cdot C_0,\ \ldots,\ S_n \cdot C_n,\ C_0 \cdot Y_0,\ \ldots,\ C_n \cdot Y_n$$

- The $\mathbf{R}$-*operation*

  $$\mathbf{R}(Y_0, \ldots, Y_n; X; Z_0, \ldots, Z_n; B_0, S_0, C_0; \ldots; B_n, S_n, C_n)$$

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

  $$Y_0 \cdot C_0,\ \ldots,\ Y_n \cdot C_n,\ C_0 \cdot S_0,\ \ldots,\ C_n \cdot S_n,\ S_0 \cdot B_0,\ \ldots,\ S_n \cdot B_n,$$
  $$B_0 \cdot Z_0,\ \ldots,\ B_n \cdot Z_n,\ (Z_0, \ldots, Z_n) \cdot X,\ X \cdot (Y_0, \ldots, Y_n)$$

Note that if $\mathcal{H}$ is the structure obtained by applying

$$\mathbf{L}(Y_0, \ldots, Y_n; X; Z_0, \ldots, Z_n; B_0, S_0, C_0; \ldots; B_n, S_n, C_n)$$

to $\mathcal{G}$ and $\mathcal{H}'$ is the structure obtained by applying

$$\mathbf{R}(Y_0, \ldots, Y_n; X; Z_0, \ldots, Z_n; B_0, S_0, C_0; \ldots; B_n, S_n, C_n)$$

to $\mathcal{G}$ then $\mathcal{H} \cong \mathcal{H}'$.

We now proceed with the construction of $\mathcal{A}^0$, $\mathcal{A}^1$, U^0, and U^1. For each $i = 0, 1$, we first define a computable structure $\mathcal{A}^i_0$. At each stage $s+1$, we perform an operation on $\mathcal{A}^i_s$ to get $\mathcal{A}^i_{s+1} \supset \mathcal{A}^i_s$ and add an element of the domain of $\mathcal{A}^i_{s+1}$ to U^i. We then let $\mathcal{A}^i = \bigcup_{s\in\omega} \mathcal{A}^i_s$. In order to guarantee that $\mathcal{A}^i$ is computable, we make it a convention that all numbers added to the domain of $\mathcal{A}^i_s$ at stage $s+1$ to get $\mathcal{A}^i_{s+1}$ are greater than s.

Let $t \geqslant s$. We say that a component L of $\mathcal{A}^i_t$ or $\mathcal{A}^i$ (resp. $\mathcal{G}_n[t]$ or $\mathcal{G}_n$) *extends* a component K of $\mathcal{A}^i_s$ ($\mathcal{G}_n[s]$) if the domain of K is contained in the domain of L, and that L *properly extends* K if this containment is proper. (Note that "L extends K" means more than just that K can be embedded in L, though it of course implies the latter.) If L extends K but not properly then we say that L is a component of $\mathcal{A}^i_s$ ($\mathcal{G}_n[s]$).

It will be the case that if K and L are distinct components of $\mathcal{A}^0_s$ and K is not a copy of $[6k+1]$ or $[6k+2]$ for any $k \in \omega$ then K and L are not extended by the same component of $\mathcal{A}^0$. Thus, since we are not interested in $\mathcal{G}_n$ unless it is isomorphic to $\mathcal{A}^0$, we may assume without loss of generality that, for each $n, s \in \omega$, there is an embedding of $\mathcal{G}_n[s]$ into $\mathcal{A}^0_s$ such that if K and L are distinct components of $\mathcal{G}_n[s]$ and K is not a copy of $[6k+1]$ or $[6k+2]$ for any $k \in \omega$ then K and L are mapped into distinct components of $\mathcal{A}^0_s$.

Suppose there is a least stage s such that $\mathcal{G}_n[s]$ has a component K isomorphic to $[6n+3]$ and let $t \geqslant s$. We call the component of $\mathcal{G}_n[t]$ (resp. $\mathcal{G}_n$) that extends K the *special component* of $\mathcal{G}_n[t]$ ($\mathcal{G}_n$).

It will be easy to check as we go along that the following are properties of the construction.

1. For each $s \in \omega$, $\mathcal{A}_s^0 \cong \mathcal{A}_s^1$ and no component of $\mathcal{A}_s^i$ is embeddable in another component of $\mathcal{A}_s^i$.

2. Let $t < s$. No component of $\mathcal{A}_t^i$ isomorphic to one of $[6a_s]$ or $[6\langle n, a_s\rangle + l]$, $l \in \{1, 2, 4, 5\}$, $n \in \omega$, participates in an operation at stage $t + 1$.

stage 0. Let $\mathcal{A}_0^0$ and $\mathcal{A}_0^1$ be computable structures with co-infinite domains, each consisting of one copy of $[k]$ for each $k \in \omega$. For each $n \in \omega$, let $r_{n,0} = 0$.

stage $s + 1$. For each $n < s + 1$, say that $s + 1$ is an *n-recovery stage* if all of the following conditions hold.

1. $\mathcal{G}_n[s]$ has a special component isomorphic to some component of $\mathcal{A}_s^0$.

2. $(\mathcal{G}_n[s])_n \cong (\mathcal{A}_s^0)_n$.

3. Let $j \notin A[s]$ be less than or equal to the number of n-recovery stages before stage $s + 1$. There is a component of $\mathcal{G}_n[s]$ isomorphic to $[6j]$ and for each $l \in \{1, 2, 4, 5\}$ there is a component of $\mathcal{G}_n[s]$ isomorphic to $[6\langle n, j\rangle + l]$.

If $s + 1$ is an n-recovery stage then, for $i = 0, 1$, let $S_{n,s}^i$ be the component of $\mathcal{A}_s^i$ that is isomorphic to the special component of $\mathcal{G}_n[s]$. If $s + 1$ is the first n-recovery stage then let $r_{n,s+1} = 0$. Otherwise, proceed as follows. Let $i = r_{n,s}$ and let $t + 1$ be the last n-recovery stage before stage $s + 1$. If $S_{n,s}^i$ extends $S_{n,t}^i$ then let $r_{n,s+1} = i$, and otherwise let $r_{n,s+1} = 1 - i$.

If $s+1$ is not an n-recovery stage then let $r_{n,s+1} = r_{n,s}$.

Now let $n_0, n_1, \ldots, n_m$ be all the numbers n_j such that a_s is less than the number of n_j-recovery stages less than or equal to $s+1$. We say that each n_j, $j \leqslant m$, is *active* at stage $s+1$. For $i = 0, 1$ and $j \leqslant m$, let X_s^i, $Y_{n_j,s}^i$, and $Z_{n_j,s}^i$ be the components of $\mathcal{A}_s^i$ isomorphic to $[6a_s]$, $[6\langle n_j, a_s\rangle + 1]$, and $[6\langle n_j, a_s\rangle + 2]$, respectively.

For each $j \leqslant m$, let $t_j + 1 \leqslant s+1$ be the last n_j-recovery stage. We say that $s+1$ is an n_j-*first stage* if it is the first stage after stage t_j at which n_j is active.

We say that $s+1$ is an n_j-*change stage* if it is an n_j-first stage and either $t_j + 1$ was the first n_j-recovery stage or $r_{n_j,t_j+1} \neq r_{n_j,t_j}$.

We say that $s+1$ is an n_j-*isomorphism recovery stage* if it is an n_j-first stage but not an n_j-change stage and one of the following conditions holds.

1. The last n_j-first stage before stage $s+1$ was an n_j-change stage.

2. There has been at least one stage at which n_j was active after the last n_j-isomorphism recovery stage and before stage $s+1$.

For each $j \leqslant m$ we define components $B_{n_j,s}^i$ and $C_{n_j,s}^i$, $i = 0, 1$. There are two cases.

1. $s+1$ is an n_j-isomorphism recovery stage. If the first condition in the definition of n_j-isomorphism recovery stage holds then let $t+1$ be the last n_j-first stage, and otherwise let $t+1$ be the first stage after the last n_j-isomorphism recovery stage at which n_j was active. There are two subcases.

   (a) If $r_{n_j,s+1} = 0$ then let $C_{n_j,s}^0$ be the component of $\mathcal{A}_s^0$ that extends $B_{n_j,t}^0$

and let $C^1_{n_j,s}$ be its isomorphic image in $\mathcal{A}^1_s$. For $i = 0, 1$, let $B^i_{n_j,s}$ be the component of $\mathcal{A}^i_s$ isomorphic to $[6\langle n_j, a_s\rangle + 4]$.

(b) If $r_{n_j,s+1} = 1$ then let $B^1_{n_j,s}$ be the component of $\mathcal{A}^1_s$ that extends $C^1_{n_j,t}$ and let $B^0_{n_j,s}$ be its isomorphic image in $\mathcal{A}^0_s$. For $i = 0, 1$, let $C^i_{n_j,s}$ be the component of $\mathcal{A}^i_s$ isomorphic to $[6\langle n_j, a_s\rangle + 5]$.

2. $s + 1$ is not an n_j-isomorphism recovery stage. For $i = 0, 1$, let $B^i_{n_j,s}$ be the component of $\mathcal{A}^i_s$ isomorphic to $[6\langle n_j, a_s\rangle + 4]$ and let $C^i_{n_j,s}$ be the component of $\mathcal{A}^i_s$ isomorphic to $[6\langle n_j, a_s\rangle + 5]$.

For each $j \leqslant m$, proceed as follows. Let $i = r_{n_j,s+1}$ and let $t + 1 \leqslant s + 1$ be the last n_j-recovery stage. Let $R^i_{n_j,s}$ be the component of $\mathcal{A}^i_s$ that extends $S^i_{n_j,t}$ and let $R^{1-i}_{n_j,s}$ be its isomorphic image in $\mathcal{A}^{1-i}_s$.

Now perform

$$\mathbf{L}(Y^0_{n_0,s}, \ldots, Y^0_{n_m,s}; X^0_s; Z^0_{n_0,s}, \ldots, Z^0_{n_m,s}; B^0_{n_0,s}, R^0_{n_0,s}, C^0_{n_0,s};$$
$$B^0_{n_1,s}, R^0_{n_1,s}, C^0_{n_1,s}; \ldots; B^0_{n_m,s}, R^0_{n_m,s}, C^0_{n_m,s})$$

on $\mathcal{A}^0_s$ to get $\mathcal{A}^0_{s+1}$ and perform

$$\mathbf{R}(Y^1_{n_0,s}, \ldots, Y^1_{n_m,s}; X^1_s; Z^1_{n_0,s}, \ldots, Z^1_{n_m,s}; B^1_{n_0,s}, R^1_{n_0,s}, C^1_{n_0,s};$$
$$B^1_{n_1,s}, R^1_{n_1,s}, C^1_{n_1,s}; \ldots; B^1_{n_m,s}, R^1_{n_m,s}, C^1_{n_m,s})$$

on $\mathcal{A}^1_s$ to get $\mathcal{A}^1_{s+1}$. (If no n is active at stage $s+1$ then, for $j = 0, 1$, let Y^j_s and Z^j_s be the components of $\mathcal{A}^j_s$ isomorphic to $[6\langle 0, a_s\rangle + 1]$ and $[6\langle 0, a_s\rangle + 2]$, respectively. Perform $\mathbf{L}(Y^0_s, X^0_s, Z^0_s)$ on $\mathcal{A}^0_s$ to get $\mathcal{A}^0_{s+1}$ and perform $\mathbf{R}(Y^1_s, X^1_s, Z^1_s)$ on $\mathcal{A}^1_s$ to get $\mathcal{A}^1_{s+1}$.)

Put the coding location of the copy of $[6a_s]$ in $\mathcal{A}^0_0$ into U^0 and put the coding location of the copy of $[6a_s]$ in $\mathcal{A}^1_{s+1} - \mathcal{A}^1_s$ into U^1.

This completes the construction. Let $\mathcal{A}^0 = \bigcup_{s\in\omega} \mathcal{A}^0_s$ and $\mathcal{A}^1 = \bigcup_{s\in\omega} \mathcal{A}^1_s$. Since for each $s \in \omega$ and $i = 0, 1$, all numbers in $\mathcal{A}^i_{s+1} - \mathcal{A}^i_s$ are greater than s, $\mathcal{A}^0$ and $\mathcal{A}^1$ are computable. We now wish to argue that properties (3.2.1)–(3.2.3) are satisfied. Theorem 3.1.3 will then follow immediately.

Property (3.2.2) is easy to establish, so we deal with it first.

3.2.7. Lemma. *$U^0 \equiv_m A$ and U^1 is computable.*

Proof. The numbers in U^0 are all coding locations of components of $\mathcal{A}^0_0$ of the form $[6j]$, $j \in \omega$, and the coding location of the copy of $[6j]$ in $\mathcal{A}^0_0$ is in U^0 if and only if $j \in A$. Since given any number we can computably determine whether it is a coding location in $\mathcal{A}^0_0$ and if so, for what $[k]$, this means that $U^0 \equiv_{\mathrm{m}} A$.

Any number put into U^1 at a stage $s + 1$ is a new number, i.e. one not in the domain of $\mathcal{A}^1_s$, and hence is greater than s. Thus U^1 is computable. $\square$

In showing that (3.2.1) and (3.2.3) are satisfied, we will need a few facts about the construction. The more obvious ones are given without proof, while the remaining ones are broken down into easily checked properties of the construction. Figures 3.3 and 3.4 should be helpful here.

We say that a component of $\mathcal{A}^i$ participates in an operation at stage $s + 1$ if it extends a component of $\mathcal{A}^i_s$ that participates in an operation at stage $s + 1$.

3.2.8. Lemma. *Let $\mathcal{G} \cong A^0$ be computable. Given x in the domain of $\mathcal{G}$, we can computably determine if x is the coding location of a copy of some $[k]$, $k \in \omega$, and if so, for what k. In particular, the set of coding locations of copies of $[6j]$, $j \in \omega$, in $\mathcal{G}$ is computable.*

3.2.9. Lemma. *Let K and L be distinct components of $\mathcal{A}^i_s$ such that K is not a copy of $[6k+1]$ or $[6k+2]$ for any $k \in \omega$. K and L are not extended by the same component of $\mathcal{A}^i$.*

Lemma 3.2.9 will be used without explicit mention several times below.

3.2.10. Lemma. *A component of $\mathcal{A}^i$ is infinite if and only if it participates in operations infinitely often.*

3.2.11. Lemma. *Let $k, n \in \omega$. Any component of $\mathcal{A}^i$ containing a copy of $[6k]$, $[6\langle n,k\rangle + 1]$, or $[6\langle n,k\rangle + 2]$ can participate in an operation at most once. Any component of $\mathcal{A}^i$ containing a copy of $[6n+3]$, $[6\langle n,k\rangle + 4]$, or $[6\langle n,k\rangle + 5]$ can participate in operations only at stages at which n is active.*

3.2.12. Lemma. *Suppose that $r_{n,s} = i \neq r_{n,s+1}$. Of all the components of $(\mathcal{A}^i)_n$ 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 $S^i_{n,s}$.*

Proof. Suppose that a component of $(\mathcal{A}^i)_n$ participates in operations at stages $t < u$ and does not participate in an operation at any stage in (t, u), and let v be the last n-first stage before stage u. It is not hard to check that it must then be the case that $t \geqslant v$.

Now let t be the first stage after stage s at which n is active. Then t is an n-change stage, and hence not an n-isomorphism recovery stage. It follows that, of all the components of $(\mathcal{A}^i)_n$ that participate in operations at stages before stage $s+1$, the only one that participates in an operation at stage t is the one that extends $S^i_{n,s}$. The lemma now follows by induction, using the fact mentioned in the previous paragraph. $\square$

3.2.13. Lemma. *For each $s \in \omega$, $\mathcal{A}^0_s \cong \mathcal{A}^1_s$ and no component of $\mathcal{A}^i_s$ is embeddable in another component of $\mathcal{A}^i_s$. Furthermore, if a component of $\mathcal{A}^i_s$ participates in an operation at stage $s+1$ then so does the (unique) isomorphic component of $\mathcal{A}^{1-i}_s$.*

3.2.14. Lemma. *Suppose that $r_{n,s} = i$ for all $s > t$ and n is active at stages s_0+1 and s_1+1, where $s_1 > s_0 \geqslant t$. Then R^i_{n,s_1} extends R^i_{n,s_0}.*

3.2.15. Lemma. *Let $s+1$ be an n-recovery stage that is not the first such stage. Let $t+1$ be the last n-recovery stage before stage $s+1$. If $r_{n,s} = 0 \neq r_{n,s+1}$ then $S^0_{n,s}$ extends $B^0_{n,u}$ for some $u \in [t,s)$. Similarly, if $r_{n,s} = 1 \neq r_{n,s+1}$ then $S^1_{n,s}$ extends $C^1_{n,u}$ for some $u \in [t,s)$.*

Proof. The two cases, $i = 0$ and $i = 1$, are similar. We do the case $i = 0$.

Since $S^0_{n,s}$ contains a copy of $S^0_{n,t}$ and $r_{n,t+1} = r_{n,s} = 0$, either $S^0_{n,s}$ extends $S^0_{n,t}$ or $S^0_{n,s}$ extends $B^0_{n,u}$ for some u such that $t \leqslant u < s$. But it cannot be the case that $S^0_{n,s}$ extends $S^0_{n,t}$, since that would imply that $r_{n,s+1} = 0$. $\square$

3.2.16. Lemma. *Suppose that $r_{n,t} = 0$ (resp. $r_{n,t} = 1$) for all $t \geqslant s_0$. Then no component of $(\mathcal{A}^0)_n$ ($(\mathcal{A}^1)_n$) can participate in an operation more than twice after* *stage s_0 unless it extends $R^0_{n,t}$ ($R^1_{n,t}$) for some $t \geqslant s_0$, while no component of $(\mathcal{A}^1)_n$ ($(\mathcal{A}^0)_n$) can participate in an operation more than twice after stage s_0 unless it extends $C^1_{n,t}$ ($B^0_{n,t}$) for some $t \geqslant s_0$ such that $t+1$ is an n-isomorphism recovery stage.*

Proof. The two cases, $i = 0$ and $i = 1$, are similar. We do the case $i = 0$.

Suppose that component K of $(\mathcal{A}^0)_n$ participates in operations at stages $s+1 < t+1 < u+1$, where $s+1 \geqslant s_0$, but not at any stage in $(t+1, u+1)$. Then either K extends $R^0_{n,u}$ or $u+1$ is an n-isomorphism recovery stage and K extends $C^0_{n,u}$. We claim that the latter case cannot hold. Indeed, if K extends $C^0_{n,u}$ then K extends $B^0_{n,t}$. But since $r_{n,t+1} = 0$, $B^0_{n,t}$ is a singleton component. Thus K does not participate in an operation at stage $s+1$, contrary to hypothesis.

Now suppose that component L of $(\mathcal{A}^1)_n$ participates in operations at stages $s+1 < t+1 < u+1$, where $s+1 \geqslant s_0$, but not at any stage in $(t+1, u+1)$. Then either L extends $R^1_{n,t}$ or $t+1$ is an n-isomorphism recovery stage and L extends $C^1_{n,t}$. But in the former case, $u+1$ is an n-isomorphism recovery stage and L extends $C^1_{n,u}$. □

3.2.17. Lemma. *Suppose that $s < t < v$ are such that $s+1$ is an n-isomorphism recovery stage, $r_{n,u} = r_{n,s+1}$ for all $u > s$, $t+1$ is the next stage after stage $s+1$ at which n is active, and $v+1$ is the next n-isomorphism recovery stage after stage $s+1$. For $i = 0, 1$, let B^i, R^i, and C^i be the components of $\mathcal{A}^i_{t+1}$ that extend $B^i_{n,t}$, $R^i_{n,t}$, and $C^i_{n,t}$, respectively, and let $\widehat{B}^i$, $\widehat{R}^i$, and $\widehat{C}^i$ be the components of $\mathcal{A}^i_v$ that extend B^i, R^i, and C^i, respectively. If $r_{n,s+1} = 0$ then $\widehat{B}^0 \cong B^0$ and $\widehat{R}^1 \cong R^1$, while if*

$r_{n,s+1} = 1$ *then* $\widehat{C}^1 \cong C^1$ *and* $\widehat{R}^0 \cong R^0$.

Proof. The two cases, $i = 0$ and $i = 1$, are similar. We do the case $i = 0$. It is enough to show that the components of $(\mathcal{A}^0)_n$ and $(\mathcal{A}^1)_n$ that extend B^0 and R^1, respectively, do not participate in operations at any stage in $(t+1, v+1)$.

Suppose that component K of $(\mathcal{A}^0)_n$ participates in operations at stages $t+1$ and $u+1$, where $t < u < v$. Since no stage in $(t+1, v+1)$ is an n-isomorphism recovery stage, K extends $R^0_{n,u}$, which in turn extends $R^0_{n,t}$. Thus K does not extend B^0.

Now suppose that component L of $(\mathcal{A}^1)_n$ participates in operations at stages $t+1$ and $u+1$, where $t < u < v$. Again, no stage in $(t+1, v+1)$ is an n-isomorphism recovery stage, so L extends $R^1_{n,u}$, which in turn extends $C^1_{n,t}$. Thus L does not extend R^1. $\square$

3.2.18. Lemma. *Let x be the coding location of a copy of $[6a_s]$ in component K of $\mathcal{A}^i$. Either K contains a copy of $[6\langle n, a_s\rangle + 1]$ for some $n \in \omega$, in which case $x \notin U^i$, or K contains a copy of $[6\langle n, a_s\rangle + 2]$ for some $n \in \omega$, in which case $x \in U^i$.*

We now wish to show that (3.2.1) holds. It follows from Lemmas 3.2.10, 3.2.13, and 3.2.18 that it is enough to show that for each infinite component of $\mathcal{A}^i$ there is an isomorphic component of $\mathcal{A}^{1-i}$. The first step in establishing this result is characterizing the infinite components of $\mathcal{A}^i$.

3.2.19. Lemma. *If $r_{n,s}$ does not have a limit then no component of $(\mathcal{A}^i)_n$ is infinite.*

Proof. Suppose that $r_{n,s} = 0 \neq r_{n,s+1}$ and let $t+1$ be the last n-recovery stage before stage $s+1$. By Lemma 3.2.12, of all the components of $(\mathcal{A}^0)_n$ 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 $S^0_{n,s}$. By Lemma 3.2.15, L extends $B^0_{n,u}$ for some $u \in [t, s)$. But the fact that $r_{n,t+1} = 0$ means that for all $u \in [t, s)$, $B^0_{n,u}$ is a singleton component, and hence did not participate in an operation at any stage before stage $t+1$.

Thus, no component of $(\mathcal{A}^0)_n$ that participates in an operation before stage $t+1$ can do so again after stage s. A similar argument shows that if $r_{n,s} = 1 \neq r_{n,s+1}$ and $t+1$ is the last n-recovery stage before stage $s+1$ then no component of $(\mathcal{A}^1)_n$ that participates in an operation before stage $t+1$ can do so again after stage s.

The lemma now follows from Lemma 3.2.10. $\square$

Thus, the only components of $\mathcal{A}^i$ that can be infinite are those components that are in $(\mathcal{A}^i)_n$ for some n such that $r_{n,s}$ has a limit and n is active infinitely often. So, by the comments preceding Lemma 3.2.19, to establish that (3.2.1) holds, it is enough to show that if $r_{n,s}$ has a limit and n is active infinitely often then, for each $i = 0, 1$, there is exactly one infinite component S^i_n of $(\mathcal{A}^i)_n$ and $S^0_n \cong S^1_n$. This is what we do in the next few lemmas.

3.2.20. Lemma. *There are infinitely many n-recovery stages if and only if n is active infinitely often.*

Proof. By definition, n is active at a stage $s+1$ if and only if a_s is less than the number of n-recovery stages less than or equal to $s+1$. Thus, if there are finitely many n-recovery stages then n cannot be active infinitely often.

For the other direction, suppose that there are infinitely many n-recovery stages but only finitely many stages at which n is active. Let s be the last stage at which n is active. Now given $x \in \omega$, let $t+1$ be the first stage after stage s by which there have been $x+1$ many n-recovery stages. Then $x \in A \Leftrightarrow x \in A[t]$. But this means that A is computable, contrary to hypothesis. $\square$

3.2.21. Lemma. *If n is active infinitely often and $r_{n,s}$ has a limit then there are infinitely many n-isomorphism recovery stages.*

Proof. If n is active infinitely often then, by Lemma 3.2.20, there are infinitely many n-recovery stages, and thus infinitely many n-first stages. The fact that $r_{n,s}$ has a limit implies that only finitely many of these can be n-change stages. The lemma now follows directly from the definition of n-isomorphism recovery stage. $\square$

3.2.22. Lemma. *Suppose that n is active infinitely often and s and i are such that $r_{n,t} = r_{n,s} = i$ for all $t \geqslant s$. By Lemma 3.2.21, there are infinitely many n-isomorphism recovery stages. Let $s_0 + 1 < s_1 + 1 < \cdots$ be the n-isomorphism recovery stages after stage s. For each $j \in \omega$, let $t_j + 1$ be the next stage after stage $s_j + 1$ at which n is active. (Note that $t_j < s_{j+1}$ for all $j \in \omega$.) For $t \geqslant t_0$, let K^l_t be the component of $\mathcal{A}^l_t$ that extends R^l_{n,t_0}. Then $K^l_{t_j} = R^l_{n,t_j}$ for all $j \in \omega$.*

Proof. The two cases, $i = 0$ and $i = 1$, are similar. We do the case $i = 0$.

That $K^0_{t_j} = R^0_{n,t_j}$ for all $j \in \omega$ follows from Lemma 3.2.14.

Now assume by induction that $K^1_{t_j} = R^1_{t_j}$. Let B be the component of $\mathcal{A}^0_{t_j+1}$ that extends B^0_{n,t_j}. By construction, $B \cong K^1_{t_j+1}$. Since $s_{j+1}+1$ is an n-isomorphism recovery stage, $C^0_{s_{j+1}}$ extends B. Thus, by Lemma 3.2.17, $C^0_{s_{j+1}} \cong B$. By the same lemma, $K^1_{s_{j+1}} \cong K^1_{t_j+1}$. So $C^0_{s_{j+1}} \cong K^1_{s_{j+1}}$, and thus $C^1_{s_{j+1}} = K^1_{s_{j+1}}$. Let R be the component of $\mathcal{A}^0_{s_{j+1}+1}$ that extends $R^0_{n,s_{j+1}}$. Then $R \cong K^1_{s_{j+1}+1}$. But, by Lemma 3.2.11, $R^0_{n,t_{j+1}} \cong R$ and $K^1_{t_{j+1}} \cong K^1_{s_{j+1}+1}$. So $K^1_{t_{j+1}} \cong R^0_{n,t_{j+1}}$, and thus $K^1_{t_{j+1}} = R^1_{n,t_{j+1}}$. □

For the next two lemmas, we assume the hypotheses of Lemma 3.2.22 and adopt its notation. Let S^l_n be the component of $\mathcal{A}^l$ that extends R^l_{n,s_0}.

3.2.23. Lemma. *S^l_n is the only infinite component of $(\mathcal{A}^l)_n$.*

Proof. This follows immediately from Lemmas 3.2.10, 3.2.16, and 3.2.22 and the observation that, for all $j \in \omega$, if $i = 0$ in the hypotheses of Lemma 3.2.22 then R^1_{n,t_j} extends C^1_{n,s_j}, while if $i = 1$ then R^0_{n,t_j} extends B^0_{n,s_j}. □

3.2.24. Lemma. *$S^0_n \cong S^1_n$.*

Proof. Directly from Lemma 3.2.22, since, by definition, $R^0_{n,t_j} \cong R^1_{n,t_j}$ for all $j \in \omega$, and $S^i_n = \bigcup_{j\in\omega} R^i_{n,t_j}$ for $i = 0, 1$. □

As we have argued above, Lemmas 3.2.23 and 3.2.24 suffice to establish that (3.2.1) holds.

3.2.25. Lemma. *$\mathcal{A}^0 \cong \mathcal{A}^1$ via an isomorphism that carries U^0 to U^1.*

We are left with showing that property (3.2.3) holds. This will break down into three steps. Suppose that $\mathcal{G}_n \cong \mathcal{A}^0$ and let U be the image of U^0 in $\mathcal{G}_n$.

1. We show that $r_{n,s}$ reaches a limit r_n.

2. Let t be such that for all $u \geqslant t$, $r_{n,u} = r_n$. Let A' be the set of all a_s such that either $s < t$ or the number of n-recovery stages less than or equal to $s+1$ is less than or equal to a_s. Let N be the set of all $x \in \mathcal{G}_n$ such that x is the coding location of a copy of $[6a]$, $a \in A'$. We show that A', N, and $U \cap N$ are computable.

3. Let C be the set of coding locations of copies of graphs of the form $[6j]$, $j \in \omega$, in $\mathcal{G}_n$ and let $M = C - N$. Note that M is computable. We show that

   (a) if $r_n = 0$ then an element x of M is in U if and only if, for some $j \in A$, x is the coding location of the first copy of $[6j]$ to appear in $\mathcal{G}_n$, so that $U \cap M \equiv_{\mathrm{m}} A$, while

   (b) if $r_n = 1$ then an element x of M is in U if and only if, for some $j \in \omega$, x is the coding location of the second copy of $[6j]$ to appear in $\mathcal{G}_n$, so that $U \cap M$ is computable.

Since $U = (U \cap N) \cup (U \cap M)$, this is enough to establish that (3.2.3) holds.

3.2.26. Lemma. *If $\mathcal{G}_n \cong \mathcal{A}^0$ then there are infinitely many n-recovery stages, and hence the special component of $\mathcal{G}_n$ is infinite.*

Proof. If $\mathcal{G}_n \cong A^0$ then $\mathcal{G}_n$ has a special component. Now suppose that there are only m many n-recovery stages. Let s_0 be the last n-recovery stage. (If there are no n-recovery stages then let s_0 be the first stage at which $\mathcal{G}_n$ has a special component.) By Lemma 3.2.20, there is a stage $s_1 > s_0$ such that n is not active at any stage $t \geqslant s_1$. If $m = a_u$ for some $u > s_1$ then let $s = u + 1$; otherwise, let $s = s_1$.

Consider the components of $\mathcal{A}^0$ that contain a copy of the special component of $\mathcal{G}_n$. By Lemma 3.2.11, each such component is finite. Thus, if the first condition in the definition of n-recovery stage is not eventually satisfied after stage s then the special component of $\mathcal{G}_n$ is not isomorphic to any component of $\mathcal{A}^0$.

Now consider $(\mathcal{A}^0)_n$. Again by Lemma 3.2.11, $(\mathcal{A}^0)_n$ is finite. So if the second condition in the definition of n-recovery stage is not eventually satisfied after stage s then $(\mathcal{G}_n)_n \ncong (\mathcal{A}^0)_n$.

Finally, let $j \notin A[s]$, $j \leqslant m$, and $l \in \{1, 2, 4, 5\}$ and consider the components of $\mathcal{A}^0$ that contain a copy of $[6\langle n, j\rangle + l]$. By the choice of s, $j \notin A[s] \Rightarrow j \notin A$, so there is only one such component and it is isomorphic to $[6\langle n, j\rangle + l]$. Similarly, there is only one component that contains a copy of $[6j]$ and it is isomorphic to $[6j]$.

Thus, if the third condition in the definition of n-recovery stage is not eventually satisfied after stage s then there is a component of $\mathcal{A}^0$ that is not isomorphic to any component of $\mathcal{G}_n$.

In any case, $\mathcal{G}_n$ cannot be isomorphic to $\mathcal{A}^0$, contradicting the hypothesis of the lemma. So there are infinitely many n-recovery stages.

Now, given any two n-recovery stages $t + 1 < u + 1$ such that there is a stage in $(t, u]$ at which n is active, the special component of $\mathcal{G}_n[u]$ properly extends the special component of $\mathcal{G}_n[t]$. But, by Lemma 3.2.20, n is active at infinitely many stages. This establishes the second part of the lemma. $\square$

3.2.27. Lemma. *If $\mathcal{G}_n \cong \mathcal{A}^0$ then $r_n = \lim_s r_{n,s}$ is well-defined.*

Proof. This follows immediately from Lemmas 3.2.19 and 3.2.26. □

3.2.28. Lemma. *Suppose that $\mathcal{G}_n \cong \mathcal{A}^0$. Let U be the image of U^0 under this isomorphism. By Lemma 3.2.27, $r_n = \lim_s r_{n,s}$ is well-defined. Let t be such that for all $u \geqslant t$, $r_{n,u} = r_n$. Let A' be the set of all a_s such that either $s < t$ or the number of n-recovery stages less than or equal to $s+1$ is less than or equal to a_s. Let N be the set of all $x \in \mathcal{G}_n$ such that x is the coding location of a copy of $[6a]$, $a \in A'$. Then A', N, and $U \cap N$ are computable.*

Proof. By Lemma 3.2.8, given x in the domain of $\mathcal{G}_n$, we can computably determine if x is the coding location of a copy of some $[k]$, $k \in \omega$, and if so, for what k.

By Lemma 3.2.26, there are infinitely many n-recovery stages, so the set of all a_s such that the number of n-recovery stages less than or equal to $s+1$ is less than or equal to a_s is computable. Thus A' and N are computable.

Now, if $x \in N$ then x is the coding location of a copy of $[6a_s]$ for some $s \in \omega$. Let K be the component of $\mathcal{G}_n$ that contains x. By Lemma 3.2.18, K contains either a copy of $[6\langle m, a_s\rangle + 1]$ for some $m \in \omega$ or a copy of $[6\langle m, a_s\rangle + 2]$ for some $m \in \omega$, but not both, and $x \in U \cap N$ if and only if K contains a copy of $[6\langle m, a_s\rangle + 2]$ for some $m \in \omega$. Thus $U \cap N$ is computable. □

3.2.29. Lemma. *Suppose that $s+1$ is an n-recovery stage, but not the first such stage, and that $r_{n,s+1} = r_{n,s} = i$. Let $t+1$ be the last n-recovery stage before stage $s+1$ and let $s_0 + 1 < s_1 + 1 < \cdots < s_m + 1$ be the stages in the interval $(t, s]$ at which n is active. For each $k \leqslant m$, let Y_k, X_k, Z_k, B_k, R_k and C_k be Y^i_{n,s_k}, $X^i_{s_k}$, Z^i_{n,s_k}, B^i_{n,s_k}, R^i_{n,s_k}, and C^i_{n,s_k}, respectively, and let Y'_k, X'_k, Z'_k, B'_k, R'_k and C'_k be* *the components of $\mathcal{A}_s^i$ that extend Y_k, X_k, Z_k, B_k, R_k and C_k, respectively. Then the following hold.*

1. *For every $k \leqslant m$, Y_k, X_k, Z_k, B_k, and C_k are components of $\mathcal{A}_t^i$, and so is R_0. For every $k, l \leqslant m$, $R'_k = R'_l$.*

2. *There exists a component $\widehat{R}_0$ of $\mathcal{G}_n[t]$ such that $\widehat{R}_0 \cong R_0$ and, for each $k \leqslant m$, there exist components $\widehat{Y}_k$, $\widehat{X}_k$, $\widehat{Z}_k$, $\widehat{B}_k$, and $\widehat{C}_k$ of $\mathcal{G}_n[t]$ such that $\widehat{Y}_k \cong Y_k$, $\widehat{X}_k \cong X_k$, $\widehat{Z}_k \cong Z_k$, $\widehat{B}_k \cong B_k$, and $\widehat{C}_k \cong C_k$.*

3. *Let $\widehat{R}'_0$ be the component of $\mathcal{G}_n[s]$ that extends $\widehat{R}_0$ and, for each $k \leqslant m$, let $\widehat{Y}'_k$, $\widehat{X}'_k$, $\widehat{Z}'_k$, $\widehat{B}'_k$, and $\widehat{C}'_k$ be the components of $\mathcal{G}_n[s]$ that extend $\widehat{Y}_k$, $\widehat{X}_k$, $\widehat{Z}_k$, $\widehat{B}_k$, and $\widehat{C}_k$, respectively. $\widehat{R}'_0 \cong R'_0$ and, for each $k \leqslant m$, $\widehat{Y}'_k \cong Y'_k$, $\widehat{X}'_k \cong X'_k$, $\widehat{Z}'_k \cong Z'_k$, $\widehat{B}'_k \cong B'_k$, and $\widehat{C}'_k \cong C'_k$.*

Proof. The first part of the lemma follows from the way Y^i_{n,s_k}, $X^i_{s_k}$, Z^i_{n,s_k}, B^i_{n,s_k}, R^i_{n,s_k}, and C^i_{n,s_k} are defined and Lemma 3.2.14. The second part of the lemma follows from the definition of n-recovery stage. We prove the third part of the lemma.

The two cases, $i = 0$ and $i = 1$, are similar. We do the case $i = 0$. Figure 3.3 might be helpful here.

By definition, $\widehat{R}_0$ and $\widehat{R}'_0$ are the special components of $\mathcal{G}_n[t]$ and $\mathcal{G}_n[s]$, respectively. Thus, since $r_{n,s+1} = r_{n,s} = 0$ and $s + 1$ is an n-recovery stage, $\widehat{R}'_0 \cong R'_0$. We now proceed by reverse induction, beginning with m.

It follows from the construction and the first part of the lemma that if K is taken from among $\widehat{R}'_0$, $\widehat{Y}'_k$, $\widehat{X}'_k$, $\widehat{Z}'_k$, $\widehat{B}'_k$, and $\widehat{C}'_k$, $k \leqslant m$, and $L \neq K$ is taken from among $\widehat{R}_0'$, $\widehat{Y}_l'$, $\widehat{X}_l'$, $\widehat{Z}_l'$, $\widehat{B}_l'$, and $\widehat{C}_l'$, $l \leqslant m$, then $K \ncong L$. Furthermore, if K is one of $\widehat{Y}_k'$, $\widehat{X}_k'$, $\widehat{Z}_k'$, $\widehat{B}_k'$, or $\widehat{C}_k'$, and L is a component of $\mathcal{A}_s^0$ such that $K \cong L$ then L is one of R_0', Y_l', X_l', Z_l', B_l', or C_l', $l \geqslant k$.

Thus, since we assume by induction that for all $j > k$, $\widehat{Y}_j' \cong Y_j'$, $\widehat{X}_j' \cong X_j'$, $\widehat{Z}_j' \cong Z_j'$, $\widehat{B}_j' \cong B_j'$, and $\widehat{C}_j' \cong C_j'$, we may assume that if K is one of $\widehat{Y}_k'$, $\widehat{X}_k'$, $\widehat{Z}_k'$, $\widehat{B}_k'$, or $\widehat{C}_k'$ and L is a component of $\mathcal{A}_s^0$ such that $K \cong L$ then L is one of R_0', Y_k', X_k', Z_k', B_k', or C_k'.

The only components among R_0', Y_k', X_k', Z_k', B_k', or C_k' that contain copies of $\widehat{C}_k$ are R_0' and C_k'. Since $\widehat{R}_0' \cong R_0'$, it must be the case that $\widehat{C}_k' \cong C_k'$.

The only components among R_0', Y_k', X_k', Z_k', B_k', or C_k' that contain copies of $\widehat{Y}_k$ are C_k' and Y_k'. Since $\widehat{C}_k' \cong C_k'$, it must be the case that $\widehat{Y}_k' \cong Y_k'$.

The only components among R_0', Y_k', X_k', Z_k', B_k', or C_k' that contain copies of $\widehat{X}_k$ are Y_k' and X_k'. Since $\widehat{Y}_k' \cong Y_k'$, it must be the case that $\widehat{X}_k' \cong X_k'$.

The only components among R_0', Y_k', X_k', Z_k', B_k', or C_k' that contain copies of $\widehat{Z}_k$ are X_k' and Z_k'. Since $\widehat{X}_k' \cong X_k'$, it must be the case that $\widehat{Z}_k' \cong Z_k'$.

