TRANSACTIONS OF THE
AMERICAN MATHEMATICAL SOCIETY
Volume 00, Number 0, Pages 000–000
S 0002-9947(XX)0000-0

ON KIERSTEAD'S CONJECTURE

KENG MENG NG AND MAXIM ZUBKOV

ABSTRACT. We settle the longstanding Kierstead's Conjecture in the negative. We do this by constructing a computable linear order with no rational subintervals, where every block has order type finite or ζ, and where every computable copy has a strongly nontrivial Π^0_1 automorphism. We also construct a strongly η-like linear order where every block has size at most 4 with no rational subinterval such that every Δ^0_2 isomorphic computable copy has a nontrivial Π^0_1 automorphism.

1. INTRODUCTION

This paper is concerned with the longstanding Kierstead's conjecture. In this paper, we settle the conjecture by showing that the conjecture is false. This conjecture is about the problem of characterizing the order types of Π^0_1-rigid computable linear orders. Downey's survey paper [1] provides an extensive exposition and describes the motivation for this problem. As usual, we let ω, ζ, η denote the order types of the natural numbers, the integers and the rational numbers respectively. We write $\mathbb{N}, \mathbb{Q}$ for the set of natural numbers and the set of rational numbers respectively.

L. Hay and J. Rosenstein proved that the effective version of the well-known Dushnik-Miller theorem ([4]), which says that an infinite countable linear order has a nontrivial self-embedding is false:

Theorem 1.1 (L. Hay, J. Rosenstein (in [11])). *There is a computable copy of ω with no nontrivial computable self-embedding.*

By using a standard back-and-forth argument, it is easy to see that if a linear order $\mathcal{L}$ has a subinterval of type η then every computable copy of $\mathcal{L}$ has a nontrivial computable automorphism. S. Schwarz gave a characterization of linear orders with nontrivial computable automorphisms.

Theorem 1.2 (S. Schwarz [12, 13]). *Let $\mathcal{L}$ be a non-rigid computable linear order. Then $\mathcal{L}$ has a computably rigid computable copy if and only if it contains no interval of order type η. Here a linear order is computably rigid if it has no nontrivial computable automorphism.*

The investigation into η-like linear orders was initiated by H. Kierstead in his paper [10]. Recall that a linear order $\mathcal{L}$ is η-like if $\mathcal{L}$ is isomorphic to $\sum\limits_{q\in\mathbb{Q}} F(q)$

Received by the editors March 29, 2019.
2010 *Mathematics Subject Classification.* Primary 03C57, 03D45.
Key words and phrases. Computable linear order, automorphism, Kierstead's Conjecture .
The first author is partially supported by the grants MOE-RG131/17 and MOE2015-T2-2-055.
The second author is supported by the RSF grant (project No. 18-11-00028).

©XXXX American Mathematical Society

for some function $F : \mathbb{Q} \to \mathbb{N}$, and we say that the order type of $\mathcal{L}$ is defined by function F, and $\mathcal{L}$ is strongly η-like if F is bounded. Kierstead considered $2 \cdot \eta$, the simplest nontrivial instance of an η-like computable linear order, where he used an infinite injury argument to construct a computable copy of $2 \cdot \eta$ with no nontrivial Π^0_1 automorphism. A total function f is called Π^0_1 if $Graph(f) = \{(x, y) \mid f(x) = y\}$ is Π^0_1. Note that a total function having a c.e. graph is equivalent to being computable, so Π^0_1-rigidity is the next level of effective rigidity for which the classification of rigid computable linear orders is open.

Theorem 1.3 (H. Kierstead [10])**.** *There is a computable linear order of order type $2 \cdot \eta$ which has no nontrivial Π^0_1 automorphism.*

Kierstead [10] called an automorphism f is *fairly trivial* if for all $x \in L$, there are only finitely many elements between x and $f(x)$. A nontrivial automorphism f is called *strongly nontrivial*, if it is not fairly trivial, i.e. there exists some $x \in L$ where x and $f(x)$ are in different blocks. H. Kierstead's paper [10] concluded with three conjectures, with the main one as follows.

Conjecture 1.4 (H. Kierstead [10])**.** Every computable copy of a linear order $\mathcal{L}$ has a strongly nontrivial Π^0_1 automorphism if and only if $\mathcal{L}$ contains an interval of order type η.

By Theorem 1.3, this conjecture is true for the order type $2 \cdot \eta$. Later, R. Downey and M. Moses proved that Kierstead's conjecture also holds for discrete linear orders. Recall that a linear order is discrete if every element has an immediate predecessor and an immediate successor, except for possibly the greatest and least elements.

Theorem 1.5 (R. Downey, M. Moses [3])**.** *Every computable discrete linear order has a computable copy with no strongly nontrivial Π^0_1 self-embedding.*

C. Harris, K. Lee and S. B. Cooper has in recent work [8] extended Kierstead's result, where they proved that Kierstead's conjecture is true for a rather large subclass of the η-like computable linear orders. Recall the following definition:

Definition 1.6. A function F is called *X-limitwise monotonic*, abbreviated as X-l.m.f., if there is an X-computable function $f(x, s)$ such that

(1) $(\forall x)(\forall s)[f(x, s) \leq f(x, s+1)]$;
(2) $(\forall x)[F(x) = \lim_{s\to\infty} f(x, s)]$.

C. Harris, K. Lee and S. B. Cooper [8] proved that every η-like linear order with no η-interval and whose order type is defined by a $\mathbf{0}'$-l.m.f function has a Π^0_1-rigid computable copy. Obviously, for η-like linear orders, an automorphism is nontrivial if and only if it is strongly nontrivial.

Theorem 1.7 (C. Harris, K. Lee, S. B. Cooper [8])**.** *Suppose that $F : \mathbb{Q} \to \mathbb{N}$ is a $\mathbf{0}'$-limitwise monotonic function and the linear order $\mathcal{L} \cong \sum_{q\in\mathbb{Q}} F(q)$ has no η-interval. Then $\mathcal{L}$ has a Π^0_1-rigid computable copy.*

Later, G. Wu and M. Zubkov considered non-η-like linear orders by generalizing this concept. They allowed $F(x) = \lim_{s\to\infty} f(x,s)$ to take value ζ, but still required that $\lim_{s\to\infty} f(x,s)$ exists for all x.

Theorem 1.8 (G. Wu, M. Zubkov [14])**.** *Kierstead's conjecture holds for all linear orders $\mathcal{L}$ of the form $\sum_{q\in\mathbb{Q}} F(q)$, where $F:\mathbb{Q}\to\mathbb{N}\cup\{\zeta\}$ satisfies the following. There is a $\mathbf{0}'$-computable function $f:\mathbb{Q}\times\mathbb{N}\to\mathbb{N}\cup\{\zeta\}$ such that:*

(1) *for all $q\in\mathbb{Q}$, $\lim_{s\to\infty} f(q,\,s)=F(q)$;*
(2) *for all $q\in\mathbb{Q}$, $s\in\mathbb{N}$, $f(q,\,s)\leq f(q,\,s+1)$;*
(3) *if $\lim_{s\to\infty} f(q,\,s)=\zeta$ then there is s_0 such that for all $s\geq s_0$, $f(q,\,s)=\zeta$.*

Here we consider ζ as a formal symbol and order $\zeta>n$ for all $n\in\mathbb{N}$.

We remark that condition (3) in Theorem 1.8 appears to be very strong. However, our first main theorem show that it is in fact necessary and that Theorem 1.8 fails if condition (3) is removed.

The following proposition is easy to check:

Proposition 1.9. Let $F:\mathbb{Q}\to\mathbb{N}\cup\{\zeta\}$ be a function. Then the following conditions are equivalent.

(1) There is a $\mathbf{0}'$-computable function $f(q,s)$ such that
   - for all $q\in\mathbb{Q}$, $\lim_{s\to\infty} f(q,\,s)=F(q)$.
   - for all $q\in\mathbb{Q}$, $s\in\mathbb{N}$, $f(q,\,s)\leq f(q,\,s+1)$.
(2) There is a computable function $g(q,s)$ such that:
   - for all $q\in\mathbb{Q}$, $F(q)=\liminf_{s\to\infty} g(q,s)$.

   In the above we identify ζ with ∞. If $F(q)=\zeta$ then the lim and lim inf are both infinite.

The difference between F in Proposition 1.9 and Theorem 1.8 is that in the case $F(q)=\zeta$ we allow $\lim_{s\to\infty} f(x,s)$ and $\liminf_{s\to\infty} g(x,s)$ to be ∞ in Proposition 1.9, while in Theorem 1.8 $\lim_{s\to\infty} f(x,s)$ must actually exist. We remark that Proposition 1.9 can also be phrased in terms of functions that are limitwise monotonic relative to the Kleene's Ordinal Notation System O studied by A. Frolov and M. Zubkov [6], and is thus a very natural extension of $\mathbf{0}'$-l.m.f. functions. We call the functions F in Proposition 1.9 *generalized* $\mathbf{0}'$*-l.m.f. functions*.

Kierstead's conjecture has been verified for a large class of linear orders, and for a long time many have believed it to be true. The remaining cases appear to be intractable and the usual tools of computability theory do not seem to help in proving Kierstead's conjecture for these remaining cases. For this reason, the conjecture has remained open for thirty years. Our first main theorem in this paper proves the astonishing result that Kierstead's conjecture is in fact false:

First Main Theorem. There exists a generalized $\mathbf{0}'$-l.m.f. function $G:\mathbb{Q}\to\mathbb{N}\cup\{\zeta\}$ such that the linear order $\mathcal{L}\cong\sum_{q\in\mathbb{Q}} G(q)$ has no subinterval of type η and every computable copy of $\mathcal{L}$ has a strongly nontrivial Π^0_1-automorphism.

Our first main theorem builds an order type of the form $\sum_{q\in\mathbb{Q}}\liminf_{s\to\infty} g(q,s)$ for some computable g. We now compare and contrast our construction with that of an η-like linear order with a $\mathbf{0}'$-l.m.f. block function, where Kierstead's conjecture *has* been verified. That is, we wish to point out how allowing ζ as the order type of a block in our construction overcomes the difficulties present in the case when all blocks have finite size. Imagine that we are building $\mathcal{L}$ and we are watching an isomorphic copy $\mathcal{L}_e$. In $\mathcal{L}_e$ perhaps we have identified $x_0 < x_1 < x_2 < \cdots < x_n$ to be adjacent elements in the same $\mathcal{L}_e$-block, and similarly $y_0 < y_1 < y_2 < \cdots < y_n$. We of course would like to build a Π^0_1 automorphism f taking x_i to y_i. Now suppose we had to make each block finite, for example, we wish to make both blocks have size $n+1$. If a new element shows up between x_0 and x_n, then $\mathcal{L}_e$ will change its interpretation of a maximal block, and perhaps now make $x_1 < x_2 < \cdots < x_n < x_{n+1}$ a new maximal block of size $n+1$. Now as both blocks are finite, we have to map the endpoints x_1 and x_{n+1} of one block to the corresponding endpoints y_0 and y_n of the other block. Since f is Π^0_1 we might have already excluded this definition of f and thus we cannot correct f. This difficulty is precisely what is used to verify Kierstead's Conjecture for η-like linear orders with a $\mathbf{0}'$-l.m.f. block function. However, in our construction, we make our blocks have order type ζ. The discrete nature of each block means that any definition of f cannot go wrong simply because we were not matching up limit points correctly. The only threat to the correctness of f is when adjacent elements on which f or f^{-1} have already been defined are no longer adjacent. In the proof, we shall describe how we can correct f in this situation.

As we have seen, having blocks of discrete order type helps in reducing the number of situations in which we have to correct an automorphism f. Thus, we do not know if Kierstead's conjecture holds in the case where blocks might have order type ω or ω^*.

Open Question 1.10. Does Kierstead's Conjecture hold for computable linear orders with no maximal blocks of type ζ in which some (or all) maximal blocks have order type ω or ω^*, and no rational subintervals?

As we have pointed out, our counterexample to Kierstead's Conjecture is a non-η-like linear order. We do not know if Kierstead's Conjecture holds for *all* η-like linear orders.

Open Question 1.11. Does Kierstead's Conjecture hold for all computable η-like linear orders with no rational subintervals?

Question 1.11 seems extremely difficult, especially in the case where $\mathcal{L}$ is a given computable η-like linear order with no $\mathbf{0}'$-l.m.f. block function. (See [2, 15] for more discussions on these). Since guessing at the block sizes is unfeasible (the block function is Δ^0_3 in general), one would expect that in order to verify Kierstead's conjecture for such $\mathcal{L}$, we would have to construct a computable copy $\mathcal{L}_e$ directly. However, making $\mathcal{L}_e$ Δ^0_2 isomorphic to $\mathcal{L}$ will not work if we want to verify Kierstead's conjecture this way. We prove this in the second main theorem of the paper:

Second Main Theorem. There exists a strongly η-like computable linear order $\mathcal{L}$ with no η subinterval such that every computable linear order $\mathcal{L}'$ which is Δ^0_2 isomorphic to $\mathcal{L}$ has a strongly nontrivial Π^0_1-automorphism. Furthermore, every block of $\mathcal{L}$ has size at most 4.

The second main theorem says that even for strongly η-like linear orders, we cannot verify Kierstead's conjecture by constructing a Π^0_1-rigid copy which is Δ^0_2-isomorphic to a given computable copy. Thus, it tells us that in order to solve Question 1.11, we will have to construct at least Π^0_2 isomorphic copies which are Π^0_1-rigid, but this is beyond the reach of current technology.

The remainder of this paper is devoted to the proof of the two main theorems. In §2 we prove the first main theorem, and in §3 we prove the second main theorem.

2. The proof of the first main theorem

In this section we will prove our first main theorem. The proof in this section is organized as follows. In Section 2.1 we provide an informal description of the strategy. In Section 2.2 we give the formal construction for meeting a single requirement. In Section 2.3 we verify that the construction for a single requirement works. Finally in Section 2.4 we apply the uniformity of the construction in Section 2.2 to provide a solution to the first main theorem.

2.1. **An informal description of the strategy.**

2.1.1. *Requirements.* We fix $\{L_e\}_{e\in\omega}$ to be the family of r.e. subsets of $\mathbb{Q}$. We write $\mathcal{L}_e = \langle L_e, <_{\mathbb{Q}}\rangle$ and let $L_{e,s}$ be the enumeration of L_e at stage s. To prove the theorem we will construct a computable linear order $\mathcal{L}$ and satisfy the following requirements:

$$R_e : \mathcal{L} \cong \mathcal{L}_e \Rightarrow (\exists f : L_e \to L_e)\left[f \text{ is a strongly nontrivial } \Pi^0_1\text{-automorphism}\right].$$

The fact that $\mathcal{L}$ corresponds to a generalized $\mathbf{0}'$-l.m.f. function will be verified later.

We note that the requirements do indeed prove the theorem: Suppose that $\mathcal{M}$ is a computable linear order such that there are no strongly nontrivial Π^0_1 automorphisms of $\mathcal{M} = \langle M; <_{\mathcal{M}}\rangle$. It is well-known that every computable linear order can be represented as an r.e. subset of the rationals preserving all effective properties. Hence, there is e such that $\mathcal{M} \cong \mathcal{L}_e$ and $\mathcal{L}_e$ has no strongly nontrivial Π^0_1 automorphism.

The main complexity in our proof lies in the strategy for a single requirement. For this reason we will first describe the strategy to meet a single requirement in isolation and prove that this strategy works. We will then observe that the strategy is uniform (in an index for $\mathcal{L}_e$) and then take $\mathcal{L}$ to be the disjoint union of the different orderings built to satisfy each R_e, using appropriate separators to distinguish between the different locations. Unfortunately as we also have to recognize within each $\mathcal{L}_e$ the appropriate interval in which we are meeting R_e, the global construction will introduce some feedback to the basic strategy. We will address this when discussing the global construction.

The construction of the linear order $\mathcal{L} = \langle L; <_{\mathcal{L}}\rangle$ uses ideas from the work of A. Frolov and M. Zubkov [5] and [6]. We give the formal construction in §2.2.1.

2.1.2. *Overview of a single requirement.* Fix e and we now describe the strategy to meet R_e in isolation. We will construct a linear order $\mathcal{L}$ such that either:

- $\mathcal{L} \cong \zeta \cdot \eta$ or $\mathcal{L} \cong m \cdot \eta$ for some $m \in \omega$.
- $\mathcal{L} \cong \mathcal{L}_e$ implies that $\mathcal{L} \cong \zeta \cdot \eta$ and $\mathcal{L}_e$ has a strongly nontrivial Π^0_1 automorphism f_e.

Clearly not every computable copy of $\zeta \cdot \eta$ will have a strongly nontrivial Π^0_1 automorphism, since it is discrete, so it is not enough to simply take $\mathcal{L}$ to be $\zeta \cdot \eta$. Similarly we cannot always take $\mathcal{L}$ to be $m \cdot \eta$ since Kierstead's conjecture holds for strongly η-like linear orders. We have to observe how $\mathcal{L}_e$ responds to our actions, build f_e as we go along, and only decide on the isomorphism type of $\mathcal{L}$ in the limit. To simplify notations we will refer to f_e as f. At the end when we consider all requirements, we will put the outputs corresponding to the different requirements into different subintervals, so our final linear order will be neither discrete nor η-like, and it will not be effective to figure out the isomorphism type in each subinterval. In fact, the oracle needed to compute the isomorphism type of each subinterval is at least as much as an oracle needed to compute an isomorphism between any two computable copies of $\mathcal{L}$. Thus there are no contradictions in allowing each R_e to produce either $m \cdot \eta$ or $\zeta \cdot \eta$, even though Kierstead's conjecture has been verified for both strongly η-like and discrete linear orders.

Suppose that f is an automorphism of a linear order $\mathcal{L}_e$. Let $f^n(x)$, $n \in \mathbb{Z}$, be given by the following inductive definition: $f^0(x) = x$, $f^{n+1} = f(f^n(x))$, $f^{n-1} = f^{-1}(f^n(x))$. The orbit of an element x (relative to f) is the set $Orb(x) = \{y = f^n(x) \mid n \in \mathbb{Z}\}$. It easy to see that for all x and y either $Orb(x) = Orb(y)$ or $Orb(x) \cap Orb(y) = \emptyset$. Thus, $L = \bigcup\limits_{i \in \mathbb{N}} Orb(x_i)$, where the x_i's are representatives of each distinct orbit. To assist us in constructing a strongly nontrivial automorphism f of $\mathcal{L}_e$, we will construct a family of sets $\{Orb_i\}_{i\in\omega}$ which satisfy the following conditions.

- **The orbit condition:** Every set in the family is infinite, and has order type ζ.
- **The order-preserving condition:** Every pair of sets is consistent, i.e. if $x, y \in Orb_i, z \in Orb_j$ such that $x <_{\mathcal{L}_e} z <_{\mathcal{L}_e} y$ and there are no elements from Orb_i between x and y then there are $x', y' \in Orb_j$ such that $x' <_{\mathcal{L}_e} x <_{\mathcal{L}_e} z <_{\mathcal{L}_e} y <_{\mathcal{L}_e} y'$ and there are no elements from Orb_j between x' and z and between z and y' (see Figure 1).
- **The totality condition:** From every block of $\mathcal{L}_e$ there is the unique pair i and x such that $x \in Orb_i$ and x is in the block.