The only components among R_0', Y_k', X_k', Z_k', B_k', or C_k' that contain copies of $\widehat{B}_k$ are Z_k' and B_k'. Since $\widehat{Z}_k' \cong Z_k'$, it must be the case that $\widehat{B}_k' \cong B_k'$. $\square$

3.2.30. Lemma. *Suppose that $s+1$ is an n-recovery stage such that $r_{n,s+1} = r_{n,s}$. Let $t+1$ be the last n-recovery stage before stage $s+1$ and let $j \in A[s] - A[t]$ be less than the number of n-recovery stages less than or equal to $t+1$. By the definition of n-recovery stage, there is a unique component K of $\mathcal{G}_n[t]$ isomorphic to $[6j]$. Let L be the component of $\mathcal{G}_n$ that extends K. Then L contains a copy of $[6\langle n, j\rangle + 2]$ if and only if $r_{n,s+1} = 0$.*

Proof. Let $i = r_{n,s+1}$. Let u be such that $j = a_u$. Since $t+1 \leqslant u < s$ and j is less than the number of n-recovery stages less than or equal to $t+1$, n is active at stage $u+1$. So, adopting the notation of Lemma 3.2.29, $K = \widehat{X}_k$ for some k. By Lemma 3.2.11, $L \cong \widehat{X}'_k$. Thus, by Lemma 3.2.29, $L \cong X'_k$. But X'_k is the component of $\mathcal{A}^i_s$ that extends X^i_u, so, by construction, X'_k contains a copy of $[6\langle n,j\rangle + 2]$ if and only if $i = 0$. □

3.2.31. Lemma. *Suppose that $\mathcal{G}_n \cong \mathcal{A}^0$. Let U be the image of U^0 under this isomorphism. Then either U is computable or $U \equiv_m A$.*

Proof. Let N be as in Lemma 3.2.28. Let C be the set of coding locations of copies of graphs of the form $[6j]$, $j \in \omega$, in $\mathcal{G}_n$ and let $M = C - N$. By Lemmas 3.2.8 and 3.2.28, C and N are computable, and hence so is M. By Lemma 3.2.28 and the fact that $U = (U \cap N) \cup (U \cap M)$, it is enough to show that either $U \cap M \equiv_{\mathrm{m}} A$ or $U \cap M$ is computable.

But, combining Lemmas 3.2.18 and 3.2.30, we conclude that

1. if $r_n = 0$ then an element x of M is in U if and only if, for some $j \in A$, x is the coding location of the first copy of $[6j]$ to appear in $\mathcal{G}_n$, so that $U \cap M \equiv_{\mathrm{m}} A$, while

2. if $r_n = 1$ then an element x of M is in U if and only if, for some $j \in \omega$, x is the coding location of the second copy of $[6j]$ to appear in $\mathcal{G}_n$, so that $U \cap M$ is computable.

□

Theorem 3.1.3 follows from Lemmas 3.2.7, 3.2.25, and 3.2.31. ■

3.3 Proof of Theorem 3.1.4

In this section we prove the following theorem.

3.1.4. Theorem. *Let* $\mathbf{a} > \mathbf{0}$ *be a c.e. degree. There exists an intrinsically c.e. relation* U *on the domain of a computable structure* $\mathcal{A}$ *of computable dimension* 2 *such that* $DgSp_{\mathcal{A}}(U) = \{\mathbf{0}, \mathbf{a}\}$*. In addition,* $\mathcal{A}$ *can be picked so that every c.e. presentation of* $\mathcal{A}$ *is computable, which implies that* $\mathcal{A}$ *has c.e. dimension* 2.

Proof. Let A be a c.e. set that is not computable and let $a_0, a_1, \ldots$ be a computable enumeration of A. Let $A[0] = \emptyset$, $A[s+1] = \{a_0, \ldots, a_s\}$. We wish to construct computable structures $\mathcal{A}^0$ and $\mathcal{A}^1$ and unary relations U^0 and U^1 on the domains of $\mathcal{A}^0$ and $\mathcal{A}^1$, respectively, so that the following properties hold.

(3.3.1) $\mathcal{A}^0 \cong \mathcal{A}^1$ via an isomorphism that carries U^0 to U^1.

(3.3.2) $U^0 \equiv_{\mathrm{m}} A$ and U^1 is computable.

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

(3.3.4) $\mathcal{A}^0$ is rigid.

(3.3.5) Every c.e. presentation of $\mathcal{A}^0$ with computable equality relation is computable.

The reason that (3.3.5) is enough to establish the last part of Theorem 3.1.4 is that we can let $\mathcal{A}$ be the result of adding to $\mathcal{A}^0$ the binary relation Q that holds of x and y if and only if $x \neq y$. Clearly, $\mathcal{A}$ shares all the relevant computable properties of $\mathcal{A}^0$, and any c.e. presentation of $\mathcal{A}$ restricts to a c.e. presentation of $\mathcal{A}^0$ with computable equality relation.

The construction in this section will be similar to the one in Section 3.2, as will the proof that properties (3.3.1) and (3.3.2) hold. (The construction in Section 3.2 also satisfied (3.3.4), but we did not mention this fact in that section because it was not needed to prove Theorem 3.1.3.) We will adopt the notation and conventions of Section 3.2 unless otherwise specified.

We now discuss the basic idea for satisfying (3.3.3). The construction in Section 3.2 was an injury-free one in which the satisfaction of (3.2.3) for a given $\mathcal{G}_n$ was handled by a single strategy, which worked with the components of $(\mathcal{A}^i)_n$ and acted independently from strategies for the satisfaction of (3.2.3) for other $\mathcal{G}_m$. The trade-off was forgoing any control of $(\mathcal{G}_n)_m$ for $m \neq n$.

In order to satisfy (3.3.3), we need to control more of $\mathcal{G}_n$ than just $(\mathcal{G}_n)_n$. In order to illustrate how we do this, we consider the following sample situation. We have two graphs $\mathcal{G}_0$ and $\mathcal{G}_1$. We proceed with a construction like that of Section 3.2, except that, in order for $\mathcal{G}_0$ to recover at stage $s+1$, we require not only that $\mathcal{G}_0[s]$ have the components that were necessary for 0-recovery in Section 3.2, but also those that were necessary for 1-recovery, and we do not allow 1-recovery unless there is 0-recovery, which means that 1 is not active unless 0 is active. We claim that we will succeed in controlling $(\mathcal{G}_0)_1$ in the same sense that we controlled $(\mathcal{G}_0)_0$

in Section 3.2.

An example should be helpful here. Suppose that $s < t < u < v$ are such that $s+1$ is a first stage, $v+1$ is the next recovery stage after stage $s+1$, $r_{0,s+1} = r_{0,v+1} = 0$, and $t+1$ and $u+1$ are the only two stages in the interval $(s+1, v+1)$ at which 0 is active. Suppose further that 1 is also active at stages $t+1$ and $u+1$. Notice that, since we do not allow 1 to be active unless 0 is active, $t+1$ and $u+1$ are the only two stages in the interval $(s+1, v+1)$ at which 1 is active.

Figures 3.6–3.8 picture what happens on the $\mathcal{A}^0$ side of the construction, depending on whether $r_{1,s+1} = 0$ or $r_{1,s+1} = 1$.

We are assuming the definition of recovery stage is such that the special component of $\mathcal{G}_0[s]$ is isomorphic to R^0_s, $\mathcal{G}_0[s]$ has a component $R_{1,s}$ isomorphic to $R^0_{1,s}$, and, for each $w = s, t, u$ and $i = 0, 1$, $\mathcal{G}_0[s]$ has components $Y_{i,w}$, X_w, $Z_{i,w}$, $B_{i,w}$, and $C_{i,w}$ isomorphic to $Y^0_{i,w}$, X^0_w, $Z^0_{i,w}$, $B^0_{i,w}$, and $C^0_{i,w}$, respectively.

Since $\mathcal{G}_0$ recovers at stage $v+1$ and $r_{0,v+1} = 0$, the special component of $\mathcal{G}_0[v]$ is isomorphic to $R^0_{0,s} \cdot C^0_{0,s} \cdot C^0_{0,t} \cdot C^0_{0,u}$. So, arguing as in Section 3.2, we see that, for each $w = s, t, u$, the components of $\mathcal{G}_0[v]$ that extend $Y_{0,w}$, X_w, $Z_{0,w}$, $B_{0,w}$, and $C_{0,w}$ are isomorphic to the components of $\mathcal{A}^0_v$ that extend $Y^0_{0,w}$, X^0_w, $Z^0_{0,w}$, $B^0_{0,w}$, and $C^0_{0,w}$, respectively. In other words, all of $(\mathcal{G}_0)_0$ goes in the same direction as $(\mathcal{A}^0)_0$.

We wish to show that $(\mathcal{G}_0)_1$ also goes in the same direction as $(\mathcal{A}^0)_1$. Let $R'_{1,s}$ be the component of $\mathcal{G}_0[v]$ that extends $R_{1,s}$ and, for each $w = s, t, u$, let $Y'_{1,w}$, X'_w, $Z'_{1,w}$, $B'_{1,w}$, and $C'_{1,w}$ be the components of $\mathcal{G}_0[v]$ that extend $Y_{1,w}$, X_w, $Z_{1,w}$, $B_{1,w}$, and $C_{1,w}$, respectively.

In the $r_{1,s+1} = 0$ case, we can argue as follows.

$$\begin{tikzcd}
Y^0_{0,s} \arrow[d] & X^0_s \arrow[d] & Z^0_{0,s} \arrow[d] \\
(Y^0_{0,s}, Y^0_{1,s}) \cdot X^0_s & X^0_s \cdot (Z^0_{0,s}, Z^0_{1,s}) & Z^0_{0,s} \cdot B^0_{0,s} \\
Y^0_{0,t} \arrow[d] & X^0_t \arrow[d] & Z^0_{0,t} \arrow[d] \\
(Y^0_{0,t}, Y^0_{1,t}) \cdot X^0_t & X^0_t \cdot (Z^0_{0,t}, Z^0_{1,t}) & Z^0_{0,t} \cdot B^0_{0,t} \\
Y^0_{0,u} \arrow[d] & X^0_u \arrow[d] & Z^0_{0,u} \arrow[d] \\
(Y^0_{0,u}, Y^0_{1,u}) \cdot X^0_u & X^0_u \cdot (Z^0_{0,u}, Z^0_{1,u}) & Z^0_{0,u} \cdot B^0_{0,u}
\end{tikzcd}$$

$$\begin{tikzcd}
B^0_{0,s} \arrow[d] & R^0_{0,s} \arrow[dd] & C^0_{0,s} \arrow[d] \\
B^0_{0,s} \cdot R^0_{0,s} & & C^0_{0,s} \cdot Y^0_{0,s} \\
B^0_{0,t} \arrow[d] & R^0_{0,s} \cdot C^0_{0,s} \arrow[dd] & C^0_{0,t} \arrow[d] \\
B^0_{0,t} \cdot R^0_{0,s} \cdot C^0_{0,s} & & C^0_{0,t} \cdot Y^0_{0,t} \\
B^0_{0,u} \arrow[d] & R^0_{0,s} \cdot C^0_{0,s} \cdot C^0_{0,t} \arrow[d] & C^0_{0,u} \arrow[d] \\
B^0_{0,u} \cdot R^0_{0,s} \cdot C^0_{0,s} \cdot C^0_{0,t} & R^0_{0,s} \cdot C^0_{0,s} \cdot C^0_{0,t} \cdot C^0_{0,u} & C^0_{0,u} \cdot Y^0_{0,u}
\end{tikzcd}$$

Figure 3.6: Recovery in a two-strategy scenario: $(\mathcal{A}^0)_0$

$$
\begin{tikzcd}
Y^0_{1,s} \arrow[d] & X^0_s \arrow[d] & Z^0_{1,s} \arrow[d] \\
(Y^0_{0,s}, Y^0_{1,s}) \cdot X^0_s & X^0_s \cdot (Z^0_{0,s}, Z^0_{1,s}) & Z^0_{1,s} \cdot B^0_{1,s} \\
Y^0_{1,t} \arrow[d] & X^0_t \arrow[d] & Z^0_{1,t} \arrow[d] \\
(Y^0_{0,t}, Y^0_{1,t}) \cdot X^0_t & X^0_t \cdot (Z^0_{0,t}, Z^0_{1,t}) & Z^0_{1,t} \cdot B^0_{1,t} \\
Y^0_{1,u} \arrow[d] & X^0_u \arrow[d] & Z^0_{1,u} \arrow[d] \\
(Y^0_{0,u}, Y^0_{1,u}) \cdot X^0_u & X^0_u \cdot (Z^0_{0,u}, Z^0_{1,u}) & Z^0_{1,u} \cdot B^0_{1,u}
\end{tikzcd}
$$

$$
\begin{tikzcd}
B^0_{1,s} \arrow[d] & R^0_{1,s} \arrow[dd] & C^0_{1,s} \arrow[d] \\
B^0_{1,s} \cdot R^0_{1,s} & & C^0_{1,s} \cdot Y^0_{1,s} \\
B^0_{1,t} \arrow[d] & R^0_{1,s} \cdot C^0_{1,s} \arrow[dd] & C^0_{1,t} \arrow[d] \\
B^0_{1,t} \cdot R^0_{1,s} \cdot C^0_{1,s} & & C^0_{1,t} \cdot Y^0_{1,t} \\
B^0_{1,u} \arrow[d] & R^0_{1,s} \cdot C^0_{1,s} \cdot C^0_{1,t} \arrow[d] & C^0_{1,u} \arrow[d] \\
B^0_{1,u} \cdot R^0_{1,s} \cdot C^0_{1,s} \cdot C^0_{1,t} & R^0_{1,s} \cdot C^0_{1,s} \cdot C^0_{1,t} \cdot C^0_{1,u} & C^0_{1,u} \cdot Y^0_{1,u}
\end{tikzcd}
$$

Figure 3.7: Recovery in a two-strategy scenario: $(\mathcal{A}^0)_1$ in case $r_{1,s+1} = 0$

$$
\begin{tikzcd}
Y^0_{1,s} \arrow[d] & X^0_s \arrow[d] & Z^0_{1,s} \arrow[d] \\
(Y^0_{0,s}, Y^0_{1,s}) \cdot X^0_s & X^0_s \cdot (Z^0_{0,s}, Z^0_{1,s}) & Z^0_{1,s} \cdot B^0_{1,s} \\
Y^0_{1,t} \arrow[d] & X^0_t \arrow[d] & Z^0_{1,t} \arrow[d] \\
(Y^0_{0,t}, Y^0_{1,t}) \cdot X^0_t & X^0_t \cdot (Z^0_{0,t}, Z^0_{1,t}) & Z^0_{1,t} \cdot B^0_{1,t} \\
Y^0_{1,u} \arrow[d] & X^0_u \arrow[d] & Z^0_{1,u} \arrow[d] \\
(Y^0_{0,u}, Y^0_{1,u}) \cdot X^0_u & X^0_u \cdot (Z^0_{0,u}, Z^0_{1,u}) & Z^0_{1,u} \cdot B^0_{1,u}
\end{tikzcd}
$$

$$
\begin{tikzcd}
B^0_{1,s} \arrow[ddr] & R^0_{1,s} \arrow[d] & C^0_{1,s} \arrow[d] \\
& R^0_{1,s} \cdot C^0_{1,s} & C^0_{1,s} \cdot Y^0_{1,s} \\
B^0_{1,t} \arrow[ddr] & B^0_{1,s} \cdot R^0_{1,s} \arrow[d] & C^0_{1,t} \arrow[d] \\
& B^0_{1,s} \cdot R^0_{1,s} \cdot C^0_{1,t} & C^0_{1,t} \cdot Y^0_{1,t} \\
B^0_{1,u} \arrow[d] & B^0_{1,t} \cdot B^0_{1,s} \cdot R^0_{1,s} \arrow[d] & C^0_{1,u} \arrow[d] \\
B^0_{1,u} \cdot B^0_{1,t} \cdot B^0_{1,s} \cdot R^0_{1,s} & B^0_{1,t} \cdot B^0_{1,s} \cdot R^0_{1,s} \cdot C^0_{1,u} & C^0_{1,u} \cdot Y^0_{1,u}
\end{tikzcd}
$$

Figure 3.8: Recovery in a two-strategy scenario: $(\mathcal{A}^0)_1$ in case $r_{1,s+1} = 1$

As we have mentioned above, for each $w = s, t, u$, $X'_w \cong X^0_w \cdot (Z^0_{0,w}, Z^0_{1,w})$, which implies that $Z'_{1,w} \cong Z^0_{1,w} \cdot B^0_{1,w}$. This in turn implies that $B'_{1,s} \cong B^0_{1,s} \cdot R^0_{1,s}$, $B'_{1,t} \cong B^0_{1,t} \cdot R^0_{1,s} \cdot C^0_{1,s}$, and $B'_{1,u} \cong B^0_{1,u} \cdot R^0_{1,s} \cdot C^0_{1,s} \cdot C^0_{1,t}$.

So the only component of $\mathcal{A}^0_v$ left for $R'_{1,s}$ to be isomorphic to is $R^0_{1,s} \cdot C^0_{1,s} \cdot C^0_{1,t} \cdot C^0_{1,u}$. This implies that, for each $w = s, t, u$, $C'_{1,w} \cong C^0_{1,w} \cdot Y^0_{1,w}$, which in turn implies that $Y'_{1,w} \cong (Y^0_{0,w}, Y^0_{1,w}) \cdot X^0_w$.

Thus, in this case, we see that $(\mathcal{G}_0)_1$ goes in the same direction as $(\mathcal{A}^0)_1$.

In the $r_{1,s+1} = 1$ case, the argument that $(\mathcal{G}_0)_1$ goes in the same direction as $(\mathcal{A}^0)_1$ is as follows. As before, for each $w = s, t, u$, $X'_w \cong X^0_w \cdot (Z^0_{0,w}, Z^0_{1,w})$, which implies that $Z'_{1,w} \cong Z^0_{1,w} \cdot B^0_{1,w}$.

This implies that $B'_{1,u} \cong B^0_{1,u} \cdot B^0_{1,t} \cdot B^0_{1,s} \cdot R^0_{1,s}$, which implies that $B'_{1,t} \cong B^0_{1,t} \cdot B^0_{1,s} \cdot R^0_{1,s} \cdot C^0_{1,u}$, which implies that $B'_{1,s} \cong B^0_{1,s} \cdot R^0_{1,s} \cdot C^0_{1,t}$, which implies that $R'_{1,s} \cong R^0_{1,s} \cdot C^0_{1,s}$.

Now, for for each $w = s, t, u$, we have $C'_{1,w} \cong C^0_{1,w} \cdot Y^0_{1,w}$, which implies that $Y'_{1,w} \cong (Y^0_{0,w}, Y^0_{1,w}) \cdot X^0_w$.

Thus, in this case also, $(\mathcal{G}_0)_1$ goes in the same direction as $(\mathcal{A}^0)_1$.

In either case, we have the same kind of control over $(\mathcal{G}_0)_1$ as over $(\mathcal{G}_0)_0$. Now assume that $\mathcal{G}_0 \cong \mathcal{A}^0$ and $\lim_s r_{0,s} = 0$. We claim that, if there are no other elements to the construction, so that, from some stage s on, all of $\mathcal{G}_0$ goes in the same direction as $\mathcal{A}^0$, then the unique isomorphism $f : \mathcal{A}^0 \to \mathcal{G}_0$ is computable. Indeed, the following is an effective procedure for computing $f(x)$ given $x \in \mathcal{A}^0$. Find the least stage $t \geqslant s$ such that x is contained in a component K of $\mathcal{A}^0_t$ and there is an isomorphism g from K to some component L of $\mathcal{G}_0[t]$. Such a stage must exist by the definition of 0-recovery, and, since all of $\mathcal{G}_0$ goes in the same direction as $\mathcal{A}^0$ from stage s on, $f(x) = g(x)$.

Of course, the strategy for $\mathcal{G}_0$ that we have just described works at the expense of the corresponding strategy for $\mathcal{G}_1$. Indeed, if $\mathcal{G}_0$ does not recover infinitely often then $\mathcal{G}_1$ is not allowed to recover infinitely often, even though it might be the case that $\mathcal{G}_1 \cong \mathcal{A}^0$.

We solve this problem in the standard way, by having multiple strategies for satisfying (3.3.3) for a given $\mathcal{G}_n$ and organizing these in a tree. More specifically, for each finite binary string σ, there will be a strategy for satisfying (3.3.3) for $\mathcal{G}_{|\sigma|}$. The string σ represents a guess about 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.

We will denote by (k) the component of $\mathcal{A}^i$ that extends the unique copy of $[k]$ in $\mathcal{A}^i_0$. By $\langle A^i \rangle_\sigma$ we will mean the union of the components of $\mathcal{A}^i$ that might potentially be used by the strategy for satisfying (3.3.3) for $\mathcal{G}_{|\sigma|}$ corresponding to σ. Once we give the formal details of the construction, it will be clear which components these are. As in Section 3.2, the notations $(A^i)_\sigma$ and $(\mathcal{G}_n)_\sigma$, $n \in \omega$, will refer to the union of those components that are actually used by the strategy corresponding to σ. By $\langle A^i \rangle$ we will mean the union of the components of $\mathcal{A}^i$ of the form $(6k)$, $k \in \omega$. (These are the components that might not be in $\langle \mathcal{A}^i \rangle_\sigma$ for any σ.)

We will not allow σ-recovery unless there is τ-recovery for all τ such that $\tau^\frown 0 \subseteq \sigma$. In this way, we will be able to control not only $(\mathcal{G}_{|\sigma|})_\sigma$, but also $(\mathcal{G}_{|\sigma|})_\tau$ for all τ such that $\sigma^\frown 0 \subseteq \tau$.

Now fix σ on the true path of the construction (which will be defined, as usual, as the leftmost path visited infinitely often) such that $\mathcal{G}_{|\sigma|} \cong \mathcal{A}^0$. These conditions on σ will imply that, for all $\tau \subsetneq \sigma$, τ recovers infinitely often if and only if $\tau^\frown 0 \subseteq \sigma$. They will also imply that $\sigma^\frown 0$ is on the true path, so that σ recovers infinitely often, and that $\lim_s r_{\sigma,s}$ exists (where $r_{\sigma,s}$ will be defined analogously to $r_{n,s}$). Let $i = \lim_s r_{\sigma,s}$ and let f be the unique isomorphism from $\mathcal{A}^i$ to $\mathcal{G}_{|\sigma|}$. By the comments in the previous paragraph, we will be able to compute both $f \upharpoonright \langle \mathcal{A}^i \rangle_\sigma$ and $f \upharpoonright \bigcup_{\tau \supseteq \sigma^\frown 0} \langle \mathcal{A}^i \rangle_\tau$. The argument is similar to that in the two-strategy scenario above.

Of course, this leaves the problem of uniformly computing $f \upharpoonright \langle \mathcal{A}^i \rangle_\tau$ for other τ, as well as $f \upharpoonright \langle A^i \rangle$. The key observation here is that, for a union T of finite components of $\mathcal{A}^i$, if there exists a computable bound on the last stage (if any) at which each component of T participates in an operation then $f \upharpoonright T$ is computable. This is because if a component K of T does not participate in operations after stage s then K is a component of $\mathcal{A}^i_s$, and hence the unique embedding from K into $\mathcal{G}_n$ can be found effectively.

Thus our strategy will be to break up the domain of $\mathcal{A}^i$ into finitely many c.e. sets and show that the restriction of f to each of these sets is computable. Most of the cases will be handled as indicated in the previous paragraph.

We begin by looking at $\langle A^i \rangle$. As in Section 3.2, we will have a computable bound $h(k)$ such that if $(6k)$ has not participated in an operation by stage $h(k)$ then, whenever it does participate in an operation, σ is active. Let T_0 be the union of those components $(6k)$ of $\langle A^i \rangle$ that do not participate in an operation by stage $h(k)$. Then $f \upharpoonright T_0$ will be computable for the same reason as $f \upharpoonright \langle \mathcal{A}^i \rangle_\sigma$.

On the other hand, since no component of $\langle A^i \rangle$ will participate in operations more than once, $f \restriction (\langle A^i \rangle - T_0)$ will be computable because $h(k)$ will be a computable bound on the last stage at which a component $(6k)$ of $\langle A^i \rangle - T_0$ participates in an operation. Thus $f \restriction \langle A^i \rangle$ will be computable.

Now let T_1 be the union of all $\langle \mathcal{A}^i \rangle_\tau$ such that τ is to the left of σ. By the definition of the true path, only finitely many components of T_1 will ever participate in operations, and those that do, will do so only finitely often. Thus there will exist a computable bound on the last stage at which each component of T_1 participates in an operation, and hence $f \restriction T_1$ will be computable.

Let T_2 be the union of all $\langle \mathcal{A}^i \rangle_\tau$ such that $\tau^\frown 1 \subseteq \sigma$. As in Section 3.2, the fact that there are only finitely many τ-recovery stages will imply that only finitely many components of T_2 participate in operations, and those that do, do so only finitely often. Thus there will exist a computable bound on the last stage at which each component of T_2 participates in an operation, and hence $f \restriction T_2$ will be computable.

Let T_3 be the union of all $\langle \mathcal{A}^i \rangle_\tau$ such that τ is to the right of $\sigma^\frown 0$. Every time the construction moves to the left of τ, we will guarantee that a certain set of components of $\langle \mathcal{A}^i \rangle_\tau$ will never again participate in an operation, in such a way that if the construction moves to the left of τ infinitely often then every component of $\langle \mathcal{A}^i \rangle_\tau$ will eventually be guaranteed never again to participate in an operation. (We call this process *initialization.*) Since $\sigma^\frown 0$ is on the true path, this will mean that there exists a computable bound on the last stage at which each component of T_3 participates in an operation, and thus $f \restriction T_3$ will be computable.

We are left with the case of $\langle \mathcal{A}^i \rangle_\tau$ such that $\tau^\frown 0 \subseteq \sigma$. We will show, much as in

Section 3.2, that, for each such τ, if $r_{\tau,s}$ has a limit then $(\mathcal{A}^i)_\tau$ has a unique infinite component S^i_τ, while if $r_{\tau,s}$ does not have a limit then all components of $(\mathcal{A}^i)_\tau$ are finite. Let T_4 be the union of the S^i_τ, $\tau^\frown 0 \subseteq \sigma$, $r_{\tau,s}$ has a limit. Given a copy K of $[m]$ contained in a component C of T_4 with top x, we will be able to find effectively the unique copy L of $[m]$ in the component of $\mathcal{G}_\sigma$ with top $f(x)$, and f will extend the unique isomorphism from K to L. Since T_4 has only finitely many components, this will mean that $f \restriction T_4$ is computable.

Finally, let T_5 be the union of all finite components of $\langle \mathcal{A}^i \rangle_\tau$, $\tau^\frown 0 \subseteq \sigma$. Examining the construction in Section 3.2, we see that, once a finite component K of $(\mathcal{A}^i)_n$ participates in an operation at a stage s, we can effectively find a stage t such that K does not participate in an operation after stage t. Indeed, we can take t to be the least n-isomorphism recovery stage such that, for some $u < t$, every component of $(\mathcal{A}^i)_n$ that participates in an operation at stage t has participated in an operation in the interval $[u, t)$ and K does not participate in an operation in the interval $[u, t]$.

The analogous situation will hold in this section, but this will not be quite enough to show that $f \restriction T_5$ is computable. We will also need an effective procedure that, for each component K of T_5, gives us a stage s such that if K has not participated in an operation by stage s then it will not participate in an operation after stage s. In order to do this, every time τ recovers, we will guarantee that a certain set of components of $\langle \mathcal{A}^i \rangle_\tau$ that have not yet participated in an operation will never participate in an operation, in such a way that if τ recovers infinitely often then every singleton component of $\langle \mathcal{A}^i \rangle_\tau$ will eventually be guaranteed never to participate in an operation. Thus $f \restriction T_5$ will be computable.

We now give the formal definitions that will be used below.

For the sake of satisfying (3.3.5), we need a new kind of building block, whose use will be made clear shortly. (Basically, if $\mathcal{G}$ is a c.e. graph with computable equality relation and K and L are different components of $\mathcal{G}[s]$, $s \in \omega$, then it cannot be the case that K and L are both extended by a component of the form $K \cdot L$ in $\mathcal{G}$. However, K and L might both be extended by the same component of $\mathcal{G}$ if this component is of the form $K \cdot (L)$, for example, since the fact that there is no edge from the top of K to the top of L in $\mathcal{G}[s]$ does not mean that the same is true in $\mathcal{G}$. We will avoid this possibility by only performing operations of the form $K \cdot (L_0, \dots, L_k)$ when K is of the form $[n]^+$, $n \in \omega$, as defined below.)

3.3.1. Definition. The directed graph $[n]^+$ consists of the following nodes and edges.

1. A copy of $[n]$ with top x.

2. For each $i \leqslant n$, $i+1$ many nodes $x_{i,0}, \dots, x_{i,i}$, with an edge from x to $x_{i,0}$ and, for each $j < i$, an edge from $x_{i,j}$ to $x_{i,j+1}$. We call $x_{i,i}$ the *i-attachment node* of $[n]^+$.

Figure 3.9 shows $[2]^+$ as an example.

We also need a new version of Definition 3.2.5.

3.3.2. Definition. Let $\mathcal{G}$ be a computable structure in the language of directed graphs whose domain is co-infinite.

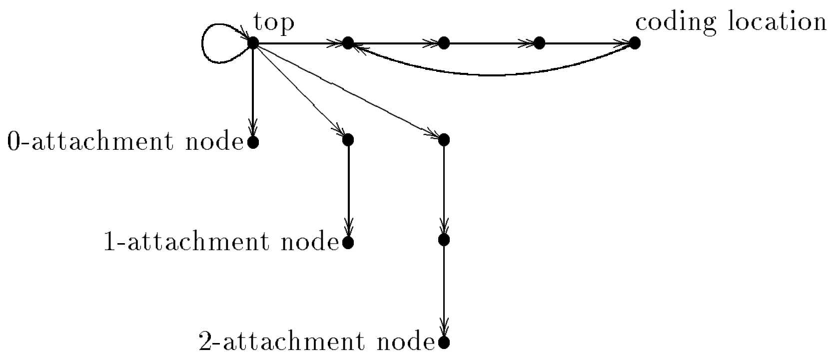


Figure 3.9: $[2]^+$

Let $K_0, K_1, \ldots, K_n$ and L be components of $\mathcal{G}$ isomorphic to $[k_0], [k_1], \ldots, [k_n]$ and $[l]^+$, respectively, where $k_0, k_1, \ldots, k_n, l \in \omega$ and $n \leqslant l$. We define two operations, each of which takes $\mathcal{G}$ to a new computable structure extending $\mathcal{G}$.

- The operation $(K_0, K_1, \ldots, K_n) \cdot L$ consists of creating a new copy of $[l]^+$, using the top of K_i as the i-attachment node for $i \leqslant n$ and numbers not in the domain of $\mathcal{G}$ as the other nodes, and otherwise leaving $\mathcal{G}$ unchanged.

- The operation $L \cdot (K_0, K_1, \ldots, K_n)$ consists of creating a new copy of $[k_i]$ for each $i \leqslant n$, using the i-attachment node of L as the top and numbers not in the domain of $\mathcal{G}$ as the other nodes, and otherwise leaving $\mathcal{G}$ unchanged.

We define the **L**- and **R**-operations as in Definition 3.2.6, except that we now require that X be of the form $[k]^+$, $k \in \omega$.

Fix a computable one-to-one function from $2^{<\omega}$ onto $\omega - \{0\}$ and let $\ulcorner\sigma\urcorner$ denote the image under this function of the string σ.

3.3.3. 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 $[x]$ or $[x]^+$ for any $x \in \omega$.

2. C contains a copy of $[6\langle\ulcorner\sigma\urcorner, j\rangle+3]$, $j \in \omega$, or a copy of $[6\langle\ulcorner\sigma\urcorner, j, k\rangle+l]$, $j, k \in \omega$, $l \in \{1, 2, 4, 5\}$.

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

Let k be the number of times σ has been initialized (defined below) before stage t. Suppose there is a least stage $s \leqslant t$ such that $\mathcal{G}_{|\sigma|}[s]$ has a component K isomorphic to $[6\langle\ulcorner\sigma\urcorner, k\rangle + 3]$. 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 component K isomorphic to $[6\langle\ulcorner\sigma\urcorner, k\rangle + 3]$ then we call the component of $\mathcal{G}_{|\sigma|}$ that extends K the *σ-special component of* $\mathcal{G}_{|\sigma|}$.

For $\sigma, \tau \in 2^{\leqslant\omega}$, $\sigma \leqslant_{\mathrm{L}} \tau$ means that either $\sigma \subseteq \tau$ or there exists an $n < |\sigma|, |\tau|$ such that $\sigma(m) = \tau(m)$ for all $m < n$, $\sigma(n) = 0$, and $\tau(n) = 1$. If $\sigma \leqslant_{\mathrm{L}} \tau$ and $\sigma \nsubseteq \tau$ then we say that σ is *to the left* of τ and that τ is *to the right* of σ.

We now proceed with the construction of $\mathcal{A}^0$, $\mathcal{A}^1$, U^0, and U^1. As before, we make it a convention that all numbers added to the domain of $\mathcal{A}^i_s$ at stage $s+1$ to get $\mathcal{A}^i_{s+1}$ are greater than s. It will be easy to check as we go along that the following are properties of the construction.

1. For each $s \in \omega$, $\mathcal{A}^0_s \cong \mathcal{A}^1_s$ and no component of $\mathcal{A}^i_s$ is embeddable in another component of $\mathcal{A}^i_s$.

2. Let $t < s$. No component of $\mathcal{A}^i_t$ isomorphic to one of $[6a_s]^+$ or $[6\langle j, a_s, k\rangle + l]$, $j, k \in \omega$, $l \in \{1, 2, 4, 5\}$, participates in an operation at stage $t + 1$.

stage 0. Let $\mathcal{A}^0_0$ and $\mathcal{A}^1_0$ be computable structures with co-infinite domains, each consisting of one copy of $[6k + l]$ and one of $[6k]^+$ for each $k \in \omega$ and $0 < l < 6$. For each $\sigma \in 2^{<\omega}$, let $r_{\sigma,0} = 0$.

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

Define the string $\sigma[s + 1] \in 2^{[0,s]}$ by recursion as follows, beginning with $n = 0$. Let $\sigma = \sigma[s + 1] \upharpoonright n$. Say that $s + 1$ is a *σ-recovery stage* if all of the following conditions hold.

1. Every τ such that $\tau^\frown 0 \subseteq \sigma$ has recovered at least $|\sigma| + 1$ many times.

2. $\mathcal{G}_n[s]$ has a σ-special component isomorphic to some component of $\mathcal{A}^0_s$.

3. If $\tau \supseteq \sigma^\frown 0$ has not yet recovered since the last time it was initialized and $|\tau| \leqslant recov(\sigma, s)$ then $\mathcal{G}_n[s]$ has a component isomorphic to $[6\langle \ulcorner\tau\urcorner, init(\tau, s)\rangle + 3]$.

4. $(\mathcal{G}_n[s])_\sigma \cong (\mathcal{A}^0_s)_\sigma$.

5. $(\mathcal{G}_n[s])_{\supseteq\sigma^\frown 0} \cong (\mathcal{A}^0_s)_{\supseteq\sigma^\frown 0}$.

6. Let τ be such that either $\tau = \sigma$ or both $\tau \supseteq \sigma^\frown 0$ and $|\tau| \leqslant recov(\sigma, s)$. Let $j \notin A[s]$ be less than or equal to $recov(\tau, s)$. There is a component of $\mathcal{G}_n[s]$

isomorphic to $[6j]^+$ and, for each $l \in \{1, 2, 4, 5\}$, there is a component of $\mathcal{G}_n[s]$ isomorphic to $[6\langle \ulcorner \tau \urcorner, j, c(\tau, s)\rangle + l]$.

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

For each σ such that $s+1$ is a σ-recovery stage, proceed as follows. For $i = 0, 1$, let $S^i_{\sigma,s}$ be the component of $\mathcal{A}^i_s$ that is isomorphic to the σ-special component of $\mathcal{G}_{|\sigma|}[s]$. If $s+1$ is either the first σ-recovery stage ever or the first σ-recovery stage since the last time σ was initialized then let $r_{\sigma,s+1} = 0$. Otherwise, proceed as follows. Let $i = r_{\sigma,s}$ and let $t+1$ be the last σ-recovery stage before stage $s+1$. If $S^i_{\sigma,s}$ extends $S^i_{\sigma,t}$ then let $r_{\sigma,s+1} = i$, and otherwise let $r_{\sigma,s+1} = 1 - i$.

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

Declare each σ to the right of $\sigma[s+1]$ to have been *initialized*. For each $\sigma \leqslant_{\mathrm{L}} \sigma[s+1]$, if there has been a σ-recovery stage since the last time σ was initialized, $a_s \geqslant |\sigma|$, and a_s is less than the number of σ-recovery stages less than or equal to $s+1$ then say that σ is *active* at stage $s+1$.

For $i = 0, 1$, let X^i_s be the component of $\mathcal{A}^i_s$ isomorphic to $[6a_s]^+$.

Let $\sigma_0, \ldots, \sigma_m$ be all the strings that are active at stage $s+1$. For $i = 0, 1$ and $j \leqslant m$, let $Y^i_{\sigma_j,s}$ and $Z^i_{\sigma_j,s}$ be the components of $\mathcal{A}^i_s$ isomorphic to $[6\langle \ulcorner \sigma_j \urcorner, a_s, c(\sigma_j, s)\rangle + 1]$ and $[6\langle \ulcorner \sigma_j \urcorner, a_s, c(\sigma_j, s)\rangle + 2]$, respectively.

For each $j \leqslant m$, let $t_j + 1 \leqslant s + 1$ be the last σ_j-recovery stage. We say that $s+1$ is a σ_j-*first stage* if it is the first stage after stage t_j at which σ_j is active.

We say that $s+1$ is a σ_j-*change stage* if it is a σ_j-first stage and one of the following holds: $t_j + 1$ was the first σ_j-recovery stage ever, $t_j + 1$ was the first σ_j-recovery stage since the last time σ_j was initialized, or $r_{\sigma_j,t_j+1} \neq r_{\sigma_j,t_j}$.

We say that $s+1$ is a *σ_j-isomorphism recovery stage* if it is a σ_j-first stage but not a σ_j-change stage and one of the following conditions holds.

1. The last σ_j-first stage before stage $s+1$ was a σ_j-change stage.

2. There has been at least one stage at which σ_j was active after the last σ_j-isomorphism recovery stage and before stage $s+1$.

For each $j \leqslant m$ we define components $B^i_{\sigma_j,s}$ and $C^i_{\sigma_j,s}$, $i = 0,1$. There are two cases.

1. $s+1$ is a σ_j-isomorphism recovery stage. If the first condition in the definition of σ_j-isomorphism recovery stage holds then let $t+1$ be the last σ_j-first stage, and otherwise let $t+1$ be the first stage after the last σ_j-isomorphism recovery stage at which σ_j was active. There are two subcases.

   (a) If $r_{\sigma_j,s+1} = 0$ then let $C^0_{\sigma_j,s}$ be the component of $\mathcal{A}^0_s$ that extends $B^0_{\sigma_j,t}$ and let $C^1_{\sigma_j,s}$ be its isomorphic image in $\mathcal{A}^1_s$. For $i = 0,1$, let $B^i_{\sigma_j,s}$ be the component of $\mathcal{A}^i_s$ isomorphic to $[6\langle \sigma_j, a_s, c(\sigma_j,s)\rangle + 4]$.

   (b) If $r_{\sigma_j,s+1} = 1$ then let $B^1_{\sigma_j,s}$ be the component of $\mathcal{A}^1_s$ that extends $C^1_{\sigma_j,t}$ and let $B^0_{\sigma_j,s}$ be its isomorphic image in $\mathcal{A}^0_s$. For $i = 0,1$, let $C^i_{\sigma_j,s}$ be the component of $\mathcal{A}^i_s$ isomorphic to $[6\langle \sigma_j, a_s, c(\sigma_j,s)\rangle + 5]$.

2. $s+1$ is not a σ_j-isomorphism recovery stage. For $i = 0,1$, let $B^i_{\sigma_j,s}$ be the component of $\mathcal{A}^i_s$ isomorphic to $[6\langle \sigma_j, a_s, c(\sigma_j,s)\rangle + 4]$ and let $C^i_{\sigma_j,s}$ be the component of $\mathcal{A}^i_s$ isomorphic to $[6\langle \sigma_j, a_s, c(\sigma_j,s)\rangle + 5]$.

For each $j \leqslant m$, proceed as follows. Let $i = r_{\sigma_j,s+1}$ and let $t+1 \leqslant s+1$ be the last σ_j-recovery stage. Let $R^i_{\sigma_j,s}$ be the component of $\mathcal{A}^i_s$ that extends $S^i_{\sigma_j,t}$ and let $R^{1-i}_{\sigma_j,s}$ be its isomorphic image in $\mathcal{A}^{1-i}_s$.

Now perform

$$\mathbf{L}(Y^0_{\sigma_0,s}, \ldots, Y^0_{\sigma_m,s}; X^0_s; Z^0_{\sigma_0,s}, \ldots, Z^0_{\sigma_m,s}; B^0_{\sigma_0,s}, R^0_{\sigma_0,s}, C^0_{\sigma_0,s};$$
$$B^0_{\sigma_1,s}, R^0_{\sigma_1,s}, C^0_{\sigma_1,s}; \ldots; B^0_{\sigma_m,s}, R^0_{\sigma_m,s}, C^0_{\sigma_m,s})$$

on $\mathcal{A}^0_s$ to get $\mathcal{A}^0_{s+1}$ and perform

$$\mathbf{R}(Y^1_{\sigma_0,s}, \ldots, Y^1_{\sigma_m,s}; X^1_s; Z^1_{\sigma_0,s}, \ldots, Z^1_{\sigma_m,s}; B^1_{\sigma_0,s}, R^1_{\sigma_0,s}, C^1_{\sigma_0,s};$$
$$B^1_{\sigma_1,s}, R^1_{\sigma_1,s}, C^1_{\sigma_1,s}; \ldots; B^1_{\sigma_m,s}, R^1_{\sigma_m,s}, C^1_{\sigma_m,s})$$

on $\mathcal{A}^1_s$ to get $\mathcal{A}^1_{s+1}$. (If no σ is active at stage $s+1$ then, for $j = 0, 1$, let Y^j_s, Z^j_s, B^j_s, R^j_s, and C^j_s be the components of $\mathcal{A}^j_s$ isomorphic to $[6\langle 0, a_s, s\rangle + 1]$, $[6\langle 0, a_s, s\rangle + 2]$, $[6\langle 0, a_s, s\rangle + 4]$, $[6\langle 0, s\rangle + 3]$, and $[6\langle 0, a_s, s\rangle + 5]$, respectively. Perform $\mathbf{L}(Y^0_s; X^0_s; Z^0_s; B^0_s; R^0_s; C^0_s)$ on $\mathcal{A}^0_s$ to get $\mathcal{A}^0_{s+1}$ and perform $\mathbf{R}(Y^1_s; X^1_s; Z^1_s; B^1_s; R^1_s; C^1_s)$ on $\mathcal{A}^1_s$ to get $\mathcal{A}^1_{s+1}$.)

Put the coding location of the copy of $[6a_s]$ in $\mathcal{A}^0_0$ into U^0 and put the coding location of the copy of $[6a_s]$ in $\mathcal{A}^1_{s+1} - \mathcal{A}^1_s$ into U^1.

This completes the construction. Let $\mathcal{A}^0 = \bigcup_{s\in\omega} \mathcal{A}^0_s$ and $\mathcal{A}^1 = \bigcup_{s\in\omega} \mathcal{A}^1_s$. Since, for each $s \in \omega$ and $i = 0, 1$, all numbers in $\mathcal{A}^i_{s+1} - \mathcal{A}^i_s$ are greater than s, $\mathcal{A}^0$ and $\mathcal{A}^1$ are computable. We now wish to argue that properties (3.3.1)–(3.3.5) are satisfied. Theorem 3.1.4 will then follow immediately.

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 $\sigma[s] \in TP$.

The following lemma, which shows that (3.3.2) holds, has the same proof as Lemma 3.2.7.

3.3.4. Lemma. *$U^0 \equiv_m A$ and U^1 is computable.*

Property (3.3.5) is also easy to establish.

3.3.5. Lemma. *Let $\mathcal{G}$ be a c.e. presentation of $\mathcal{A}^0$ with computable equality relation. Then $\mathcal{G}$ is computable.*

Proof. Let $S = \{6k \mid k \in \omega\}$. Let $x_0, x_1 \in \mathcal{G}$. Wait until a stage s in the enumeration of $\mathcal{G}$ such that, for each $i = 0, 1$, x_i is in a copy of either $[n_i]$ for some $n_i \notin S$ or $[n_i]^+$ for some $n_i \in S$. It is easy to check from the definition of $\mathcal{A}^0$ that, for each $i = 0, 1$, there is an edge from x_i to x_{1-i} if and only if there already is such an edge at stage s. □

Lemmas 3.2.8–3.2.10 and 3.2.13 still hold, as do the following versions of Lemmas 3.2.11, 3.2.12, and 3.2.14–3.2.18. In all cases, the reasoning is basically the same as before.

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

3.3.7. 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* $S^i_{\sigma,s}$*.*

3.3.8. Lemma. *Suppose that* $r_{\sigma,s} = i$ *for all* $s > t$*,* σ *is not initialized at any stage after stage* t*, and* σ *is active at stages* $s_0 + 1$ *and* $s_1 + 1$*, where* $s_1 > s_0 \geqslant t$*. Then* R^i_{σ,s_1} *extends* R^i_{σ,s_0}*.*

3.3.9. 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 stage* u*. Let* $t+1$ *be the last* σ*-recovery stage before stage* $s+1$*. If* $r_{\sigma,s} = 0 \neq r_{\sigma,s+1}$ *then* $S^0_{\sigma,s}$ *extends* $B^0_{\sigma,v}$ *for some* $v \in [t,s)$*. Similarly, if* $r_{\sigma,s} = 1 \neq r_{\sigma,s+1}$ *then* $S^1_{\sigma,s}$ *extends* $C^1_{\sigma,v}$ *for some* $v \in [t,s)$*.*

3.3.10. Lemma. *Suppose that* $r_{\sigma,t} = 0$ *(resp.* $r_{\sigma,t} = 1$*) for all* $t \geqslant s_0$*. Then no component of* $(\mathcal{A}^0)_\sigma$ *(*$(\mathcal{A}^1)_\sigma$*) can participate in an operation more than twice after stage* s_0 *unless it extends* $R^0_{\sigma,t}$ *(*$R^1_{\sigma,t}$*) for some* $t \geqslant s_0$*, while no component of* $(\mathcal{A}^1)_\sigma$ *(*$(\mathcal{A}^0)_\sigma$*) can participate in an operation more than twice after stage* s_0 *unless it extends* $C^1_{\sigma,t}$ *(*$B^0_{\sigma,t}$*) for some* $t \geqslant s_0$ *such that* $t+1$ *is a* σ*-isomorphism recovery stage.*

3.3.11. Lemma. *Let* s_0 *be a stage after which* σ *is never initialized. Suppose that* $s_0 \leqslant s < t < v$ *are such that* $s+1$ *is a* σ*-isomorphism recovery stage,* $r_{\sigma,u} = r_{\sigma,s+1}$ *for all* $u > s$*,* $t+1$ *is the next stage after stage* $s+1$ *at which* σ *is active, and* $v+1$ *is the next* σ*-isomorphism recovery stage after stage* $s+1$*. For* $i = 0,1$*, let* B^i*,* R^i*, and* C^i *be the components of* $\mathcal{A}^i_{t+1}$ *that extend* $B^i_{\sigma,t}$*,* $R^i_{\sigma,t}$*, and* $C^i_{\sigma,t}$*, respectively, and* *let* $\widehat{B}^i$, $\widehat{R}^i$, *and* $\widehat{C}^i$ *be the components of* $\mathcal{A}^i_v$ *that extend* B^i, R^i, *and* C^i, *respectively. If* $r_{\sigma,s+1} = 0$ *then* $\widehat{B}^0 \cong B^0$ *and* $\widehat{R}^1 \cong R^1$, *while if* $r_{\sigma,s+1} = 1$ *then* $\widehat{C}^1 \cong C^1$ *and* $\widehat{R}^0 \cong R^0$.

3.3.12. Lemma. *Let* x *be the coding location of a copy of* $[6a_s]$ *in component* K *of* $\mathcal{A}^i$. *Either* K *contains a copy of* $[6\langle n, a_s, k\rangle + 1]$ *for some* $n, k \in \omega$, *in which case* $x \notin U^i$, *or* K *contains a copy of* $[6\langle n, a_s, k\rangle + 2]$ *for some* $n, k \in \omega$, *in which case* $x \in U^i$.

The following lemmas will also be useful.

3.3.13. Lemma. *If* σ *is to the left of* TP *then* $(\mathcal{A}^i)_{\supseteq\sigma}$ *is finite.*

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

Proof. It is easy to check that if a component of $(\mathcal{A}^i)_\sigma$ participates in operations at stages $s < t$ and not at any stage in the interval (s, t) then there are no σ-change stages in (s, t).

Now let t be the first stage after stage s at which σ is active. (If there are no such stages then we are done.) Then t is a σ-change stage, and hence not a σ-isomorphism recovery stage. It is easy to check that, together with the fact that σ is initialized at stage $s+1$, this implies that none of the components of $(\mathcal{A}^i)_\sigma$ that participate in operations at stages before stage $s+1$ can participate in an operation at stage t. The lemma now follows from the remark in the previous paragraph. $\square$

3.3.15. Lemma. *If σ is to the right of TP then $(\mathcal{A}^i)_\sigma$ has no infinite components.*

We now wish to show that (3.3.1) holds. In the course of doing so, we will also be able to show that (3.3.4) holds. It follows from Lemmas 3.2.10, 3.2.13, and 3.3.12 that, to show that (3.3.1) holds, it is enough to show that for each infinite component of $\mathcal{A}^i$ there is an isomorphic component of $\mathcal{A}^{1-i}$. As before, the first step in establishing this result is characterizing the infinite components of $\mathcal{A}^i$. By Lemmas 3.3.13 and 3.3.15, if σ is not on TP then no component of $(\mathcal{A}^i)_\sigma$ is infinite, so we can restrict our attention to the components of $(\mathcal{A}^i)_\sigma$, $\sigma \in TP$.

The following lemma has the same proof as Lemma 3.2.19, with Lemmas 3.3.7 and 3.3.9 in place of Lemmas 3.2.12 and 3.2.15, respectively.

3.3.16. Lemma. *Let $\sigma \in TP$. If $r_{\sigma,s}$ does not have a limit then no component of $(\mathcal{A}^i)_\sigma$ is infinite.*

Thus, the only components of $\mathcal{A}^i$ that can be infinite are those components that are in $(\mathcal{A}^i)_\sigma$ for some $\sigma \in TP$ such that $r_{\sigma,s}$ has a limit and σ is active infinitely often. So, by the comments preceding Lemma 3.3.16, to establish that (3.3.1) holds, it is enough to show that if $\sigma \in TP$, $r_{\sigma,s}$ has a limit, and σ is active infinitely often, then, for each $i = 0, 1$, there is exactly one infinite component S^i_σ of $(\mathcal{A}^i)_\sigma$ and $S^0_\sigma \cong S^1_\sigma$. As we will argue below, this will also be enough to show that (3.3.4) holds.

3.3.17. Lemma. *Let $\sigma \in TP$. There are infinitely many σ-recovery stages if and only if σ is active infinitely often.*

Proof. By definition, σ is not active at a stage $s+1$ unless a_s is less than the number of σ-recovery stages less than or equal to $s+1$. Thus, if there are finitely many σ-recovery stages then σ cannot be active infinitely often.

For the other direction, suppose that there are infinitely many σ-recovery stages but only finitely many stages at which σ is active. Let s be a stage after which σ is never active or initialized and such that there has been a σ-recovery stage since the last time σ was initialized. Now, given $x > |\sigma|$, let $t+1$ be the first stage after stage s by which there have been $x+1$ many σ-recovery stages. Then $x \in A \Leftrightarrow x \in A[t]$. But this means that A is computable, contrary to hypothesis. $\square$

3.3.18. Lemma. *If $\sigma \in TP$ is active infinitely often and $r_{\sigma,s}$ has a limit then there are infinitely many σ-isomorphism recovery stages.*

Proof. If σ is active infinitely often then, by Lemma 3.3.17, there are infinitely many σ-recovery stages, and thus infinitely many σ-first stages. The fact that $r_{\sigma,s}$ has a limit and that σ is initialized only finitely often implies that only finitely many of these can be σ-change stages. The lemma now follows directly from the definition of σ-isomorphism recovery stage. $\square$

The next lemma has the same proof as Lemma 3.2.22, with Lemmas 3.3.6, 3.3.8, and 3.3.11 in place of Lemmas 3.2.11, 3.2.14, and 3.2.17.

3.3.19. Lemma. *Suppose that $\sigma \in TP$ is active infinitely often and s and i are such that σ is not initialized after stage s and $r_{\sigma,t} = r_{\sigma,s} = i$ for all $t \geqslant s$. By Lemma 3.3.18, there are infinitely many σ-isomorphism recovery stages. Let $s_0+1 < s_1+1 < \cdots$ be the σ-isomorphism recovery stages after stage s. For each $j \in \omega$, let*

$t_j + 1$ be the next stage after stage $s_j + 1$ at which σ is active. (Note that $t_j < s_{j+1}$ for all $j \in \omega$.) For $t \geqslant t_0$, let K_t^l be the component of $\mathcal{A}_t^l$ that extends R_{σ,t_0}^l. Then $K_{t_j}^l = R_{\sigma,t_j}^l$ for all $j \in \omega$.

Now assume the hypotheses of Lemma 3.3.19 and adopt its notation. Let S_σ^l be the component of $\mathcal{A}^l$ that extends R_{σ,s_0}^l.

3.3.20. Lemma. *S_σ^l is the only infinite component of $(\mathcal{A}^l)_\sigma$.*

Proof. This follows immediately from Lemmas 3.2.10, 3.3.10, and 3.3.19 and the observation that, for all $j \in \omega$, if $i = 0$ in the hypotheses of Lemma 3.3.19 then R_{σ,t_j}^1 extends C_{σ,s_j}^1, while if $i = 1$ then R_{σ,t_j}^0 extends B_{σ,s_j}^0. □

3.3.21. Lemma. $S_\sigma^0 \cong S_\sigma^1$.

Proof. Directly from Lemma 3.3.19, since, by definition, $R_{\sigma,t_j}^0 \cong R_{\sigma,t_j}^1$ for all $j \in \omega$, and $S_\sigma^i = \bigcup_{j\in\omega} R_{\sigma,t_j}^i$ for $i = 0, 1$. □

As we have argued above, Lemmas 3.3.20 and 3.3.21 suffice to establish that (3.3.1) holds.

3.3.22. Lemma. *$\mathcal{A}^0 \cong \mathcal{A}^1$ via an isomorphism that carries U^0 to U^1.*

We are now also in a position to show that (3.3.4) holds.

3.3.23. Lemma. *$\mathcal{A}^0$ is rigid.*

Proof. It is easy to check that every component of $\mathcal{A}^0$ is rigid. Thus it is enough to show that no two components of $\mathcal{A}^0$ are isomorphic. By Lemma 3.2.13, for each $s \in \omega$, no component of $\mathcal{A}_s^0$ is embeddable in another component of $\mathcal{A}_s^0$, which implies that no two finite components of $\mathcal{A}^0$ are isomorphic. Since the only infinite components of $\mathcal{A}^0$ are the S^0_σ defined above and S^0_σ contains a copy of $[6\langle\ulcorner\tau\urcorner, k\rangle + 3]$ for some $k \in \omega$ if and only if $\tau = \sigma$, it is also the case that no two infinite components of $\mathcal{A}^0$ are isomorphic. $\square$

We are left with showing that property (3.3.3) is satisfied. This is where this proof differs significantly from that in Section 3.2. We begin by showing that if $\sigma \in TP$ and $\mathcal{G}_{|\sigma|} \cong \mathcal{A}^0$ then $\lim_s r_{\sigma,s}$ is well-defined.

3.3.24. Lemma. *If $\sigma \in TP$ and $\mathcal{G}_{|\sigma|} \cong \mathcal{A}^0$ 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 $\mathcal{G}_{|\sigma|} \cong A^0$ then $\mathcal{G}_{|\sigma|}$ has a σ-special component. Now 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 $s_0 = 0$.) By Lemma 3.3.17, there is a stage $s_1 > s_0$ such that σ is not active at any stage $t \geqslant s_1$. By the definition of TP, there is a stage $s_2 \geqslant s_1$ satisfying the following conditions: every τ such that $\tau^\frown 0 \subseteq \sigma$ has recovered at least $|\sigma| + 1$ many times by stage s_2, $\mathcal{G}_{|\sigma|}[s_2]$ has a σ-special component, and σ is not initialized at any stage greater than or equal to s_2. If $m = a_u$ for some $u > s_2$ then let $s = u + 1$; otherwise, let $s = s_2$.

By the definition of s, the first condition in the definition of 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 3.3.6, each such component is finite. Thus, if the second condition in the definition of 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 $\mathcal{A}^0$.

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 recovered since the last time it was initialized never again recovers, and hence there is a component of $\mathcal{A}^0$ isomorphic to $[6\langle\ulcorner\tau\urcorner, init(\tau,t)\rangle + 3]$. Since there are only finitely many τ such that $|\tau| \leqslant recov(\sigma, s)$, if the third condition in the definition of σ-recovery stage is not eventually satisfied after stage s then $\mathcal{G}_{|\sigma|} \ncong \mathcal{A}^0$.

Now consider $(\mathcal{A}^0)_\sigma$. Again by Lemma 3.3.6, $(\mathcal{A}^0)_\sigma$ is finite. So if the fourth condition in the definition of σ-recovery stage is not eventually satisfied after stage s then $(\mathcal{G}_{|\sigma|})_\sigma \ncong (\mathcal{A}^0)_\sigma$.

Since we are assuming that there are only finitely many σ-recovery stages, $\sigma^\frown 1 \in TP$. Thus, by Lemma 3.3.13, $(\mathcal{A}^0_s)_{\supseteq\sigma^\frown 0}$ is finite. So if the fifth condition in the definition of σ-recovery stage is not eventually satisfied after stage s then $(\mathcal{G}_{|\sigma|})_{\supseteq\sigma^\frown 0} \ncong (\mathcal{A}^0)_{\supseteq\sigma^\frown 0}$.

Finally, let τ be such that either $\tau = \sigma$ or both $\tau \supseteq \sigma^\frown 0$ and $|\tau| \leqslant recov(\sigma, s)$. Let $j \notin A[s]$ be less than or equal to $recov(\tau, s)$. Clearly, $c(\tau, t)$ reaches a limit $c(\tau)$. By the choice of s, $j \notin A[s] \Rightarrow j \notin A$. So, for each $l \in \{1, 2, 4, 5\}$, there is a unique component of $\mathcal{A}^0$ that contains a copy of $[6\langle\ulcorner\tau\urcorner, j, c(\tau)\rangle + l]$, and it is isomorphic to $[6\langle\ulcorner\tau\urcorner, j, c(\tau)\rangle + l]$. Similarly, there is a unique component of $\mathcal{A}^0$ that contains a copy of $[6j]^+$, and it is isomorphic to $[6j]^+$. Thus, if the last condition in the definition of σ-recovery stage is not eventually satisfied after stage s then there is a component of $\mathcal{A}^0$ that is not isomorphic to any component of $\mathcal{G}_{|\sigma|}$.

In any case, $\mathcal{G}_{|\sigma|}$ cannot be isomorphic to $\mathcal{A}^0$, 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 σ-recovery stages $v < t+1 < u+1$ such that there is a stage in $(t, u]$ at which σ is active, the σ-special component of $\mathcal{G}_{|\sigma|}[u]$ properly extends the σ-special component of $\mathcal{G}_{|\sigma|}[t]$. But, by Lemma 3.3.17, σ is active at infinitely many stages. This establishes the second part of the lemma. $\square$

3.3.25. Lemma. *If $\sigma \in TP$ and $\mathcal{G}_{|\sigma|} \cong \mathcal{A}^0$ then $\lim_s r_{\sigma,s}$ is well-defined.*

Proof. This follows immediately from Lemmas 3.3.16 and 3.3.24. $\square$

Now fix $\sigma \in TP$ such that $\mathcal{G}_{|\sigma|} \cong \mathcal{A}^0$. Let $n = |\sigma|$. By Lemma 3.3.25, $r = \lim_s r_{\sigma,s}$ is well-defined. We wish to show that $\mathcal{G}_n$ is computably isomorphic to $\mathcal{A}^r$. The two cases, $r = 0$ and $r = 1$, are essentially the same. We will assume that $r = 0$.

Let $f : \mathcal{A}^0 \cong \mathcal{G}_n$. Since $\mathcal{A}^0$ is rigid, f is the unique isomorphism from $\mathcal{A}^0$ to $\mathcal{G}_n$, so we need to show that f is computable.

As outlined at the beginning of this section, our strategy will be to break up the domain of $\mathcal{A}^0$ into a finite number of c.e. sets and show that the restriction of f to each of these sets is computable. (If P is c.e. then we say that $f \restriction P$ is computable if there exists a partial computable function Φ such that $x \in P \Rightarrow \Phi(x)\!\downarrow = f(x)$.) We will need the following definition.

3.3.26. Definition. Let $k, s \in \omega$. We denote by (k) and $(k)_s$ the components of $\mathcal{A}^0$ and $\mathcal{A}^0_s$, respectively, that extend the unique copy of $[k]$ in $\mathcal{A}^0_0$.

For $D \subseteq \omega$, let $P_D = \bigcup_{k \in D} (k)$.

Note that, for any $k, s \in \omega$, $(k)_s$ is finite. Note also that, since every component of $\mathcal{A}^0$ extends some component of $\mathcal{A}^0_0$, $\bigcup_{k\in\omega}(k) = \mathcal{A}^0$; similarly, $\bigcup_{k\in\omega}(k)_s = \mathcal{A}^0_s$. It is not the case that $k \neq l \Rightarrow (k) \neq (l)$, but, as we will see, this will not matter for our purposes.

The following result justifies our approach.

3.3.27. Lemma. *Let $D_0, \ldots, D_m$ be computable subsets of ω such that $\bigcup_{i=0}^m D_i = \omega$. If for each $i \leqslant m$, $f \restriction P_{D_i}$ is computable, then f is computable.*

Proof. Since $D_0, \ldots, D_m$ are computable, $P_{D_0}, \ldots, P_{D_m}$ are c.e.. Since $\bigcup_{i=0}^m D_i = \omega$, $\bigcup_{i=0}^m P_{D_i} = \mathcal{A}^0$. Thus, to compute $f(x)$ for some $x \in \mathcal{A}^0$, all we need to do is wait until x is enumerated into some P_{D_i} and then compute $(f \restriction P_{D_i})(x)$. □

We will partition ω into the pairwise disjoint computable sets $D_0, \ldots, D_6$ shown in Table 3.1. (The corresponding P_{D_i} will not be pairwise disjoint, but this does not matter, since it was not required to prove Lemma 3.3.27.) We will then show that, for each $i \leqslant 6$, $f \restriction P_{D_i}$ is computable, which will enable us to apply Lemma 3.3.27 to conclude that f is computable. The following two lemmas provide a useful tool for our task.

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

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

Table 3.1: $D_0, \dots, D_6$

D_0	$\{6\langle 0,k\rangle+3,\ 6\langle 0,j,k\rangle+l \mid j,k\in\omega,\ l\in\{1,2,4,5\}\}$
D_1	$\{6\langle\ulcorner\tau\urcorner,k\rangle+3,\ 6\langle\ulcorner\tau\urcorner,j,k\rangle+l \mid \tau$ to the left of σ or $\tau^\frown 1\subseteq\sigma,$ $j,k\in\omega,\ l\in\{1,2,4,5\}\}$
D_2	$\{6\langle\ulcorner\tau\urcorner,k\rangle+3,\ 6\langle\ulcorner\tau\urcorner,j,k\rangle+l \mid \tau$ to the right of $\sigma^\frown 0,$ $j,k\in\omega,\ l\in\{1,2,4,5\}\}$
D_3	$\{m\in\omega \mid (m)$ is the unique infinite component of some $(\mathcal{A}^0)_\tau,\ \tau^\frown 0\subseteq\sigma\}$
D_4	$\{6\langle\ulcorner\tau\urcorner,j\rangle+3,\ 6\langle\ulcorner\tau\urcorner,j,k\rangle+l \mid \tau^\frown 0\subseteq\sigma,\ j,k\in\omega,\ l\in\{1,2,4,5\}\}-D_3$
D_5	$\{6k \mid k<n\}\cup\{6a_s \mid a_s\geqslant recov(\sigma,s+1)$ or s is less than the first σ-recovery stage after the last time σ is initialized$\}$
D_6	$\{6\langle\ulcorner\tau\urcorner,j\rangle+3,\ 6\langle\ulcorner\tau\urcorner,j,k\rangle+l \mid \tau=\sigma$ or $\sigma^\frown 0\subseteq\tau,\ j,k\in\omega,\ l\in\{1,2,4,5\}\}\cup$ $\{6k \mid k\in\omega\}-D_5$

3.3.29. Lemma. *Let $D\subseteq\omega$ and $h:D\to\omega$ be computable. Suppose that, for each $k\in D$ and $t\geqslant h(k)$, $(k)_t$ does not participate in an operation at stage $t+1$. Then $f\restriction P_D$ is computable.*

Proof. Let $x\in P_D$ and let $k\in D$ be such that $x\in(k)$. By Lemma 3.3.28, $(k)_{h(k)}\cong(k)$, so (k) is finite. Since no component of $\mathcal{A}^0$ is embeddable in another component of $\mathcal{A}^0$, there is a unique finite set $T\subset\mathcal{G}_n$ such that there is an isomorphism $g_x:(k)\cong T$. Clearly, g_x can be extended to an isomorphism from $\mathcal{A}^0$ to $\mathcal{G}_n$. By the uniqueness of f, $f(x)=g_x(x)$. It is easy to see that g_x can be computably determined given $x\in P_D$. Thus $f\restriction P_D$ is computable. $\square$

3.3.30. Lemma. *Let D_0 consist of all numbers of the form $6\langle 0,k\rangle+3$ or $6\langle 0,j,k\rangle+l$, $j,k\in\omega$, $l\in\{1,2,4,5\}$. Then $f\restriction P_{D_0}$ is computable.*

Proof. Let m be of the form $6\langle 0,k\rangle+3$ or $6\langle 0,j,k\rangle+l$, $j,k\in\omega$, $l\in\{1,2,4,5\}$. Recall that, for all $\tau\in 2^{<\omega}$, $\ulcorner\tau\urcorner\neq 0$. Thus, the only time (m) can participate in an operation is at stage $k+1$. (This happens if no element of $2^{<\omega}$ is active at stage $k+1$.) So if we define $h(m)=k+1$ then the hypotheses of Lemma 3.3.29 are satisfied. $\square$

3.3.31. Lemma. *There exists a stage s such that if τ is either to the left of σ or such that $\tau^\frown 1\subseteq\sigma$ then τ is not active after stage s.*

Proof. Let $\tau\in 2^{<\omega}$. By definition, τ is not active at a stage t unless a_s is less than the number of τ-recovery stages less than or equal to t. Thus, τ cannot be active more often than it recovers.

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. The lemma now follows by the remark in the previous paragraph. $\square$

3.3.32. Lemma. *Let D_1 be the set of all numbers of the form $6\langle\ulcorner\tau\urcorner,k\rangle+3$ or $6\langle\ulcorner\tau\urcorner,j,k\rangle+l$, τ to the left of σ or $\tau^\frown 1\subseteq\sigma$, $j,k\in\omega$, $l\in\{1,2,4,5\}$. Then $f\restriction P_{D_1}$ is computable.*

Proof. Let s be as in Lemma 3.3.31. By Lemma 3.3.6, for each $m\in D_1$ and $t\geqslant s$, $(m)_t$ does not participate in an operation at stage $t+1$. So if we let $h(m)=s$ for all $m \in D_1$ then the hypotheses of Lemma 3.3.29 are satisfied. □

3.3.33. Lemma. *Let τ be to the right of $\sigma^\frown 0$. Let m be of the form $6\langle \ulcorner\tau\urcorner, k\rangle + 3$ or $6\langle \ulcorner\tau\urcorner, j, k\rangle + l$, $l \in \{1, 2, 4, 5\}$. Let $s+1$ be the stage at which τ is initialized for the $(k+1)^{st}$ time. Then (m) does not participate in an operation after stage s.*

Proof. If a singleton component of $\mathcal{A}_t^0$ of the form $[6\langle \ulcorner\tau\urcorner, p\rangle + 3]$, participates in an operation at a stage $t+1 > s$ then $p = init(\tau, t) \geqslant k+1$. If a singleton component of $\mathcal{A}_t^0$ of the form $[6\langle \ulcorner\tau\urcorner, j, p\rangle + l]$, $l \in \{1, 2, 4, 5\}$, participates in an operation at a stage $t+1 > s$ then $p = c(\tau, t) \geqslant init(\tau, t) \geqslant k+1$. So if (m) 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 (m) 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 3.3.14. □

3.3.34. Lemma. *Let D_2 be the set of all numbers of the form $6\langle \ulcorner\tau\urcorner, k\rangle + 3$ or $6\langle \ulcorner\tau\urcorner, j, k\rangle + l$, τ to the right of $\sigma^\frown 0$, $j, k \in \omega$, $l \in \{1, 2, 4, 5\}$. Then $f \upharpoonright P_{D_2}$ is computable.*

Proof. If $m \in D_2$ is of the form $6\langle \ulcorner\tau\urcorner, k\rangle + 3$ or $6\langle \ulcorner\tau\urcorner, j, k\rangle + l$ 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 3.3.33, the hypotheses of Lemma 3.3.29 are satisfied. □

If $\tau^\frown 0 \subseteq \sigma$ and $r_{\tau,s}$ has a limit then, by Lemma 3.3.20, $(\mathcal{A}^0)_\tau$ has a unique infinite component. On the other hand, if $\tau^\frown 0 \subseteq \sigma$ and $r_{\tau,s}$ does not have a limit then, by Lemma 3.3.16, all components of $(\mathcal{A}^0)_\tau$ are finite. Let D_3 be the set of all $m \in \omega$ such that (m) is the unique infinite component of some $\tau^\frown 0 \subseteq \sigma$ such that $r_{\tau,s}$ has a limit. Note that D_3 is finite.

3.3.35. Lemma. *$f \restriction P_{D_3}$ is computable.*

Proof. Let $T = \{x_0, \dots, x_m\}$ be the tops of the components of P_{D_3}. Given $x \in P_{D_3} - T$, find the unique k such that x is in a copy K of $[k]$. The top of K is x_i for some $i \leqslant m$. Let L be the unique copy of $[k]$ in $\mathcal{G}_n$ with top $f(x_i)$ and let g_x be the unique isomorphism form K to L. Then $f(x) = g_x(x)$. It is easy to see that g_x can be computably determined given $x \in P_{D_3} - T$. Since T is finite, this implies that $f \restriction P_{D_3}$ is computable. $\square$

The following lemma is easy to check from the construction.

3.3.36. Lemma. *Let $s \in \omega$. If all the components of $(\mathcal{A}^0_t)_\tau$ that participate in an operation at a τ-isomorphism recovery stage $t+1 > s$ have participated in an operation in the interval $(s,t]$ then no component of $(\mathcal{A}^0_t)_\tau$ that participates in an operation in the interval $(0,s]$ but not in the interval $(s,t+1]$ can participate in an operation after stage t.*

3.3.37. Lemma. *Let D_4 be the set of all numbers not in D_3 that are of the form $6\langle\ulcorner\tau\urcorner, j\rangle + 3$ or $6\langle\ulcorner\tau\urcorner, j, k\rangle + l$, $\tau^\frown 0 \subseteq \sigma$, $j,k \in \omega$, $l \in \{1,2,4,5\}$. Then $f \restriction P_{D_4}$ is computable.*

Proof. Let $m \in D_4$. If m is of the form $6\langle\ulcorner\tau\urcorner, j, k\rangle + l$, $l \in \{1, 2, 4, 5\}$, then let s be the first stage by which τ has recovered $k + 1$ many times. If (m) has not participated in an operation before stage s then, by the same reasoning as in the proof of Lemma 3.3.33, it does not participate in an operation after stage s. In this case, let $h(m) = s$.

Now suppose that m is of the form $6\langle\ulcorner\tau\urcorner, j\rangle + 3$. Let $init(\tau) = \lim_s init(\tau, s)$. If $j < init(\tau)$ then let s be the least stage by which τ has been initialized $k + 1$ many times. Arguing as in the proof of Lemma 3.3.33, we see that (m) does not participate in an operation after stage s. In this case, let $h(m) = s$. If $j > init(\tau)$ then (m) never participates in an operation. In this case, let $h(m) = 0$.

If $h(m)$ has not yet been defined then (m) participates in an operation at least once. However, since (m) is finite, (m) participates in operations only finitely often. So there exist stages $s < t + 1$ such that $t + 1$ is a τ-isomorphism recovery stage, all the components of $(\mathcal{A}^0_t)_\tau$ that participate in an operation at stage $t + 1$ have participated in an operation in the interval $(s, t]$, and (m) does not participate in an operation in the interval $(s, t + 1]$. Then, by Lemma 3.3.36, (m) does not participate in an operation after stage t. In this case, let $h(m) = t$.

Now the hypotheses of Lemma 3.3.29 are satisfied. $\square$

3.3.38. Lemma. *Let D'_5 be the set of all numbers of the form $6k$, $k < n$. Let D''_5 be set of all numbers of the form $6a_s$ such that $a_s \geqslant recov(\sigma, s+1)$ or s is less than the first σ-recovery stage after the last time σ is initialized. Let $D_5 = D'_5 \cup D''_5$. Then $f \restriction P_{D_5}$ is computable.*

Proof. By Lemma 3.3.6, there is a stage t such that no $(6k)$, $k < n$, participates in an operation after stage t. For $k < n$, let $h(6k) = t$.

For $6a_s \in D''_5$, let $h(6a_s) = s + 1$. Again by Lemma 3.3.6, $(6a_s)$ does not participate in an operation after stage $h(6a_s)$.

Since D'_5 is finite, h is computable, and hence the hypotheses of Lemma 3.3.29 are satisfied. $\square$

Let D'_6 be the set of all numbers of the form $6\langle\ulcorner\tau\urcorner, j\rangle + 3$ or $6\langle\ulcorner\tau\urcorner, j, k\rangle + l$, $\tau = \sigma$ or $\sigma^\frown 0 \subseteq \tau$, $j, k \in \omega$, $l \in \{1, 2, 4, 5\}$. Let D''_6 be the set of all numbers of the form $6k$ that are not in D_5. Let $D_6 = D'_6 \cup D''_6$. In order to show that f is computable, we are left with showing that $f \restriction P_{D_6}$ is computable. Roughly, the idea is to show that, once $r_{\sigma,s}$ has reached its final value, $\mathcal{G}_n$ and $\mathcal{A}^0$ always go in the same direction at stages at which components of P_{D_6} participate in operations. The argument is similar to that in the proof of Lemma 3.2.29, but we now need to worry about the $\tau \supseteq \sigma^\frown 0$ case as well as the $\tau = \sigma$ case.

3.3.39. 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, for all $s \geqslant u$, $r_{\sigma,s} = 0$. Let $s + 1 > u$ be a σ-recovery stage, but not the first such stage after stage u. Let $t + 1$ be the last σ-recovery stage before stage $s + 1$ and let $s_0 + 1 < s_1 + 1 < \cdots < s_m + 1$ be the stages in the interval $(t, s]$ at which τ is active. For each $k \leqslant m$, let Y_k, X_k, Z_k, B_k, R_k and C_k be Y^0_{τ,s_k}, $X^0_{s_k}$, Z^0_{τ,s_k}, B^0_{τ,s_k}, R^0_{τ,s_k}, and C^0_{τ,s_k}, respectively, and let Y'_k, X'_k, Z'_k, B'_k, R'_k and C'_k be the components of $\mathcal{A}^0_s$ that extend Y_k, X_k, Z_k, B_k, R_k and C_k, respectively. Then the following hold.*

1. *For every* $k \leqslant m$, Y_k, X_k, Z_k, B_k, *and* C_k *are components of* $\mathcal{A}_t^0$, *and so is* R_0. *If* $r_{\tau,t+1} = 0$ *then, for every* $k, l \leqslant m$, $R'_k = R'_l$. *If* $r_{\tau,t+1} = 1$ *then, for every* $0 < k \leqslant m$, $R'_k = B'_{k-1}$.

2. *There exists a component* $\widehat{R}_0$ *of* $\mathcal{G}_n[t]$ *such that* $\widehat{R}_0 \cong R_0$ *and, for each* $k \leqslant m$, *there exist components* $\widehat{Y}_k$, $\widehat{X}_k$, $\widehat{Z}_k$, $\widehat{B}_k$, *and* $\widehat{C}_k$ *of* $\mathcal{G}_n[t]$ *such that* $\widehat{Y}_k \cong Y_k$, $\widehat{X}_k \cong X_k$, $\widehat{Z}_k \cong Z_k$, $\widehat{B}_k \cong B_k$, *and* $\widehat{C}_k \cong C_k$.

3. *Let* $\widehat{R}'_0$ *be the component of* $\mathcal{G}_n[s]$ *that extends* $\widehat{R}_0$ *and, for each* $k \leqslant m$, *let* $\widehat{Y}'_k$, $\widehat{X}'_k$, $\widehat{Z}'_k$, $\widehat{B}'_k$, *and* $\widehat{C}'_k$ *be the components of* $\mathcal{G}_n[s]$ *that extend* $\widehat{Y}_k$, $\widehat{X}_k$, $\widehat{Z}_k$, $\widehat{B}_k$, *and* $\widehat{C}_k$, *respectively.* $\widehat{R}'_0 \cong R'_0$ *and, for each* $k \leqslant m$, $\widehat{Y}'_k \cong Y'_k$, $\widehat{X}'_k \cong X'_k$, $\widehat{Z}'_k \cong Z'_k$, $\widehat{B}'_k \cong B'_k$, *and* $\widehat{C}'_k \cong C'_k$.

Proof. There are no τ-recovery stages in the interval $(t+1, s]$, which implies that if τ is initialized in the interval $(t, s]$ then this initialization happens after stage $s_m + 1$. So the first part of the lemma follows from the way Y^0_{τ,s_k}, $X^0_{s_k}$, Z^0_{τ,s_k}, B^0_{τ,s_k}, R^0_{τ,s_k}, and C^0_{τ,s_k} are defined. The second part of the lemma follows from the definition of σ-recovery stage. We prove the third part of the lemma.

In the $\tau = \sigma$ case, the proof is the same as the proof of the third part of Lemma 3.2.29. We handle the $\tau \supseteq \sigma^\frown 0$ case. There are two subcases.

First suppose that $r_{\tau,t+1} = 0$.

Let $k \leqslant m$. Since σ is active whenever τ is active, it follows from the $\tau = \sigma$ case that $\widehat{X}'_k \cong X'_k$.

The only components of $\mathcal{A}_s^0$ that contain copies of $\widehat{Z}_k$ are X'_k and Z'_k. Since $\widehat{X}'_k \cong X'_k$, it must be the case that $\widehat{Z}'_k \cong Z'_k$.

The only components of $\mathcal{A}_s^0$ that contain copies of $\widehat{B}_k$ are Z_k' and B_k'. Since $\widehat{Z}_k' \cong Z_k'$, it must be the case that $\widehat{B}_k' \cong B_k'$.

The only components of $\mathcal{A}_s^0$ that contain copies of $\widehat{R}_0$ are R_0' and $B_0', \dots, B_m'$. We have shown that, for every $k \leqslant m$, $\widehat{B}_k' \cong B_k'$. Thus it must be the case that $\widehat{R}_0' \cong R_0'$.

We now proceed by reverse induction, beginning with m. Let $k \leqslant m$. Assume by induction that, for all $j > k$, $\widehat{Y}_j' \cong Y_j'$, $\widehat{X}_j' \cong X_j'$, $\widehat{Z}_j' \cong Z_j'$, $\widehat{B}_j' \cong B_j'$, and $\widehat{C}_j' \cong C_j'$. As in the $\tau = \sigma$ case, we may assume that if K is one of $\widehat{Y}_k'$, $\widehat{X}_k'$, $\widehat{Z}_k'$, $\widehat{B}_k'$, or $\widehat{C}_k'$ and L is a component of $\mathcal{A}_s^0$ such that $K \cong L$ then L is one of R_0', Y_k', X_k', Z_k', B_k', or C_k'.

We have already seen that $\widehat{X}_k' \cong X_k'$, $\widehat{Z}_k' \cong Z_k'$, $\widehat{B}_k' \cong B_k'$, and $\widehat{R}_0' \cong R_0'$.

The only components among R_0', Y_k', X_k', Z_k', B_k', or C_k' that contain copies of $\widehat{C}_k$ are R_0' and C_k'. Since $\widehat{R}_0' \cong R_0'$, it must be the case that $\widehat{C}_k' \cong C_k'$.