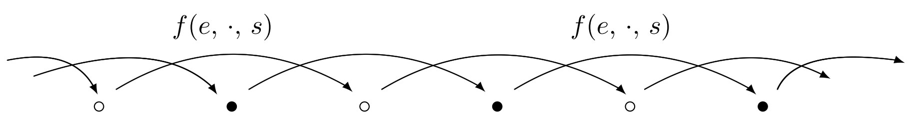

Figure 1. Two consistent orbits

Notice that the totality condition is rather strong, as it not only implies that the family of sets is pairwise disjoint, but that Orb_i and Orb_j cannot even contain elements from the same block, unless $i = j$. It also implies that each Orb_i contains at most one element from each block. Note that $\{Orb_i\}_{i\in\omega}$ is not required to cover *every* f-orbit, but the totality condition ensures that every block is covered by some orbit and so f can be uniquely extended to elements outside $Orb = \cup_i Orb_i$. (See Lemma 2.1 below).

Lemma 2.1. *Suppose that $\mathcal{L}_e \cong \zeta \cdot \eta$ and $\{Orb_i\}_{i\in\omega}$ is a family of sets satisfying the three conditions above. Then there is a unique $f : L_e \to L_e$ such that f is a strongly nontrivial automorphism such that for any $x \in Orb_i$ $f(x)$ is a successor of x inside Orb_i.*

Proof. We first define f on the elements of each Orb_i: By the orbit condition, Orb_i has order type ζ, and so we will obviously take $f(x)$ to be the successor of x inside Orb_i. Now we extend f to elements outside $Orb = \cup_i Orb_i$. If $x \notin Orb$ we find some y and i such that x and y are in the same block and $y \in Orb_i$. We find $y' <_{\mathcal{L}_e} f(y)$ if $x <_{\mathcal{L}_e} y$ or $y' >_{\mathcal{L}_e} f(y)$ if $x >_{\mathcal{L}_e} y$ such that the number of elements between x and y is equal to the number of elements between y' and $f(y)$ and define $f(x) = y'$ (see Figure 2). The totality condition guarantees that y and i can always be found (and will be unique) for each x, and the fact that each block of $\mathcal{L}_e$ is of order type ζ means that y' can be found. Therefore f is total. The second (order-preserving) condition imply that f is order-preserving on Orb. The third (totality) condition imply that a single block cannot contain elements from distinct orbits, hence f is in fact order-preserving on L_e. The fact that f is surjective also follows from the third condition. Finally, f is strongly nontrivial because each Orb_i contains at most one element of each block. □

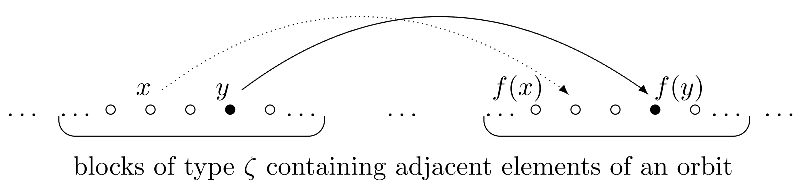

FIGURE 2. A reconstruction of an automorphism by orbits

We remark that simply constructing the orbits to have the desired properties is not quite enough, as we need both Orb and f to be effective in some way. The construction of the family $\{Orb_i\}$ will be by stages and we use the notation $Orb_i(s)$ for the effective approximation of Orb_i at stage s. Obviously we will take $\lim_{s\to\infty} Orb_i(s) = Orb_i$. However, we don't want $\{Orb_i\}$ to be just any Δ^0_2 family of sets, since the induced automorphism f might not necessarily be Π^0_1. We need to restrict how we approximate $\{Orb_i\}$. To do this we will add new elements to each Orb_i only to the left of the leftmost element, or to the right of the rightmost element of $Orb_i(s)$, but never on the inside of $Orb_i(s)$. Furthermore each Orb_i will have elements leaving (the approximation is Δ^0_2), but we always do this by splitting Orb_i at some point $x \in Orb_i$, and in two halves.

Even with a nice approximation to the family $\{Orb_i\}$, it is still not immediate that we can define a Π^0_1 approximation to f (like in lemma 2.1). We shall need to force f to have a Π^0_1 approximation, and this will be the main problem the basic strategy has to address. The entire construction will be devoted to ensuring that we can force L_e to grow in a certain way so as to allow us to define a Π^0_1 approximation to f. This approximation to f will be defined explicitly in the construction.

2.1.3. *Ensuring the properties for the orbits.* As mentioned earlier, the construction defines a computable approximation to the family $\{Orb_i\}$ during the construction.

We must ensure that at the end the constructed family satisfies the three conditions for orbits; this will be described soon. But there is a more basic requirement the family $\{Orb_i\}$ has to satisfy, namely, that every pair of elements $x, y \in Orb$ have to be in different $\mathcal{L}_e$ blocks. Since we do not control $\mathcal{L}_e$, how do we force the distance between each pair of elements in Orb to be infinite?

We define the distance between $x, y \in L_e$ to be $d(x, y) = 1 + |\{z \in L_e : z$ lies strictly between x and $y\}|$. We agree that $d(x, x) = 0$ and $d(x, y) = 1$ if x and y are adjacent. $d(x, y) = \infty$ if they are in different blocks. A similar definition holds for distance in $\mathcal{L}$, and $d(x, y, s)$ is the distance measured at stage s. Often when the context is clear we write $d(x, y)$ instead of $d(x, y, s)$.

At stage s suppose we see that $x, y \in Orb(s)$ and $d(x, y, s) = j$. To ensure that $d(x, y) = \infty$ at the end, we might want to make our copy $\mathcal{L} \cong j \cdot \eta$. Since $\mathcal{L}$ does not have a block of size $\geq j + 1$, either $\mathcal{L}_e \ncong \mathcal{L}$ or $d(x, y) > j$, which in the latter case means that $d(x, y) = \infty$. Therefore, the only mechanism we have to force $d(x, y) = \infty$ for a pair of elements $x, y \in L_e$ is for us to play a certain block size in $\mathcal{L}$. We have to be a little careful with this approach, since we cannot always promise to make $\mathcal{L}$ strongly η-like. In the case where $\mathcal{L}_e \cong \mathcal{L}$, we actually wish to make $\mathcal{L} \cong \zeta \cdot \eta$. Therefore, we have to continuously monitor $d(x, y, s)$. We will build $\mathcal{L} \cong j \cdot \eta$ until $d(x, y, s)$ increases to a new value $j' > j$. At that point we increase the size of blocks in $\mathcal{L}$ and switch to making $\mathcal{L} \cong j' \cdot \eta$, and wait for $d(x, y)$ to further increase, and so on. At the end if $d(x, y) = m < \infty$ then we end up with $\mathcal{L} \cong m \cdot \eta \ncong \mathcal{L}_e$; otherwise we will succeed in forcing $d(x, y) = \infty$ and with no obstructions to making $\mathcal{L} \cong \zeta \cdot \eta$.

To implement the above mentioned, we will control the construction of $\mathcal{L}$ and Orb using two important parameters: $g(s)$ and $h_1(s)$. The parameter $g(s)$ represents the size of blocks we currently wish for $\mathcal{L}$ to have; as long as $g(s) = m$ is stable we will proceed to build $\mathcal{L} \cong m \cdot \eta$. The parameter $h_1(s)$ is a number such that if $d(x, y, s) < h_1(s)$ then we currently believe that x and y are in the same $\mathcal{L}_e$ block. We also demand that $\min\{d(x, y, s) \mid x \neq y \in Orb(s)\} > 2h_1(s)$ and we will obviously have to keep $g(s) \leq 2h_1(s)$ in order for the above mentioned to work. We will increase $h_1(s)$ only if $\min\{d(x, y, s) \mid x \neq y \in Orb(s)\}$ grows beyond $16h_1(s)$. As long as we increase $g(s)$ together with $h_1(s)$, we have that $\mathcal{L} \cong \mathcal{L}_e$ iff $\liminf_s g(s) = \infty$ iff $\liminf_s h_1(s) = \infty$. Note that the fact that the quantities $h_1(s), 2h_1(s)$ and $16h_1(s)$ are all used are due to mere technicalities, which we will not explain at this time.

Now that we have seen the basic mechanism of how we can force two distinct points $x, y \in L_e$ to increase their distance, we are now ready to describe how we intend to satisfy the three conditions for orbits. The orbits will be extended by one of two modules during the construction:

- **The first extension of orbits:** The primary purpose of this module is to add elements to an existing orbit, and to ensure that the orbit and order-preserving conditions hold.
- **The second extension of orbits:** The main purpose of this module is to start a new orbit and to make sure that the totality condition holds.

In order to carefully control the growth of $\mathcal{L}$, we will not allow the first extension module to work at every stage, but only when triggered by the parameters h_1 and w_s. In particular, the first extension module will work on stages such that $h_1(s + 1) \neq h_1(s)$ and $w_s \uparrow$. We now elaborate on this.

A typical scenario will see the second extension module wait for h_1 to increase and then pick an element x which is currently sufficiently far from all elements in $Orb(s)$; i.e. $d(x, Orb(s)) > h_1(s)$. The second extension module would of course like to add x to a new fresh Orb_i, but obviously cannot do so until $d(x, Orb) > 2h_1$. So the second extension module will set $w_s = x$, recording the fact that it is now waiting for the element x to further separate from $Orb(s)$. While waiting we keep $g = h_1(s) + 1$ and do nothing else. Unless $\mathcal{L} \not\cong \mathcal{L}_e$, we must later discover that $d(x, Orb) > 2h_1$; at this point, the second extension module is happy to add x to a fresh Orb_i and declare the current $w_s \uparrow$.