The only components among R_0', Y_k', X_k', Z_k', B_k', or C_k' that contain copies of $\widehat{Y}_k$ are C_k' and Y_k'. Since $\widehat{C}_k' \cong C_k'$, it must be the case that $\widehat{Y}_k' \cong Y_k'$.

Now suppose that $r_{\tau,t+1} = 1$.

As before, it follows from the $\tau = \sigma$ case that $\widehat{X}_k' \cong X_k'$ for all $k \leqslant m$.

We first proceed by reverse induction, beginning with m, to show that $\widehat{Z}_k' \cong Z_k'$, $\widehat{B}_k' \cong B_k'$, and $\widehat{R}_0' \cong R_0'$. Let $k \leqslant m$. We may assume by induction that, for all $k < j \leqslant m$, $\widehat{B}_j' \cong B_j'$.

The only components of $\mathcal{A}_s^0$ that contain copies of $\widehat{Z}_k$ are X_k' and Z_k'. Since $\widehat{X}_k' \cong X_k'$, it must be the case that $\widehat{Z}_k' \cong Z_k'$.

The only components of $\mathcal{A}_s^0$ that contain copies of $\widehat{B}_k$ are Z_k', and B_j', $k \leqslant j \leqslant m$.

Since $\widehat{Z}'_k \cong Z'_k$ and, for all $k < j \leqslant m$, $\widehat{B}'_j \cong B'_j$, it must be the case that $\widehat{B}'_k \cong B'_k$.

The only components of $\mathcal{A}^0_s$ that contain copies of $\widehat{R}_0$ are R'_0 and $B'_0, \ldots, B'_m$. We have shown that, for every $k \leqslant m$, $\widehat{B}'_k \cong B'_k$. Thus it must be the case that $\widehat{R}'_0 \cong R'_0$.

Now let $0 < k \leqslant m$. The only components of $\mathcal{A}^0_s$ that contain copies of $\widehat{C}_k$ are B'_{k-1} and C'_k. Since $\widehat{B}'_{k-1} \cong B'_{k-1}$, it must be the case that $\widehat{C}'_k \cong C'_k$.

The only components of $\mathcal{A}^0_s$ that contain copies of $\widehat{C}_0$ are R'_0 and C'_0. Since $\widehat{R}'_0 \cong R'_0$, it must be the case that $\widehat{C}'_0 \cong C'_0$.

Let $k \leqslant m$. The only components of $\mathcal{A}^0_s$ that contain copies of $\widehat{Y}_k$ are C'_k and Y'_k. Since $\widehat{C}'_k \cong C'_k$, it must be the case that $\widehat{Y}'_k \cong Y'_k$. □

The following lemma can be easily checked.

3.3.40. Lemma. *Let $m \in \omega$ be of the form $6\langle\ulcorner\tau\urcorner, j\rangle + 3$ or $6\langle\ulcorner\tau\urcorner, j, k\rangle + l$, $\tau = \sigma$ or $\sigma^\frown 0 \subseteq \tau$, $j, k \in \omega$, $l \in \{1, 2, 4, 5\}$. If $(m)_s$ participates in an operation at stage $s+1$ then it is one of $Y^0_{\tau,s}$, $Z^0_{\tau,s}$, $B^0_{\tau,s}$, $R^0_{\tau,s}$, or $C^0_{\tau,s}$.*

Let $m \in D''_6$. If $(m)_s$ participates in an operation at stage $s+1$ then it is X^0_s and σ is active at stage $s+1$.

3.3.41. Lemma. *Let u be a stage after which σ is never initialized and such that, for all $s \geqslant u$, $r_{\sigma,s} = 0$. Let $s+1 > u$ be a σ-recovery stage and let $t+1$ be the next σ-recovery stage after stage $s+1$. Let $m \in D_6$. Suppose there exists a component L of $\mathcal{G}_n[s]$ that is isomorphic to $(m)_s$. Then the component L' of $\mathcal{G}_n[t]$ that extends L is isomorphic to $(m)_t$.*

Proof. If (m) does not participate in an operation in the interval $(s, t]$ then $(m)_t \cong (m)_s$. Since $L' \supseteq L$, $(m)_t$ is not embeddable in another component of $\mathcal{A}^0_t$, and, by convention (see page 58), $\mathcal{G}_n[t]$ is embeddable in $\mathcal{A}^0_t$, this means that $L' \cong (m)_t$.

Otherwise, the lemma follows from Lemmas 3.3.39 and 3.3.40. $\square$

3.3.42. Lemma. *Let u be a stage after which σ is never initialized and such that, for all $s \geqslant u$, $r_{\sigma,s} = 0$. Let $x \in P_{D_6}$. There exists a σ-recovery stage $s + 1 \geqslant u$ such that x is contained in $(k)_s$ for some $k \in D_6$ and there exists an isomorphic component L of $\mathcal{G}_n[s]$. For any such s, if we let g be the unique isomorphism from $(k)_s$ to L then $f(x) = g(x)$.*

Proof. If x is contained in a finite component of $\mathcal{A}^0$ then the existence of s follows from the fact that $\mathcal{G}_n \cong \mathcal{A}^0$. Otherwise, there are $t \geqslant s > u$ such that $s + 1$ is a σ-recovery stage, there are no σ-recovery stages in the interval $(s + 1, t + 1]$, x is contained in $(k)_t$, $k \in D_6$, and $(k)_t$ is involved in an operation at stage $t + 1$. Now it follows from Lemma 3.3.39 that x is contained in $(k)_s$ and there exists an isomorphic component L of $\mathcal{G}_n[s]$.

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 3.3.41 and induction, we see that, for each $i \geqslant 0$, there exists a unique isomorphism $g_i : (k)_{s_i} \cong L_i$. 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 (k) to L'. By the uniqueness of f, $f(x) = g'(x) = g(x)$. $\square$

3.3.43. Lemma. *$f \restriction P_{D_6}$ is computable.*

Proof. Let u be a stage after which σ is never initialized and such that, for all $s \geqslant u$, $r_{\sigma,s} = 0$. Given $x \in P_{D_6}$, find the least σ-recovery stage $s+1 \geqslant u$ such that x is contained in a component $(m)_s$, $m \in D_6$, of $\mathcal{A}^0_s$ and there exists a component L of $\mathcal{G}_n[s]$ isomorphic to $(m)_s$. Such a stage exists by Lemma 3.3.42. Let g_x be the unique isomorphism from $(m)_s$ to L. Again by Lemma 3.3.42, $f(x) = g_x(x)$. Since g_x can be computably determined given $x \in P_{D_6}$, $f \restriction P_{D_6}$ is computable. □

By Lemmas 3.3.30, 3.3.32, 3.3.34, 3.3.35, 3.3.37, 3.3.38, and 3.3.43, $f \restriction P_{D_i}$ is computable for each $i \leqslant 6$. As can be easily checked by referring to Table 3.1, $D_0, \dots, D_6$ are computable and $\bigcup_{i=0}^{6} D_i = \omega$. Thus, by Lemma 3.3.27, we have the following result.

3.3.44. Lemma. *The unique isomorphism $f : \mathcal{A}^0 \cong \mathcal{G}_n$ is computable.*

Theorem 3.1.4 follows from Lemmas 3.3.4, 3.3.5, 3.3.22, 3.3.23, and 3.3.44. ■

3.4 Proof of Theorem 3.1.6

In this section we prove the following theorem.

3.1.6. Theorem. *Let $\{A_i\}_{i\in\omega}$ be a uniformly c.e. (u.c.e.) collection of sets. There exists an intrinsically c.e. relation U on the domain of a computable structure $\mathcal{A}$ such that $DgSp_{\mathcal{A}}(U) = \{\deg(A_i) \mid i \in \omega\}$.*

Proof. Let $\{A_i\}_{i\in\omega}$ be a u.c.e. collection of sets. Let $A = \bigoplus_{i\in\omega} A_i = \{\langle i, x\rangle \mid x \in A_i\}$ and let $a_0, a_1, \dots$ be a computable enumeration of A. Let $A[0] = \emptyset$, $A[s+1] = \{a_0, \dots, a_s\}$.

We wish to construct computable structures $\mathcal{A}^i$, $i \in \omega$, and for each such structure a corresponding unary relation U^i on the domain of $\mathcal{A}^i$, so that for all $i, j \in \omega$, the following properties hold.

(3.4.1) $\mathcal{A}^i \cong \mathcal{A}^j$ via an isomorphism that carries U^i to U^j.

(3.4.2) $U^i \equiv_{\mathrm{m}} A_i$.

(3.4.3) If $\mathcal{G} \cong \mathcal{A}^0$ is a computable structure then the image of U^0 in $\mathcal{G}$ is m-equivalent to A_k for some $k \in \omega$.

The construction will be similar to the one in Section 3.2, as will the proof that the above properties hold. In this section, we restrict ourselves to pointing out the necessary changes.

We assume without loss of generality that, for all $i \in \omega$, $A_i \neq \emptyset$ and $A_i \neq \omega$. We also assume that A is not computable. (If A is computable then $\{\deg(A_i) \mid i \in \omega\} = \{\mathbf{0}\}$, and it is obvious that there exists a relation on the domain of a computable structure with degree spectrum $\{\mathbf{0}\}$.)

The basic idea is the following. Suppose that at stage $s+1$ we perform an **L**-operation involving copies of $[6a_s]$ and appropriate special components on $\mathcal{A}_s^{\pi_0(a_s)}$ and perform the corresponding **R**-operation on each $\mathcal{A}_s^j$, $j \neq \pi_0(a_s)$, and that we then put the coding location of the old copy of $[6a_s]$ in $\mathcal{A}_s^{\pi_0(a_s)}$ into $U^{\pi_0(a_s)}$ and, for each $j \neq \pi_0(a_s)$, we put the coding location of the new copy of $[6a_s]$ in $\mathcal{A}_s^j$ into U^j. Then the coding location x of a copy of $[6\langle i, k\rangle]$, $k \in \omega$, in $\mathcal{A}^i$ is in U^i if and only if $x \in \mathcal{A}_0^i$ and $k \in A_i$. On the other hand, the coding location of a copy of $[6\langle j, k\rangle]$, $k \in \omega$, $j \neq i$, is in U^i if and only if it is not in $\mathcal{A}_0^i$. Thus (3.4.2) is satisfied.

However, there is a problem in defining the isomorphism recovery mechanism used to satisfy (3.4.1), due to the fact that both **L**- and **R**-operations are applied to a given $\mathcal{A}^i$ during the construction. We deal with this by separating the stages at which elements enter the U^i from the stages at which isomorphism recovery can happen, reserving the even stages for the former purpose and the odd ones for the latter.

We now give the full description of the construction of the $\mathcal{A}^i$ and U^i.

stage 0. Let each $\mathcal{A}^i_0$, $i \in \omega$, be a computable structure with co-infinite domain, consisting of one copy of $[k]$ for each $k \in \omega$. For each $n \in \omega$, let $r_{n,0} = 0$.

stage $2s+1$. For each $n < s+1$, say that $2s+1$ is an n-recovery stage if all of the following conditions hold.

1. $\mathcal{G}_n[2s]$ has a special component isomorphic to some component of $\mathcal{A}^0_{2s}$. (Here "special component" has the same meaning as in the previous section.)

2. $(\mathcal{G}_n[2s])_n \cong (\mathcal{A}^0_{2s})_n$.

3. Let $j \notin A[s]$ be less than or equal to the number of n-recovery stages before stage $2s+1$. There is a component of $\mathcal{G}_n[2s]$ isomorphic to $[6j]$, for each $l \in \{1,2\}$ there is a component of $\mathcal{G}_n[2s]$ isomorphic to $[6\langle n,j\rangle + l]$, and for each $l \in \{10,11\}$ there is a component of $\mathcal{G}_n[2s]$ isomorphic to $[12\langle n,j\rangle + l]$.

4. Let c be the number of n-recovery stages before stage $2s+1$. For each $l \in \{4,5\}$ there is a component of $\mathcal{G}_n[2s]$ isomorphic to $[12\langle n,c\rangle + l]$.

If $2s+1$ is an n-recovery stage then, for $i \in \omega$, let $S^i_{n,2s}$ be the component of $\mathcal{A}^i_{2s}$ that is isomorphic to the special component of $\mathcal{G}_n[2s]$. If $2s+1$ is the first n-recovery stage then let $r_{n,2s+1} = 0$. Otherwise, proceed as follows. Let $i = r_{n,2s}$ and let $2t+1$ be the last n-recovery stage before stage $2s+1$. If $S^i_{n,2s}$ extends $S^i_{n,2t}$ then let $r_{n,2s+1} = i$. Otherwise, let c be the number of n-change stages (defined below) before stage $2s+1$ and let $r_{n,2s+1} = \pi_0(c)$.

If $2s+1$ is not an n-recovery stage then let $r_{n,2s+1} = r_{n,2s}$.

We say that $2s+1$ is an n-change stage if it is the first n-recovery stage or $r_{n,2s+1} \neq r_{n,2s}$. We say that $2s+1$ is an n-isomorphism recovery stage if it is an n-recovery stage but not an n-change stage and one of the following conditions holds.

1. The last n-recovery stage before stage $2s+1$ was an n-change stage.

2. There has been at least one stage at which n was active after the last n-isomorphism recovery stage and before stage $2s+1$.

Let $n_0, n_1, \ldots, n_m$ be all the numbers n_k such that $2s+1$ is an n_k-recovery stage. We say that each n_k, $k \leqslant m$, is active at stage $2s+1$. For each $k \leqslant m$, proceed as follows. Let $i = r_{n_k,2s+1}$ and let $2t+1 \leqslant 2s+1$ be the last n_k-recovery stage. Let $R^i_{n_k,2s}$ be the component of $\mathcal{A}^i_{2s}$ that extends $S^i_{n_k,2t}$ and, for each $j \neq i$, let $R^j_{n_k,2s}$ be the isomorphic image of $R^i_{n_k,2s}$ in $\mathcal{A}^j_{2s}$. Let c_k be the number of n_k-recovery stages before stage $2s+1$.

For each $k \leqslant m$, we define components $B^j_{n_k,2s}$ and $C^j_{n_k,2s}$, $j \in \omega$. There are two cases.

1. $2s+1$ is an n_k-isomorphism recovery stage. If the first condition in the definition of n_k-isomorphism recovery stage holds then let $t+1$ be the last n_k-recovery stage, and otherwise let $t+1$ be the first stage after the last n_k-isomorphism recovery stage at which n_k was active. Let $C^i_{n_k,2s}$ be the component of $\mathcal{A}^i_{2s}$ that extends $B^i_{n_k,t}$ and, for $j \neq i$, let $C^j_{n_k,2s}$ be the isomorphic image of $C^i_{n_k,2s}$ in $\mathcal{A}^j_{2s}$. For $j \in \omega$, let $B^j_{n_k,2s}$ be the component of $\mathcal{A}^j_{2s}$ isomorphic to $[12\langle n_k, c_k\rangle + 4]$.

2. $2s+1$ is not an n_k-isomorphism recovery stage. For $j \in \omega$, let $B^j_{n_k,2s}$ and $C^j_{n_k,2s}$ be the components of $\mathcal{A}^j_{2s}$ isomorphic to $[12\langle n_k, c_k\rangle + 4]$ and $[12\langle n_k, c_k\rangle + 5]$, respectively.

For each $i \in \omega$, we define operations $\mathbf{O}^i_0, \ldots, \mathbf{O}^i_m$ as follows. If $i = r_{n_k,2s+1}$ then let $\mathbf{O}^i_k = \mathbf{L}(B^i_{n_k,2s}, R^i_{n_k,2s}, C^i_{n_k,2s})$. Otherwise, let $\mathbf{O}^i_k = \mathbf{R}(B^i_{n_k,2s}, R^i_{n_k,2s}, C^i_{n_k,2s})$.

For each $i \in \omega$, perform the sequence of operations $\mathbf{O}^i_0, \ldots, \mathbf{O}^i_m$ on $\mathcal{A}^i_{2s}$ to get $\mathcal{A}^i_{2s+1}$.

stage $2s+2$. For each $n \in \omega$, let $r_{n,2s+2} = r_{n,2s+1}$.

Let $l = \pi_0(a_s)$. Let $n_0, n_1, \ldots, n_m$ be all the numbers n_j such that a_s is less than the number of n_j-recovery stages before stage $2s+2$. We say that each n_j, $j \leqslant m$, is active at stage $2s+2$. For $i \in \omega$ and $j \leqslant m$, let X^i_{2s+1}, $Y^i_{n_j,2s+1}$, $Z^i_{n_j,2s+1}$, $B^i_{n_j,2s+1}$, and $C^i_{n_j,2s+1}$ be the components of $\mathcal{A}^i_{2s+1}$ isomorphic to $[6a_s]$, $[6\langle n_j, a_s\rangle+1]$, $[6\langle n_j, a_s\rangle + 2]$, $[12\langle n_j, a_s\rangle + 10]$, and $[12\langle n_j, a_s\rangle + 11]$, respectively.

For each $k \leqslant m$, proceed as follows. Let $i = r_{n_k,2s+2}$ and let $2t+1$ be the last n_k-recovery stage before stage $2s+2$. Let $R^i_{n_k,2s+1}$ be the component of $\mathcal{A}^i_{2s+1}$ that extends $S^i_{n_k,2t}$ and, for each $j \neq i$, let $R^j_{n_k,2s+1}$ be the isomorphic image of $R^i_{n_k,2s+1}$ in $\mathcal{A}^j_{2s+1}$.

Now perform

$$\mathbf{L}(Y^l_{n_0,2s+1},\dots,Y^l_{n_m,2s+1};X^l_{2s+1};Z^l_{n_0,2s+1},\dots,Z^l_{n_m,2s+1};B^l_{n_0,2s+1},R^l_{n_0,2s+1},C^l_{n_0,2s+1};$$
$$B^l_{n_1,2s+1},R^l_{n_1,2s+1},C^l_{n_1,2s+1};\dots;B^l_{n_m,2s+1},R^l_{n_m,2s+1},C^l_{n_m,2s+1})$$

on $\mathcal{A}^l_{2s+1}$ to get $\mathcal{A}^l_{2s+2}$ and, for each $j \neq l$, perform

$$\mathbf{R}(Y^j_{n_0,2s+1},\dots,Y^j_{n_m,2s+1};X^j_{2s+1};Z^j_{n_0,2s+1},\dots,Z^j_{n_m,2s+1};B^j_{n_0,2s+1},R^j_{n_0,2s+1},C^j_{n_0,2s+1};$$
$$B^j_{n_1,2s+1},R^j_{n_1,2s+1},C^j_{n_1,2s+1};\dots;B^j_{n_m,2s+1},R^j_{n_m,2s+1},C^j_{n_m,2s+1})$$

on $\mathcal{A}^j_{2s+1}$ to get $\mathcal{A}^j_{2s+2}$. (If no n is active at stage $2s+2$ then, for each $i \in \omega$, let Y^i_{2s+1} and Z^i_{2s+1} be the components of $\mathcal{A}^i_{2s+1}$ isomorphic to $[6\langle 0, a_s\rangle + 1]$ and $[6\langle 0, a_s\rangle + 2]$, respectively. Perform $\mathbf{L}(Y^l_{2s+1}, X^l_{2s+1}, Z^l_{2s+1})$ on $\mathcal{A}^l_{2s+1}$ to get $\mathcal{A}^l_{2s+2}$ and, for each $j \neq l$, perform $\mathbf{R}(Y^j_{2s+1}, X^j_{2s+1}, Z^j_{2s+1})$ on $\mathcal{A}^j_{2s+1}$ to get $\mathcal{A}^j_{2s+2}$.)

Put the coding location of the copy of $[6a_s]$ in $\mathcal{A}^l_0$ into U^l and, for each $j \neq l$, put the coding location of the copy of $[6a_s]$ in $\mathcal{A}^j_{2s+2} - \mathcal{A}^j_{2s+1}$ into U^j.

This completes the construction. For each $i \in \omega$, let $\mathcal{A}^i = \bigcup_{s\in\omega} \mathcal{A}^i_s$. As previously remarked, the proof that (3.4.1)–(3.4.3) are satisfied is similar to what we did in Section 3.2. We begin by showing that (3.4.2) is satisfied.

3.4.1. Lemma. *For each $i \in \omega$, $U^i \equiv_m A_i$.*

Proof. If k is the coding location of a copy of $[6\langle i, x\rangle]$ in $\mathcal{A}^i$ then $k \in U^i$ if and only if $k \in \mathcal{A}^i_0$ and $x \in A_i$. On the other hand, if k is the coding location of a copy of

$[6\langle j,x\rangle]$ in $\mathcal{A}^i$ for some $x \in \omega$, $j \neq i$, and k enters U^i at stage $s+1$ then k is a new number at that stage, and hence is greater than s. $\square$

Lemmas 3.2.8, 3.2.9, 3.2.10, 3.2.11, 3.2.12, 3.2.14, and 3.2.18 still hold, as do the following versions of Lemmas 3.2.13, 3.2.15, 3.2.16, and 3.2.17. In all cases, the reasoning is basically the same as in Section 3.2.

3.4.2. Lemma. *Let $i,j,s \in \omega$. $\mathcal{A}^i_s \cong \mathcal{A}^j_s$ and no component of $\mathcal{A}^i_s$ is embeddable in another component of $\mathcal{A}^i_s$. Furthermore, if a component of $\mathcal{A}^i_s$ participates in an operation at stage $s+1$ then so does the (unique) isomorphic component of $\mathcal{A}^j_s$.*

3.4.3. Lemma. *Let $2s+1$ be an n-recovery stage that is not the first such stage. Let $2t+1$ be the last n-recovery stage before stage $2s+1$ and suppose that $r_{n,2t+1} = i \neq r_{n,2s+1}$. Then for some $u \in [t,s)$, $S^i_{n,2s}$ extends one of $B^i_{n,2u}$, $B^i_{n,2u+1}$, or $C^i_{n,2u+1}$.*

3.4.4. Lemma. *Suppose that $r_{n,t} = i$ for all $t \geqslant s$. Then no component of $(\mathcal{A}^i)_n$ can participate in an operation more than twice after stage s unless it extends $R^i_{n,t}$ for some $t \geqslant s$, while for $j \neq i$, no component of $(\mathcal{A}^j)_n$ can participate in an operation more than twice after stage s unless it extends $C^i_{n,t}$ for some $t \geqslant s$ such that $t+1$ is an n-isomorphism recovery stage.*

3.4.5. Lemma. *Suppose that $s < t < v$ are such that $s+1$ is an n-isomorphism recovery stage, $r_{n,u} = r_{n,s+1}$ for all $u > s$, $t+1$ is the next stage after stage $s+1$ at which n is active, and $v+1$ is the next n-isomorphism recovery stage after stage $s+1$. For $j \in \omega$, let B^j and R^j be the components of $\mathcal{A}^j_{t+1}$ that extend $B^j_{n,t}$ and $R^j_{n,t}$, respectively, and let $\widehat{B}^j$ and $\widehat{R}^j$ be the components of $\mathcal{A}^j_v$ that extend B^j and R^j, respectively. Then $\widehat{B}^i \cong B^i$ and, for $j \neq i$, $\widehat{R}^j \cong R^j$.*

We now wish to show that that (3.4.3) is satisfied. Lemma 3.2.20 still holds, and hence so does Lemma 3.2.26. In both cases the proofs are essentially the same as in Section 3.2. Using Lemma 3.4.3 in place of Lemma 3.2.15, we can prove Lemma 3.2.19 in much the same way as before. (Notice that the way we define $r_{n,2s+1}$ guarantees that if $r_{n,s}$ does not have a limit then for each $i \in \omega$ there are infinitely many stages s such that $r_{n,s} = i$.) Now Lemma 3.2.27 follows, as before, from Lemmas 3.2.19 and 3.2.26.

Lemmas 3.2.28 and 3.2.29 still hold, with essentially the same proofs as in Section 3.2, provided that, in the latter lemma, we make the obvious changes arising from the fact that if n is active at stage $2s+1$ then the components $B^i_{n,2s}$, $R^i_{n,2s}$, and $C^i_{n,2s}$ are defined but the components $Y^i_{n,2s}$, X^i_{2s}, and $Z^i_{n,2s}$ are not.

Now the following lemma can be proved in much the same way as Lemma 3.2.30.

3.4.6. Lemma. *Suppose that $2s+1$ is an n-recovery stage such that $r_{n,2s+1} = r_{n,2s}$. Let $2t+1$ be the last n-recovery stage before stage $2s+1$ and let $j \in A[s] - A[t]$ be less than the number of n-recovery stages less than or equal to $2t+1$. By the definition of n-recovery stage, there is a unique component K of $\mathcal{G}_n[2t]$ isomorphic to $[6j]$. Let L be the component of $\mathcal{G}_n$ that extends K. Then L contains a copy of $[6\langle n,j\rangle + 2]$ if and only if $r_{n,2s+1} = \pi_0(j)$.*

The previous lemma allows us to establish that (3.4.3) is satisfied.

3.4.7. Lemma. *Suppose that $\mathcal{G}_n \cong \mathcal{A}^0$. Let U be the image of U^0 under this isomorphism. Then $U \equiv_m A_i$ for some $i \in \omega$.*

Proof. Let N and M be as in the proof of Lemma 3.2.31. By Lemma 3.2.28, it is enough to show that $U \cap M \equiv_{\mathrm{m}} A_i$ for some $i \in \omega$. By Lemma 3.2.27, $r_{n,s}$ has a limit i. Let M_0 be the set of elements of M that are coding locations of copies of graphs of the form $[6n]$, $\pi_0(n) = i$, and let $M_1 = M - M_0$. Note that M_0 and M_1 are computable.

Now, combining Lemmas 3.2.18 and 3.4.6, we see that

1. an element x of M_0 is in U if and only if, for some $j \in A_i$, x is the coding location of the first copy of $[6\langle i, j\rangle]$ to appear in $\mathcal{G}_n$, while

2. an element x of M_1 is in U if and only if, for some $k \in \omega$, x is the coding location of the second copy of $[6k]$ to appear in $\mathcal{G}_n$.

So $U \cap M_0 \equiv_{\mathrm{m}} A_i$ and $U \cap M_1$ is computable, and thus $U \cap M \equiv_{\mathrm{m}} A_i$. □

We are left with showing that (3.4.1) is satisfied. Lemma 3.2.21 still holds, with basically the same proof as before. Lemma 3.2.22 still holds, but the proof needs to be slightly modified, so we restate the lemma and give the new proof.

3.4.8. Lemma. *Suppose that n is active infinitely often and s and i are such that $r_{n,t} = r_{n,s} = i$ for all $t \geqslant s$. By Lemma 3.2.21, there are infinitely many n-isomorphism recovery stages. Let $s_0 + 1 < s_1 + 1 < \cdots$ be the n-isomorphism recovery stages after stage s. For each $j \in \omega$, let $t_j + 1$ be the next stage after stage $s_j + 1$ at which n is active. (Note that $t_j < s_{j+1}$ for all $j \in \omega$.) For $t \geqslant t_0$, let K_t^l be the component of $\mathcal{A}_t^l$ that extends R_{n,t_0}^l. Then $K_{t_j}^l = R_{n,t_j}^l$ for all $j \in \omega$.*

Proof. That $K_{s_j}^i = R_{n,s_j}^i$ for all $j \in \omega$ follows from Lemma 3.2.14.

Now let $l \neq i$ and assume by induction that $K^l_{t_j} = R^l_{t_j}$. Let B be the component of $\mathcal{A}^i_{t_j+1}$ that extends B^i_{n,t_j}. By construction, $B \cong K^l_{t_j+1}$. Since $s_{j+1}+1$ is an n-isomorphism recovery stage, $C^i_{s_{j+1}}$ extends B. Thus, by Lemma 3.4.5, $C^i_{s_{j+1}} \cong B$. By the same lemma, $K^l_{s_{j+1}} \cong K^l_{t_j+1}$. So $C^i_{s_{j+1}} \cong K^l_{s_{j+1}}$, and thus $C^l_{s_{j+1}} = K^l_{s_{j+1}}$. Let R be the component of $\mathcal{A}^i_{s_{j+1}+1}$ that extends $R^i_{n,s_{j+1}}$. Then $R \cong K^l_{s_{j+1}+1}$. But, by Lemma 3.2.11, $R^i_{n,t_{j+1}} \cong R$ and $K^l_{t_{j+1}} \cong K^l_{s_{j+1}+1}$. So $K^l_{t_{j+1}} \cong R^i_{n,t_{j+1}}$, and thus $K^l_{t_{j+1}} = R^l_{n,t_{j+1}}$. □

If we assume the hypotheses of Lemma 3.4.8 and let S^l_n be the component of $\mathcal{A}^l$ that extends R^l_{n,s_0} then we can prove Lemma 3.2.23 in the same way as before, using Lemma 3.4.4 in place of Lemma 3.2.16. Furthermore, the following version of Lemma 3.2.24 follows directly from Lemma 3.4.8.

3.4.9. Lemma. *Assume the hypotheses of Lemma 3.4.8 and let S^l_n be the component of $\mathcal{A}^l$ that extends R^l_{n,s_0}. Then $S^k_n \cong S^l_n$ for all $k, l \in \omega$.*

Reasoning as in Section 3.2 (using Lemma 3.4.2 in place of Lemma 3.2.13), we see that Lemmas 3.2.23 and 3.4.9 suffice to establish that (3.4.1) is satisfied.

3.4.10. Lemma. *For each $i, j \in \omega$, $\mathcal{A}^i \cong \mathcal{A}^j$ via an isomorphism that carries U^i to U^j.*

Theorem 3.1.6 follows from Lemmas 3.4.1, 3.4.7, and 3.4.10. ■

3.5 Proof of Theorem 3.1.7

In this section we prove the following theorem.

3.1.7. Theorem. *Let $\alpha \in \omega \cup \{\omega\}$ and let $\mathbf{b} > \mathbf{0}$ be an α-c.e. degree. There exists an intrinsically α-c.e. relation V on the domain of a computable structure $\mathcal{B}$ of computable dimension 2 such that $DgSp_{\mathcal{B}}(V) = \{\mathbf{0}, \mathbf{b}\}$.*

Proof. Let $\alpha \in \omega \cup \{\omega\}$ and let B be an α-c.e. set that is not computable. It is well-known (see [6]) that there exist a computable sequence $b_0, b_1, \ldots \in \omega$ and a function f such that

1. either $\alpha < \omega$ and $f(x) = \alpha$ for all $x \in \omega$ or $\alpha = \omega$ and f is computable,
2. $|\{s \mid b_s = x\}| \leqslant f(x)$ for all $x \in \omega$, and
3. $x \in B \Leftrightarrow |\{s \mid b_s = x\}| \equiv 1 \bmod 2$.

We wish to construct computable structures $\mathcal{B}^0$ and $\mathcal{B}^1$ and unary relations V^0 and V^1 on the domains of $\mathcal{B}^0$ and $\mathcal{B}^1$, respectively, so that the following properties hold.

(3.5.1) $\mathcal{B}^0 \cong \mathcal{B}^1$ via an isomorphism that carries V^0 to V^1.

(3.5.2) $V^0 \equiv_{\mathrm{m}} B$ and V^1 is computable.

(3.5.3) If $\mathcal{G} \cong \mathcal{B}^0$ is a computable structure then $\mathcal{G}$ is computably isomorphic to either $\mathcal{B}^0$ or $\mathcal{B}^1$.

(3.5.4) $\mathcal{B}^0$ is rigid.

For each $s \in \omega$, let $c_s = |\{t < s \mid b_t = b_s\}|$ and let $a_s = \langle b_s, c_s \rangle$. Let $A = \{a_0, a_1, \ldots\}$. A is clearly c.e. but not computable, so we can follow the construction in Section 3.3 to obtain computable structures $\mathcal{A}^0$ and $\mathcal{A}^1$ and relations U^0 and U^1 on the domains of $\mathcal{A}^0$ and $\mathcal{A}^1$, respectively, satisfying properties (3.3.1)–(3.3.4). (We assume that the construction has been carried out in such a way that the domains of $\mathcal{A}^0$ and $\mathcal{A}^1$ are co-infinite.)

Now, for $i = 0, 1$, proceed as follows. Add an element, which we will call the *identifying node* of $\mathcal{B}^i$ to the domain of $\mathcal{A}^i$ and add an edge from this node to each node of $\mathcal{A}^i$. For each $j \in \omega$ and each sequence of components $L_0, L_1, \ldots, L_{f(j)-1}$ such that each L_k contains a copy of $[6\langle j, k\rangle]$, add an element x (which will be said to be a *j-coding node*) to the domain of $\mathcal{A}^i$ and, for each $k < f(j)$, add an edge from x to the coding location of the copy of $[6\langle j, k\rangle]$ in L_k. The resulting graph is $\mathcal{B}^i$.

Clearly, each $\mathcal{B}^i$ is a computable graph, and the following lemma can be easily checked, using the fact that $\mathcal{A}^0$ is rigid.

3.5.1. Lemma. *$\mathcal{B}^0$ is rigid.*

We now define a relation V^i on the domain of $\mathcal{B}^i$. Let K^i be the set of coding nodes in $\mathcal{B}^i$. Let $j \in \omega$ and let x be a j-coding node in $\mathcal{B}^i$. By construction, there exist components $L_0, \ldots, L_{f(j)-1}$ of $\mathcal{A}^i$ such that, for each $k < f(j)$, L_k contains a copy of $[6\langle j, k\rangle]$ whose coding location y_k is attached to x. Let $c^i(x)$ be the least $k < f(j)$ such that $y_k \notin U^i$, if such a k exists, and let $c^i(x) = f(j)$ otherwise. Now let $V^i = \{x \in K^i \mid c^i(x) \text{ is odd}\}$.

3.5.2. Lemma. *$\mathcal{B}^0 \cong \mathcal{B}^1$ via an isomorphism that carries V^0 to V^1.*

Proof. By (3.2.1), $\mathcal{A}^0 \cong \mathcal{A}^1$ via an isomorphism that carries U^0 to U^1. It is straightforward to extend this isomorphism to an isomorphism $h : \mathcal{B}^0 \cong \mathcal{B}^1$. The fact that $h(U^0) = (U^1)$ implies that if $x \in K^0$ then $c^0(x) = c^1(h(x))$. Thus $h(V^0) = V^1$. $\square$

3.5.3. Lemma. *V_1 is computable and $V_0 \equiv_m B$.*

Proof. Since U^1 is computable, there is a computable procedure for determining $c(x)$ given $x \in K^0$, and thus V_1 is computable.

Let $x \in K^0$. By construction, there exist components $L_0, \ldots, L_{f(j)-1}$ of $\mathcal{A}^0$ such that, for each $k < f(j)$, L_k contains a copy of $[6\langle j,k\rangle]$ whose coding location y_k is attached to x. Let $d(x)$ be the least k such that, for all $m \geqslant k$, y_m is the coding location of the copy of $[6\langle j,m\rangle]$ in $\mathcal{A}^0_0$, if such a k exists, and let $d(x) = f(j)$ otherwise. Note that there is a computable procedure for determining $d(x)$ given $x \in K^0$.

If $d(x) > 0$ then clearly $\langle j, d(x)-1\rangle \in A$. But this means that, in fact, $\langle j,k\rangle \in A$ for all $k < d(x)$. It follows that we can computably determine whether $y_k \in U^0$ for $k < d(x)$. So $S = \{x \in K^0 \mid c(x) < d(x)\}$, $T = K^0 - S$, and $V^0 \cap S$ are computable.

Now let $x \in T$ be a j-coding node and let $y_0, \ldots, y_{f(j)-1}$ be as above. By the definition of T, $y_0, \ldots, y_{d(x)-1} \in U^0$, so $\langle j,k\rangle \in A$ for all $k < d(x)$. But, by the definition of $d(x)$, for each $k \geqslant d(x)$, $y_k \in U^0$ if and only if $\langle j,k\rangle \in A$. So $c(x) = |\{k \mid \langle j,k\rangle \in A\}| = |\{t \mid b_t = j\}|$. Thus $x \in V^0$ if and only if $j \in B$, and hence $V^0 \cap T \equiv_{\mathrm{m}} B$. Since $V^0 = (V^0 \cap S) \cup (V^0 \cap T)$, it follows that $V^0 \equiv_{\mathrm{m}} B$. $\square$

3.5.4. Lemma. *If $\mathcal{G} \cong \mathcal{B}^0$ is a computable structure then $\mathcal{G}$ is computably isomorphic to either $\mathcal{B}^0$ or $\mathcal{B}^1$.*

Proof. Let z be the image of the identifying node of $\mathcal{B}^0$ in $\mathcal{G}$. Let $\mathcal{G}'$ be the computable subgraph of $\mathcal{G}$ consisting of all elements y of $\mathcal{G}$ such that there is an edge from z to y. By the definition of $\mathcal{B}^0$, $\mathcal{G}'$ is isomorphic to $\mathcal{A}^0$. Thus, by the results of Section 3.3, for some $i = 0, 1$ there exists a computable isomorphism $h : \mathcal{A}^i \cong \mathcal{G}'$.

To extend this isomorphism to a computable isomorphism $\hat{h} : \mathcal{B}^i \cong \mathcal{G}$, we first define $\hat{h} \upharpoonright \mathcal{A}^i \equiv h$. Now let $x \in \mathcal{B}^i - \mathcal{A}^i$. Then x is a j-coding node for some $j \in \omega$, and we can computably determine the $f(j)$ many coding locations $y_0, \ldots, y_{f(j)-1}$ attached to x. There is a unique $w \in \mathcal{G} - \mathcal{G}'$ attached to $h(y_0), \ldots, h(y_{f(j)-1})$. Define $\hat{h}(x) = w$. It is now easy to check that $\hat{h}$ is a computable isomorphism from $\mathcal{B}^i$ to $\mathcal{G}$. $\square$

Theorem 3.1.7 follows from Lemmas 3.5.1, 3.5.2, 3.5.3, and 3.5.4. $\blacksquare$

Chapter 4

Expansions of Computably Categorical Structures

4.1 Introduction

In classical model theory, 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 [30] showed that, with a relatively small additional amount of decidability, computable categoricity is preserved under expansion by finitely many constants.

4.1.1. 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 [4].

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

This raises the following question, left open in [4], as well as in [25], where an easier proof of Theorem 4.1.2 is given: Does there exist a computably categorical structure whose expansion by some set of finitely many constants has computable dimension ω? In this chapter, which reports on joint work with Bakhadyr Khoussainov and Richard Shore, we apply the methods of Chapter 3 to give the following positive answer to this question.

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

4.2 Proof of Theorem 4.1.3

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.

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

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

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

The ideas used in this proof borrow heavily from those of Chapter 3, and we will assume familiarity with the methods and definitions in that chapter.

The structure $\mathcal{A}$ will be a *leveled* graph, in the sense of the following definition.

4.2.1. Definition. The *backbone graph* is the directed graph, shown in Figure 4.1, 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. Here a *cycle* is a copy of $[k]$ for some $k \in \omega$, where $[k]$ is as in Definition 3.2.1.

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 $i \neq j \in \mathbb{Z}$. 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 $\mathcal{G}$. 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 $i \in \mathbb{Z}$, $\mathcal{G}^i$ will denote 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, for each $i \in \omega$, L_i has a level $g(i)$. For each component M of $\mathcal{G}^*$ as above, let $\widehat{M}$ 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 $\widehat{M}$ such that M is a component of $\mathcal{G}^*$. In case $r = 0$, we write simply $(\mathcal{G}^*)^n$.

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 (4.2.1)–(4.2.3) 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 (4.2.3). 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

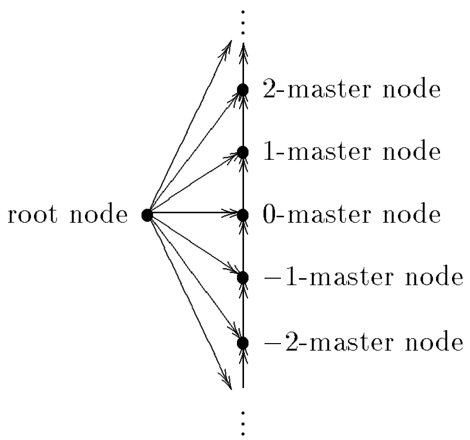


Figure 4.1: The backbone graph

cycle in 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 of B such that $x \in |\mathcal{G}[s]|$ and there is an edge in $\mathcal{G}[s]$ from x to the top of K, and 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 (4.2.3). As in Chapter 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.

In order to satisfy (4.2.2), we will satisfy the following requirement for each $e \in \omega$ and $i \in \mathbb{Z}$, $i \neq 0$:

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

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

The basic idea for satisfying $R_{\langle e,i \rangle}$ is simple. We begin with a leveled graph $\mathcal{A}_0$ with following properties. All levels are isomorphic and consist of singleton components (in the sense of Definition 3.2.6), A_0^* consists of singleton components that have no level, no two components of the same level are isomorphic, and no component that has a level is isomorphic to a component that has no level. 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 copy of $E^0 \cdot F^0$, while E^i is extended by a copy of $D^i \cdot E^i$.

Of course, in order 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 sequence of operations $\mathbf{L}(D^0, E^0, F^0)$ and $\mathbf{R}(D^j, E^j, F^j)$, $j \in \mathbb{Z}$, $j \neq 0$, where these operations are as described in Definition 3.2.3.

However, as in Chapter 3, the satisfaction of (4.2.3) will require us to involve more components than just the D^j, E^j, and F^j in our operations; as in that chapter, whenever we perform an operation, we will have several rows of components involved in the operation. These rows will share a component, which is what will allow us to satisfy (4.2.3) in much the same way as we satisfied the corresponding property in Section 3.3.

In contrast to what we did before, a single operation on $\mathcal{A}$ will have certain rows of components going to the left and other rows going to the right, in the sense of Chapter 3. All of our operations will be periodic, in the sense that there is an $n > 0$ such that, if a row of level-i components participates in the operation by going to the left then so does the isomorphic row of level-$(i+nj)$ components for each $j \in \mathbb{Z}$, and similarly for rows that go to the right.

As an illustration, Figure 4.2, 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 $i = 3$. As in the figures in Chapter 3, an arrow from, say, 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 that acts as the link between different rows of components participating in the operation.

In order to understand Figure 4.2, we need to define two new kinds of basic operations.

4.2.2. 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

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

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

$$\begin{array}{cc} E^2 & F_1^2 \\ \downarrow & \downarrow \\ E^2 \cdot D^2 & F_1^2 \odot (E^2, F_0^2, F_2^2) \end{array} \qquad \begin{array}{cc} E^3 & F_2^3 \\ \downarrow & \downarrow \\ E^3 \cdot D^3 & F_2^3 \odot (E^3, F_0^3, F_1^3) \end{array}$$

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

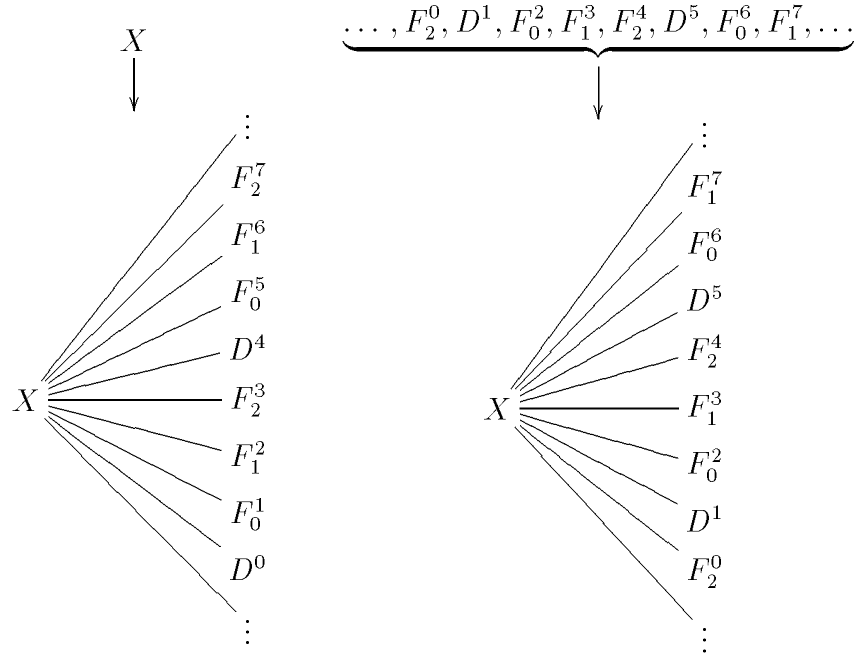
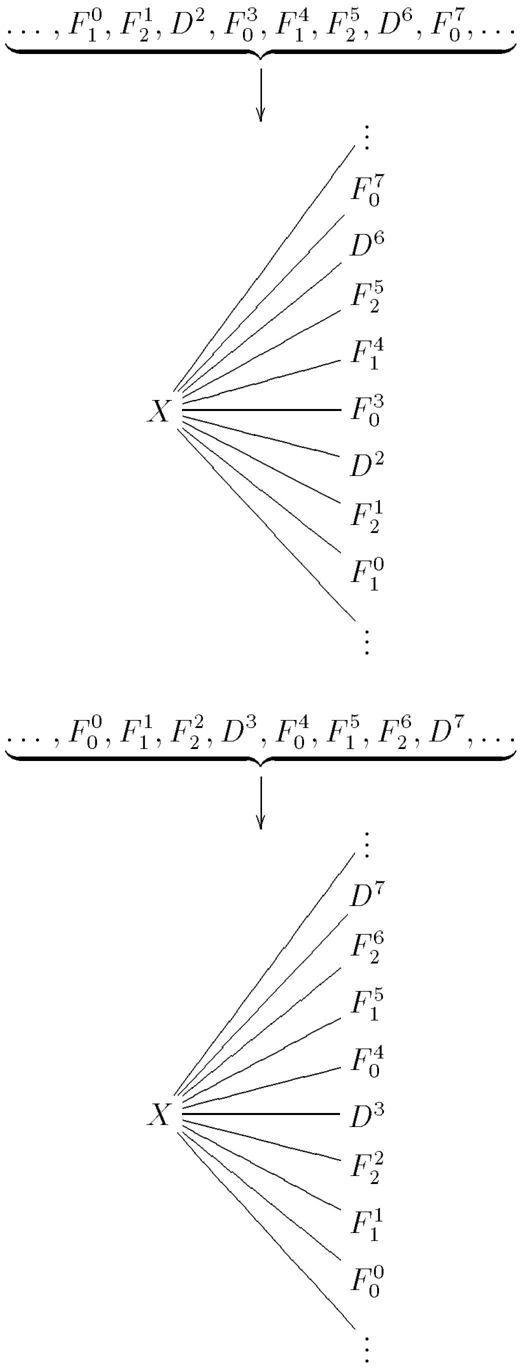


Figure 4.2: The basic diagonalization strategy

Figure 4.2 (Continued)



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 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 $[y_i]$.

4.2.3. 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.

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 $[y_j]$.

The operations in Definition 4.2.2 are infinite versions of those in Definition 3.2.5. As an example of the operation in Definition 4.2.3, 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 4.3, 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.

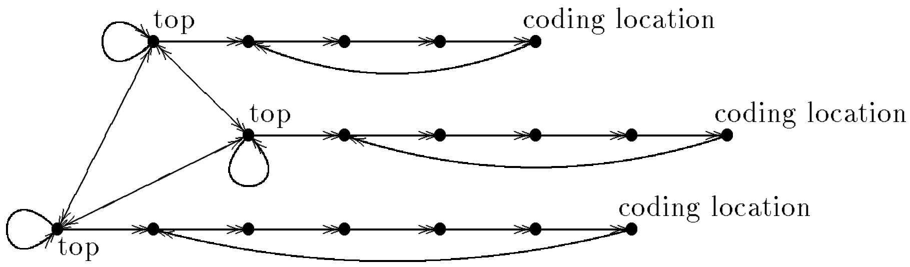


Figure 4.3: The result of any of $[2] \odot ([3], [4])$, $[3] \odot ([2], [4])$, or $[4] \odot ([2], [3])$

In Figure 4.2, 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 $L \odot (K_0, K_1, \ldots, K_n)$. (The important difference between $L \cdot (K_0, K_1, \ldots, K_n)$ and $L \odot (K_0, K_1, \ldots, K_n)$ is that, for any partial computable graph $\mathcal{G}$ and $s < t \in \omega$, if K_i and K_j, $i, j \leqslant n$, are different components of $\mathcal{G}[s]$ then they can be extended by the same component of $\mathcal{G}[t]$ if this component results from an operation of the form $L \cdot (K_0, K_1, \ldots, K_n)$, but not if it results from an operation of the form $L \odot (K_0, K_1, \ldots, K_n)$.)

As we have 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 Chapter 3, we will use special components to satisfy (4.2.3). The idea is similar to what we did in Section 3.3. For each finite binary string σ, there will be a strategy for satisfying (4.2.3) for $\mathcal{G}_{|\sigma|}$. As before, the string σ will represent 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. We will not allow σ-recovery unless there is τ-recovery for all τ such that $\tau^\frown 0 \subseteq \sigma$, where σ-recovery will be defined much as before, with a few differences which will be discussed below.

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 3.3, $r_{\sigma,s}$ was a guess as to which copy of the structure constructed in that section behaved like $\mathcal{G}_{|\sigma|}$.

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, $r_{\sigma,t+1} \neq r_{\sigma,s+1}$.

Because we are diagonalizing against certain potential automorphisms of $\mathcal{A}$ rather than, as in Chapter 3, coding a given set into a relation, we will be able to arrange the construction so that σ is not active more than once between σ-recovery stages. In fact, σ will not be active except at σ-recovery stages. This will simplify the analysis of what happens when σ recovers. We will comment further on this below.

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 $j \in \mathbb{Z}$, it will be the case that 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, as we did in Chapter 3. 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.

In order to illustrate the recovery process, consider Figure 4.4, which illustrates an operation that might be performed at some stage $s+1$ of our construction, ignoring for now all components indexed by τ.

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 4.4, 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 $r_{\sigma,s+1} = 0$ and we perform the operation pictured in Figure 4.4 on $\mathcal{A}_s$ to obtain $\mathcal{A}_{s+1}$ and 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 $\langle \mathcal{A}_s, a_0 \rangle \cong \langle \mathcal{A}_s, a_i \rangle$ then $\langle \mathcal{A}_{s+1}, a_0 \rangle \cong \langle \mathcal{A}_{s+1}, a_i \rangle$.

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 $i \in \mathbb{Z}$, $|i| \leqslant k$, $\mathcal{G}_{|\sigma|}[s]$ 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 $\overline{\mathcal{G}}_n[t]$ be the union of $(\mathcal{G}_n^*[t])^k$ and $\mathcal{G}_n^i[t]$ 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

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

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 4.4: A $3, (0, 1)$-operation

Figure 4.4 (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 4.4 (Continued)

$$\begin{array}{cccc} B_\tau^0 & S_\tau^0 & C_\tau^0 & Y_{\tau,2}^0 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ B_\tau^0 \cdot Z_\tau^0 & S_\tau^0 \cdot B_\tau^0 & C_\tau^0 \cdot S_\tau^0 & Y_{\tau,2}^0 \odot (C_\tau^0, Y_{\tau,0}^0, Y_{\tau,1}^0) \end{array}$$

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

$$\begin{array}{cccc} Z_\tau^1 & B_\tau^1 & S_\tau^1 & C_\tau^1 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ Z_\tau^1 \cdot B_\tau^1 & B_\tau^1 \cdot S_\tau^1 & S_\tau^1 \cdot C_\tau^1 & C_\tau^1 \odot (Y_{\tau,0}^1, Y_{\tau,1}^1, Y_{\tau,2}^1) \end{array}$$

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

$$\begin{array}{cccc} B_\tau^2 & S_\tau^2 & C_\tau^2 & Y_{\tau,1}^2 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ B_\tau^2 \cdot Z_\tau^2 & S_\tau^2 \cdot B_\tau^2 & C_\tau^2 \cdot S_\tau^2 & Y_{\tau,0}^2 \odot (C_\tau^2, Y_{\tau,1}^2, Y_{\tau,2}^2) \end{array}$$

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

$$\begin{array}{cccc} B_\tau^3 & S_\tau^3 & C_\tau^3 & Y_{\tau,1}^3 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ B_\tau^3 \cdot Z_\tau^3 & S_\tau^3 \cdot B_\tau^3 & C_\tau^3 \cdot S_\tau^3 & Y_{\tau,1}^3 \odot (C_\tau^3, Y_{\tau,0}^3, Y_{\tau,2}^3) \end{array}$$

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

Figure 4.4 (Continued)

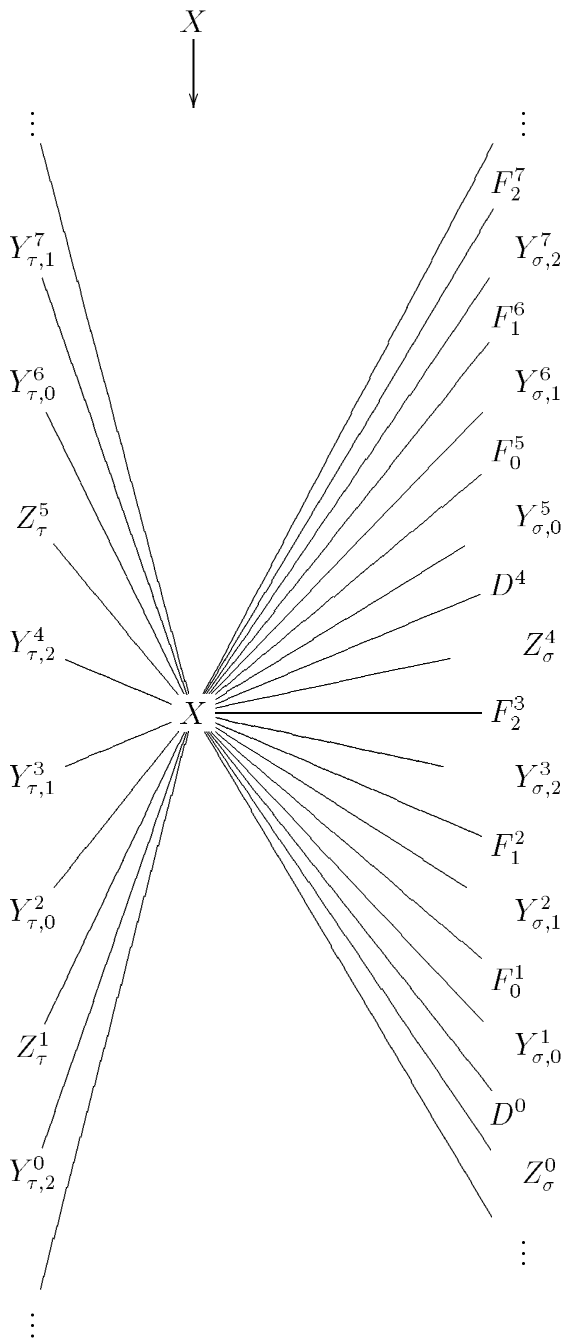


Figure 4.4 (Continued)

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

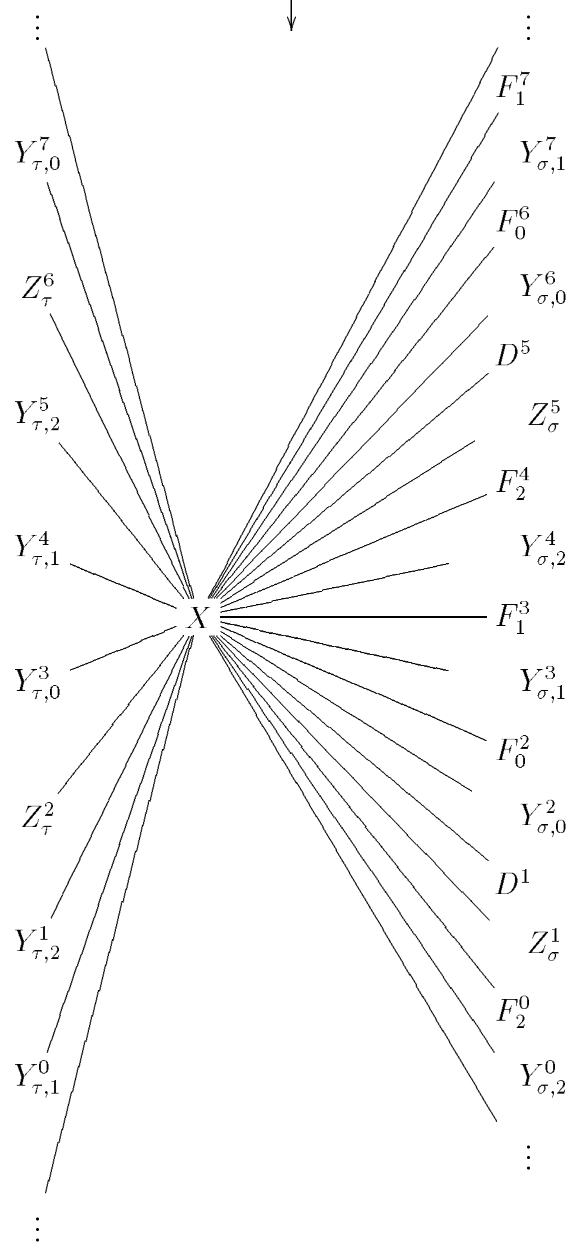


Figure 4.4 (Continued)

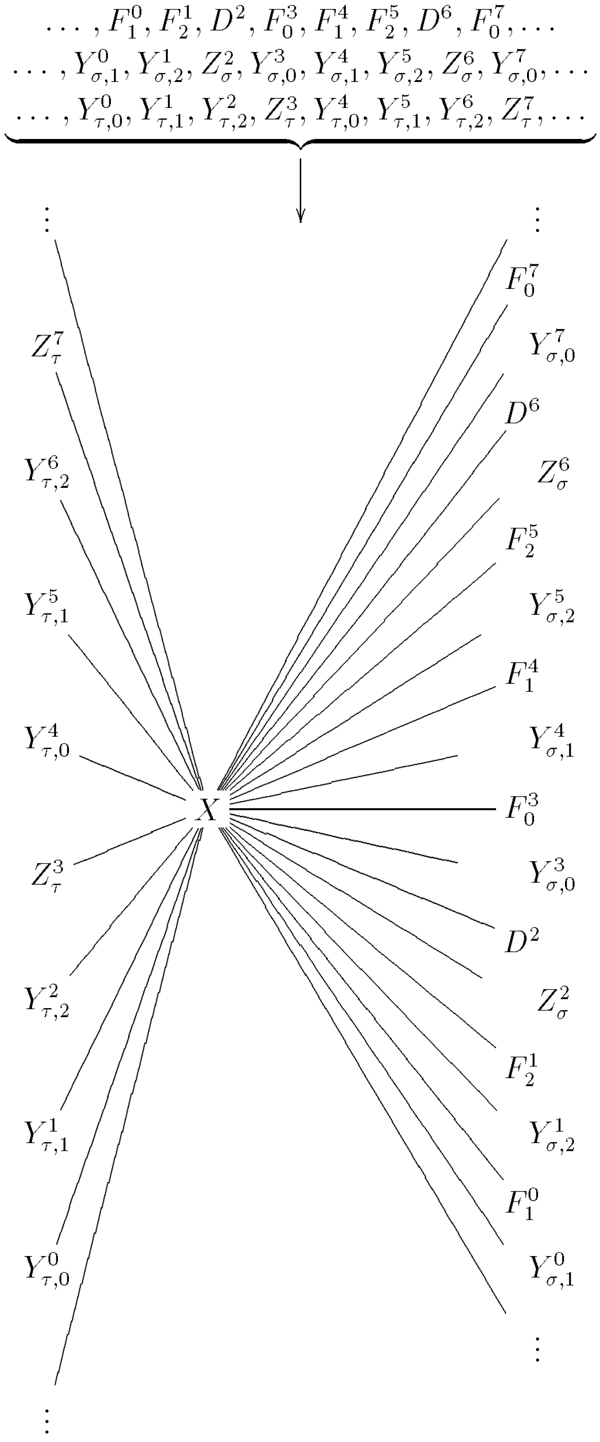
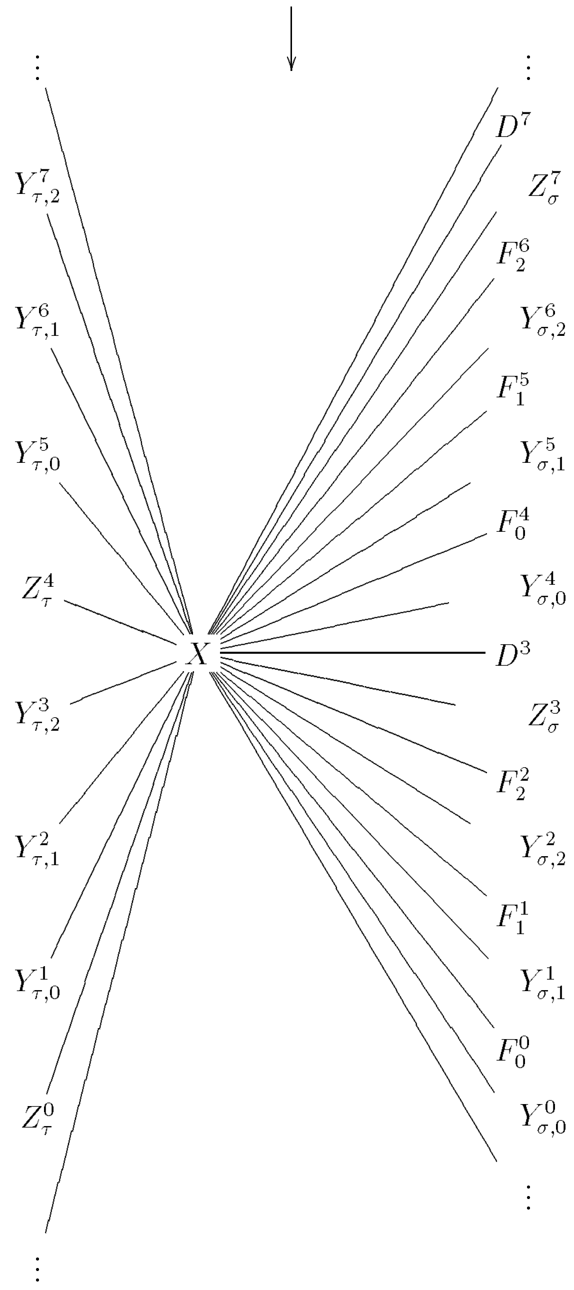


Figure 4.4 (Continued)

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



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 $\overline{\mathcal{G}}_n[t]$.

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 $\widetilde{F}_2^i$, $|i| \leqslant k$, of $\mathcal{G}_{|\sigma|}[t]$ 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 $v+1$, $r_{\sigma,v+1} \neq r_{\sigma,u+1}$, the copy of the σ-special component of $\mathcal{G}_{|\sigma|}[v]$ in $\mathcal{A}_v$ will extend a singleton component of $\mathcal{A}_u$. As in Section 3.4, 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 $\check{X}$, $\check{Z}^i$, $\check{B}^i$, $\check{S}^i$, $\check{C}^i$, $\check{Y}_0^i$, $\check{Y}_1^i$, $\check{Y}_2^i$, $\check{D}^i$, $\check{E}^i$, $\check{F}_0^i$, $\check{F}_1^i$, and $\check{F}_2^i$ be the intersection of the components of $\mathcal{A}_t$ that extend X, 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, with $\overline{\mathcal{A}}_t$. We are assuming that $\widetilde{S}^0 \cong \check{S}^0$, from which it follows that $\widetilde{C}^0 \cong \check{C}^0$, since all the components of $\mathcal{A}_t$ that contain copies of C^0_σ are isomorphic to either $\check{S}^0$ or $\check{C}^0$ and, since $\widetilde{S}^0$ and $\widetilde{C}^0$ have the same level, it cannot be the case that $\widetilde{S}^0 \cong \widetilde{C}^0$.

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 $i \equiv 0 \bmod 4$, $|i| \leqslant k$, $\widetilde{Z}^i \cong \check{Z}^i$, 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 $l < 3$.

For $i \equiv l + 1 \bmod 4$, $l < 3$, $|i| \leqslant k$, 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 $i \in \mathbb{Z}$, $|i| \leqslant k$, $\widetilde{D}^i \cong \check{D}^i$, $\widetilde{E}^i \cong \check{E}^i$, 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, $|i| \leqslant k$, 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, in order 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 4.4. 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.

4.2.4. 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 $i \equiv l+1 \bmod n+1$, $l < n$, in $\mathbb{Z}$:

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

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

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

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.

Before proceeding with the full construction, there are two more differences between this construction and the ones in Chapter 3 that should be mentioned.

The first one is that we will not be able to use the isomorphism recovery mechanism described in that chapter. Recall that, in Section 3.3, in order to compute the isomorphism from $\mathcal{A}^0$ to some $\mathcal{G}_n \cong \mathcal{A}^0$, we had to guess at the images of finitely many infinite special components. In the construction in this section, however, each special component will have infinitely many images, one for each level of $\mathcal{A}$. This creates a problem, which we solve as follows.

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.)

4.2.5. Definition. Let $\mathcal{G}$ be a computable leveled graph whose domain is co-infinite and 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 $L_{0,0}, \dots, L_{0,n}$ 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 $P_m \subset \omega$. Suppose further that, for each $m \leqslant n$, $P_m \nsubseteq \bigcup_{k \neq m} P_k$, and let l_m be the largest element of $P_m - \bigcup_{k \neq m} P_k$. Let $P = \bigcup_{m \leqslant n} P_m - \{l_0, \dots, l_n\}$.

For $i \in \mathbb{Z}$, let $L_{i,0}, \dots, L_{i,n}$ be the components of $\mathcal{G}^i$ isomorphic to $L_{0,0}, \dots, L_{0,n}$, 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 $\mathcal{A}_s$ counts as σ being active, which means that we must then wait for σ to recover 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 (4.2.2) is satisfied, while at the same time helping us to construct the computable isomorphisms needed to satisfy (4.2.3), 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 $\mathcal{G}_{|\sigma|}$.

Much as in Section 3.4, 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. In order 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.

Another difference between the construction in this section and the ones in

Chapter 3, which has already been noted above, is that σ will not be active except at σ-recovery stages. 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 that 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. (Note that this definition of $\ulcorner\sigma\urcorner$ is slightly different from the one in Section 3.3.) Fix a computable function ξ from ω onto $\mathbb{Z}$ such that, for each $i \in \mathbb{Z}$, there are infinitely many $n \in \omega$ such that $\xi(n) = i$.

4.2.6. 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.

4.2.7. 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 $[10\langle\ulcorner\sigma\urcorner, j\rangle]$, $j \in \omega$. In case $r = 0$, we write simply $(\mathcal{G}^*)^n_\sigma$.

Define $(\mathcal{G}^*)_{\supseteq\sigma} = \bigcup_{\tau\supseteq\sigma}(\mathcal{G}^*)_\tau$ and $(\mathcal{G}^*)^{n,r}_{\supseteq\sigma} = \bigcup_{\tau\supseteq\sigma}(\mathcal{G}^*)^{n,r}_\tau$. In case $r = 0$, 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* $\mathcal{G}_{|\sigma|}$.

We now proceed with the construction of $\mathcal{A}$. We will start with a computable structure $\mathcal{A}^0$ with co-infinite domain. In order 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. In order to ensure that $\mathcal{A}$ is computable, we require that the collection of sets $(|\mathcal{A}_{s+1}| - |\mathcal{A}_s|)_{s\in\omega}$ is 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 4.2.1.

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 $|i| \leqslant recov(\sigma, 2s)$, $(\mathcal{G}^i_n[2s])_\sigma \cong (\mathcal{A}^0_{2s})_\sigma$.

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

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

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 $\mathcal{A}^i_{2s}$ 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 $\mathcal{A}^0_{2s+1}$.

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

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

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 $|i| \leqslant recov(\sigma, 2s+1)$, $\mathcal{G}_n^i[2s+1]$ 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 $\mathcal{G}_n^i[2s+1]$ 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 $\mathcal{G}_n^i[2s+1]$ 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 $\mathcal{G}_n^i[s]$ 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 $\mathcal{G}_n^i[s]$ 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 $\sigma_0, \dots, \sigma_{k-1}$ 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 (4.2.1)–(4.2.3) are satisfied. Theorem 4.1.3 will then follow immediately.

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 (4.2.2) is satisfied.

4.2.8. 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. $\square$

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

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 $\langle\mathcal{A}, a_0\rangle$ to $\langle\mathcal{A}, a_i\rangle$ can take x to y. $\square$

4.2.10. Lemma. *For every $i \neq j \in \mathbb{Z}$, $\langle\mathcal{A}, a_i\rangle$ is not computably isomorphic to $\langle\mathcal{A}, a_j\rangle$.*

Proof. Since, given $k \neq l \in \mathbb{Z}$, 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 $\langle\mathcal{A}, a_0\rangle$ to $\langle\mathcal{A}, a_i\rangle$.

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 4.2.9, 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 $\langle\mathcal{A}, a_0\rangle$ to $\langle\mathcal{A}, a_i\rangle$ must take x to y. $\square$

In showing that (4.2.1) and (4.2.3) are satisfied, we will need a few easily checked facts about the construction.

As in Chapter 3, 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 $f(x)$.

4.2.11. 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 $\mathcal{A}^*$.*

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

4.2.13. Lemma. *For each $s \in \omega$ and $i, j \in \mathbb{Z}$, $\langle \mathcal{A}_s, a_i \rangle \cong \langle \mathcal{A}_s, a_j \rangle$ 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 $\mathcal{A}_s^j$.*

4.2.14. 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.*

4.2.15. 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 4.2.15 implies that, for each $k \in \omega$, there is at most one stage at which a copy of $[10k]$ participates in an operation.

4.2.16. 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 $i \in \mathbb{Z}$, $i \neq 0$.*

4.2.17. 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.*

4.2.18. 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 $S_{\sigma,s}$.*

4.2.19. 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 $2u+1 \in [t, s)$.*

4.2.20. 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 $K \cong L$.*

4.2.21. 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 (4.2.1) is satisfied. As in Chapter 3, this requires us to characterize the infinite components of $\mathcal{A}$.

4.2.22. 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 4.2.14 and 4.2.15. □

4.2.23. 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 4.2.14 and 4.2.17. □

4.2.24. 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 4.2.12, 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 4.2.18, 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 4.2.19, 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 4.2.14. $\square$

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

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 4.2.15, 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 $2s+2$. $\square$

4.2.26. 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, for all $t > u$, $r_{\sigma,t} = \lim_s r_{\sigma,s}$. 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 $2t+1 > 2s+1$, $T_{\sigma,2t+1} = T$, so $K_0, K_1, \ldots$ are involved in T-catch-up operations infinitely often. Thus, by Lemma 4.2.20, $K_0, K_1, \ldots$ are infinite and, for all $j, k \in \omega$, $K_j \cong K_k$.

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 4.2.14, 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 that therefore contain a copy of T. $\square$

4.2.27. Lemma. *For every $i \in \mathbb{Z}$, $\langle \mathcal{A}, a_i \rangle \cong \langle \mathcal{A}, a_0 \rangle$.*

Proof. By Lemma 4.2.13, 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 $K_0, \dots, K_n$ be as in Lemma 4.2.25. 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 $K_0^j, K_1^j, \dots$ 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 4.2.26, 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 4.2.22, 4.2.23, and 4.2.24, we have defined f on all infinite components of $\mathcal{A}$. By Lemmas 4.2.25 and 4.2.26, f is 1–1 and onto, and, for each infinite component K of $\mathcal{A}$, $f(K)$ is i-isomorphic to K. $\square$

We are left with showing that property (4.2.3) 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.

4.2.28. 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 4.2.15, 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 4.2.15, $(\mathcal{A}^0)_\sigma$ 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 4.2.14 and 4.2.15 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. So 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 any 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. □

4.2.29. 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 4.2.24 and 4.2.28. □

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 4.2.29, $r = \lim_s r_{\sigma,s}$ is well-defined. We wish to show that $\langle \mathcal{G}_n, g \rangle$ is computably isomorphic to $\langle \mathcal{A}, a_r \rangle$. 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 $\mathcal{G}_n$ to $\mathcal{A}$ extending b. As in Section 3.3, 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 a little more careful here, because $\mathcal{A}$ is not rigid, but Lemma 4.2.32 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 $\pi(k) = k$.

4.2.30. Definition. Let $p = (k,i) \in N_0$, $s \in \omega$, and $i \in \mathbb{Z}$. 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 $\mathcal{A}_0^i$.

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 $\widetilde{H} = \{(k,i) \mid k \in H,\ k \not\equiv 0 \bmod 10,\ i \in \mathbb{Z}\} \cup \{k \in H \mid k \equiv 0 \bmod 10\}$ and let P_H be the graph obtained by restricting the domain of $\mathcal{A}$ to $|\mathcal{B}| \cup \bigcup_{p \in \widetilde{H}} |(p)|$.

4.2.31. Lemma. *Let H and H' be disjoint subsets of ω^- and let f and f' be embeddings of P_H and $P_{H'}$, respectively, into $\mathcal{G}_n$. Then f and f' agree on $P_H \cap P_{H'}$.*

Proof. By Lemma 4.2.11, if $p \neq q \in N$ are such that $(p) = (q)$ then (p) is a component of $\mathcal{A}^*$. This clearly implies that $f^{-1} \circ (f' \restriction P_H \cap P_{H'})$ 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 $P_H \cap P_{H'}$. □

4.2.32. Lemma. *Let $H_0, \ldots, H_m$ be pairwise disjoint computable subsets of ω^- such that $\bigcup_{i=0}^m H_i = \omega^-$ and, for $i \neq j \leqslant m$, if K and L are components of P_{H_i} and P_{H_j}, respectively, and $f_i(K) = f_j(L)$, then $K = L$. Suppose that, for each $i \leqslant m$, there exists a computable embedding $f_i \supset b$ from P_{H_i} into $\mathcal{G}_n$, such that $\bigcup_{i=0}^m \operatorname{rng}(f_i) = |\mathcal{G}_n|$. Then there exists a computable isomorphism $f \supset b$ from $\mathcal{A}$ to $\mathcal{G}_n$.*

Proof. Since $H_0, \ldots, H_m$ are computable, $P_{H_0}, \ldots, P_{H_m}$ are c.e.. Since $\bigcup_{i=0}^m H_i = \omega^-$, $\bigcup_{i=0}^m P_{H_i} = \mathcal{A}$. Define f as follows. Given $x \in \mathcal{A}^0$, wait until x is enumerated into some P_{H_i}, $i \leqslant m$, and then let $f(x) = f_i(x)$. It is easy to check that the conditions imposed on $H_0, \ldots, H_m$ and $f_0, \ldots, f_m$, together with Lemma 4.2.31, imply that f is an isomorphism from $\mathcal{A}$ to $\mathcal{G}_n$. □

We will partition ω^- into the pairwise disjoint computable sets $H_0, \ldots, H_6$ shown in Table 4.1.

Table 4.1: $H_0, \dots, H_6$

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	$\{m \in \omega \mid$ for any $i \in \mathbb{Z}$, $((m,i))$ is an infinite component of $(\mathcal{A})_\tau,\ \tau \subset \sigma,$ that does not contain a copy of $[10k]$ for any $k \in \omega\}$
H_4	$\{10\langle\ulcorner\tau\urcorner, k, j\rangle + 1,\ 10\langle\ulcorner\tau\urcorner, k\rangle + l \mid \tau^\frown 0 \subseteq \sigma,\ j, k \in \omega,\ l \in \{2,3,4,5\}\} - H_3$
H_5	$\{10\langle\ulcorner\tau\urcorner, k, j\rangle + 8,\ 10\langle\ulcorner\tau\urcorner, k\rangle + l \mid \tau \subseteq \sigma,\ j, k \in \omega,\ l \in \{0,6,7\}\}$
H_6	$\{10\langle\ulcorner\tau\urcorner, k, j\rangle + 1,\ 10\langle\ulcorner\tau\urcorner, k\rangle + l \mid \tau = \sigma$ or $\sigma^\frown 0 \subseteq \tau,$ $j, k \in \omega,\ l \in \{2,3,4,5\}\} \cup$ $\{10\langle\ulcorner\tau\urcorner, k, j\rangle + 8,\ 10\langle\ulcorner\tau\urcorner, k\rangle + l \mid \sigma^\frown 0 \subseteq \tau,\ j, k \in \omega,\ l \in \{0,6,7\}\}$

The following two lemmas are versions of Lemmas 3.3.28 and 3.3.29.

4.2.33. 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. □

4.2.34. Lemma. *Let $H \subseteq \omega^-$ and $h : H \to \omega$ be computable. Suppose that, for each $p \in \widetilde{H}$ 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_H into $\mathcal{G}_n$, and f is computable.

Proof. Let $x \in P_H$ and let $p \in \widetilde{H}$ be such that $x \in (p)$. By Lemma 4.2.33, $(p)_{h(\pi(p))} \cong (p)$, so either (p) is finite or it contains a copy of $[10k]$ for some $k \in \omega$.

First suppose that (p) is finite. Since, by Lemma 4.2.13, 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 $i \in \mathbb{Z}$, $i \neq 0$, there is a unique finite set $T \subset \mathcal{G}_n$ such that 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 $x \in (p) \cap (\mathcal{A}^*)^m$. Since, by Lemma 4.2.16, $(p) \cap (\mathcal{A}^*)^m$ is not embeddable in any component $L \neq K$ of $\mathcal{A}$ unless (p) and L are i-isomorphic for some $i \in \mathbb{Z}$, $i \neq 0$, there is a unique finite set $T \subset \mathcal{G}_n$ such that 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_H into $\mathcal{G}_n$ extending b. Furthermore, it is easy to see that g_x can be computably determined given $x \in P_H$, which implies that f is computable.

$\square$

4.2.35. Lemma. *Let H_0 consist of all numbers of the form $10\langle \ulcorner\tau\urcorner, k, j\rangle + 1$ or $10\langle \ulcorner\tau\urcorner, k\rangle + l$, τ to the left of σ or $\tau^\frown 1 \subseteq \sigma$, $j, k \in \omega$, $l \in \{2,3,4,5\}$. Then there is a unique embedding $f_0 \supset b$ of P_{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 4.2.34 are satisfied. $\square$

4.2.36. Lemma. *Let H_1 consist of all numbers of the form $10\langle\ulcorner\tau\urcorner, k, j\rangle + 8$ or $10\langle\ulcorner\tau\urcorner, k\rangle + l$, τ to the left of σ, $j, k \in \omega$, $l \in \{0, 6, 7\}$. Then there is a unique embedding $f_1 \supset b$ of P_{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 4.2.34 are satisfied. $\square$

4.2.37. 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 $10\langle\ulcorner\tau\urcorner, k\rangle + l$, $j, k \in \omega$, $d \in \{1, 8\}$, $l \in \{0, 2, 3, 4, 5, 6, 7\}$. Let $s+1$ be the stage at which τ is initialized for the $(k+1)^{st}$ time. Let $i \in \mathbb{Z}$. If $m \in N_0$ then let $p = (m, i)$; otherwise, let $p = m$. Then (p) does not participate in an operation after stage s.*

Proof. If a singleton component of $\mathcal{A}_t$ of the form $[10\langle\ulcorner\tau\urcorner, q\rangle + l]$, $l \in \{0, 3, 6, 7\}$, 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 $[10\langle\ulcorner\tau\urcorner, q\rangle + l]$, $l \in \{2, 4, 5\}$, 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 4.2.17. $\square$

4.2.38. Lemma. *Let H_2 consist of all numbers of the form $10\langle \ulcorner\tau\urcorner, k, j\rangle + d$ or $10\langle \ulcorner\tau\urcorner, k\rangle + l$, τ to the right of $\sigma^\frown 0$, $j, k \in \omega$, $d \in \{1, 8\}$, $l \in \{0, 2, 3, 4, 5, 6, 7\}$. Then there is a unique embedding $f_2 \supset b$ of P_{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 4.2.37, the hypotheses of Lemma 4.2.34 are satisfied. $\square$

Let H_3 be the set of all $m \in \omega$ such that, for any $i \in \mathbb{Z}$, $((m, i))$ is an infinite component of $(\mathcal{A})_\tau$, $\tau \subset \sigma$, that does not contain a copy of $[10k]$ for any $k \in \omega$.

4.2.39. Lemma. *H_3 is computable.*

Proof. By Lemmas 4.2.15, 4.2.22, and 4.2.24, every element of H_3 must be of the form $10\langle \ulcorner\tau\urcorner, k\rangle + l$, $\tau^\frown 0 \subseteq \sigma$, $\lim_s r_{\tau,s}$ exists, $k \in \omega$, $l \in \{3, 4, 5\}$. Let $m \in \omega$ be of this form. We will describe an effective procedure for deciding whether $m \in H_3$.