Only when the second extension module has successfully added a new Orb_i will we allow the first extension module to act. The first extension module will pick the index of an existing orbit which needs to be extended and add a new element to it. Each orbit Orb_m will have a distinguished "central" element x_m^0, and a number of elements to the left and to the right of x_m^0. The central element x_m^0 stipulates the priority of the orbit Orb_m, and will be used to determine how the orbit Orb_m is split (under the splitting module). Basically the first extension module tries to keep the number of elements on each side of x_m^0 balanced, and adds a new element to Orb_m on the deficient side. In the actual construction we shall also have another parameter, $Cand(s)$, which is a finite set of elements that are supposed to be placed into different fresh orbits. For this reason, the first extension module will also make sure that the new element it picks to add to the existing orbit Orb_m is far from $Orb(s) \cup Cand(s)$. The second extension module will of course ignore $Cand(s)$. Note that the set $Cand$ records the set of elements on which f or f^{-1} has previously been defined and is now re-defined. We need to stream these elements into different new orbits in order to keep f Π_1^0.

2.1.4. *The x-module, simplified.* With the functions $g(s)$ and $h_1(s)$ we can establish some control over the distance of points in L_e. We have just discussed how this can be used to define the family $\{Orb_i\}$ of points in L_e with the desired properties. Unfortunately, this is not quite enough to ensure that f is Π_1^0. Suppose that the opponent was defining $\mathcal{L}_e$ and of course would like to keep $\mathcal{L}_e \cong \mathcal{L}$. Now the cunning opponent *knew* that if he made his copy $\mathcal{L}_e$ isomorphic to our copy $\mathcal{L}$, we must end up making $\mathcal{L} \cong \zeta \cdot \eta$. Armed with this knowledge, the opponent could attempt some sort of strategy similar to that for a general discrete linear order to defeat our definition of f. Our solution to this is to observe that we are not obliged to *always* produce $\mathcal{L} \cong \zeta \cdot \eta$; we only have to ensure this in the limit and only if $\mathcal{L}_e \cong \mathcal{L}$. Thus, we could, at every stage of the construction, make $\mathcal{L}$ appear to be strongly η-like. The opponent *must* reduce the block size of every block in $\mathcal{L}_e$, otherwise he risks allowing $\mathcal{L}_e \not\cong \mathcal{L}$. All we have to do is to ensure that $\liminf_s g(s) = \infty$, but we will interrupt the opponent's strategy infinitely often and make it impossible for him to keep $\mathcal{L}_e \cong \mathcal{L}$ while simultaneously running the strategy to defeat f in a discrete linear order.

We now provide more details of the strategy above. Suppose that we have defined $x < y$ to be in Orb_i for some i, such that no z in Orb_i is between x and y. Then according to our choice of f in Lemma 2.1 we will send $f(x) = y$. Suppose that $x_0 < x < x_1$ are currently adjacent and $y_0 < y < y_1$ are also currently adjacent in $\mathcal{L}_e$. Suppose also that $h_1(s) > 4$ so that we currently believe that $\{x_0, x, x_1\}$ are in the same block, and similarly for $\{y_0, y, y_1\}$. (Note that $d(x, y) > 2h_1(s)$). Thus we currently have no choice but to define $f(x_0) = y_0$ and $f(x_1) = y_1$.

The strategy the opponent would use to defeat f is the following. The opponent first enumerates a new point y_2 such that $y_0 < y_2 < y < y_1$. Suppose Orb_i is already stable and we have now fixed $f(x) = y$; this means that $f(x_0)$ must be updated to y_2. If the opponent now enumerates a new point x_2 in the corresponding position, $x_0 < x_2 < x < x_1$, then we are in a bind; since $f \in \Pi^0_1$ we cannot return $f(x_0)$ to y_0, and the other alternative is to split the orbit Orb_i at x and redefine $f(x)$. The latter alternative is not desirable, since the opponent has not yet done anything that might cause $\mathcal{L}_e \not\cong \mathcal{L}$, therefore we can only redefine $f(x)$ finitely often.

As mentioned above, our response to this is to play a strongly η-like linear order at every stage and force the opponent to reduce the size of every block in $\mathcal{L}_e$. Once the opponent enumerates the first point y_2 and we redefine $f(x_0) = y_2$, we note that it is dangerous to allow y_0 to be adjacent to, or even to remain in the same block as y_2. Therefore we will attempt to force $d(y_0, y)$ to increase enough so that we can put y_0 into a new orbit. We achieve this by first waiting for a new point to appear on the right of y; let's say that the point y_3 is enumerated so that $y_0 < y_2 < y < y_3 < y_1$. Symmetrically we redefine $f(x_1) = y_3$ and we now wish to increase $d(y_1, y)$. We now set $g(s) = 2$ and play $\mathcal{L} \cong 2 \cdot \eta$. Now notice that $d(y_0, y)$ and $d(y_1, y)$ have to both increase, unless $\mathcal{L}_e$ contains a block of size at least 3, and thus $\mathcal{L}_e \not\cong \mathcal{L}$. Once they have both increased sufficiently, we can put y_0, y_1 and y into three different orbits, and thus we will never have to worry about having to return $f(x_0)$ to y_0, or $f(x_1)$ to y_1.

2.1.5. *The x-module, in full.* The above section describes the problem that the x-module is designed to overcome, as well as the working of an x-module in a simplified situation, where we play $g(s) = 2$. The full x-module will have several other features which we will describe now.

The x-module defines a function $g(x, s)$; the function $g(s)$ will be taken to be the minimum of all $g(x, s)$. We will have an x-module for each $x \in Orb(s)$. We also keep a parameter W_s to record the progress of each x-module. The elements of W_s are 5-tuples of the form $\langle x, d_l, d_r, z_l, z_r\rangle$. For every x there is at most one such tuple in W_s.

There are two possible cases at stage $s + 1$ for the x-module.

Case 1. $\langle x, d_l, d_r, z_l, z_r\rangle \notin W_s$ **for any** d_l, d_r, z_l, z_r**:** This represents the situation where the x-module has been previously concluded successfully, and f corrected. We now wait for a new element to be enumerated close to x.

Suppose we find a new element $y > x$ enumerated into L_e such that $d(x, y) < h_1(s)$. We define z_r to be the element immediately to the right of y. Therefore we would like to increase $d(z_r, x)$ and force z_r to be put into a new orbit. We call the elements strictly between x and y *interior elements.* We next wait for a new element $y' < x$ to be enumerated into L_e (this must exist, otherwise $\mathcal{L}_e \not\cong \mathcal{L}$). We then define z_l to be the element immediately to the left of y', and also call the elements strictly between y' and x interior elements (see Figure 3). We enumerate $\langle x, d(y', x), d(x, y), z_l, z_r\rangle$ into W. Note that the number of interior elements is equal to $d(y', x) + d(x, y) - 2$. We now set $g(x) = d(y', y)$; for now we assume that no new element is enumerated between x and y while waiting for y', so that $d(y', y)$ equals to 2 + the number of interior elements.

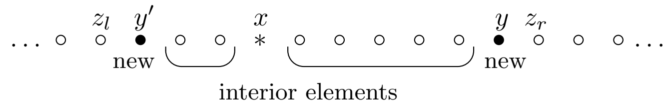

FIGURE 3. Case 1

Case 2. $\langle x, d_l, d_r, z_l, z_r \rangle \in W_s$ **for some** d_l, d_r, z_l, z_r**:** This means that we are now trying to force z_l, x and z_r into different orbits. In this situation we set $g(x) = 2 +$ the number of interior elements. Notice that one of the three alternatives must hold: Either $\mathcal{L}_e \ncong \mathcal{L}$, or $d(z_l, x)$ and $d(x, z_r)$ must both increase, or we will find a new element enumerated into L_e between two interior elements. In the first alternative we win the entire requirement. In the second alternative we will eventually see $d(z_l, x) > 2h_1$ and $d(x, z_r) > 2h_1$ so we can now add z_l and z_r to the set $Cand$. Recall that the set $Cand$ contains elements that we do not wish to add to existing orbits, hence putting z_l and z_r into $Cand$ ensures that they will eventually be placed into new orbits. We can remove $\langle x, d_l, d_r, z_l, z_r \rangle$ from W_s and declare a successful completion of the x-module.

The third alternative requires further elaboration. Suppose we discover a new element $y'' > x$ enumerated between two interior elements. Let z be the interior element immediately to the right of the new point y'' (see Figure 4). In this case it is possible that $d(z_l, x)$ remains unchanged. We will set z as our new z_r and update $\langle x, d_l, d(x, y''), z_l, z \rangle$ in W_s. We also adjust the set of interior elements as being only those which are $\leq y''$. We now decrease $g(x)$ to $d_l + d(x, y'')$, and wait again.

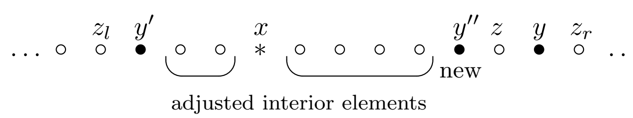

FIGURE 4. Case 2, third alternative

We do have to be careful about allowing $g(x)$ to be too small. If $\liminf_s g(x, s) < \infty$ for some fixed x, then as we will soon see in §2.1.6, this will imply that $\mathcal{L}_e \ncong \mathcal{L}$. However we have to avoid the situation where $\liminf_s g(s) < \infty$ but $\liminf_s g(x, s) = \infty$ for every x. In order to do this we impose a minimum *threshold* for each x-module; a reasonable choice would be to insist that $g(x) \geq x$ for every x. Due to global considerations, this particular requirement for defeating $\mathcal{L}_e$ will also have a parameter k, which we can think of as being constant for the requirement. We will run the above strategy for the x-module as long as the strategy does not request for $g(x) < \max\{k, x\}$. If ever the x-module discovers that $d_l + d_r < \max\{k, x\}$, it will not proceed to request for $g(s) \leq d_l + d_r$; instead it will split the orbit at the point x: What this means is to take all elements of Orb_i which are $\leq x$ in one orbit, and take all elements of Orb_i which are $> x$ in a different orbit. As observed in §2.1.4 this will automatically allow f to be corrected; more details are given in §2.1.7.

2.1.6. *Analyzing the outcomes of the x-module.* We now analyze the outcomes of the x-module. We have to ensure the global requirement $\liminf_s g(s) < \infty \Rightarrow \mathcal{L}_e \ncong \mathcal{L}$. Since each x-module adopts the threshold $\max\{k, x\}$, we see that $\liminf_s g(s) < \infty \Rightarrow \liminf_s g(x, s) < \infty$ for some x. Therefore we have to check that if an x-module infinitely often requests for $g(x) = m$ then $\mathcal{L}_e \ncong \mathcal{L}$.

Suppose that an x-module infinitely often requests for $g(x) = m$ and it is not stuck in any subcases. We claim that the size of the $\mathcal{L}_e$-block containing x is at most $m-1$. Otherwise consider a stage after which at least m many elements of the block containing x are stable. After such a stage all new elements of $\mathcal{L}_e$ are added outside of this interval of m many elements; which means that the x-module must have at least $m-1$ many interior elements. Hence the x-module will always request for $g(x) \geq 2 + (m-1) > m$, a contradiction. Therefore $\mathcal{L}_e$ will contain a block of size at most $m-1$, while $\mathcal{L} \cong m \cdot \eta$, which means that $\mathcal{L}_e \ncong \mathcal{L}$. If x-module infinitely often requests for $g(x) = m$ and it is stuck somewhere. In this case we will have two points in $\mathcal{L}_e$ such that the distance between of them is greater that m and stabilized. Hence, this two points in the same block and $\mathcal{L}_e$ has a block of size bigger than m, while $\mathcal{L} \cong m \cdot \eta$, which means that $\mathcal{L}_e \ncong \mathcal{L}$.

There is one other outcome of an x-module to consider, which is the case where the module infinitely often requests for $g(x) < \max\{k, x\}$. (Notice that this outcome has no bearing on $\liminf_s g(s)$, so the global requirement is met independently of this outcome). In this case we will split the orbit at x infinitely often, which must be avoided unless $\mathcal{L}_e \ncong \mathcal{L}$, as this case violates the first orbit condition. If this is the case then $\mathcal{L}_e$ contains a block of size less than $\max\{k, x\}$. If $\mathcal{L}_e \cong \mathcal{L}$ then by the global requirement, $\mathcal{L} \cong \zeta \cdot \eta$, which is a contradiction.

2.1.7. *Correcting f.* During the formal construction we will define an approximation to f, which has to be correct only if $\mathcal{L}_e \cong \mathcal{L} \cong \zeta \cdot \eta$. We do so by enumerating a c.e. set G which is intended to be $Cograph(f)$. At the end, we will verify that for every pair of elements x, y, $f(x) = y$ if and only if $(x, y) \notin G$, and thus f is Π^0_1. First of all, notice that we always extend an orbit by adding an element which is (and never was) part of an orbit; therefore, the approximation to f is total. Suppose x_0 is currently near to $x \in Orb$ (i.e. $0 < d(x_0, x) \leq h_1(s)$) and we have defined $f(x_0)$ such that $d(f(x_0), f(x)) = d(x_0, x)$. Now notice that the Π^0_1 requirement on $f(x_0)$ isn't in danger unless $d(x_0, x)$ increases, but if a new point is enumerated in $\mathcal{L}_e$ between x_0 and x, then the x-module would act to force x_0 to be far from x. This means that the element z that we next find such that $z \in Orb$ and $d(x_0, z) \leq h_1$ will be in an orbit distinct from any element having anything to do with x or the previous values of $f(x_0)$. Hence we can arrange for the approximation of f to be consistent with G.

We describe one more problem which we have to overcome. In the above discussion we had implied that for each $x \in Orb_i$ we will always define $f(x)$ to be the least element of Orb_i larger than x. Using the definition of f on the elements of Orb we can extend the definition of f to all elements of $\mathcal{L}_e$. However, requiring that $f(Orb_i) = Orb_i$ is a little too much. Consider the following situation. Suppose that we have defined $f(x) = y$ where $x < y$ are successive elements in some Orb_i. Suppose that $x_0 < x_1 < x$ and $y_0 < y_1 < y$ are currently adjacent, and we have defined $f(x_0) = y_0$ and $f(x_1) = y_1$. Now we must also enumerate (x_0, y_1) into G. However, suppose after we do this we see a new point y_2 show up such that $y_0 < y_1 < y_2 < y$. Notice that we have already enumerated (x_0, y_1) into G and thus we cannot update $f(x_0)$ to be y_1. Our mechanism for dealing with this was to rely on the actions of the y-module and hope that we will be able to force y_1 to be in a different block from y; if the y-module is able to do this successfully, then we can rescue f by redefining $f(x_0)$ to be one of the new elements appearing before y_2. Unfortunately, it could be that the y-module finishes unsuccessfully by splitting the orbit at y, and having no new elements show up between y_1 and y_2.

What are our options in this case? We certainly do not wish to split x and y into different orbits, since no new element might have appeared close to x. We wish to ensure the orbit condition for x and that the orbit containing x is of type ζ. In particular, x should not be allowed to lose its Orb-successor unless the orbit is also split at x. Therefore we have to keep x and y in the same orbit. However, if we also demand that $f(x) = y$ then we cannot consistently have $f(x_0) = y_1$, as this violates G. Our solution is to pick some number $q > 0$ large enough and redefine $f(x) = y'$ such that $y' > y$ and $d(y, y') = q$. Obviously we have to wait until $h_1 > q$ before we can do this, otherwise y and y' are not yet believed to be in the same block, but we can certainly stop enumerating into G until we see this. Now q is chosen large enough such that $f(x_0)$ and $f(x_1)$ can be consistently defined together with $f(x) = y'$ (notice that we need to freeze G until we are able to redefine f). Thus we have to allow $f(x)$ to be not necessarily the Orb-successor y of x, but still require $f(x)$ to be in the same block as y. In this way, $f(x)$ is redefined if the orbit is split at y, but if $\mathcal{L} \cong \mathcal{L}_e$ then the orbit is split at y only finitely often, and so $f(x)$ is redefined this way only finitely often. All other definitions of $f(x')$ where x' and x are in the same block can then reference $f(x)$.

We remark here that it seems rather critical in our proof that $\mathcal{L} \cong \zeta\cdot\eta$ in order for f to be built successfully. This is because our definition of $f(x_0)$ for an arbitrary element x_0 depends on the definition of $f(x)$ where x is the unique element in Orb in the same block as x_0. In turn, our choice of $f(x)$ was dependent on our implementation of the totality condition for orbits, which, in our construction was rather arbitrary. In other words, we are exploiting the fact that in ζ, any element is automorphic to any other element. If our constructed linear order $\mathcal{L}$ contains blocks of rigid order types, then our strategy will not work: For instance, if we wanted to anchor our definition of f at the leftmost point of each block, then we would have to redefine f whenever the leftmost point of a block appears to shift, and we would be stuck if the leftmost point of a block shifts back to a previous point. For example, our strategy will not be able to define f correctly if we instead promise to make $\mathcal{L}$ η-like if $\mathcal{L}_e \cong \mathcal{L}$. We do not know if a counter example to Kierstead's conjecture is possible if we replace $\zeta \cdot \eta$ with, say, $\omega \cdot \eta$.

2.2. **The formal construction for a single requirement.** In this section we fix a single requirement R_e and we describe the formal construction to meet R_e. We fix the associated given $\mathcal{L}_e$. As explained in the previous section, §2.1, the construction will consist of several modules and parts, and are controlled by the main module (§2.2.8). These will be presented over the different subsections of 2.2. As mentioned before, the number $k > 1$ will be assumed to be fixed for this requirement R_e, and is used by the global construction to exert control over each requirement.

2.2.1. *Construction of the linear order $\mathcal{L}$.* We construct a uniformly computable sequence $\{I_s(q)\}_{s\in\omega,\, q\in\mathbb{Q}}$ of finite linear orders such that $I(q) = \lim\limits_{s\to\infty} I_s(q)$ is a computable linear order, and $\mathcal{L} = \sum\limits_{q\in\mathbb{Q}} I(q)$ is a computable linear order. Every interval $I_s(q)$ has a special element $U(q)$, which gives the interval $I(q)$ its identity. We have the following properties: $U(q)$ is defined at the least stage s_0 when the approximation of $I(q)$ is nonempty; for every $s > s_0$ we have $U(q) \in I_s(q)$. All other elements of $I_s(q)$ may leave the interval, but will not be allowed to rejoin.

At stage $s = 0$, $I_0(q) = \emptyset$ for all $q \in \mathbb{Q}$. At stage $s+1$, we do the following. For all $q \in \mathbb{Q}$ such that the Gödel number of q is $\le s$, we modify $I_{s+1}(q)$, according to the following two cases.

Case 1. $I_s(q)$ **is not empty:** We write $I_s(q) = L_s(q) + \{U(q)\} + R_s(q) = \{l_{i_l(q,\,s)} < \ldots < l_1 < U(q) < r_1 < \ldots < r_{i_r(q,\,s)}\}$. We note that we will always either have $i_l(q,s) = i_r(q,\,s)$ or $i_l(q,s) = i_r(q,\,s) - 1$. Our actions for the block $I(q)$ will obviously depend on the current value of $g(s+1)$.