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 $m \notin H_3$. Otherwise, by Lemma 4.2.21, for any $i \in \mathbb{Z}$, $m \in H_3$ if and only if $((m, i))_u$ 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 4.2.37, we see that if $((m,i))$, $i \in \mathbb{Z}$, has not participated in an operation by stage t then it will never participate in an operation, in which case $m \notin H_3$. On the other hand, if $((m,i))$, $i \in \mathbb{Z}$, has participated in an operation by stage t then, by Lemma 4.2.21, $m \in H_3$ if and only if $((m,i))_t$ contains a copy of T. $\square$

For $\tau \subset \sigma$, let H_3^τ be the set of all $m \in H_3$ such that, for any $i \in \mathbb{Z}$, $((m,i))$ is a component of $(\mathcal{A})_\tau$. The following lemma is easily checked.

4.2.40. Lemma. *For each $\tau \subset \sigma$, $P_{H_3^\tau}$ 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 $M = \bigcup_{\tau \subset \sigma} M_\tau$.

4.2.41. Lemma. *Each M_τ, $\tau \subset \sigma$, is c.e..*

Proof. By Lemmas 4.2.15, 4.2.22, and 4.2.24, it is enough to show that, for each $\tau^\frown 0 \subseteq \sigma$ such that $\lim_s r_{\tau,s}$ exists, M_τ is c.e..

Fix such a τ. Since $\lim_s r_{\tau,s}$ exists, $T_{\tau,s}$ comes to a limit T. By Lemma 4.2.21, the components of M_τ are exactly those that contain a copy of T. Since T is finite, we can effectively enumerate such components. $\square$

4.2.42. Lemma. *There exists a computable isomorphism $f_3 \supset b$ from P_{H_3} to the graph obtained by restricting the domain of $\mathcal{G}_n$ to $|B| \cup |M|$.*

Proof. Let $\tau \subset \sigma$. By Lemma 4.2.40, $P_{H_3^\tau}$ is c.e.. By Lemma 4.2.41, so is M_τ. Thus there exists a computable 1–1 and onto map d_τ from the tops of components of $P_{H_3^\tau}$ to the tops of components of M_τ such that if x is the top of a level-i component of $P_{H_3^\tau}$, $i \in \mathbb{Z}$, then $d_\tau(x)$ is the top of a level-$(i-r)$ component of M_τ. By

Lemma 4.2.26, d_τ can be extended to a computable isomorphism f_3^τ from $P_{H_3^\tau}$ to M_τ. Now define $f_3 = b \cup \bigcup_{\tau \subset \sigma} f_3^\tau$. $\square$

4.2.43. Lemma. *Let H_4 consist of all numbers of the form $10\langle \ulcorner \tau \urcorner, k, j\rangle + 1$ or $10\langle \ulcorner \tau \urcorner, k\rangle + l$, $\tau^\frown 0 \subseteq \sigma$, $j, k \in \omega$, $l \in \{2, 3, 4, 5\}$, that are not in H_3. Then there is a unique embedding $f_4 \supset b$ of P_{H_4} into $\mathcal{G}_n$, and f_4 is computable.*

Proof. Let $m \in H_4$ be of the form $10\langle \ulcorner \tau \urcorner, k\rangle + 3$, let $i \in \mathbb{Z}$, and let $p = (m, i)$. 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 4.2.37, 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. 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 $m \in H_4$ be of the form $10\langle \ulcorner \tau \urcorner, k, j\rangle + 1$ or $10\langle \ulcorner \tau \urcorner, k\rangle + l$, $l \in \{2, 4, 5\}$, let $i \in \mathbb{Z}$, and let $p = (m, i)$. 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. 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 4.2.34 are satisfied. $\square$

4.2.44. Lemma. *Let H_5 consist of all numbers of the form $10\langle\ulcorner\tau\urcorner,k,j\rangle+8$ or $10\langle\ulcorner\tau\urcorner,k\rangle+l$, $\tau\subseteq\sigma$, $j,k\in\omega$, $l\in\{0,6,7\}$. Then there is a unique embedding $f_5\supset b$ of P_{H_5} into $\mathcal{G}_n$, and f_5 is computable.*

Proof. Let $m\in H_5$ 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 $m\in N_0$ then let $p=(m,i)$; otherwise, let $p=m$. 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 4.2.34 are satisfied. $\square$

Let H_6' consist of all numbers of the form $10\langle\ulcorner\tau\urcorner,k,j\rangle+1$ or $10\langle\ulcorner\tau\urcorner,k\rangle+l$, $\tau=\sigma$ or $\sigma^\frown 0\subseteq\tau$, $j,k\in\omega$, $l\in\{2,3,4,5\}$. Let H_6'' consist of all numbers of the form $10\langle\ulcorner\tau\urcorner,k,j\rangle+8$ or $10\langle\ulcorner\tau\urcorner,k\rangle+l$, $\sigma^\frown 0\subseteq\tau$, $j,k\in\omega$, $l\in\{0,6,7\}$. Let $H_6=H_6'\cup H_6''$.

4.2.45. 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, for all $s\geqslant u$, $r_{\sigma,s}=r$. 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 $K_0,\dots,K_m$ be the components of $(\mathcal{A}^i_{2s})_\tau$ that participate in an operation at stage $2s+1$. Let $K_0',\dots,K_m'$ be the components of $\mathcal{A}_{2t+1}$ that extend $K_0,\dots,K_m$, respectively. Then the following hold.*

1. *There exist components $\widehat{K}_0,\dots,\widehat{K}_m$ of $\mathcal{G}^{i-r}_n[2s]$ such that $\widehat{K}_0\cong K_0,\dots,\widehat{K}_m\cong K_m$.*

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 $\widehat{K}_j$. Since, 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, it must be the case that $\widehat{K}'_j \cong K'_j$. $\square$

4.2.46. 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, for all $s \geqslant u$, $r_{\sigma,s} = r$. 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 Definition 4.2.1) 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 $Y_0, \ldots, Y_{m-1}$, X, Z, B, S, C, D, E, and $F_0, \ldots, F_{m-1}$ be $Y^i_{\tau,0,2s+1}, \ldots, Y^i_{\tau,m,2s+1}$, 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 $F^i_{0,2s+1}, \ldots, F^i_{m,2s+1}$, respectively. Let $Y'_0, \ldots, Y'_{m-1}$, X', Z', B', S', C', D', E', and $F'_0, \ldots, F'_{m-1}$ 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* $\mathcal{G}_n^{i-r}[2s+1]$ *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* $\mathcal{G}_n^{i-r}[2s+1]$ *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. 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 $Y_0', \ldots, Y_{m-1}'$ and X'. Since, for each $j < m$, $\widehat{Y}_j' \cong Y_j'$, 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', $j < m$, $j \neq l$, 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', $j < m$, $j \neq l$, 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 $\widehat{D}' \cong D'$. $\square$

The following lemma can be easily checked.

4.2.47. Lemma. *Let $p \in \widetilde{H}_6$. If $(p)_{2s+1}$ participates in an operation at stage $2s+2$ then it 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 $F^i_{m,2s+1}$, $\tau = \sigma$ or $\sigma\frown 0 \subseteq \tau$, $m \in \omega$, $i \in \mathbb{Z}$, and σ is active at stage $2s+2$.*

4.2.48. Lemma. *Let u be a stage after which σ is never initialized and such that, for all $s \geqslant u$, $r_{\sigma,s} = r$. Let $s+1 > u$ be a σ-recovery stage and let $t+1$ be the next σ-recovery stage after stage $s+1$. Let $p \in \widetilde{H}_6$. 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 4.2.45, 4.2.46, and 4.2.47. $\square$

4.2.49. Lemma. *Let u be a stage after which σ is never initialized and such that, for all $s \geqslant u$, $r_{\sigma,s} = r$. Let $x \in P_{H_6}$. There exists a σ-recovery stage $s+1 > u$ such that x is contained in $(p)_s$ for some $p \in \widetilde{H}_6$ 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 $(p)_s$, $p \in \widetilde{H}_6$, and $(p)_s$ is involved in an operation at stage $s+1$. Now it follows from Lemmas 4.2.45, and 4.2.46 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 4.2.48 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'. $\square$

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 $m \in H_6$.

4.2.50. Lemma. *There exists a computable isomorphism $f_6 \supset b$ from P_{H_6} to T.*

Proof. We begin by defining $f_6 \restriction \mathcal{B} \equiv b$. Now let u be a stage after which σ is never initialized and such that, for all $s \geqslant u$, $r_{\sigma,s} = r$. Given $x \in P_{H_6}$, find the least σ-recovery stage $s+1 > u$ such that x is contained in a component $(p)_s$, $p \in \widetilde{H}_6$, 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 4.2.49. Let d_x be the unique isomorphism from $(p)_s$ to L and define $f_6(x) = d_x(x)$.

We need to show that f_6 is computable, that it is an embedding, and that its range is all of T.

Since d_x can be computably determined given $x \in P_{H_6}$, f_6 is computable.

By Lemma 4.2.49, all we need to do to show that f_6 is an embedding is to show that if x and y are both contained in a component (p), $p \in \widetilde{H}_6$, then $f_6(x)$ and $f_6(y)$ are contained in the same component of $\mathcal{G}_n$. But this follows from Lemma 4.2.48, 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 $m \in H_6$ then K is $(p)_s$ for some $p \in \widetilde{H}_6$.

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 $p \in \widetilde{H}_6$, L is in the range of f_6.

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 $m \in H_6$. 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 $p \in \widetilde{H}_6$, x is in the range of f_6. Thus L is in the range of f_6. $\square$

Now $H_0, \ldots, H_6$ are computable subsets of ω^- such that $\bigcup_{i=0}^{6} H_i = \omega^-$. It is not hard to check that, for $i \neq j \leqslant 6$, if K and L are components of P_{H_i} and P_{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_4, and f_5, together with the surjectivity of f_3 and f_6, imply that $\bigcup_{i=0}^{m} \mathrm{rng}(f_i) = |\mathcal{G}_n|$. So, combining Lemmas 4.2.35, 4.2.36, 4.2.38, 4.2.42, 4.2.43, 4.2.44, and 4.2.50 with Lemma 4.2.32, we have the following result.

4.2.51. Lemma. *There exists a computable isomorphism from* $\langle \mathcal{A}, a_r \rangle$ *to* $\langle \mathcal{G}_n, g \rangle$.

Theorem 4.1.3 follows from Lemmas 4.2.10, 4.2.27, and 4.2.51. ■

Chapter 5

Relations on Algebraic Structures I: Positive Results

5.1 Introduction

Whenever a computable structure with a particularly interesting property is found, it is natural to ask whether similar examples can be found within well-known classes of algebraic structures, such as groups, rings, lattices, and so forth. As an example, let us consider the computable dimension of computable structures.

It is easy to construct computable structures with computable dimension 1 or ω. Indeed, most familiar structures and even all members of many classes of familiar structures have computable dimension 1 or ω. Nurtazin [31], for example, showed that all decidable structures fall into this category. Goncharov [8] later extended this result to 1-decidable structures, and there have been several other familiar classes of structures for which similar results have been established.

5.1.1. Theorem. *All structures in each of the following classes have computable dimension* 1 *or* ω.

- **(Nurtazin; Metakides and Nerode)** *algebraically closed fields*
- **(Nurtazin)** *real closed fields*
- **(Goncharov)** *Abelian groups*
- **(Goncharov and Dzgoev; Remmel)** *linear orderings*
- **(Goncharov; LaRoche)** *Boolean algebras*
- **(Goncharov)** Δ_2^0-*categorical structures*

The result for algebraically closed and real closed fields is implied by the results in [31]; the result for algebraically closed fields was also independently proved in [29]. The result for Abelian groups appears in [12], that for linear orderings independently in [15] and [33], and that for Δ_2^0-categorical structures in [13]. The result for Boolean algebras appears in full in [14], though it is implicit in earlier work of Goncharov and, independently, in [27].

Thus, an important question early in the development of computable model theory was whether there exist computable structures of finite computable dimension greater than 1. This question was answered positively by Goncharov [11].

5.1.2. Theorem (Goncharov). *For each* $n > 0$ *there is a computable structure with computable dimension* n.

Further investigation led to examples of computable structures with finite computable dimension greater than 1 in several classes of algebraic structures. In each case, the proof consists of coding families of c.e. sets with a finite number of computable enumerations (up to a suitable notion of computable equivalence of enumerations) in a sufficiently effective way.

5.1.3. Theorem. *For each $n > 0$ there are structures with computable dimension n in each of the following classes.*

- **(Goncharov)** *graphs, partial orderings, and lattices*
- **(Goncharov, Molokov, and Romanovskii)** *2-step nilpotent groups*
- **(Kudinov)** *integral domains*

The results for partial orderings and (implicitly) graphs appear in [11], and the result for lattices is an easy consequence of the results in that paper. The result for 2-step nilpotent groups (which improves a result in [12]) appears in [17], and that for integral domains in [26].

In the original proofs of Theorems 3.1.2 and 4.1.2, the structures in question were directed graphs, and the relation mentioned in Theorem 3.1.2 was unary. The same holds of the results of Chapters 3 and 4. It is natural to ask, in the spirit of what was done for structures of finite computable dimension, for which theories these theorems remain true if we also require that $\mathcal{A}$ be a model of the given theory.

In this chapter, we present a method for showing that Theorems 3.1.2 and 4.1.2, as well as related results, including the results of Chapters 3 and 4, remain true if we also require that $\mathcal{A}$ be a model of a given theory, and apply it to the cases of symmetric, irreflexive graphs; integral domains of arbitrary characteristic; and commutative semigroups. These results also appear in [22], where the following cases are also dealt with: partial orderings, lattices, rings (with zero-divisors) of arbitrary characteristic, and 2-step nilpotent groups.

Our method is based on coding computable graphs with the desired properties into models of the given theory in a way that is effective enough to preserve these properties. This approach is much simpler than attempting to adapt the original proofs of the theorems under consideration. Furthermore, our codings are sufficiently effective to make other similar results that might be proved for graphs in the future carry over to the classes of structures mentioned above without additional work.

Notice that, by Theorem 5.1.1, most of the results mentioned above cannot be extended from partial orderings to linear orderings, from lattices to Boolean algebras, or from commutative semigroups and 2-step nilpotent groups to Abelian groups. We will say more about this in Chapter 6. A natural open question is what is the situation for fields. It is not even known whether there exist fields of finite computable dimension greater than 1.

5.2 A Sufficient Condition

In this section, we give a sufficient condition for a coding of a graph into a structure to be effective enough for our purposes. This condition is not the most general one we could give, but it is sufficient for our needs. It corresponds to an especially effective version of interpretations of theories (in the standard model-theoretic sense). (See, e.g., chapter 5 of [23] for more on interpretations of theories.)

If Q is an equivalence relation on a set D then by a *set of Q-representatives* we mean a set of elements of D containing exactly one member of each Q-equivalence class.

5.2.1. Theorem. *Let $\mathcal{G}$ be a computably presentable directed graph and let $\mathcal{A}$ be a computably presentable structure. Suppose there exist intrinsically computable, invariant relations $D(x)$, $Q(x,y)$, and $R(x,y)$ on $|\mathcal{A}|$ and a map $G \mapsto A_G$ from the set of computable presentations of $\mathcal{G}$ to the set of computable presentations of $\mathcal{A}$ with the following properties.*

(P1) For each computable presentation G of $\mathcal{G}$, there is a computable map $g_G : D(A_G) \xrightarrow{onto} |G|$ such that, for $x, y \in D(A_G)$, $R^{A_G}(x,y) \Leftrightarrow E^G(g_G(x), g_G(y))$ and $Q^{A_G}(x,y) \Leftrightarrow g_G(x) = g_G(y)$. (Note that this implies that Q is an equivalence relation and that if $Q(x,x')$ and $Q(y,y')$ then $R(x,y) \Leftrightarrow R(x',y')$.)

(P2) For every pair S, S' of sets of Q-representatives, if $f : S \xrightarrow[onto]{1\text{-}1} S'$ is such that for every $x, y \in S$, $R(x,y) \Leftrightarrow R(f(x), f(y))$, then f can be extended to an automorphism of $\mathcal{A}$.

(P3) If G is a computable presentation of $\mathcal{G}$ and S is a computable set of Q^{A_G}-representatives then there is a computable set of existential formulas $\{\varphi_0(\vec{a}, \vec{b}_0, x), \varphi_1(\vec{a}, \vec{b}_1, x), \ldots\}$ such that $\vec{a}$ is a tuple of elements of $|A_G|$, for each $i \in \omega$, $\vec{b}_i$ is a tuple of elements of S, each $x \in |A_G|$ satisfies some φ_i, and no two

elements of $|A_G|$ satisfy the same φ_i. (Such a set of formulas is known as a defining family *for $\langle A_G, a\rangle_{a\in S}$.)*

Then the following hold.

1. *$\mathcal{A}$ has the same computable dimension as $\mathcal{G}$.*

2. *If $x \in |\mathcal{G}|$ then there exists an $a \in D(\mathcal{A})$ such that $\langle \mathcal{A}, a\rangle$ has the same computable dimension as $\langle \mathcal{G}, x\rangle$.*

3. *If $V \subseteq |\mathcal{G}|$ then there exists a $U \subseteq D(\mathcal{A})$ such that $\mathrm{DgSp}_{\mathcal{A}}(U) = \mathrm{DgSp}_{\mathcal{G}}(V)$.*

Proof. We begin with two remarks. First, if G is a computable presentation of $\mathcal{G}$ and S is a set of Q^{A_G}-representatives then $g_G \upharpoonright S$ is one-to-one. Second, if S and S' are sets of Q-representatives and $f : S \xrightarrow[\text{onto}]{1\text{–}1} S'$ is such that, for every $x \in S$, $Q(x, f(x))$, then (P1) implies that, for every $x, y \in S$, $R(x, y) \Leftrightarrow R(f(x), f(y))$, so that, by (P2), f can be extended to an automorphism of $\mathcal{A}$.

We now need a few lemmas.

5.2.2. Lemma. *Let A and G be computable presentations of $\mathcal{A}$ and $\mathcal{G}$, respectively, let S be a computable set of Q^A-representatives, and let $f : A \cong A_G$. If $f \upharpoonright S$ is computable then so is f.*

Proof. It is enough to show that f^{-1} is computable. Given $x \in |A_G|$, find an $i \in \omega$ such that $A_G \vDash \varphi_i(\vec{a}, \vec{b}_i, x)$, where $\varphi_i(\vec{a}, \vec{b}_i, x)$ is as in (P3). By definition, x is the only element of $|A_G|$ that satisfies φ_i. Thus, there exists a unique $y \in |A|$ such that $A \vDash \varphi_i(f^{-1}(\vec{a}), f^{-1}(\vec{b}_i), y)$, and $f^{-1}(x) = y$. $\square$

5.2.3. Lemma. *Let A and G be computable presentations of $\mathcal{A}$ and $\mathcal{G}$, respectively. Let S be a computable set of Q^A-representatives and let S' be a computable set of Q^{A_G}-representatives. Suppose that there exists a (computable) map $f : S \xrightarrow[onto]{1-1} S'$ such that, for each $x, y \in S$, $R^A(x, y) \Leftrightarrow R^{A_G}(f(x), f(y))$. Then f can be extended to a (computable) isomorphism $\hat{f} : A \cong A_G$.*

Proof. Since A and A_G are both computable presentations of $\mathcal{A}$, there exists an isomorphism $h : A \cong A_G$. By our second remark above, h can be chosen so that $h(S) = S'$. Let $d \equiv h \restriction S$. Then $c \equiv f \circ d^{-1}$ is a one-to-one map from S' onto itself such that, for each $x, y \in S$, $R^{A_G}(x, y) \Leftrightarrow R^{A_G}(c(x), c(y))$. So, by (P2), c can be extended to $\hat{c} : A_G \cong A_G$. Now let $\hat{f} \equiv \hat{c} \circ h$. Then $\hat{f} : A \cong A_G$ and $\hat{f} \restriction S \equiv f \circ d^{-1} \circ d \equiv f$. If f is computable then Lemma 5.2.2 implies that $\hat{f}$ is computable. □

We now need a few definitions. Let A be a computable presentation of $\mathcal{A}$. Let $\widehat{D}(A) = \{x \in D(A) \mid y < x \Rightarrow \neg Q^A(x, y)\}$, where $<$ is the usual ordering on ω. Note that $\widehat{D}(A)$ is a computable set of Q^A-representatives. Define G_A to be the computable graph whose universe is $\widehat{D}(A)$, with an edge between x and y if and only if $R^A(x, y)$. For any computable presentation G of $\mathcal{G}$, let $d_G = g_G \restriction \widehat{D}(A_G)$. Note that, by our first remark above, d_G is one-to-one and hence invertible.

5.2.4. Lemma. *If G and G' are computable presentations of $\mathcal{G}$ and $h : G \cong G'$ is a (computable) isomorphism then there exists a (computable) isomorphism $\hat{f} : A_G \cong A_{G'}$ such that $\hat{f} \restriction \widehat{D}(A_G) \equiv d_{G'}^{-1} \circ h \circ d_G$.*

Proof. Let $f : \widehat{D}(A_G) \xrightarrow[\text{onto}]{1\text{–}1} \widehat{D}(A_{G'})$ be defined by $f \equiv d_{G'}^{-1} \circ h \circ d_G$. If h is computable then so is f. Furthermore, for each $x, y \in \widehat{D}(A_G)$, $R^{A_G}(x, y) \Leftrightarrow E^G(d_G(x), d_G(y)) \Leftrightarrow E^{G'}(h \circ d_G(x), h \circ d_G(y)) \Leftrightarrow R^{A_{G'}}(f(x), f(y))$. So, by Lemma 5.2.3, there exists an isomorphism $\hat{f} : A_G \cong A_{G'}$ extending f and if h is computable then so is $\hat{f}$. $\square$

5.2.5. Lemma. *If A and A' are computable presentations of $\mathcal{A}$ and $f : A \cong A'$ is a (computable) isomorphism then there exists a map $h : f(\widehat{D}(A)) \xrightarrow[\text{onto}]{1\text{–}1} \widehat{D}(A')$ such that $h \circ (f \restriction \widehat{D}(A))$ is a (computable) isomorphism from G_A to $G_{A'}$.*

Proof. For $x \in f(\widehat{D}(A))$, let $h(x)$ be the unique $y \in \widehat{D}(A')$ such that $Q^{A'}(x, y)$. Then $E^{G_A}(x, y) \Leftrightarrow R^A(x, y) \Leftrightarrow R^{A'}(f(x), f(y)) \Leftrightarrow R^{A'}(h \circ f(x), h \circ f(y)) \Leftrightarrow E^{G_{A'}}(h \circ f(x), h \circ f(y))$. Thus $h \circ (f \restriction \widehat{D}(A))$ is an isomorphism from G_A to $G_{A'}$. Furthermore, if f is computable then clearly so is $h \circ (f \restriction \widehat{D}(A))$. $\square$

5.2.6. Lemma. *If G is a computable presentation of $\mathcal{G}$ then d_G is a computable isomorphism from G_{A_G} to G.*

Proof. If $x, y \in |G_{A_G}|$ then $E^{G_{A_G}}(x, y) \Leftrightarrow R^{A_G}(x, y) \Leftrightarrow E^G(d_G(x), d_G(y))$. Thus d_G is a computable isomorphism from G_{A_G} to G. $\square$

5.2.7. Lemma. *If A is a computable presentation of $\mathcal{A}$ then there exists a computable isomorphism $f : A \cong A_{G_A}$ such that $f \restriction \widehat{D}(A) \equiv d_{G_A}^{-1}$.*

Proof. The map $d_{G_A}^{-1}$ is computable. Furthermore, for each $x, y \in D(A)$, $R^A(x, y) \Leftrightarrow E^{G_A}(x, y) \Leftrightarrow R^{A_{G_A}}(d_{G_A}^{-1}(x), d_{G_A}^{-1}(y))$. So, by Lemma 5.2.3, $d_{G_A}^{-1}$ can be extended to a computable isomorphism from A to A_{G_A}. $\square$

We are now ready to show that 1–3 in the statement of Theorem 5.2.1 hold.

5.2.8. Proposition. *$\mathcal{A}$ has the same computable dimension as $\mathcal{G}$.*

Proof. Let G and G' be computable presentations of $\mathcal{G}$ that are not computably isomorphic. By Lemma 5.2.6, G_{A_G} and $G_{A_{G'}}$ are not computably isomorphic. Thus, by Lemma 5.2.5, A_G and $A_{G'}$ are not computably isomorphic. So the computable dimension of $\mathcal{A}$ is at least the same as that of $\mathcal{G}$.

Now let A and A' be computable presentations of $\mathcal{A}$ that are not computably isomorphic. By Lemma 5.2.7, A_{G_A} and $A_{G_{A'}}$ are not computably isomorphic. Thus, by Lemma 5.2.4, G_A and $G_{A'}$ are not computably isomorphic. So the computable dimension of $\mathcal{G}$ is at least the same as that of $\mathcal{A}$. ☐

5.2.9. Proposition. *Let $x \in |\mathcal{G}|$. There exists an $a \in D(\mathcal{A})$ such that $\langle \mathcal{A}, a \rangle$ has the same computable dimension as $\langle \mathcal{G}, x \rangle$.*

Proof. Let $f : \mathcal{G} \cong G$ be a computable presentation of $\mathcal{G}$, let $h : \mathcal{A} \cong A_G$ be an isomorphism, and let $a = h^{-1} \circ d_G^{-1} \circ f(x)$. By Lemma 5.2.4, for every computable presentation $f' : \mathcal{G} \cong G'$ of $\mathcal{G}$ there exists an isomorphism $k : \mathcal{A} \cong A_{G'}$ such that $a = k^{-1} \circ d_{G'}^{-1} \circ f'(x)$. The rest of the proof is similar to the proof of Proposition 5.2.8.

Let $\langle G, x^G \rangle$ and $\langle G', x^{G'} \rangle$ be computable presentations of $\langle \mathcal{G}, x \rangle$ that are not computably isomorphic. By Lemma 5.2.6, $\langle G_{A_G}, d_G^{-1}(x^G) \rangle$ and $\langle G_{A_{G'}}, d_{G'}^{-1}(x^{G'}) \rangle$ are not computably isomorphic. Thus, by Lemma 5.2.5, $\langle A_G, d_G^{-1}(x^G) \rangle$ and $\langle A_{G'}, d_{G'}^{-1}(x^{G'}) \rangle$ are not computably isomorphic. So the computable dimension of $\langle \mathcal{A}, a \rangle$ is at least the same as that of $\langle \mathcal{G}, x \rangle$.

Now let $\langle B, a^B \rangle$ and $\langle B', a^{B'} \rangle$ be computable presentations of $\langle \mathcal{A}, a \rangle$ that are not computably isomorphic. By Lemma 5.2.3, there exist computable presentations

$\langle A, a^A\rangle$ and $\langle A', a^{A'}\rangle$ of $\langle \mathcal{A}, a\rangle$ such that $\langle A, a^A\rangle$ is computably isomorphic to $\langle B, a^B\rangle$, $\langle A', a^{A'}\rangle$ is computably isomorphic to $\langle B', a^{B'}\rangle$, $a^A \in \widehat{D}(A)$, and $a^{A'} \in \widehat{D}(A')$. By Lemma 5.2.7, $\langle A_{G_A}, d^{-1}_{G_A}(a^A)\rangle$ and $\langle A_{G_{A'}}, d^{-1}_{G_{A'}}(a^{A'})\rangle$ are not computably isomorphic. Thus, by Lemma 5.2.4, $\langle G_A, a^A\rangle$ and $\langle G_{A'}, a^{A'}\rangle$ are not computably isomorphic. So the computable dimension of $\langle \mathcal{G}, x\rangle$ is at least the same as that of $\langle \mathcal{A}, a\rangle$. $\square$

5.2.10. Proposition. *Let $V \subseteq |\mathcal{G}|$. There is a $U \subseteq D(\mathcal{A})$ such that* $\mathrm{DgSp}_{\mathcal{A}}(U) = \mathrm{DgSp}_{\mathcal{G}}(V)$.

Proof. Let $f : \mathcal{G} \cong G$ be a computable presentation of $\mathcal{G}$, let $h : \mathcal{A} \cong A_G$ be an isomorphism, and let $U = \{x \in D \mid \exists y[Q(x,y) \wedge y \in h^{-1} \circ d_G^{-1} \circ f(V)]\}$. Notice that this definition guarantees that $Q(x,y) \Rightarrow (U(x) \Leftrightarrow U(y))$.

By Lemma 5.2.4, for every computable presentation $f' : \mathcal{G} \cong G'$ of $\mathcal{G}$ there exists an isomorphism $k : \mathcal{A} \cong A_{G'}$ such that $U = \{x \in D \mid \exists y[Q(x,y) \wedge y \in k^{-1} \circ d_{G'}^{-1} \circ f'(V)]\} = \{x \in D \mid \neg\exists y[Q(x,y) \wedge y \in k^{-1} \circ d_{G'}^{-1} \circ f'(|\mathcal{G}| - V)]\}$, which implies that $\mathrm{DgSp}_{\mathcal{G}}(V) \subseteq \mathrm{DgSp}_{\mathcal{A}}(U)$.

On the other hand, for every computable presentation $k : \mathcal{A} \cong A$ of $\mathcal{A}$, our second remark at the beginning of the proof of Theorem 5.2.1 implies that there exists an automorphism $p : \mathcal{A} \cong \mathcal{A}$ such that $p \circ h^{-1}(\widehat{D}(A_G)) = k^{-1}(\widehat{D}(A))$. It is not hard to check that $m = k \circ p \circ h^{-1} \circ d_G^{-1} \circ f : \mathcal{G} \cong G_A$ is an isomorphism and that $V = m^{-1} \circ k(U \restriction k^{-1}(\widehat{D}(A)))$. This implies that $\mathrm{DgSp}_{\mathcal{A}}(U) \subseteq \mathrm{DgSp}_{\mathcal{G}}(V)$. $\square$

Theorem 5.2.1 follows from Propositions 5.2.8, 5.2.9, and 5.2.10. $\blacksquare$

5.3 Undirected Graphs

It is usually easier to code symmetric, irreflexive graphs into structures than arbitrary directed graphs. Thus, as a simple but useful example of the application of Theorem 5.2.1, we exhibit a coding of an arbitrary directed graph into a symmetric, irreflexive graph that allows us to restrict our attention to symmetric, irreflexive graphs when applying Theorem 5.2.1.

Let G be a computable infinite directed graph with edge relation E.

Let G' be a computable presentation of G. The computably presentable symmetric, irreflexive graph $H_{G'} = \langle |H_{G'}|, F \rangle$ is defined as follows.

1. $|H_{G'}| = \{a, a', b\} \cup \{c_i, d_i, e_i \mid i \in |G'|\}$.
2. $F(x, y)$ holds only in the following cases.
   (a) $F(a, a')$ and $F(a', a)$.
   (b) For all $i \in |G'|$,
      i. $F(a, c_i)$ and $F(c_i, a)$,
      ii. $F(b, e_i)$ and $F(e_i, b)$,
      iii. $F(c_i, d_i)$ and $F(d_i, c_i)$.
      iv. $F(d_i, e_i)$ and $F(e_i, d_i)$.
   (c) If $E^{G'}(i, j)$ then $F(c_i, e_j)$ and $F(e_j, c_i)$.

As an example, Figure 5.1 shows part of H_G in the case in which $E(0, 1)$, $E(1, 0)$, $E(1, 2)$, $E(2, 2)$, $\neg E(0, 0)$, $\neg E(0, 2)$, $\neg E(1, 1)$, $\neg E(2, 0)$, and $\neg E(2, 1)$.

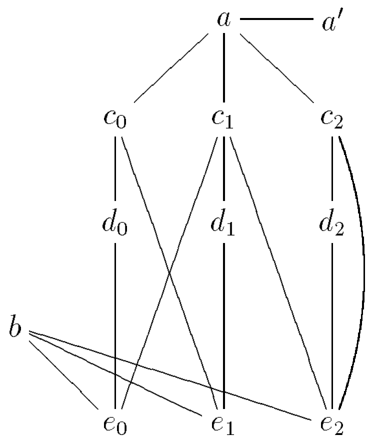


Figure 5.1: Part of the graph H_G

Fix a computable presentation of $H_{G'}$ for which the map $g_{G'} : c_i \mapsto i$ is computable and identify $H_{G'}$ with this presentation.

It is easy to see that, for any computable presentation G' of G, $H_G \cong H_{G'}$. Now let a, a', and b be as in the definition of H_G and define

$$D(x) = \{x \in |H_G| \mid x \neq a' \wedge F(a, x)\},$$

$$Q(x, y) = \{(x, x) \mid D(x)\},$$

and

$$R(x, y) = \{(x, y) \mid D(x) \wedge D(y) \wedge \exists d, e(F(b, e) \wedge F(y, d) \wedge F(d, e) \wedge F(x, e))\}.$$

Clearly, D and Q are intrinsically computable relations, and so is R, since, for $x, y \in D(H_G)$,

$$\exists d, e(F(b, e) \wedge F(y, d) \wedge F(d, e) \wedge F(x, e)) \Leftrightarrow$$
$$\neg \exists d, e(F(b, e) \wedge F(y, d) \wedge F(d, e) \wedge \neg F(x, e)).$$

Furthermore, if G' is a computable presentation of G then $D(H_{G'}) = \mathrm{dom}(g_{G'})$, $Q^{H_{G'}}(x,y) \Leftrightarrow g_{G'}(x) = g_{G'}(y)$, and $R^{H_{G'}}(x,y) \Leftrightarrow E^{G'}(g_{G'}(x), g_{G'}(y))$.

To see that D, Q, and R are invariant, it is enough to notice that a is the only element of H_G that satisfies the formula

$$\exists^\infty y(F(x,y)) \wedge \exists z(F(x,z) \wedge \forall w(F(w,z) \rightarrow w = x)),$$

a' is the only element of H_G that satisfies

$$F(x,a) \wedge \forall y(F(x,y) \rightarrow y = a)$$

and $x = b$ is the only element of H_G that satisfies

$$\exists^\infty y(F(x,y)) \wedge \neg F(a,x) \wedge \neg\exists z(F(a,z) \wedge F(x,z)).$$

The only set of Q-representatives is $D(H_G)$ itself. If $f : D(H_G) \xrightarrow[\text{onto}]{1\text{–}1} D(H_G)$ is such that $R(x,y) \Leftrightarrow R(f(x), f(y))$ then we can extend f as follows. Let a, a', b, d_i, and e_i be as in the definition of H_G. Let $f(a) = a$, $f(a') = a'$, $f(b) = b$, $f(d_i) = d_{g_G \circ f(i)}$, and $f(e_i) = e_{g_G \circ f(i)}$. It can be easily checked that this extended map is an automorphism of H_G.

Finally, given a computable presentation G' of G, let a and b be as in the definition of $H_{G'}$ and consider the computable set of formulas

$$\begin{aligned}&\{x = a, x = a', x = b\} \cup \{x = c \mid c \in D(H_{G'})\} \cup \\ &\qquad \{x \neq a \wedge F(c,x) \wedge \neg F(b,x) \mid c \in D(H_{G'})\} \cup \\ &\qquad\qquad \{F(b,x) \wedge \exists d(F(c,d) \wedge F(d,x)) \mid c \in D(H_{G'})\}.\end{aligned}$$

Clearly, every $x \in |H_{G'}|$ satisfies some formula in this set, with no two elements satisfying the same formula.

It now follows from Theorem 5.2.1 that Theorems 3.1.2 and 4.1.2 and the results of Chapters 3 and 4 remain true if we require that $\mathcal{A}$ be a symmetric, irreflexive graph and that U be a subgraph of $\mathcal{A}$.

5.4 Integral Domains and Commutative Semigroups

In this section, we present a coding of a graph into an integral domain inspired by Kudinov's coding [26] of a family of c.e. sets into an integral domain of characteristic 0 and show how this leads to a proof that Theorems 3.1.2 and 4.1.2 and the results of Chapters 3 and 4 remain true if we also require that $\mathcal{A}$ be a integral domain of arbitrary characteristic and that U be a subring of $\mathcal{A}$. Because our coding does not make use of the additive structure of the domain, we will simultaneously handle the case of commutative semigroups.

Let p be either 0 or a prime. We adopt the convention that $\mathbb{Z}_0 = \mathbb{Z}$. If $p = 0$ then let $\mathbb{F} = \mathbb{Q}$; otherwise, let $\mathbb{F} = \mathbb{Z}_p$. Let I be the set of invertible elements of $\mathbb{Z}_p$. Note that I is finite.

The graphs constructed in Section 5.3 have the following property: For every finite set of nodes S there exist nodes $x, y \notin S$ that are connected by an edge. Thus, in applying Theorem 5.2.1, we can restrict our attention to graphs with this property.

Let G be a symmetric, irreflexive, infinite computable graph with edge relation E, having the property mentioned in the previous paragraph.

Let G' be a computable presentation of G. We assume without loss of generality that $|G'| = \omega$. The computably presentable integral domain with unit $A_{G'}$ is defined to be

$$\mathbb{Z}_p[x_i \mid i \in \omega]\left[\frac{y}{x_i x_j} \mid E^{G'}(i,j)\right]\left[\frac{z}{x_i x_j} \mid \neg E^{G'}(i,j)\right]\left[\frac{y}{x_i^n} \mid i, n \in \omega\right].$$

Note that, since G is irreflexive, $\frac{z}{x_i^2}$ is included as a generator for each $i \in \omega$.

It is easy to see that $A_{G'}$ is computably presentable. In fact, if we fix a computable presentation P of the ring $\mathbb{F}(x_i \mid i \in \omega)[y, z]$ then $A_{G'}$ has an obvious presentation induced from that of P. (Just take as the domain of this presentation a computable copy of the set of all elements of P that can be generated from the generators of $A_{G'}$.) In what follows, we will identify $A_{G'}$ with this presentation. We will also assume that we have chosen P so that the map $g_{G'} : ax_i \mapsto i$, $a \in I$, is computable.

It is easy to check that if G' is a computable presentation of G then $A_{G'} \cong A_G$.

Let y and z be as in the definition of A_G. Let

$$D(x) = \{x \mid x \notin I \wedge \exists r(x^2 r = z)\},$$

$$Q(x, x') = \{(x, ax) \mid D(x) \wedge a \in I\},$$

and

$$R(x, x') = \{(x, x') \mid D(x) \wedge D(x') \wedge \neg Q(x, x') \wedge \exists r(rxx' = y)\}.$$

Since A_G is a subring of $\mathbb{F}(x_i \mid i \in \omega)[y, z]$, it makes sense to talk of the degree in y or z of an element r of A_G. We will denote these by $\deg_y(r)$ and $\deg_z(r)$, respectively. Let

$$Gen = \{\pm 1\} \cup \{x_i \mid i \in \omega\} \cup \left\{ \frac{y}{x_i x_j} \mid E(i,j) \right\} \cup \left\{ \frac{z}{x_i x_j} \mid \neg E(i,j) \right\} \cup \left\{ \frac{y}{x_i^n} \mid i, n \in \omega \right\}.$$

It will be useful to think of elements of A_G as sums of products of elements of Gen. (Of course, such a representation is not unique, but this will not matter for our purposes.)

Whenever we mention another ring B, such as $\mathbb{Z}_p[x_i, \frac{1}{x_i} \mid i \in \omega][y, z]$ or $\mathbb{Z}_p[x_i \mid i \in \omega]$, for example, we will think of A_G as a subring of B or of B as a subring of A_G, as appropriate. The relationships between such rings should be clear. For instance, if $\deg_y(r) = \deg_z(r) = 0$ then r can be expressed as a sum of products of the generators x_i, $i \in \omega$, so that r is in the subring $\mathbb{Z}_p[x_i \mid i \in \omega]$ of A_G. In this case, it makes sense to talk of the degree in x_i of r, denoted by $\deg_{x_i}(r)$, for any $i \in \omega$. We will make frequent use of these and similar facts.

Let $M = \mathbb{Z}_p[x_i, \frac{1}{x_i} \mid i \in \omega][y, z]$.