Subcase $g(s+1) = i_r(q,\,s) + i_l(q,s) + 1$**:** The requested block size is exactly right, so we leave the parameters $L_{s+1}(q)$ and $R_{s+1}(q)$ unchanged.

Subcase $g(s+1) > i_r(q,\,s) + i_l(q,s) + 1$**:** We need to add $g(s+1) - (i_l(q,\,s) + i_r(q,\,s) + 1)$ many new elements. We add the new elements on the outside, i.e. we add the new elements to the right of $r_{i_r(q,\,s)}$ and to the left of $l_{i_l(q,\,s)}$. The $g(s+1) - (i_l(q,\,s) + i_r(q,\,s) + 1)$ many new elements are distributed in a way to keep both sides balanced, i.e. we keep $i_l(q,s+1) = i_r(q,\,s+1)$ or $i_l(q,s+1) = i_r(q,\,s+1) - 1$.

Subcase $g(s+1) < i_r(q,\,s) + i_l(q,s) + 1$**:** Then we need to trim the block $I_s(q)$. If $g(s+1) = 2k$ then take $R_{s+1}(q) = \{r_1 < \ldots < r_k\}$ and $L_{s+1}(q) = \{l_{k-1} < \ldots < l_1\}$. Otherwise if $g(s+1) = 2k+1$ then take $R_{s+1}(q) = \{r_1 < \ldots < r_k\}$ and $L_{s+1}(q) = \{l_k < \ldots < l_1\}$. In either subcase define $I_{s+1}(q) = L_{s+1} + U(q) + R_{s+1}$ and call all elements from the set $I_s(q) - I_{s+1}(q)$ *free*. These free elements are already in $\mathcal{L}$ but are now not associated with any block $I(q)$, so we have to find a new block for each free element.

Case 2. $I_s(q)$ **is empty:** For uniformity we add a new point at ∞ and a new point at $-\infty$ and agree that $I(\pm\infty) = \{\pm\infty\}$. Now search for $q_1, q_2 \in \mathbb{Q}\bigcup\{-\infty, +\infty\}$ such that $q_1 <_{\mathbb{Q}} q <_{\mathbb{Q}} q_2$, $I_s(q_1)$ and $I_s(q_2)$ are not empty and there is no q_3 such that $q_1 <_{\mathbb{Q}} q_3 <_{\mathbb{Q}} q_2$ and $I_s(q_3)$ is not empty. Find the least x (in the standard order on $\mathbb{N}$) such that x *is free* and lies between $I_s(q_1)$ and $I_s(q_2)$. If there is no such x then we add a new *free* element x between $I_s(q_1)$ and $I_s(q_2)$. Define $U(q) = x$. Populate the rest of $I_{s+1}(q)$ according to Case 1.

Clearly $|I(q)| = \liminf_s |I_s(q)|$. Therefore, $\mathcal{L} \cong \sum\limits_{q\in\mathbb{Q}} \liminf_s |I_s(q)|$. It has no subinterval of type η since $|I(q)| \ge \liminf_s g(s) \ge k > 1$ for every q. Hence $\mathcal{L}$ is of the desired type. The inequality $\liminf_s g(s) \ge k$ directly follows from the construction of g.

2.2.2. *The x-module.* Formally we define, for $x, y \in L_e$ and $s \in \omega$,

$$d_e(x,y,s) = \begin{cases} |\{z \in L_e \mid z < s \ \wedge \ x <_{\mathcal{L}_e} z <_{\mathcal{L}_e} y\}| + 1, & \text{if } x <_{\mathcal{L}_e} y, \\ |\{z \in L_e \mid z < s \ \wedge \ y <_{\mathcal{L}_e} z <_{\mathcal{L}_e} x\}| + 1, & \text{if } y <_{\mathcal{L}_e} x, \\ 0, & \text{if } x = y. \end{cases}$$

We will simply write $<$ instead of $<_{\mathcal{L}_e}$ when the context is clear. The x-module will work only if $x \in Orb(s)$. We assume that at every stage at most one new element is enumerated into L_e.

First of all, if $h_1(s) \leq \max\{k, x\} + 1$, we keep $g(x, s+1) \uparrow$ and say that the x-module is *inactive*. Otherwise we declare the x-module *active* and consider the following cases.

Case 1. $\langle x, d_l, d_r, z_l, z_r \rangle \notin W_s$ **for any** d_l, d_r, z_l, z_r**:** This means that the x-module is currently not attending to any instructions and is ready to start again. There are three subcases.

Case 1.1: There is $y \in L_{e,s+1} - L_{e,s}$ such that $y > x$ and $h_1(s) > d(x,y,s+1)$ and there is some least $z > x$ such that $d(x,z,s+1) > d(x,z,s)$. We put $\langle x, -\infty, d(x,y,s+1), -\infty, z \rangle$ in W_{s+1}, and define $g(x,s+1) = h_1(s) + d(x,y,s+1) - 2$.

Case 1.2: There is $y \in L_{e,s+1} - L_{e,s}$ such that $y < x$ and $h_1(s) > d(x,y,s+1)$ and there is some greatest $z < x$ such that $d(x,z,s+1) > d(x,z,s)$. We put $\langle x, d(x,y,s+1), +\infty, z, +\infty \rangle$ in W_{s+1}, and define $g(x,s+1) = h_1(s) + d(x,y,s+1) - 2$.

Case 1.3: Otherwise, define $g(x,s+1) = 2h_1(s) + 1$.

Case 2. $\langle x, d_l, d_r, z_l, z_r \rangle \in W_s$ **and** $d_l = -\infty$ **or** $d_r = +\infty$**:** This means that the x-module has found either $z_r > x$ or $z_l < x$ (but not both) that needs to be forced to be in a different block as x.

Case 2.1: $d_l = -\infty$ and there is $y \in L_{e,s+1} - L_{e,s}$ such that $y < x$ and $d(x,y,s+1) < h_1(s)$.

(i) If $y < z$ for all $z \in L_{e,s}$ then set $g(x,s+1) = h_1(s) + d_r - 2$.

(ii) If there is some greatest $z \in L_{e,s}$ such that $z < y$ and $d_r + d(x,y,s+1) \geq \max\{k,x\}$ then set $g(x,s+1) = d(x,y,s+1) + d_r$ and update $\langle x, d(x,y,s+1), d_r, z, z_r \rangle$ in W_{s+1}.

(iii) Otherwise, i.e. the new element $y \in L_{e,s}$ is not added to the extreme left of $L_{e,s}$ and $d_r + d(x,y,s+1) < \max\{k,x\}$. Set $g(x,s+1) = \max\{k,x\} + 1$ and update $\langle x, 0, 0, 0, 0 \rangle$ in W_{s+1}. Additionally we split the orbit at point x.

Case 2.2: $d_l = -\infty$ and there is $y \in L_{e,s+1} - L_{e,s}$ such that $y > x$ and $d(x,y,s+1) < d_r$. Hence there has now appeared a new element y between x and an interior element, and we need to update z_r. Let $z > x$ be the least such that $d(x,z,s+1) > d(x,z,s)$. We define $g(x,s+1) = h_1(s) + d(x,y,s+1) - 2$ and replace $\langle x, -\infty, d_r, -\infty, z_r \rangle$ by $\langle x, -\infty, d(x,y,s+1), -\infty, z \rangle$ in W_{s+1}.

Case 2.3: $d_r = +\infty$ and there is $y \in L_{e,s+1} - L_{e,s}$ such that $y > x$ and $d(x,y,s+1) < h_1(s)$. This case is symmetric with Case 2.1.

(i) If $y > z$ for all $z \in L_{e,s}$ then set $g(x,s+1) = h_1(s) + d_l - 2$.

(ii) If there is some least $z \in L_{e,s}$ such that $z > y$ and $d_l + d(x,y,s+1) \geq \max\{k,x\}$ then set $g(x,s+1) = d(x,y,s+1) + d_l$ and update $\langle x, d_l, d(x,y,s+1), z_l, z \rangle$ in W_{s+1}.

(iii) Otherwise, i.e. the new element $y \in L_{e,s}$ is not added to the extreme right of $L_{e,s}$ and $d_l + d(x, y, s+1) < \max\{k, x\}$. Set $g(x, s+1) = \max\{k, x\} + 1$ and update $\langle x, 0, 0, 0, 0\rangle$ in W_{s+1}. Additionally we split the orbit at point x.

Case 2.4: $d_r = +\infty$ and there is $y \in L_{e,s+1} - L_{e,s}$ such that $y < x$ and $d(x, y, s+1) < d_l$. This case is symmetric with Case 2.2. There is now a new element y between an interior element and x, and we need to update z_l. Let $z < x$ be the greatest such that $d(x, z, s+1) > d(x, z, s)$. We define $g(x, s+1) = h_1(s) + d(x, y, s+1) - 2$ and replace $\langle x, d_l, +\infty, z_l, +\infty\rangle$ by $\langle x, d(x, y, s+1), +\infty, z, +\infty\rangle$ in W_{s+1}.

Case 2.5: Otherwise, do nothing and leave all parameters relating to the x-module unchanged.

Case 3. $\langle x, d_l, d_r, z_l, z_r\rangle \in W_s$ **and** $d_l \neq -\infty$**,** $d_r \neq +\infty$**:** This means that the x-module has found both $z_r > x$ and $z_l < x$, and now wants to force both to leave the block containing x. If $\langle x, 0, 0, 0, 0\rangle \in W_s$ then we say that *the x-module is split*.

Case 3.1: There is $y \in L_{e,s+1} - L_{e,s}$ such that $y > x$ and $d_r > d(x, y, s+1)$, and the x-module is not yet split. This means that a new element has appeared between x and an interior element, and we need to update z_r if we can.

(i) If $d_l + d(x, y, s+1) \geq \max\{k, x\}$, set $g(x, s+1) = d_l + d(x, y, s+1)$. Let $z > x$ be the least such that $d(x, z, s+1) > d(x, z, s)$. We update $\langle x, d_l, d_r, z_l, z_r\rangle$ with $\langle x, d_l, d(x, y, s+1), z_l, z\rangle$ in W_{s+1}.

(ii) If $d_l + d(x, y, s+1) < \max\{k, x\}$, set $g(x, s+1) = \max\{k, x\} + 1$ and update $\langle x, 0, 0, 0, 0\rangle$ in W_{s+1}. Additionally we split the orbit at point x.

Case 3.2: There is $y \in L_{e,s+1} - L_{e,s}$ such that $y < x$ and $d_l > d(x, y, s+1)$, and the x-module is not yet split. This means that a new element has appeared between an interior element and x, and we need to update z_l if we can.

(i) If $d_r + d(x, y, s+1) \geq \max\{k, x\}$, set $g(x, s+1) = d_r + d(x, y, s+1)$. Let $z < x$ be the greatest such that $d(x, z, s+1) > d(x, z, s)$. We update $\langle x, d_l, d_r, z_l, z_r\rangle$ with $\langle x, d(x, y, s+1), d_r, z, z_r\rangle$ in W_{s+1}.

(ii) If $d_r + d(x, y, s+1) < \max\{k, x\}$, set $g(x, s+1) = \max\{k, x\} + 1$ and update $\langle x, 0, 0, 0, 0\rangle$ in W_{s+1}. Additionally we split the orbit at point x.

Case 3.3: No such $y \in L_{e,s+1} - L_{e,s}$ exists, or the x-module is split. Then no updates of z_l or z_r are necessary. If $d\left(Orb(s+1), z, s+1\right) > 16h_1(s)$ for every old element z, we set $g(x, s+1) = 2h_1(s) + 1$. If we find that some old element does not satisfy the above inequality and the x-module is already split, we set $g(x, s+1) = \max\{k, x\} + 1$. Otherwise we do nothing and leave all parameters relating to the x-module unchanged.

Let t be the first stage where the current x-cycle entered case 2. We define an old element z to be one where $z \in \mathcal{L}_e[t]$, $d(z, x, t+1) \leq h_1(t)$, and where either (i) the x-module is not split and either $z \leq z_l$ or $z \geq z_r$ holds, or (ii) the x-module is split and $d(z, x, s+1) > \max\{k, x\} + 1$.

If $\langle x, -\rangle$ is removed from W, then we say that the x-module *completes a x-cycle.*

2.2.3. *Splitting the orbit at the point x.* Suppose that x is currently in $Orb_m(s) = \{x_m^{-l_s} < \ldots < x_m^0 < \ldots < x_m^{r_s}\}$. There are three cases, depending on the position of x relative to the center x_m^0. Suppose that $x = x_m^j$.

$j = -l_s$ or r_s: Then we do nothing.

$0 < j < r_s$: Then we find the least k such that $Orb_k(s) = \emptyset$ and define
$Orb_m(s+1) = \{x_m^{-l_s} < \ldots < x_m^0 < \ldots < x_m^j\}$, and
$Orb_k(s+1) = \{x_m^{j+1} < \ldots < x_m^{r_s}\} = \{x_k^0 < \ldots < x_k^{r_s-j-1}\}$.

$-l_s < j < 0$: Then we find the least k such that $Orb_k(s) = \emptyset$ and define
$Orb_m(s+1) = \{x_m^j < \ldots < x_m^0 < \ldots < x_m^{r_s}\}$.
$Orb_k(s+1) = \{x_m^{-l_s} < \ldots < x_m^{j-1}\} = \{x_k^{-l_s-j+1} < \ldots < x_k^0\}$.

$j = 0$: Then we find the least k, k' such that $Orb_k(s) = Orb_{k'}(s) = \emptyset$ and define
$Orb_m(s+1) = \{x_m^0\}$.
$Orb_k(s+1) = \{x_m^{-l_s} < \ldots < x_m^{-1}\} = \{x_k^{-l_s+1} < \ldots < x_k^0\}$.
$Orb_{k'}(s+1) = \{x_m^1 < \ldots < x_m^{r_s}\} = \{x_{k'}^0 < \ldots < x_{k'}^{r_s-1}\}$.

2.2.4. *The definition of h_1.* We use h_2 as a local parameter to help us define h_1. At stage $s = 0$, set $h_1(0) = h_2(0) = k$. At stage $s+1$, we define the quantity

$$h_2(s+1) = \min\left\{\frac{d(x,y,s+1)}{16} : x \in Orb(s+1) \cup \{-\infty, \infty\},\right.$$
$$\left. y \in Orb(s+1) \cup Cand^*(s+1) \cup \{-\infty, \infty\},\ x \neq y\right\}.$$

Here, $Cand^*(s) = \{x \in Cand(s) : d(x, Orb(s)) > 2h_1(s)\}$. Notice that $Cand$ is a c.e. set while $Cand^*$ is Δ^0_2. We check if the following conditions hold:

(i) $h_2(s+1) > h_1(s)$.
(ii) For each active x-module, either it is still in case 1 and $\langle x, -\rangle \notin W_s$, or else it is in case 3 and $d\left(Orb(s+1), z, s+1\right) > 16h_1(s)$ for every old element z.
(iii) If $w_s \downarrow= x$ then $d(x, Orb(s+1), s+1) > 16h_1(s)$.

If all three conditions hold, we call $s+1$ a *h_1-expansionary stage* and take following actions:

(i) Set $h_1(s+1) = h_1(s) + 1$.
(ii) For each active x-module in case 3, define $g(x, s+1) = 2h_1(s)+1$, $Cand(s+1) = Cand(s) \cup \{\text{all old elements}\}$, and remove $\langle x, d_l, d_r, z_l, z_r\rangle$ from W_{s+1}.
(iii) If $w_s \downarrow= x$ find the least m such that $Orb_m(s) = \emptyset$ and define $x_m^0 = x$ and $w_{s+1} \uparrow$.

Otherwise, if $s+1$ is not h_1-expansionary take $h_1(s+1) = h_1(s)$.

2.2.5. *The first extension of orbits.* The primary aim of this module is to extend an existing orbit. We use a local parameter l for this module, which records the number of times this module has acted. At stage $s = 0$, define $l(0) = 0$. If s is not h_1-expansionary we simply define $l(s+1) = l(s)$ and do nothing else. Otherwise, we assume that s is h_1-expansionary. Fix m and j such that $l(s) = \langle m, j\rangle$ and we increment $l(s+1) = l(s) + 1$.

Case 1. $Orb_m(s) = \emptyset$**:** Do nothing.

Case 2. $Orb_m(s) = \{x_m^{-l_s} <_{\mathcal{L}_e} \ldots <_{\mathcal{L}_e} x_m^0 <_{\mathcal{L}_e} \ldots <_{\mathcal{L}_e} x_m^{r_s}\}$**:** If either $x_m^{-l_s}$ or $x_m^{r_s}$ has not been defined for at least two h_1-expansionary stages, we do nothing. Otherwise we assume that they had both received their definitions at least two expansionary stages ago.

If $l_s < r_s$ then we extend to the left. We find the least x such that $x < x_m^{-l_s}$, $x \notin Orb(s)$, $d(x, Orb(s) \cup Cand(s), s) > 2h_1(s)$ and consistent with $Orb_p(s)$ for all $p \neq m$ such that $Orb_p(s) \neq \emptyset$. This means that if $x_0 < x < x_1$ for $x_0, x_1 \in Orb_p(s)$ then $f(x_0, s) < x_m^{-l_s} < f(x_1, s)$. We define $x_m^{-l_s-1} = x$.

If $l_s \geq r_s$, then we extend to the right, in a similar way. We find the least x such that $x > x_m^{r_s}$, $x \notin Orb(s)$, $d(x, Orb(s) \cup Cand(s), s) > 2h_1(s)$ and consistent with $Orb_p(s)$ for all $p \neq m$ such that $Orb_p(s) \neq \emptyset$. This means that if $x_0 < x_m^{r_s} < x_1$ for $x_0, x_1 \in Orb_p(s)$, then $f(x_0, s) < x < f(x_1, s)$. We define $x_m^{r_s+1} = x$.

We remark here that if s is h_1-expansionary, we can always find the required x by Lemma 2.3.

2.2.6. *The second extension of orbits.* The primary aim of this module is to grow a new orbit. We search for the $\mathbb{N}$-least x such that $d(x, Orb(s), s+1) > h_1(s)$. If x exists and if $w_s \uparrow$ or $w_s \downarrow >_{\mathbb{N}} x$, we define $w_{s+1} = x$.

2.2.7. *The definition of f.* Let $x_m^i \in Orb_m(s)$. We will use the following notation. Let $x_m^{i,0} = x_m^i$, and if $x_m^i < y$ and $d(x_m^i, y, s) = j$ then $x_m^{i,j} = y$. If $y < x_m^i$ and $d(x_m^i, y, s) = j$ then $x_m^{i,-j} = y$. We also enumerate a c.e. set G with the intention that G is the complement of $Graph(f)$.

Let $x = x_m^i$. If $i < 0$, the x_m^i-module is active and s is a h_1-expansionary stage, we wish to act for x. If this is the first time that x is put in Orb, we set $f(x_m^{i,j}, s+1) = x_m^{i+1,j}$ for $-h_1(s+1) < j < h_1(s+1)$, provided that none of these definitions are already in G. Otherwise, let $t < s$ be the previous time we were able to act for x. There are two cases.