5.4.1. Lemma. *The only invertible elements of A_G are the elements of I.*

Proof. If $rs = 1$ then $\deg_y(r) = \deg_z(r) = 0$, and hence $r \in \mathbb{Z}_p[x_i \mid i \in \omega]$. Clearly, the only invertible elements of $\mathbb{Z}_p[x_i \mid i \in \omega]$ are the invertible elements of $\mathbb{Z}_p$. $\square$

5.4.2. Lemma. *Let $r, s \in A_G$. Suppose that $r^2 s = z$ and $r \notin I$. Then $r = a x_i$ for some $i \in \omega$ and $a \in I$.*

Proof. Clearly, $\deg_y(r) = \deg_z(r) = 0$. Since $r \notin I$, it must be the case that $r = x_i r_0 + r_1$ for some $i \in \omega$, $r_0 \in \mathbb{Z}_p[x_k \mid k \in \omega]$, $r_0 \neq 0$, and $r_1 \in \mathbb{Z}_p[x_k \mid k \neq i]$.

Now, $\deg_y(s) = 0$ and $\deg_z(s) = 1$, so that, working in M, we can write $s = \frac{z}{x_i^2}s_0 + \frac{z}{x_i}s_1 + s_2$, where $s_0 \in \mathbb{Z}_p[x_j \mid j \neq i]$, $s_1 \in \mathbb{Z}_p[x_j, \frac{1}{x_j} \mid j \neq i]$, and $s_2 \in \mathbb{Z}_p[x_j \mid j \in \omega][\frac{1}{x_j} \mid j \neq i][z]$.

Suppose that $r_1 \neq 0$. It is easy to check that

$$x_i^2 z = x_i^2 r^2 s = zr_1^2 s_0 + x_i(2zr_0r_1s_0 + zr_1^2 s_1) + x_i^2 t$$

for some $t \in \mathbb{Z}_p[x_j \mid j \in \omega][\frac{1}{x_j} \mid j \neq i][z]$, and hence that $zr_1^2 s_0 = x_i u$ for some $u \in \mathbb{Z}_p[x_j \mid j \in \omega][\frac{1}{x_j} \mid j \neq i][z]$. Since $\deg_{x_i}(zr_1^2 s_0) = 0$, it must be the case that $s_0 = 0$. Now $(zr_1^2 s_1)x_i = x_i^2(z - t)$. Since $\deg_{x_i}(zr_1^2 s_1) = 0$, it follows from this that $s_1 = 0$. But then $s_2 \neq 0$ and

$$x_i^2 r_0^2 s_2 = (x_i r_0 + r_1)^2 s_2 - (2x_i r_0 r_1 + r_1^2)s_2 = z - (2x_i r_0 r_1 + r_1^2)s_2.$$

Since now

$$\deg_{x_i}(x_i^2 r_0^2 s_2) = 2\deg_{x_i}(r_0) + \deg_{x_i}(s_2) + 2 > \deg_{x_i}(r_0) + \deg_{x_i}(s_2) + 1 \geqslant \deg_{x_i}(z - (2x_i r_0 r_1 + r_1^2)s_2),$$

this is a contradiction. So in fact $r_1 = 0$, and hence $r = x_i r_0$. We need to show that $r_0 \in I$.

We have

$$x_i^2 r_0^2 s_2 = x_i^2 r_0^2 s - (r_0^2 s_0 z + x_i r_0^2 s_1 z) = z - (r_0^2 s_0 z + x_i r_0^2 s_1 z).$$

Since $s_2 \neq 0$ implies that

$$\deg_{x_i}(x_i^2 r_0^2 s_2) = 2\deg_{x_i}(r_0) + \deg_{x_i}(s_2) + 2 > 2\deg_{x_i}(r_0) + 1 \geqslant \deg_{x_i}(z - (r_0^2 s_0 z + x_i r_0^2 s_1 z)),$$

it must be the case that $s_2 = 0$. Now $x_i r_0^2 s_1 z = z - r_0^2 s_0 z$. Since $s_1 \neq 0$ implies that

$$\deg_{x_i}(x_i r_0^2 s_1 z) = 2\deg_{x_i}(r_0) + 1 > 2\deg_{x_i}(r_0) \geqslant \deg_{x_i}(z - r_0^2 s_0 z),$$

it must be the case that $s_1 = 0$.

So $z = x_i^2 r_0^2 \frac{z}{x_i^2} s_0 = r_0^2 s_0 z$, and hence $r_0 \in I$. □

5.4.3. Corollary. *For any computable presentation G' of G, $D(A_{G'}) = \{ax_i \mid i \in \omega,\ a \in I\}$. Furthermore, D is intrinsically computable.*

Proof. The first statement follows immediately from Lemma 5.4.2; we prove the second.

Let A be a computable presentation of A_G. We want to show that $D(A)$ is computable. Abusing notation, we refer to the images of y and z in A as y and z, respectively. Let $\widehat{D}(A)$ be as in Section 5.2. Since I is finite and $x \in D(A) \Leftrightarrow \exists a \in I(ax \in \widehat{D}(A))$, it is enough to show that $\widehat{D}(A)$ is computable.

Clearly, $\widehat{D}(A)$ is c.e., and hence so is the set

$$Gen_A = \widehat{D}(A) \cup \left\{ r \in A \mid \exists x, x' \in \widehat{D}(A), n \in \omega (xx'r = y \vee xx'r = z \vee x^n r = y) \right\}.$$

Given $x \in |A|$, we can write x as a sum of products of elements of Gen_A and hence computably determine $\deg_y(x)$ and $\deg_z(x)$. If it is not the case that $\deg_y(x) = \deg_z(x) = 0$ then $x \notin \widehat{D}(A)$. Otherwise, x is a polynomial over the elements of $\widehat{D}(A)$ with coefficients in $\mathbb{Z}_p$, and checking whether a polynomial over a linearly independent c.e. set is an element of that set can be done computably. □

5.4.4. Lemma. *If $i \neq j$ and $\neg E(i,j)$ then there is no $r \in A_G$ such that $rx_i x_j = y$. Similarly, if $E(i,j)$ then there is no $r \in A_G$ such that $rx_i x_j = z$.*

Proof. The proofs of both statements are similar; we prove the first.

Assume for a contradiction that, for some $i \neq j \in \omega$ and $r \in A_G$, $\neg E(i,j)$ and $x_i x_j r = y$. We work in the ring M. Since $\deg_y(r) = 1$ and $\deg_z(r) = 0$, thinking of r as a sum of products of elements of Gen, we see that we can write $r = \frac{y}{x_i} r_0 + \frac{y}{x_j} r_1 + r_2$, where $r_0 \in \mathbb{Z}_p[x_k \mid k \neq i][\frac{1}{x_k} \mid k \neq j]$, $r_1 \in \mathbb{Z}_p[x_k \mid k \neq j][\frac{1}{x_k} \mid k \neq i]$, and $r_2 \in \mathbb{Z}_p[x_k \mid k \in \omega][\frac{1}{x_k} \mid k \neq i,j][y]$.

Let $n \in \omega$ be such that $x_i^n r_0, x_j^n r_2 \in \mathbb{Z}_p[x_k \mid k \in \omega][\frac{1}{x_k} \mid k \neq i,j]$. Then

$$(x_i x_j)^{n+1} r_2 = (x_i x_j)^n y - (x_i^n x_j^{n+1} r_0 y + x_i^{n+1} x_j^n r_1 y).$$

Since $\deg_{x_i}(x_i^n x_j^{n+1} r_0 y)$, $\deg_{x_j}(x_i^{n+1} x_j^n r_1 y)$, and $\deg_{x_i}((x_i x_j)^n y)$ are all less than or equal to n and $r_2 \in \mathbb{Z}_p[x_k \mid k \in \omega][\frac{1}{x_k} \mid k \neq i,j][y]$, it must be the case that $r_2 = 0$.

Now

$$(x_i x_j)^n y = x_i^n x_j^{n+1} r_0 y + x_i^{n+1} x_j^n r_1 y.$$

But

$$r_0 \neq 0 \Rightarrow \deg_{x_i}(x_i^n x_j^{n+1} r_0 y) \leqslant n \wedge \deg_{x_j}(x_i^n x_j^{n+1} r_0 y) > n$$

and

$$r_1 \neq 0 \Rightarrow \deg_{x_i}(x_i^{n+1} x_j^n r_1 y) > n \wedge \deg_{x_j}(x_i^{n+1} x_j^n r_1 y) \leqslant n.$$

Since it cannot be the case that $r_0 = r_1 = 0$, this means that at least one of $\deg_{x_i}(x_i^n x_j^{n+1} r_0 y + x_i^{n+1} x_j^n r_1 y)$ and $\deg_{x_j}(x_i^n x_j^{n+1} r_0 y + x_i^{n+1} x_j^n r_1 y)$ is greater than n. However, $\deg_{x_i}((x_i x_j)^n y) = \deg_{x_j}((x_i x_j)^n y) = n$, so this is a contradiction. □

5.4.5. Corollary. *$R(A_G) = \{(x,x') \mid D(x) \wedge D(x') \wedge \neg Q(x,x') \wedge \exists r(rxx' = y)\} = \{(x,x') \mid D(x) \wedge D(x') \wedge \neg\exists r(rxx' = z)\}$, and hence R is intrinsically computable.*

Furthermore, for any computable presentation G' of G, $R(A_{G'}) = \{(ax_i, bx_j) \mid E^{G'}(i,j) \wedge a, b \in I\}$.

We now need to show that D, Q, and R are invariant. Fix an automorphism $f : A_G \cong A_G$. We will show that $f(D) = D$, $f(Q) = Q$, and $f(R) = R$.

5.4.6. Lemma. *Suppose that $i \in \omega$ and $f(x_i) = rs$ for some $r, s \in A_G$. Then either $r \in I$ or $s \in I$.*

Proof. Since $f(I) = I$ and $x_i = f^{-1}(r)f^{-1}(s)$, it is enough to show that if $x_i = r's'$ for some $r', s' \in A_G$ then either $r' \in I$ or $s' \in I$. But this follows easily from the fact that if $x_i = r's'$ then $\deg_y(r') = \deg_z(r') = \deg_y(s') = \deg_z(s') = 0$, so that $r', s' \in \mathbb{Z}_p[x_j : j \in \omega]$. □

5.4.7. Lemma. *$f(D) = D$, which implies that $f(Q) = Q$.*

Proof. It is enough to show that $f(D) \subseteq D$. Since f is an arbitrary automorphism of A_G, the same proof will show that $f^{-1}(D) \subseteq D$, and hence that $D \subseteq f(D)$.

Let $i \in \omega$. Let $n = \deg_y(f(y))$ and let $r = f(\frac{y}{x_i^{n+1}})$. Then $f(x_i)^{n+1}r = f(y)$, and hence $n = \deg_y(f(y)) \geqslant \deg_y(f(x_i)^{n+1}) = (n+1)\deg_y(f(x_i))$. Thus it must be the case that $\deg_y(f(x_i)) = 0$. A similar argument shows that $\deg_z(f(x_i)) = 0$. Since $f(x_i) \notin I$, this means that $f(x_i) = x_j s_0 + s_1$ for some $j \in \omega$, $s_0 \in \mathbb{Z}_p[x_l : l \in \omega]$, $s_0 \neq 0$, and $s_1 \in \mathbb{Z}_p[x_l : l \neq j]$.

Let k be such that $x_j^k f(y) \in \mathbb{Z}_p[x_l \mid l \in \omega][\frac{1}{x_l} \mid l \neq j][y, z]$ and let $n = \deg_{x_j}(x_j^k f(y)) + 1$. For some $r \in A_G$, $x_j^k f(x_i)^n r = x_j^k f(y)$. Working in M, we can write

$$r = \frac{1}{x_j^{k+1}} r_0 + \frac{1}{x_j^k} r_1 + \cdots + r_{k+1},$$

where $r_0 \in \mathbb{Z}_p[x_l \mid l \neq j][\frac{1}{x_l} \mid l \in \omega][y,z]$, $r_1, \ldots, r_k \in \mathbb{Z}_p[x_l, \frac{1}{x_l} \mid l \neq j][y,z]$, and $r_{k+1} \in \mathbb{Z}_p[x_l \mid l \in \omega][\frac{1}{x_l} \mid l \neq j][y,z]$.

Now

$$x_j^k(x_js_0+s_1)^n r_{k+1} = x_j^k(x_js_0+s_1)^n r - x_j^k(x_js_0+s_1)^n(r-r_{k+1}) = \\ x_j^k f(y) - \left(x_j^k(x_js_0+s_1)^n \left(\frac{1}{x_j^{k+1}} r_0 + \frac{1}{x_j^k} r_1 + \cdots + \frac{1}{x_j} r_k \right)\right).$$

But it is easy to check that if $r_{k+1} \neq 0$ then

$$\deg_{x_j}\left(x_j^k(x_js_0+s_1)^n r_{k+1}\right) = \\ n\deg_{x_j}(s_0) + \deg_{x_j}(r_{k+1}) + k + n > n\deg_{x_j}(s_0) + k + n - 1 \geqslant \\ \deg_{x_j}\left(x_j^k f(y) - \left(x_j^k(x_js_0+s_1)^n \left(\frac{1}{x_j^{k+1}} r_0 + \frac{1}{x_j^k} r_1 + \cdots + \frac{1}{x_j} r_k \right)\right)\right).$$

It follows that $r_{k+1} = 0$.

It is not hard to see that we can now repeat the above argument with k in place of $k+1$ (assuming $k > 0$). Proceeding in this fashion, we see that $r_1 = \cdots = r_{k+1} = 0$.

So

$$\frac{s_1^n r_0}{x_j} = x_j^k(x_js_0+s_1)^n \frac{1}{x_j^{k+1}} r_0 - x_j^k((x_js_0+s_1)^n - s_1^n)\frac{1}{x_j^{k+1}} r_0 = \\ x_j^k f(y) - ((x_js_0+s_1)^n - s_1^n)\frac{1}{x_j} r_0.$$

But $s_1^n r_0 \in \mathbb{Z}_p[x_l \mid l \neq j][\frac{1}{x_l} \mid l \in \omega][y,z]$, which implies that either $s_1^n r_0 = 0$ or $\frac{s_1^n r_0}{x_j} \notin \mathbb{Z}_p[x_l \mid l \in \omega][\frac{1}{x_l} \mid l \neq j][y,z]$. Since

$$x_j^k f(y) - ((x_js_0+s_1)^n - s_1^n)\frac{1}{x_j} r_0 \in \mathbb{Z}_p[x_l \mid l \in \omega]\left[\frac{1}{x_l} \mid l \neq j\right][y,z],$$

it must be the case that $s_1^n r_0 = 0$. Since $r \neq 0$, we conclude that $s_1 = 0$.

Thus $f(x_i) = s_0 x_j$. By Lemma 5.4.6, $s_0 \in I$. $\square$

5.4.8. Corollary. $f(\mathbb{Z}_p[x_i \mid i \in \omega]) = \mathbb{Z}_p[x_i \mid i \in \omega]$.

5.4.9. Lemma. *Let $r \in A_G$ be such that $r \neq 0$, $\deg_y(r) = 0$, and $\deg_z(r) \leqslant n$. Then, for all $i \in \omega$, $x_i^{2n+1} r \notin \mathbb{Z}_p[x_j, \frac{1}{x_j} \mid j \neq i][y, z]$.*

Proof. We work in the ring M. Let $i \in \omega$. Thinking of r as a sum of products of elements of Gen, each term t in this sum can be written as $\frac{z^m}{x_i^{2m}} s$, where $m \leqslant n$ and $s \in \mathbb{Z}_p[x_j \mid j \in \omega][\frac{1}{x_j} \mid j \neq i]$. So $x_i^{2n+1} t = x_i u$ for some $u \in \mathbb{Z}_p[x_j \mid j \in \omega][\frac{1}{x_j} \mid j \neq i][z]$. Thus $x_i^{2n+1} r = x_i v$ for some $v \in \mathbb{Z}_p[x_j \mid j \in \omega][\frac{1}{x_j} \mid j \neq i][z]$, and hence $x_i^{2n+1} r \notin \mathbb{Z}_p[x_j, \frac{1}{x_j} \mid j \neq i][y, z]$. $\square$

5.4.10. Lemma. $\deg_y(f(y)) = 1$ *and* $\deg_z(f(y)) = 0$.

Proof. Let $i \in \omega$ be such that $f(y) \in \mathbb{Z}_p[x_j, \frac{1}{x_j} \mid j \neq i][y, z]$. Working in M, we can write $f(y) = ys_0 + s_1$, where $s_0 \in M$, $s_1 \in A_G$, and $\deg_y(s_1) = 0$. Let $n = \deg_z(s_1)$.

By Lemma 5.4.7, there exists an $r \in A_G$ such that $x_i^{2n+1} r = f(y) = ys_0 + s_1$. We can write $r = yr_0 + r_1$, where $r_0 \in M$, $r_1 \in A_G$, and $\deg_y(r_1) = 0$. Now $x_i^{2n+1} r_1 = s_1$. Since $\deg_z(r_1) = \deg_z(s_1) = n$, it follows from Lemma 5.4.9 that either $r_1 = 0$ or $s_1 \notin \mathbb{Z}_p[x_j, \frac{1}{x_j} \mid j \neq i][y, z]$. But the latter possibility would imply that $f(y) \notin \mathbb{Z}_p[x_j, \frac{1}{x_j} \mid j \neq i][y, z]$, contradicting our choice of i. So $r_1 = 0$, and hence $s_1 = 0$.

We now have $f(y) = ys_0$. A similar argument shows that $\deg_y(f^{-1}(y)) \geqslant 1$. We now need to show that $\deg_y(s_0) = \deg_z(s_0) = 0$.

Let $t \in \mathbb{Z}_p[x_j \mid j \in \omega]$ be such that $ts_0 \in \mathbb{Z}_p[x_j \mid j \in \omega][y, z]$. Then

$$f^{-1}(t)y = f^{-1}(tf(y)) = f^{-1}(ts_0 y) = f^{-1}(ts_0) f^{-1}(y).$$

By Corollary 5.4.8, $f^{-1}(t) \in \mathbb{Z}_p[x_j \mid j \in \omega]$, which means that $\deg_y(f^{-1}(t)y) = 1$ and $\deg_z(f^{-1}(t)y) = 0$. Since $\deg_y(f^{-1}(y)) \geqslant 1$, this means that $f^{-1}(ts_0) \in \mathbb{Z}_p[x_j \mid j \in \omega]$. By Corollary 5.4.8, $ts_0 \in \mathbb{Z}_p[x_j \mid j \in \omega]$. So $\deg_y(s_0) = \deg_z(s_0) = 0$. $\square$

5.4.11. Lemma. *$f(y) = ty$ for some $t \in A_G$.*

Proof. Let $i, j, i', j' \in \omega$ be such that $i \neq j$, $f(x_{i'}) = ax_i$ and $f(x_{j'}) = bx_j$ for some $a, b \in I$, $f(y) \in \mathbb{Z}_p[x_k, \frac{1}{x_k} \mid k \neq i, j][y, z]$, and $E(i', j')$. Such numbers exist by Lemma 5.4.7 and the assumption about G that we made at the beginning of this section.

Let $r = f(\frac{aby}{x_i' x_j'})$. Then $x_i x_j r = f(y)$. By Lemma 5.4.10, $\deg_y(f(y)) = 1$ and $\deg_z(f(y)) = 0$, and hence $\deg_y(r) = 1$ and $\deg_z(r) = 0$. Working in M and thinking of r as a sum of products of elements of Gen, we see that we can write

$$r = yr_0 + \frac{y}{x_i}r_1 + \frac{y}{x_j}r_2 + \frac{y}{x_i x_j}r_3 + r_4,$$

where $r_0 \in \mathbb{Z}_p[x_k \mid k \in \omega][\frac{1}{x_k} \mid k \neq i, j]$, $r_1 \in \mathbb{Z}_p[x_k \mid k \neq i][\frac{1}{x_k} \mid k \neq j]$, $r_2 \in \mathbb{Z}_p[x_k \mid k \neq j][\frac{1}{x_k} \mid k \neq i]$, and $r_3, r_4 \in \mathbb{Z}_p[x_k \mid k \neq i, j]$.

Let $n \in \omega$ be such that $x_i^n r_1, x_j^n r_2 \in \mathbb{Z}_p[x_k \mid k \in \omega][\frac{1}{x_k} \mid k \neq i, j]$. Then

$$(x_i x_j)^{n+1} r_0 y + (x_i x_j)^{n+1} r_4 = (x_i x_j)^n f(y) - (x_i^n x_j^{n+1} r_1 y + x_i^{n+1} x_j^n r_2 y + (x_i x_j)^n r_3 y).$$

But $\deg_{x_i}(x_i^n x_j^{n+1} r_1 y)$, $\deg_{x_j}(x_i^{n+1} x_j^n r_2 y)$, $\deg_{x_i}((x_i x_j)^n r_3 y)$, and $\deg_{x_i}((x_i x_j)^n f(y))$ are all at most n, $r_0, r_4 \in \mathbb{Z}_p[x_k \mid k \in \omega][\frac{1}{x_k} \mid k \neq i, j]$, and $\deg_y((x_i x_j)^{n+1} r_4) = 0$. So it must be the case that $r_0 = r_4 = 0$.

Now

$$(x_i x_j)^n f(y) - (x_i x_j)^n r_3 y = x_i^n x_j^{n+1} r_1 y + x_i^{n+1} x_j^n r_2 y.$$

But

$$r_1 \neq 0 \Rightarrow \deg_{x_i}(x_i^n x_j^{n+1} r_1 y) \leqslant n \wedge \deg_{x_j}(x_i^n x_j^{n+1} r_1 y) > n$$

and

$$r_2 \neq 0 \Rightarrow \deg_{x_i}(x_i^{n+1} x_j^n r_2 y) > n \wedge \deg_{x_j}(x_i^{n+1} x_j^n r_2 y) \leqslant n,$$

which means that either $r_1 = r_2 = 0$ or at least one of $\deg_{x_i}(x_i^n x_j^{n+1} r_1 y + x_i^{n+1} x_j^n r_2 y)$ and $\deg_{x_j}(x_i^n x_j^{n+1} r_1 y + x_i^{n+1} x_j^n r_2 y)$ is greater than n. Since

$$\deg_{x_i}((x_i x_j)^n f(y) - (x_i x_j)^n r_3 y), \deg_{x_j}((x_i x_j)^n f(y) - (x_i x_j)^n r_3 y) \leqslant n,$$

it must be the case that $r_1 = r_2 = 0$. Thus $f(y) = x_i x_j \frac{y}{x_i x_j} r_3 = y r_3$. Since $r_3 \in A_G$, we are done. $\square$

5.4.12. Corollary. *If* $\exists r(x_i x_j r = y)$ *then* $\exists r(x_i x_j r = f(y))$.

5.4.13. Lemma. $f(R) = R$.

Proof. It is enough to show that $R \subseteq f(R)$. Since f is an arbitrary automorphism of A_G, the same proof will show that $R \subseteq f^{-1}(R)$, and hence that $f(R) \subseteq R$.

By Corollaries 5.4.5 and 5.4.12, $R(x_i, x_j) \Rightarrow \exists r(x_i x_j r = y) \Rightarrow \exists r(x_i x_j r = f(y)) \Rightarrow R(f(x_i), f(x_j))$. $\square$

Since f is an arbitrary automorphism of A_G, Lemmas 5.4.7 and 5.4.13 imply the following result.

5.4.14. Lemma. *D, Q, and R are invariant.*

In order to apply Theorem 5.2.1, we are left with showing that properties (P2) and (P3) in the statement of that theorem are satisfied.

5.4.15. Lemma. *For every pair S, S' of sets of Q-representatives, if $f : S \xrightarrow[onto]{1\text{-}1} S'$ is such that, for every $x, y \in S$, $R(x,y) \Leftrightarrow R(f(x), f(y))$, then f can be extended to an automorphism of A_G.*

Proof. Let y, z, and x_i be as in the definition of A_G. A set of Q-representatives contains one element of the form ax_i, $a \in I$, for each $i \in \omega$, and it contains no other elements. So there exist sequences $a_0, a_1, \ldots \in I$ and $b_0, b_1, \ldots \in I$ such that $S = \{a_0x_0, a_1x_1, \ldots\}$ and $S' = \{b_0x_0, b_1x_1, \ldots\}$. Thus, for some permutation π of ω, $f : a_ix_i \mapsto b_{\pi(i)}x_{\pi(i)}$.

Now, for all $i, j \in \omega$, $R(x_i, x_j) \Leftrightarrow R(x_{\pi(i)}, x_{\pi(i)})$. So it is clear from what we have previously done that the map $x_i \mapsto x_{\pi(i)}$ can be extended to an automorphism of A_G. Thus it is enough to show that the map $a_ix_i \mapsto b_{\pi(i)}x_i$, or, equivalently, the map $h : x_i \mapsto \frac{b_{\pi(i)}}{a_i}x_i$ can be extended to an automorphism of A_G. But h can clearly be extended to an automorphism of $\mathbb{F}(x_i \mid i \in \omega)[y,z]$ that fixes y and z. Since $\frac{b_{\pi(i)}}{a_i} \in I$, this automorphism restricts to an automorphism of A_G. $\square$

5.4.16. Lemma. *For every computable presentation G' of G and every computable set S of $Q^{A_{G'}}$-representatives, there exists a defining family for $\langle A_{G'}, a\rangle_{a\in S}$.*

Proof. Let y, z, and x_i be as in the definition of $A_{G'}$. As mentioned above, $S = \{a_0x_0, a_1x_1, \ldots\}$ for some sequence $a_0, a_1, \ldots \in I$. Let $s_i = a_ix_i$ and consider the sets

$$Gen' = \{\pm 1\} \cup \{s_i \mid i \in \omega\} \cup \left\{\frac{y}{s_is_j} \mid E^{G'}(i,j)\right\} \cup \left\{\frac{z}{s_is_j} \mid \neg E^{G'}(i,j)\right\} \cup \left\{\frac{y}{s_i^n} \mid i, n \in \omega\right\}$$

and

$$Gen'_k = \{\pm 1\} \cup \{s_i \mid i \leqslant k\} \cup \left\{ \frac{y}{s_i s_j} \mid E^{G'}(i,j),\ i,j \leqslant k \right\} \cup \\ \left\{ \frac{z}{s_i s_j} \mid \neg E^{G'}(i,j),\ i,j \leqslant k \right\} \cup \left\{ \frac{y}{s_i^n} \mid i,n \leqslant k \right\}.$$

For each $i,j,n \in \omega$, let the formula $\varphi_{i,j,n}$ over the language of rings with additional constants $y, z, s_0, s_1, \ldots$ be defined by

$$\varphi_{i,j,n} = \begin{cases} s_i s_j u_{i,j} = y \wedge s_i^n v_{i,n} = y & \text{if } E^{G'}(i,j), \\ s_i s_j u_{i,j} = z \wedge s_i^n v_{i,n} = y & \text{if } \neg E^{G'}(i,j). \end{cases}$$

(Here $u_{i,j}$ and $v_{i,n}$ are the free variables of $\varphi_{i,j,n}$.) For each sum t of products of elements of Gen', let t' be the result of substituting all occurrences of $\frac{y}{s_i s_j}$ or $\frac{z}{s_i s_j}$ in t by $u_{i,j}$, and all occurrences of $\frac{y}{s_i^n}$ by $v_{i,n}$. If k is the least number such that t is a sum of products of elements of Gen'_k then let $\hat{t}$ be the formula

$$\exists u_{0,0}, v_{0,0}, \ldots, u_{0,k}, v_{0,k}, \ldots, u_{k,0}, v_{k,0}, \ldots, u_{k,k}, v_{k,k} \left(t' \wedge \bigwedge_{i,j,n \leqslant k} \varphi_{i,j,n} \right).$$

Let $t_0, t_1, \ldots$ be an effective list of all sums of products of elements of Gen'. Since each s_i is a product of x_i with an element of I, each element of $A_{G'}$ is equal to t_i for some $i \in \omega$. It follows easily that $\{\hat{t}_i \mid i \in \omega\}$ is a defining family for $\langle A_{G'}, a \rangle_{a \in S}$. □

Lemmas 5.4.3, 5.4.14, 5.4.15, and 5.4.16 and Corollary 5.4.5 are enough to enable us to apply Theorem 5.2.1. It is straightforward to check that, for any computable presentation A of A_G, if U is a subset of $D(A)$ such that $Q(x,y) \Rightarrow (U(x) \Leftrightarrow U(y))$ then the subring of A generated by U has the same degree as U. Thus it follows that Theorems 3.1.2 and 4.1.2 and the results of Chapters 3 and 4 remain true if we require that $\mathcal{A}$ be an integral domain of characteristic p and that U be a subring of $\mathcal{A}$.

Now consider the commutative semigroup generated (multiplicatively) by the elements of Gen. Let

$$D(x) = \{x \mid \exists r(x^2 r = z)\},$$

$$Q(x, x') = \{(x, x) \mid D(x)\},$$

and

$$R(x, x') = \{(x, x') \mid D(x) \wedge D(x') \wedge x \neq x' \wedge \exists r(rxx' = y)\}.$$

It is not hard to check that Theorem 5.2.1 can be applied in this case, with essentially the same proof as above, to show that Theorems 3.1.2 and 4.1.2 and the results of Chapters 3 and 4 remain true if we require that $\mathcal{A}$ be a commutative semigroup and that U be a subsemigroup of $\mathcal{A}$.

Chapter 6

Relations on Algebraic Structures II: Negative Results

6.1 Introduction

As we remarked in Section 5.1, results such as Theorem 3.1.4 cannot be extended to any of the classes of structures mentioned in Theorem 5.1.1. However, since it is certainly possible for a relation on a computable structure of infinite computable dimension to have a degree spectrum of finite cardinality, this does not rule out the possibility that, for one or more of these classes, results such as Theorem 3.1.3 still hold if we also require that the structures mentioned in these results are in the given class. In this chapter, we give conditions that guarantee that the degree spectrum of a relation on a computable structure is either a singleton or infinite.

Goncharov [13] has shown that if two computable structures are Δ^0_2-isomorphic but not computably isomorphic then their computable dimension is ω. This theorem is quite useful in establishing results such as those in Theorem 3.1.3, since it reduces the task of building infinitely many noncomputably isomorphic computable presentations of a computable structure to that of building a single computable presentation that is Δ_2^0-isomorphic but not computably isomorphic to the original structure. In Section 6.2, we give an analog of this result in the case of degree spectra of relations and examine some of its consequences. In Section 6.3, we deal with linear orderings, and in Section 6.4, with 1-decidable structures.

It should be pointed out that, in the general case, there are no known restrictions on the sets of degrees that can be realized as degree spectra of relations on computable structures other than the ones that follow from the fact that the set of images of a relation on a computable structure in different computable presentations of the structure is (by definition) Σ_1^1.

6.2 General Criteria

We begin this section by proving a technical theorem that will have as an immediate corollary the analog of Goncharov's result mentioned above.

Here and below, we will need some notation to talk about "finite portions" of a computable structure $\mathcal{A}$ of (possibly infinite) signature L. Let $S \subset \omega$ be finite. Define L_S to be the language obtained by restricting L to its first $|S|$ symbols, substituting all j-ary function symbols by $(j+1)$-ary relation symbols in the obvious way, and dropping any constants whose interpretation in $\mathcal{A}$ is not in S. Define $\mathcal{A} \upharpoonright S$

to be the finite structure obtained from $\mathcal{A}$ by restricting the domain to $|\mathcal{A}| \cap S$ and restricting the language to L_S.

For $\vec{x} = (x_0, \ldots, x_{k-1}) \in \omega^k$, let $\max(\vec{x}) = \max\{x_i \mid i < k\}$.

6.2.1. Theorem. *Let $k \in \omega$. Let U^0 and U^1 be k-ary relations on the domains of computable structures $\mathcal{A}^0$ and $\mathcal{A}^1$, respectively, and let $B_0, \ldots, B_{n-1} \subset \omega^k$ be Δ^0_2 but not computable. Suppose that U^0 is not computable, U^1 is computable, and there exists a Δ^0_2 isomorphism $f : \mathcal{A}^0 \to \mathcal{A}^1$ such that $f(U^0) = U^1$. Then there exists a Δ^0_2 function $h : |\mathcal{A}^0| \to \omega$ such that $h(\mathcal{A}^0)$ is a computable structure, $h(U^0)$ is not computable, and for all $m < n$, $B_m \not\leq_T h(U^0)$.*

Proof. Let k be the arity of U^0 and U^1. Let Φ_e be the e^{th} k-ary Turing functional. We will build $h : |\mathcal{A}^0| \xrightarrow[\text{onto}]{1-1} \omega$ to satisfy the requirements

$$Q_e : \Phi_e \neq h(U^0)$$

and

$$R_{ni+j} : \Phi_i^{h(U^0)} \neq B_j$$

for each $e, i \in \omega$, $j < n$, while in addition guaranteeing that $h(\mathcal{A}^0)$ is a computable structure.

Since f is Δ^0_2, there exist sequences $S^i_0, S^i_1, \ldots$, $i = 0, 1$, of subsets of ω and a computable sequence $f_0, f_1, \ldots$ of maps so that, for each $s \in \omega$, $f_s : \mathcal{A}^0 \upharpoonright S^0_s \cong \mathcal{A}^1 \upharpoonright S^1_s$, $S^i_0 \subset S^i_1 \subset \cdots$, $\bigcup_{s \in \omega} S^i_s = |\mathcal{A}^i|$, and, for each $x \in |\mathcal{A}^0|$, $f(x) = \lim_s f_s(x)$. For each $s \in \omega$, we will denote $f_s^{-1}(U^1 \cap (S^1_s)^k)$ by $U^0[s]$.

Our construction will be similar to the standard finite injury argument that would be used to satisfy the above requirements with a Δ^0_2 set A in place of $h(U^0)$.

Of course, when building a Δ^0_2 set, we can decide at any stage whether we want the value of A at some given element to remain the same or change. In our construction, the only thing we control is h.

At each stage $s+1$, we will define the approximation h_{s+1} of h to extend either h_s or $h_s \circ f_s^{-1} \circ f_{s+1}$. If h_{s+1} extends $h_s \circ f_s^{-1} \circ f_{s+1}$ then, for all $\vec{x}$ in the range of h_s, $U^0(h_{s+1}^{-1}(\vec{x}))[s+1] = U^0(h_s^{-1}(\vec{x}))[s]$, which means that $h(U^0)$ remains unaltered at this stage. On the other hand, if $U^0(h_s^{-1}(\vec{x}))[s+1] \neq U^0(h_s^{-1}(\vec{x}))[s]$ then we can change the value of $h(U^0)$ at $\vec{x}$ by letting h_{s+1} extend h_s. The fact that f is Δ^0_2 means that, for every $\vec{x} \in \omega^k$, there exists a stage s such that, for all $t \geqslant s$, $f_t^{-1} \circ f_{t+1}(\vec{x}) = \vec{x}$. This is what will allow us to ensure that $\lim_s h_s$ exists.

We now proceed with the construction. We will use the notation $\vec{x} < y$ to mean that $\vec{x} \in \omega^k$, $y \in \omega$, and $\max(\vec{x}) < y$. The notations $\vec{x} \leqslant y$ and $y \leqslant \vec{x}$ are defined analogously.

stage 0. Let $h_0 = \emptyset$.

stage $s+1$. For each $e < s+1$, let i and j be such that $ni + j = e$ and let

$$q_{e,s} = \begin{cases} \max(f_s^{-1}(\vec{z})) & Q_e \text{ is currently satisfied} \\ & \quad \text{through } \vec{z} \text{ (defined below)} \\ \max\{y \mid \forall \vec{z} < y(\Phi_e(\vec{z})[s]\downarrow)\} & \text{otherwise,} \end{cases}$$

$$l_{e,s} = \max\left\{y \mid \forall \vec{z} < y \left(\Phi_i^{h_s(U^0[s])}(\vec{z})[s]\downarrow = B_j(\vec{z})[s]\right)\right\},$$

$$m_{e,s} = \max\{l_{e,t} \mid t \leqslant s\},$$

and

$$r_{e,s} = \max\left\{\varphi_i^{h_s(U^0[s])}(\vec{z})[s] \mid \vec{z} \leqslant m_{e,s}\right\}.$$

In order to define h_{s+1}, we will begin by defining an auxiliary function g. Let $e < s+1$ be the least number such that $f_{s+1}(y) = f_s(y)$ for all $y \leqslant e$ and all y such that $h_s(y) \leqslant e$, and one of the following holds.

1. Q_e is not satisfied and, for some $\vec{x} \in \mathrm{dom}(h_s)$, $U^0(\vec{x})[s+1] \neq U^0(\vec{x})[s]$ and $\Phi_e(h_s(\vec{x}))[s]\downarrow$.

2. Not 1 and for some $y \leqslant q_{e,s}$, $f_{s+1}(y) \neq f_s(y)$.

3. Not 1 or 2, and for some y such that $h_s(y) \leqslant r_{e,s}$, $f_{s+1}(y) \neq f_s(y)$.

If no such number exists then let $g = h_s$.

If condition 1 holds then proceed as follows. If $\Phi_e(h_s(\vec{x})) = U^0(\vec{x})[s]$ then let $g = h_s$; otherwise, let $g = h_s \circ f_s^{-1} \circ f_{s+1}$. In either case, declare Q_e to be satisfied through $f_s(\vec{x})$. We say that Q_e is active at stage $s+1$.

If condition 2 holds then proceed as follows. If Q_e is not satisfied then let $g = h_s$; otherwise, let $g = h_s \circ f_s^{-1} \circ f_{s+1}$. In either case, we say that Q_e is active at stage $s+1$.

If condition 3 holds then let $g = h_s \circ f_s^{-1} \circ f_{s+1}$. We say that R_e is active at stage $s+1$.

If either Q_e or R_e is active at stage $s+1$ then declare each Q_i, $i > e$, to be unsatisfied.

Now define h_{s+1} as follows. For $y \in \mathrm{dom}(g)$, let $h_{s+1}(y) = g(y)$. Let $y_0 < \cdots < y_m$ be the elements of $S_{s+1} - \mathrm{dom}(g)$ and let $z_0 < \cdots < z_m$ be the $m+1$ least numbers not in $\mathrm{rng}(h_s)$. For $i \leqslant m$, let $h_{s+1}(y_i) = z_i$.

This completes the construction. We now need to show that $h = \lim_s h_s$ and $h^{-1} = \lim_s h_s^{-1}$ are well-defined, all requirements are met, and $h(\mathcal{A}^0)$ is a computable structure. We begin by showing by induction that h and h^{-1} are well-defined; each requirement is active only finitely often; for each $e \in \omega$, $q_{e,s}$ and $r_{e,s}$ have finite limits; and for each $e \in \omega$, if Φ_e is total then Q_e is eventually permanently satisfied.

For the following lemmas, fix $e \in \omega$ and assume by induction that, for all $i < e$, the requirements Q_i and R_i are active only finitely often and $\lim_s h_s^{-1}(i)$ is well-defined.

6.2.2. Lemma. *$h^{-1}(e) = \lim_s h_s^{-1}(e)$ is well-defined and (if $e \in |\mathcal{A}^0|$) so is $h(e) = \lim_s h_s(e)$.*

Proof. Let s be a stage such that no requirement Q_i or R_i, $i < e$, is active after stage s. By construction, for all $t \in \omega$, h_{t+1} extends either h_t or $h_t \circ f_t^{-1} \circ f_{t+1}$. One of the conditions for a requirement Q_j or R_j, $j \geqslant e$, to be active at stage $t+1$ is that $f_{t+1}(e) = f_t(e)$ and $f_{t+1}(h_t^{-1}(e)) = f_t(h_t^{-1}(e))$. So, for all $t \geqslant s$, if $f_{t+1}(e) \neq f_t(e)$ or $f_{t+1}(h_t^{-1}(e)) \neq f_t(h_t^{-1}(e))$ then no requirement is active at stage $t+1$, and hence h_{t+1} extends h_t. Thus $h_t(e) = h_s(e)$ and $h_t^{-1}(e) = h_s^{-1}(e)$ for all $t > s$. $\square$

Let $s_0 > e$ be such that no requirement Q_i or R_i, $i < e$, is active after stage s_0 and $f_t(y) = f_{s_0}(y)$ for all $t > s_0$ and all y such that either $y \leqslant e$ or $h_{s_0}^{-1}(i) = y$ for some $i \leqslant e$.

6.2.3. Lemma. *If Φ_e is total then Q_e is eventually permanently satisfied.*

Proof. It is enough to show that if Φ_e is total then Q_e is satisfied at some stage $t > s_0$.

Suppose otherwise. We claim that we can compute U^0, which contradicts the hypothesis the U^0 is not computable. Let $\vec{x} \in \omega^k$. Since Φ_e is total and Q_e is never satisfied after stage s_0, $\lim_t q_{e,t} = \infty$. Let $t > s_0$ be such that $\vec{x} < q_{e,t}$ and $\vec{x} \in (\mathrm{dom}(h_t))^k$. As mentioned above, for all $u \in \omega$, h_{u+1} extends either h_u or $h_u \circ f_u^{-1} \circ f_{u+1}$. Furthermore, for all $u \geqslant t$, if $f_{u+1}(\vec{x}) \neq f_u(\vec{x})$ then h_{u+1} extends h_u. So $h_u(\vec{x}) = h_t(\vec{x})$ for all $u > t$. Now let $u \geqslant t$ be such that $\Phi_e(h_t(\vec{x}))[u]\downarrow$. If $U^0(\vec{x})[v+1] \neq U^0(\vec{x})[v]$ for some $v \geqslant u$ then Q_e is satisfied at stage $v+1$. Therefore, $\vec{x} \in U^0 \Leftrightarrow \vec{x} \in U^0[u]$. □

6.2.4. Lemma. $\lim_s q_{e,s} < \infty$ *and* Q_e *is active only finitely often.*

Proof. If Q_e is satisfied through $\vec{z}$ after stage s_0 then $\lim_s q_{e,s} = \max(f^{-1}(\vec{z}))$. Otherwise, by the previous lemma, Φ_e is not total, and thus $\lim_s q_{e,s}$ is equal to the largest y such that for all $\vec{x} < y$, $\Phi_e(\vec{x})\downarrow$.

Now let $t > s_0$ be such that either Q_e is satisfied at stage t or Q_e is never satisfied after stage t and, for all $u > t$ and $y \leqslant \lim_s q_{e,s}$, $q_{e,u} = q_{e,t}$ and $f_u(y) = f_t(y)$. Then Q_e is not active after stage t. □

Let $s_1 \geqslant s_0$ be such that Q_e is not active after stage s_1. Let i and j be such that $ni + j = e$.

6.2.5. Lemma. $\lim_s m_{e,s} < \infty$.

Proof. Let $t > s_1$, $x \leqslant r_{e,t}$, and $z = f_t \circ h_t^{-1}(x)$. For any $u \geqslant t$, if $f_{u+1}(x) \neq f_u(x)$ then h_{u+1} extends $h_u \circ f_{u+1}^{-1} \circ f_u$. So for all $u \geqslant t$, $f_u \circ h_u^{-1}(x) = z$. Therefore, for all $u \geqslant t$, $h_u(U^0[u])(x) = U^0(h_u^{-1}(x))[u] = U^1(f_u \circ h_u^{-1}(x)) = U^1(f_t \circ h_t^{-1}(x)) = U^0(h_t^{-1}(x))[t] = h_t(U^0[t])(x)$.

Now assume for a contradiction that $\lim_s m_{e,s} = \infty$. Then for each $\vec{x} \in \omega^k$ there is a $t_{\vec{x}} \in \omega$ such that $\vec{x} < l_{e,t_{\vec{x}}}$. Since B_j is not computable and we can computably determine $t_{\vec{x}}$ from $\vec{x}$, there exists an $\vec{x} \in \omega^k$ such that $B_j(\vec{x}) \neq B_j(\vec{x})[t_{\vec{x}}]$. Let u be such that, for all $v > u$, $B_j(\vec{x})[v] = B_j(\vec{x})[u]$. Using the result of the previous paragraph, we conclude that, for all $v \geqslant u$, $\Phi_i^{h_v(U^0[v])}(\vec{x})[v]\downarrow = \Phi_i^{h_{t_{\vec{x}}}(U^0[t_{\vec{x}}])}(\vec{x})[t_{\vec{x}}]\downarrow = B_j(\vec{x})[t_{\vec{x}}] \neq B_j(\vec{x})[v]$, which implies that $l_{e,v} \leqslant \vec{x}$, contradicting the assumption that $\lim_s m_{e,s} = \infty$. □

6.2.6. Lemma. $\lim_s r_{e,s} < \infty$ *and* R_e *is active only finitely often.*

Proof. Let $t > s_1$ be such that, for all $u > t$, $m_{e,u} = m_{e,t}$. As shown above, $h_u(U^0[u])(\vec{y}) = h_t(U^0[t])(\vec{y})$ for all $\vec{y} \leqslant r_{e,t}$ and $u > t$. Thus, for all $u > t$ and all $\vec{x} \leqslant m_{e,t}$, $\varphi_i^{h_u(U^0[u])}(\vec{x})[u] = \varphi_i^{h_t(U^0[t])}(\vec{x})[t]$. So $r_e = \lim_s r_{e,s} < \infty$.

Now let $t > s_1$ be such that, for all $u > t$, $r_{e,u} = r_{e,t}$. For $m \leqslant r_e$, let $z_m = f_t \circ h_t^{-1}(m)$. For all $u > t$ and all $m \leqslant r_e$, $f_u \circ h_u^{-1}(m) = z_m$. Let $u > t$ be such that, for all $v > u$ and all $m \leqslant r_e$, $f_v^{-1}(z_m) = f_u^{-1}(z_m)$. For $m \leqslant r_e$, let $y_m = h_u^{-1}(m)$. Now, for all $v > u$ and all $m \leqslant r_e$, $h_v^{-1}(m) = y_m$ and $f_v(y_m) = f_u(y_m)$. It follows that R_e is not active after stage u. □

This completes the induction. We now show that all requirements are met and $h(\mathcal{A}^0)$ is a computable structure.

6.2.7. Lemma. *For all* $e \in \omega$, $\Phi_e \neq h(U^0)$.

Proof. If Φ_e is not total then there is nothing to show, so assume that Φ_e is total. By Lemma 6.2.3, for some $\vec{z} \in \omega^k$ and $t \in \omega$, Q_e is permanently satisfied through

$\vec{z}$ at stage $t+1$. Let $\vec{y} = h_t \circ f_t^{-1}(\vec{z})$. It is easy to check that, by the definition of h_{t+1}, $\Phi_e(\vec{y}) \neq U^1(\vec{z})$.

We claim that, for all $u \geqslant t$, $h_{u+1} \circ f_{u+1}^{-1}(\vec{z}) = \vec{y}$. Indeed, let $u \geqslant t$ and assume by induction that $h_u \circ f_u^{-1}(\vec{z}) = \vec{y}$. There are two cases.

1. If $f_{u+1}^{-1}(\vec{z}) = f_u^{-1}(\vec{z})$ then, no matter which way h_{u+1} is defined, $h_{u+1} \circ f_{u+1}^{-1}(\vec{z}) = h_u \circ f_u^{-1}(\vec{z}) = \vec{y}$.

2. If $f_{u+1}^{-1}(\vec{z}) \neq f_u^{-1}(\vec{z})$ then, since $q_{e,u} = f_u^{-1}(\vec{z})$, h_{u+1} extends $h_u \circ f_u^{-1} \circ f_{u+1}$, and hence $h_{u+1} \circ f_{u+1}^{-1}(\vec{z}) = h_u \circ f_u^{-1} \circ f_{u+1} \circ f_{u+1}^{-1}(\vec{z}) = h_u \circ f_u^{-1}(\vec{z}) = \vec{y}$.

So, by induction, for all $u \geqslant t$, $h_{u+1} \circ f_{u+1}^{-1}(\vec{z}) = \vec{y}$, and hence $h \circ f^{-1}(\vec{z}) = \vec{y}$. Thus $h(U^0)(\vec{y}) = U^0(h^{-1}(\vec{y})) = U^0(f^{-1}(\vec{z})) = U^1(\vec{z}) \neq \Phi_e(\vec{y})$. $\square$

6.2.8. Lemma. *For all $i \in \omega$ and all $j < n$, $\Phi_i^{h(U^0)} \neq B_j$.*

Proof. If $\Phi_i^{h(U^0)} = B_j$ then $\lim_s m_{ni+j,s} = \infty$, which we have already shown not to be the case. $\square$

6.2.9. Lemma. *$h(\mathcal{A}^0)$ is a computable structure.*

Proof. For all $s \in \omega$, $\mathrm{rng}(h_{s+1}) \supset \mathrm{rng}(h_s)$ and $h_{s+1}^{-1} \circ h_s$ is an embedding from $\mathcal{A}^0 \upharpoonright S_s$ into $\mathcal{A}^0 \upharpoonright S_{s+1}$, if we restrict the latter structure to the language L_{S_s}. Furthermore, $\bigcup_{s\in\omega} \mathrm{rng}(h_s) = \omega$. So the images of h_s form a chain whose limit $h(\mathcal{A}^0)$ is a computable structure. $\square$

The theorem follows from Lemmas 6.2.7, 6.2.8, and 6.2.9. $\blacksquare$

6.2.10. Corollary. *Let U^0 and U^1 be relations on the domains of computable structures $\mathcal{A}^0$ and $\mathcal{A}^1$, respectively. Suppose that U^0 is not computable, U^1 is computable, and there exists a Δ^0_2 isomorphism $f : \mathcal{A}^0 \cong \mathcal{A}^1$ such that $f(U^0) = U^1$. Then* $\mathrm{DgSp}_{\mathcal{A}^0}(U^0)$ *is infinite.*

The following is an obvious application of Corollary 6.2.10.

6.2.11. Corollary. *Let U be an invariant computable relation on the domain of a Δ^0_2-categorical computable structure $\mathcal{A}$. Either U is intrinsically computable or* $\mathrm{DgSp}_{\mathcal{A}}(U)$ *is infinite.*

Remark. In the above corollary, both conditions on U are necessary. In Section 2.4, we saw that there exists an invariant relation on the domain of a Δ^0_2-categorical computable structure whose degree spectrum consists of exactly two degrees, neither of them computable. Now let $\mathcal{A}^0$, $\mathcal{A}^1$, U^0, and U^1 be the structures and relations built in [25] to prove the theorem that we have numbered Theorem 3.1.2. We can assume that $|\mathcal{A}^0| \cap |\mathcal{A}^1| = \emptyset$. Let P be the predicate $\{(x, y) \mid x \in U^0 \wedge y \in U^1 \wedge$ there is an isomorphism from $\mathcal{A}^0$ to $\mathcal{A}^1$ that extends the map $x \mapsto y\}$ and let E be the equivalence relation whose equivalence classes are $|\mathcal{A}^0|$ and $|\mathcal{A}^1|$. In the proof of Theorem 4.2 of [25], it is shown that if $\mathcal{B}$ is the computable structure obtained by taking the union of $\mathcal{A}^0$ and $\mathcal{A}^1$ and expanding it by P and E then $\mathcal{B}$ is computably categorical. Since $\mathcal{B}$ has exactly one nontrivial automorphism, which sends U^1 to U^0, $\mathrm{DgSp}_{\mathcal{B}}(U^1) = \{\mathbf{0}, \deg(U^0)\}$.

Even when $\mathcal{A}$ is not Δ^0_2-categorical (and U is not necessarily invariant), it is sometimes possible to use Corollary 6.2.10 to show that either U is intrinsically computable or $\mathrm{DgSp}_{\mathcal{A}}(U)$ is infinite. The concept of a *splitting relation* is a useful tool.

6.2.12. Definition. Let U be a k-ary relation on the domain of a computable structure $\mathcal{A}$. We say that U is *splitting* if there exists a uniformly Δ^0_2 collection of finite sets $S_0, S_1, \dots \subset \left(|\mathcal{A}|^k\right)^2$ such that, for each $r \in \omega$, the following conditions are satisfied.

1. For each $(\vec{x}_0, \vec{x}_1) \in S_r$, $U(\vec{x}_0) \neq U(\vec{x}_1)$.
2. There exists an $(\vec{x}_0, \vec{x}_1) \in S_r$ such that, for all $m \geqslant r, \max(\vec{x}_0{}^\frown\vec{x}_1)$, there exist embeddings $g_i : \mathcal{A} \upharpoonright [0, m] \to \mathcal{A}$, $i = 0, 1$, such that
    (a) $g_0(\vec{x}_0) = g_1(\vec{x}_1)$ and
    (b) for all $j \in |\mathcal{A} \upharpoonright [0, r]|$, $g_0(j) = g_1(j) = j$.

6.2.13. Theorem. *If U is a splitting relation on the domain of a computable structure $\mathcal{A}$ then $\mathrm{DgSp}_{\mathcal{A}}(U)$ is infinite.*

Proof. To simplify our notation, we assume without loss of generality that $|\mathcal{A}| = \omega$. For each $m \in \omega$, let $\mathcal{A}_m = \mathcal{A} \upharpoonright [0, m]$. Let $S_0, S_1, \dots$ be as in Definition 6.2.12. We adopt the convention that, for all $r, s \in \omega$, $S_r[s] \subseteq \left([0, s]^k\right)^2$.

We will construct a Δ^0_2 function $f : \omega \xrightarrow[\text{onto}]{1\text{–}1} \omega$ such that $f(\mathcal{A})$ is a computable structure and $f(U)$ is not computable. By Corollary 6.2.10, this will be enough to establish that $\mathrm{DgSp}_{\mathcal{A}}(U)$ is infinite.

Let k be the arity of U. Let Φ_e be the e^{th} k-ary partial computable function.

For each $e \in \omega$, we will want to satisfy the requirement

$$R_e : \Phi_e \neq f(U),$$

while ensuring that $f(\mathcal{A})$ is a computable structure.

We now proceed with the construction of f.

stage 0. Let $f_0 = \emptyset$ and $m_0 = 0$. For each $e \in \omega$, let $r_{e,0} = 0$.

stage $s+1$. Let $e < s+1$ be the least number, if any, such that R_e is not currently satisfied (defined below) and if $(\vec{x}_0, \vec{x}_1) \in S_r[s]$, where $r = \max(\{r_{i,s} \mid i < e\} \cup \{e\} \cup \{y \mid f_s(y) \leqslant e\})$, then $\Phi_e(f_s(\vec{x}_i))[s]\downarrow$ for $i = 0, 1$.

If no such number exists then proceed as follows. Let $m_{s+1} = m_s + 1$. For each $i \leqslant m_s$, let $f_{s+1}(i) = f_s(i)$, and let $f_{s+1}(m_{s+1}) = m_{s+1}$. For each $i \in \omega$, let $r_{i,s+1} = r_{i,s}$.

Otherwise, let r be as above and search for a number $m > m_s$ satisfying one of the following conditions.

1. There is a pair $(\vec{x}_0, \vec{x}_1) \in S_r[m] - S_r[s]$.

2. Not 1 and there exist embeddings $g_i : \mathcal{A}_{m_s} \to \mathcal{A}_m$, $i = 0, 1$, such that, for all $j \leqslant r$, $g_0(j) = g_1(j) = j$, and for some $(\vec{x}_0, \vec{x}_1) \in S_r[s]$, $g_0(\vec{x}_0) = g_1(\vec{x}_1)$.

By the definition of S_r, such a number must exist, so there are three possibilities.

1. Condition 1 holds. For each $i \leqslant m_s$, let $f_{s+1}(i) = f_s(i)$. For each $i \in [m_s + 1, m]$, let $f_{s+1}(i) = i$. Let $r_{e,s+1} = \max\{\max(\vec{x}_0{}^\frown\vec{x}_1) \mid (\vec{x}_0, \vec{x}_1) \in S_r[s+1]\}$.

2. Condition 2 holds and, for some $i = 0, 1$, $\Phi_e(f_s(\vec{x}_i)) \neq U(\vec{x}_i)$. For each $i \leqslant m_s$, let $f_{s+1}(i) = f_s(i)$, and for each $i \in [m_s + 1, m]$, let $f_{s+1}(i) = i$. Let $r_{e,s+1} = \max(\vec{x}_i)$ and declare R_e to be satisfied.

3. Condition 2 holds and, for $i = 0, 1$, $\Phi_e(f_s(\vec{x}_i)) = U(\vec{x}_i)$. Since $U(\vec{x}_0) \neq U(\vec{x}_1)$ and $g_0(\vec{x}_0) = g_1(\vec{x}_1)$, it must be the case that, for some $i = 0, 1$, $\Phi_e(f_s(\vec{x}_i)) \neq U(g_i(\vec{x}_i))$. For each $j \in g_i([0, m_s])$, let $f_{s+1}(j) = f_s \circ g_i^{-1}(j)$. Let $j_0, \ldots, j_n$ be the elements of $[0, m] - g_i([0, m_s])$. For $l \leqslant n$, let $f_{s+1}(j_l) = m_s + l + 1$. Let $r_{e,s+1} = \max(g_i(\vec{x}_i))$ and declare R_e to be satisfied.

In any case, let $m_{s+1} = m$ and say that R_e is active at stage $s + 1$. For $i < e$, let $r_{i,s+1} = r_{i,s}$; for $i > e$, let $r_{e,s+1} = 0$ and declare R_i to be unsatisfied.

This completes the construction. We now need to show that f is Δ^0_2, $f(\mathcal{A})$ is a computable structure, and $f(U)$ is not computable. We begin by showing by induction that f is Δ^0_2 and, for each $e \in \omega$, $\lim_s r_{e,s}$ is well-defined, R_e is active only finitely often, and if Φ_e is total then R_e is eventually permanently satisfied.

For the following lemmas, fix $e \in \omega$ and assume by induction that there is a stage s_0 such that, for all $i < e$, $r_{i,s}$ has achieved a limit r_i by stage s_0 and R_i is not active after stage s_0. Let $r = \max(\{r_{i,s_0} \mid i < e\} \cup \{e\} \cup \{y \mid f_{s_0}(y) \leqslant e\})$. Notice that, for all $s > s_0$, $\max(\{r_{i,s} \mid i < e\} \cup \{e\} \cup \{y \mid f_s(y) \leqslant e\}) = r$.

6.2.14. Lemma. *$\lim_s f_s(e)$ and $\lim_s f_s^{-1}(e)$ are well-defined.*

Proof. It can be easily checked from the construction that, since $r \geqslant e$, $f_{s+1}(e) = f_s(e)$ and $f_{s+1}^{-1}(e) = f_s^{-1}(e)$ for all $s \geqslant s_0$. □

6.2.15. Lemma. *If Φ_e is total then R_e is eventually permanently satisfied.*

Proof. Assume that Φ_e is total. It is enough to show that R_e is satisfied at some stage $s > s_0$.

Suppose otherwise. Let $t > s_0$ be such that, for all $u > t$, $S_r[u] = S_r$. For all $u > t$, $r_{e,u} = r_{e,t} = \max\{\max(\vec{x}_0{}^\frown\vec{x}_1) \mid (\vec{x}_0, \vec{x}_1) \in S_r\}$, so for all $u > t$ and all $(\vec{x}_0, \vec{x}_1) \in S_r$, $f_u(\vec{x}_i) = f_t(\vec{x}_i)$ for $i = 0, 1$. Thus, since Φ_e is total, there is a $u > t$ such that $\Phi_e(f_u(\vec{x}_i))[u]\downarrow$ for all $(\vec{x}_0, \vec{x}_1) \in S_r$ and $i = 0, 1$. R_e is satisfied at stage $u + 1$. $\square$

6.2.16. Lemma. $\lim_s r_{e,s}$ *is well-defined and R_e is active only finitely often.*

Proof. If R_e is never satisfied after stage s_0 then let $t > s_0$ be as in the previous lemma. Clearly, $r_{e,u} = r_{e,t}$ for all $u > t$ and R_e is not active after stage t.

On the other hand, if R_e is satisfied at stage $u > s_0$ then $r_{e,v} = r_{e,u}$ for all $v > u$ and R_e is not active after stage u. $\square$

This completes the induction. We now show that $f(\mathcal{A})$ is a computable structure and $f(U)$ is not computable.

6.2.17. Lemma. *$f(\mathcal{A})$ is a computable structure.*

Proof. For all $s \in \omega$, $\text{rng}(f_{s+1}) \supset \text{rng}(f_s)$ and $f_{s+1}^{-1} \circ f_s$ is an embedding from $\mathcal{A}_{m_s}$ into $\mathcal{A}_{m_{s+1}}$, if we restrict $\mathcal{A}_{m_{s+1}}$ to the language of $\mathcal{A}_{m_s}$. Furthermore, $\bigcup_{s\in\omega} \text{rng}(f_s) = \omega$. $\square$

6.2.18. Lemma. *For all $e \in \omega$, $\Phi_e \neq f(U)$.*

Proof. We may assume that Φ_e is total. Let s be a stage by which R_e is permanently satisfied. It is easy to check from the construction that, for some $\vec{x} \in \omega^k$, $U(f_s^{-1}(\vec{x})) \neq \Phi_e(\vec{x})$ and, for all $t > s$, $f_t^{-1}(\vec{x}) = f_s^{-1}(\vec{x})$. Thus $U(f^{-1}(\vec{x})) \neq \Phi_e(\vec{x})$, which implies that $\Phi_e \neq f(U)$. □

The theorem follows from Lemmas 6.2.14, 6.2.17, and 6.2.18. ■

6.3 Linear Orderings

In this section, we show how Theorem 6.2.13 can be used to establish the following result.

6.3.1. Theorem. *Let U be a computable relation on the domain of a computable linear ordering $\mathcal{L}$. Either U is intrinsically computable or* $\mathrm{DgSp}_{\mathcal{L}}(U)$ *is infinite.*

Proof. Suppose that U is not intrinsically computable. We will show that U is splitting.

Let k be the arity of U. Let $\prec$ be the order relation of $\mathcal{L}$. In order to simplify our notation, we will assume without loss of generality that $|\mathcal{L}| = \omega$. For $T \subset \omega$, $n \in \omega$, and $\vec{x} \in \omega^n$, let $\mathrm{tp}_T^{\Delta}(\vec{x})$ denote the atomic n-type of $\vec{x}$ in the structure $\mathcal{L}$ expanded by a constant for each element of T.

For $r \in \omega$, let S_r be the set of all $(\vec{x}_0, \vec{x}_1) \in (\omega^k)^2$ that satisfy the following conditions.

1. $U(\vec{x}_0) \neq U(\vec{x}_1)$.

2. If $(\vec{y}_0, \vec{y}_1) \in (\omega^k)^2$ satisfies condition 1 and $\mathrm{tp}^{\Delta}_{\{0,\dots,r\}}(\vec{y}_0{}^\frown\vec{y}_1) = \mathrm{tp}^{\Delta}_{\{0,\dots,r\}}(\vec{x}_0{}^\frown\vec{x}_1)$ then $\langle \vec{x}_0{}^\frown\vec{x}_1\rangle \leqslant \langle \vec{y}_0{}^\frown\vec{y}_1\rangle$.

Now $S_0, S_1, \dots$ is a uniformly Δ^0_2 collection of finite subsets of $(\omega^k)^2$, and each S_r satisfies the first condition in Definition 6.2.12. We need to show that each S_r satisfies the second condition in Definition 6.2.12.

Fix $r \in \omega$. Let $x_0, x_1, \dots, x_r$ be such that $\{x_0, \dots, x_r\} = \{0, \dots, r\}$ and $x_0 \prec x_1 \prec \cdots \prec x_r$. Let $I_0 = \{y \in |\mathcal{L}| \mid y \prec x_0\}$, $I_{r+1} = \{y \in |\mathcal{L}| \mid x_r \prec y\}$, and, for $0 < i \leqslant r$, $I_i = \{y \in |\mathcal{L}| \mid x_{i-1} \prec y \prec x_i\}$.

Let

$$F = \{0, \dots, r\} \cup \bigcup_{\substack{i=0 \\ |I_i|<\omega}}^{r+1} I_i.$$

Since F is finite, there are only finitely many atomic types over F. Thus, since we are assuming that U is not intrinsically computable, there exist $\vec{y}_0, \vec{y}_1 \in \omega^k$ such that $\mathrm{tp}^{\Delta}_{F}(\vec{y}_0) = \mathrm{tp}^{\Delta}_{F}(\vec{y}_1)$ and $U(\vec{y}_0) \neq U(\vec{y}_1)$. By the definition of S_r, there exists a pair $(\vec{x}_0, \vec{x}_1) \in S_r$ such that $\mathrm{tp}^{\Delta}_{\{0,\dots,r\}}(\vec{x}_0{}^\frown\vec{x}_1) = \mathrm{tp}^{\Delta}_{\{0,\dots,r\}}(\vec{y}_0{}^\frown\vec{y}_1)$. But it is easy to check that

$$\mathrm{tp}^{\Delta}_{F}(\vec{y}_0) = \mathrm{tp}^{\Delta}_{F}(\vec{y}_1) \wedge \mathrm{tp}^{\Delta}_{\{0,\dots,r\}}(\vec{x}_0{}^\frown\vec{x}_1) = \mathrm{tp}^{\Delta}_{\{0,\dots,r\}}(\vec{y}_0{}^\frown\vec{y}_1) \Rightarrow \mathrm{tp}^{\Delta}_{F}(\vec{x}_0) = \mathrm{tp}^{\Delta}_{F}(\vec{x}_1).$$

Thus there exists a pair $(\vec{x}_0, \vec{x}_1) \in S_r$ such that $\mathrm{tp}^{\Delta}_{F}(\vec{x}_0) = \mathrm{tp}^{\Delta}_{F}(\vec{x}_1)$. Let $m \geqslant r, \max(\vec{x}_0{}^\frown\vec{x}_1)$. We need to define embeddings $g_i : \mathcal{L} \upharpoonright [0, m] \to \mathcal{L}$, $i = 0, 1$, such that

1. $g_0(\vec{x}_0) = g_1(\vec{x}_1)$ and

2. for all $j \leqslant r$, $g_0(j) = g_1(j) = j$.

We begin by setting $g_0(j) = g_1(j) = j$ for all $j \leqslant r$. We can define the g_i independently on each $I_k \restriction [0, m]$, $k \leqslant r + 1$. As long as we embed each $I_k \restriction [0, m]$ into $I_k \restriction [0, n]$, we will have an embedding of $\mathcal{L} \restriction [0, m]$ into $\mathcal{L} \restriction [0, n]$.

If I_k is finite then, for $j \in I_k \restriction [0, m]$, let $g_0(j) = g_1(j) = j$. Otherwise, let M be the finite linear ordering of type $\mathrm{tp}^{\Delta}(\vec{x}_0 \cap I_k) = \mathrm{tp}^{\Delta}(\vec{x}_1 \cap I_k)$. Since the class of finite linear orderings has the amalgamation property, there exists a finite linear ordering N and embeddings $h_0, h_1 : I_k \restriction [0, m] \to N$ such that, for all j such that $\vec{x}_0(j) \in I_k$, $h_0(\vec{x}_0(j)) = h_1(\vec{x}_1(j))$. Since I_k is infinite, there exists an embedding $k : N \to I_k$. For $i = 0, 1$ and $j \in I_k \restriction [0, m]$, let $g_i(j) = k \circ h_i(j)$.

It is now easy to check that the g_i are embeddings of $\mathcal{L} \restriction [0, m]$ into $\mathcal{L}$ with the desired properties. $\square$

6.4 1-decidable Structures

As mentioned in Section 5.1, every 1-decidable structure has computable dimension 1 or ω. In this section, we show that a roughly analogous situation holds in the context of degree spectra of relations. The extra condition that we need to make things work is the one that appears as condition $(*)$ in [2], which is not surprising, since the proof of Theorem 6.4.3 below is similar to that of Theorem 3.2 in [2].

We will make use of the notion of a formally computable relation, which is also due to Ash and Nerode [2].

6.4.1. Definition. A k-ary relation U on a computable structure $\mathcal{A}$ is *formally c.e.* if there exists a c.e. sequence $\psi_0, \psi_1, \ldots$ of existential formulas in the language of $\mathcal{A}$ expanded by finitely many constants from $\mathcal{A}$ such that, for every $\vec{x} \in \omega^k$, $U(\vec{x}) \Leftrightarrow \bigvee_{n\in\omega} \psi_n(\vec{x})$.

A relation U on a computable structure is *formally computable* if both it and its complement are formally c.e..

6.4.2. Theorem (Ash and Nerode). *If a relation U on a computable structure is formally computable then it is intrinsically computable.*

6.4.3. Theorem. *Let U be a computable relation on a computable structure $\mathcal{A}$ such that $\mathcal{A}$ is 1-decidable and there exists a computable procedure for determining, given an existential formula $\psi(\vec{x})$ in the language of $\mathcal{A}$ expanded by finitely many constants from $\mathcal{A}$, whether $\langle \mathcal{A}, U\rangle \vDash \forall\vec{x}(\psi(\vec{x}) \rightarrow U(\vec{x}))$. (Notice that a sufficient condition for this to hold is that $\langle \mathcal{A}, U\rangle$ be 1-decidable.) Either U is intrinsically computable or* $\mathrm{DgSp}_{\mathcal{A}}(U)$ *is infinite.*

Proof. To simplify our notation, we assume without loss of generality that $|\mathcal{A}| = \omega$. For $s \in \omega$, let L_s and $\mathcal{A}_s$ be $L_{\{0,\dots,s\}}$ and $\mathcal{A} \upharpoonright \{0, \dots, s\}$, respectively, as defined in Section 6.2.

We will attempt to construct a Δ^0_2 function $f : \omega \xrightarrow[\text{onto}]{1-1} \omega$ such that $f(\mathcal{A})$ is a computable structure and $f(U)$ is not computable. If we fail then we will be able to show that U is intrinsically computable by showing that it is formally computable.

Let k be the arity of U. Let Φ_e be the e^{th} k-ary partial computable function. For each $e \in \omega$, we will want to satisfy the requirement

$$R_e : \Phi_e \neq f(U),$$

while ensuring that $f(\mathcal{A})$ is a computable structure.

We now proceed with the construction of f.

stage 0. Let $f_0 = \emptyset$ and $m_0 = 0$. For each $e \in \omega$, let $r_{e,0} = 0$.

stage $s+1$. Let $e < s+1$ be the least number, if any, such that both of the following conditions hold, where $r = \max(\{r_{i,s} \mid i < e\} \cup \{e\} \cup \{y \mid f_s(y) \leqslant e\})$.

1. R_e is not currently satisfied (defined below).
2. There exists an $\vec{x} \in \omega^k$ such that
   (a) $\Phi_e(\vec{x})[s]\downarrow$ and
   (b) for some $\vec{y} \in \omega^k$ such that $U(\vec{y}) \neq \Phi_e(\vec{x})$, there exists an embedding $g : \mathcal{A}_s \to \mathcal{A}$ relative to L_s such that $g(\vec{x}) = \vec{y}$, and, for all $j \leqslant r$, $g(j) = j$.

   (Notice that the hypotheses of Theorem 6.4.3 guarantee that we can effectively check whether (b) holds.)

If no such number exists then proceed as follows. Let $m_{s+1} = m_s + 1$. For each $i \leqslant m_s$, let $f_{s+1}(i) = f_s(i)$, and let $f_{s+1}(m_{s+1}) = m_{s+1}$. For each $i \in \omega$, let $r_{i,s+1} = r_{i,s}$.

Otherwise, let g be as in 2(b) above and let $m_{s+1} = \max(\mathrm{rng}(g)) + 1$. Let $y_0 < y_1 < \cdots < y_{m_{s+1}-m_s-1}$ be the numbers in $[0, m_{s+1}] - \mathrm{rng}(g)$. For each $i \in \mathrm{rng}(g)$, let $f_{s+1}(i) = f_s \circ g^{-1}(i)$. For each y_i, $0 \leqslant i < m_{s+1} - m_s$, let $f_{s+1}(i) = m_s + 1 + i$. Say that R_e is active at stage $s+1$, declare R_e to be satisfied, and let $r_{e,s+1} = m_{s+1}$. For $i < e$, let $r_{i,s+1} = r_{i,s}$; for $i > e$, let $r_{e,s+1} = 0$ and declare R_i to be unsatisfied.

This completes the construction. Suppose that U is not intrinsically computable. We need to show that f is Δ^0_2, $f(\mathcal{A})$ is a computable structure, and $f(U)$ is not computable. We begin by showing by induction that f is Δ^0_2 and, for each $e \in \omega$, $\lim_s r_{e,s}$ is well-defined, R_e is active only finitely often, and if Φ_e is total then R_e is eventually permanently satisfied.

For the following lemmas, fix $e \in \omega$ and assume by induction that there is a stage s_0 such that, for all $i < e$, $r_{i,s}$ has achieved a limit r_i by stage s_0 and R_i is not active after stage s_0. Let $r = \max(\{r_{i,s_0} \mid i < e\} \cup \{e\} \cup \{y \mid f_{s_0}(y) \leqslant e\})$. Notice that, for all $s > s_0$, $\max(\{r_{i,s} \mid i < e\} \cup \{e\} \cup \{y \mid f_s(y) \leqslant e\}) = r$.

6.4.4. Lemma. $\lim_s f_s(e)$ *and* $\lim_s f_s^{-1}(e)$ *are well-defined.*

Proof. It can be easily checked from the construction that, since $r \geqslant e$, $f_{s+1}(e) = f_s(e)$ and $f_{s+1}^{-1}(e) = f_s^{-1}(e)$ for all $s \geqslant s_0$. □

6.4.5. Lemma. *If* Φ_e *is total then* R_e *is eventually permanently satisfied.*

Proof. Assume that Φ_e is total. It is enough to show that R_e is satisfied at some stage $s > s_0$.

Suppose otherwise. We then claim that U is formally computable, which, by Theorem 6.4.2, contradicts the assumption that U is not intrinsically computable.

Let $\vec{x} \in \omega^k$. Let $t+1$ be the least stage after s_0 such that $\Phi_e(\vec{x})[t]\downarrow$. Since L_t is a finite relational language, there exists an existential formula $\psi_{\vec{x}}$ in L_t expanded by a constant for each $j \leqslant r$ such that $\psi_{\vec{x}}(\vec{y})$ holds if and only if there exists an embedding $g : \mathcal{A}_t \to \mathcal{A}$ relative to L_t such that $g(\vec{x}) = \vec{y}$, and, for all $j \leqslant r$, $g(j) = j$.

Since R_e is not active at stage $t+1$, it must be the case that $\psi_{\vec{x}}(\vec{y}) \Rightarrow U(\vec{y}) = \Phi_e(\vec{x})$. Thus we have $U(\vec{y}) \Leftrightarrow \bigvee_{\Phi_e(\vec{x})=1} \psi_{\vec{x}}(\vec{y}) \Leftrightarrow \neg \bigvee_{\Phi_e(\vec{x})=0} \psi_{\vec{x}}(\vec{y})$. □

6.4.6. Lemma. $\lim_s r_{e,s}$ *is well-defined and* R_e *is active only finitely often.*

Proof. If R_e is never satisfied after stage s_0 then $r_{e,t} = r_{e,s_0}$ for all $t > s_0$ and R_e is not active after stage s_0. On the other hand, if R_e is satisfied at stage $t > s_0$ then $r_{e,u} = r_{e,t}$ for all $u > t$ and R_e is not active after stage t. □

This completes the induction. We now show that $f(\mathcal{A})$ is a computable structure and $f(U)$ is not computable.

6.4.7. Lemma. $f(\mathcal{A})$ *is a computable structure.*

Proof. For all $s \in \omega$, $\operatorname{rng}(f_{s+1}) \supset \operatorname{rng}(f_s)$ and $f_{s+1}^{-1} \circ f_s$ is an embedding from $\mathcal{A}_{m_s}$ into $\mathcal{A}_{m_{s+1}}$, if we restrict $\mathcal{A}_{m_{s+1}}$ to the language of $\mathcal{A}_{m_s}$. Furthermore, $\bigcup_{s\in\omega} \operatorname{rng}(f_s) = \omega$. □

6.4.8. Lemma. *For all* $e \in \omega$, $\Phi_e \neq f(U)$.

Proof. We may assume that Φ_e is total. Let s be a stage by which R_e is permanently satisfied. It is easy to check from the construction that, for some $\vec{x} \in \omega^k$, $U(f_s^{-1}(\vec{x})) \neq \Phi_e(\vec{x})$ and, for all $t > s$, $f_t^{-1}(\vec{x}) = f_s^{-1}(\vec{x})$. Thus $U(f^{-1}(\vec{x})) \neq \Phi_e(\vec{x})$, which implies that $\Phi_e \neq f(U)$. □

The theorem follows from Lemmas 6.4.4, 6.4.7, and 6.4.8. ■

Bibliography

[1] C. J. Ash, P. Cholak, and J. F. Knight, Permitting, forcing, and copying of a given recursive relation, Ann. Pure Appl. Logic 87 (1997) 219–236.

[2] C. J. Ash and A. Nerode, Intrinsically recursive relations, in J. N. Crossley (ed.), Aspects of Effective Algebra (Clayton, 1979) (Upside Down A Book Co., Yarra Glen, Australia, 1981) 26–41.

[3] E. Barker, Intrinsically Σ^0_α relations, Ann. Pure Appl. Logic 39 (1988) 105–130.

[4] P. Cholak, S. S. Goncharov, B. Khoussainov, and R. A. Shore, Computably categorical structures and extensions by constants, to appear in J. Symbolic Logic.

[5] S. B. Cooper, L. Harrington, A. H. Lachlan, S. Lempp, and R. I. Soare, The d.r.e. degrees are not dense, Ann. Pure Appl. Logic 55 (1991) 125–151.

[6] R. L. Epstein, R. Haas, and R. L. Kramer, Hierarchies of sets and degrees below $\mathbf{0}'$, in M. Lerman, J. H. Schmerl, and R. I. Soare (eds.), Logic Year 1979–80 (Proc. Seminars and Conf. Math. Logic, Univ. Connecticut, Storrs, Conn., 1979/80), vol. 859 of Lecture Notes in Mathematics (Springer–Verlag, Heidelberg, 1981) 32–48.

[7] 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).

[8] S. S. Goncharov, The quantity of nonautoequivalent constructivizations, Algebra and Logic 16 (1977) 169–185.

[9] S. S. Goncharov, Autostability of models and Abelian groups, Algebra and Logic 19 (1980) 13–27.

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

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

[12] S. S. Goncharov, Groups with a finite number of constructivizations, Soviet Math. Dokl. 23 (1981) 58–61.

[13] S. S. Goncharov, Limit equivalent constructivizations, in Mathematical Logic and the Theory of Algorithms ("Nauka" Sibirsk. Otdel., Novosibirsk, 1982) 4–12.

[14] S. S. Goncharov, Countable Boolean Algebras and Decidability, Siberian School of Algebra and Logic (Consultants Bureau, New York, 1997).

[15] S. S. Goncharov and V. D. Dzgoev, Autostability of models, Algebra and Logic 19 (1980) 28–37.

[16] S. S. Goncharov and B. Khoussainov, On the spectrum of degrees of decidable relations, Doklady Math. 55 (1997) 55–57, research announcement.

[17] S. S. Goncharov, A. V. Molokov, and N. S. Romanovskii, Nilpotent groups of finite algorithmic dimension,, Siberian Math. J. 30 (1989) 63–68.

[18] V. S. Harizanov, Degree spectrum of a recursive relation on a recursive structure, Phd thesis, University of Wisconsin, Madison, WI (1987).

[19] V. S. Harizanov, Some effects of Ash-Nerode and other decidability conditions on degree spectra, Ann. Pure Appl. Logic 55 (1991) 51–65.

[20] V. S. Harizanov, The possible Turing degree of the nonzero member in a two element degree spectrum, Ann. Pure Appl. Logic 60 (1993) 1–30.

[21] V. S. Harizanov, Turing degrees of certain isomorphic images of computable relations, Ann. Pure Appl. Logic 93 (1998) 103–113.

[22] D. R. Hirschfeldt, B. Khoussainov, R. A. Shore, and A. M. Slinko, Degree spectra and computable dimension in algebraic structures, to appear.

[23] W. Hodges, Model Theory, vol. 42 of Encyclopedia of Mathematics and its Applications (Cambridge University Press, Cambridge, 1993).

[24] B. Khoussainov and R. A. Shore, personal communication.

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

[26] O. Kudinov, An integral domain with finite algorithmic dimension, personal communication.

[27] P. LaRoche, Recursively represented Boolean algebras, Notices Amer. Math. Soc. 24 (1977) A–552, research announcement.

[28] A. Malcev, On a correspondence between rings and groups, Amer. Math. Soc. Translations (series 2) 45 (1965) 221–232.

[29] G. Metakides and A. Nerode, Effective content of field theory, Ann. Math. Logic 17 (1979) 289–320.

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

[31] A. T. Nurtazin, Strong and weak constructivizations and enumerable families, Algebra and Logic 13 (1974) 177–184.

[32] M. Rabin and D. Scott, The undecidability of some simple theories, unpublished.

[33] J. B. Remmel, Recursively categorical linear orderings, Proc. Amer. Math. Soc. 83 (1981) 387–391.

[34] R. I. Soare, Recursively Enumerable Sets and Degrees, Perspectives in Mathematical Logic, Omega Series (Springer–Verlag, Heidelberg, 1987).