Case 1: Suppose that between t and s we did not split the x-module. Let

$$q = \begin{cases} d\left(f(x_m^i), x_m^{i+1}\right)[t], & \text{if } d\left(f(x_m^i), x_m^{i+1}\right)[t] < h_1(s+1), \\ 0, & \text{if } d\left(f(x_m^i), x_m^{i+1}\right)[t] \geq h_1(s+1). \end{cases}$$

For every j such that $-h_1(s+1) < j < h_1(s+1) - q$ we set $f(x_m^{i,j}) = x_m^{i+1,q+j}$, provided that none of these definitions are already in G. (In particular, it might be the case that $d\left(f(x_m^i, t), x_m^{i+1}, t\right) \neq d\left(f(x_m^i, t), x_m^{i+1}, s\right)$, but we will keep the position relative to x_m^{i+1}). If some of these definitions are in G, do nothing else for x.

For every m', i', j and j', if $-h_1(s+1) < j < h_1(s+1) - q$, $-h_1(s+1) < j' < h_1(s+1)$ and $x_{m'}^{i'} \downarrow \neq x_m^{i+1}$, we enumerate $\left(x_m^{i,j}, x_{m'}^{i',j'}\right)$ in G. For each j satisfying $-h_1(s+1) < j < h_1(s+1) - q$ and each j' satisfying $j' \neq q + j$ and $-h_1(s+1) < j' < h_1(s+1)$ we enumerate $\left(x_m^{i,j}, x_m^{i+1,j'}\right)$ into G.

Case 2: Between t and s we had split the x-module. Find the least number $q > 0$ such that $2h_1(t+1) < q < h_1(s+1)$ and for every j satisfying $-h_1(s+1) < j < h_1(s+1) - q$, the pair $\left(x_m^{i,j}, x_m^{i+1,q+j}\right) \notin G$. If we can find q, we set $f(x_m^{i,j}) = x_m^{i+1,q+j}$ for all j satisfying $-h_1(s+1) < j < h_1(s+1) - q$. If we cannot find q, do nothing for x and consider the current stage as not having acted for x.

Now suppose that $i \geq 0$, the x-module is active and s is a h_1-expansionary stage. If x_m^{i+1} is not defined, do nothing for x_m^i. Otherwise if this is the first time x_m^{i+1} is put in Orb, we set $f(x_m^{i,j}, s+1) = x_m^{i+1,j}$ for $-h_1(s+1) < j < h_1(s+1)$, provided that none of these definitions are already in G. Otherwise let $t < s$ be the previous time we were able to act for x. There are two cases.

Case 1: Suppose that between t and s we did not split the x_m^{i+1}-module. Then the actions for x_m^i are exactly the same as Case 1 above, for $i < 0$.

Case 2: Between t and s we had split the x_m^{i+1}-module. Then the actions for x_m^i are exactly the same as Case 2 above, for $i < 0$.

2.2.8. *The main module for R_e.* At stage $s+1$ the main module consists of the following steps.

(1) Do stage $s+1$ of the second extension of orbits (§2.2.6).
(2) Do stage $s+1$ of the first extension of orbits (§2.2.5).
(3) Do stage $s+1$ of each active x-module, for $x \in Orb(s+1)$ (§2.2.2).
(4) Take $g(s+1) = \min\{h_1(s)+1, g(x,s) \mid x \leq s+1\}$.
(5) Do stage $s+1$ of the $\mathcal{L}$-construction (§2.2.1).
(6) Update h_1 (§2.2.4).
(7) Do stage $s+1$ of the definition of f (§2.2.7).

The module in §2.2.3 is not directly called upon by the main module, because it is called by the x-module (§2.2.2) as a subroutine. We also assume that a new element is enumerated into $\mathcal{L}_e$ at the beginning of every stage (before any step of the main module is considered).

2.3. **The formal verification for a single requirement.** Throughout the rest of this proof, when we refer to a parameter, we mean the value or status of the parameter when it is mentioned. For instance, a stage is h_1-expansionary if h_1 is increased the previous time we updated h_1. Whenever we refer to a stage, we mean a particular instance or "sub-stage" within that stage. Also we assume that $h_1(s) > 16$ for every s. We also adopt the "Lachlan notation", by appending $[s]$ to an expression to mean the value of the expression evaluated at s.

Lemma 2.2. *If h_1 is incremented at s, then immediately after this step, $d(x, y, s+1) > 15h_1(s+1)$ for every distinct $x \in Orb(s+1)$ and $y \in Orb(s+1) \cup Cand^*(s+1)$, unless x is added during this step and $y \in Cand^*(s+1)$.*

Proof. Note that if $y \in Cand^*(s+1) \cap Cand(s)$ then $y \in Cand^*(s)$. Hence if neither x nor y is added during this step, then $d(x, y, s+1) \geq 16h_2(s+1) > 16h_1(s)$. Since $h_1(s+1) = h_1(s) + 1$, we have $d(x, y, s+1) > 15h_1(s+1)$. If x is added during this step, then $y \in Orb(s+1)$, and as $x \neq y$, $y \in Orb(s)$. We can then apply the condition §2.2.4(iii) (for an expansionary stage). The same goes if y is added during this step and $y \in Orb(s+1)$. Finally if y is added during this step and $y \in Cand^*(s+1)$ then $x \in Orb(s)$ and we can apply condition §2.2.4(ii) (for an expansionary stage). $\square$

Lemma 2.3. *At each h_1-expansionary stage we are able to find the element x in §2.2.5.*

Proof. Let $s+1$ be h_1-expansionary. Consider case 2 of §2.2.5. Fix an element $x_m \in Orb(s+1)$, and assume that x_m was added to Orb before the previous expansionary stage. (If this is not true then we force the first extension module to wait for one more expansionary stage before considering this orbit again). Suppose we wish to find an element $x < x_m$ with the desired property, namely, we need to find a x such that $d(x, Orb(s+1) \cup Cand(s+1)) > 2h_1(s+1)$. Let $y < x_m$ be the rightmost element such that $y \in Orb(s+1) \cup Cand^*(s+1)$. If y does not exist then clearly any $x < x_m$ such that $d(x, x_m) > 4h_1(s+1)$ will have the desired property, so x can be found as long as there are sufficiently many elements to the left of x_m (which we always assume we have by speeding up the enumeration of $\mathcal{L}_e$, if necessary).

Otherwise fix the rightmost such y. Then any $z \in Cand(s+1)$ such $y < z < x_m$ must have distance at most $2h_1(s+1)$ from y or from x_m. At the last increment of h_1, since x_m was already in $Orb(s)$, we apply Lemma 2.2 to conclude that $d(x_m, y) > 15h_1$. (Note that extensions of orbits are always done at the beginning of a stage). Therefore, x can always be found between y and x_m. □

Lemma 2.4. *At every stage s and every $x \in Orb(s)$ and every $y \in Orb(s) \cup Cand^*(s)$, if $x \neq y$ then $d(x, y, s) > 2h_1(s)$.*

Proof. We proceed by induction on s. If $y \in Cand^*(s)$ then by definition we already have $d(x, y, s) > 2h_1(s)$. So we may assume that $y \in Orb(s)$. At each stage s, h_1 is increased (see §2.2.4) only if $d(x, y, s+1) \geq 16h_2(s+1) > 15h_1(s+1)$. If a new element x is added to $Orb(s)$ then (by §2.2.4 and §2.2.5) we must have $d(x, y, s+1) > 2h_1(s+1)$. □

Lemma 2.5. *If $\liminf_{s\to\infty} g(s) = m < \infty$, then $\mathcal{L} \cong m \cdot \eta$.*

Proof. There is a stage s_0 such that for all $s \geq s_0$ we have $g(s) \geq m$. Fix $q \in \mathbb{Q}$, and examine §2.2.1. Since we always grow and trim $I_s(q)$ symmetrically about the center point $U(q)$, and $U(q)$ is never changed, we always have $i_l(q) = i_r(q)$ or $i_r(q) - 1$, this means that there must be elements $x_{-l}, \ldots, x_0, \ldots, x_r$ (where $l + r + 1 = m$ and $x_0 = U(q)$) which are permanently in $I(q)$. At the infinitely many stages where $g(s+1) = m$ we will remove all other elements from the block $I(q)$. Therefore, $I(q) \cong m$.

Now we wish to argue that every element x enumerated into $\mathcal{L}$ is permanently in some block $I(q)$. The only way for x to not belong to any block is for it to become free during the construction of $\mathcal{L}$. Suppose x is the least (in the standard order of $\mathbb{N}$) free element. Then the construction must move to set $x = U(q)$ for some yet unused q. Once x is picked as $U(q)$ for some q, then it stays in $I(q)$. Thus $\mathcal{L} \cong m \cdot \eta$. □

Lemma 2.6. *Suppose that $\liminf_{s\to\infty} g(s) = m < \infty$ and that $\lim_{s\to\infty} h_1(s) + 1 > m$, then there is x such that $\liminf_{s\to\infty} g(x, s) = m$.*

Proof. Notice that h_1 is a non-decreasing function. Since each x-module is only active if $h_1(s) > \max\{k, x\} + 1$, it is easy to check the construction to see that $g(x, s) \geq \max\{k, x\}$ for every x and s. By definition, if $\liminf_{s\to\infty} g(s) < \lim_{s\to\infty} h_1(s)+1$, then there are infinitely many stages s_i such that $g(s_i) = \min\{g(x, s_i) \mid x \leq s_i\} = m$. Since $g(x, s_i) \geq \max\{k, x\}$ for each of these x, thus $\min\{g(x, s_i) \mid x \leq s_i\} = \min\{g(x, s_i) \mid x \leq m\}$, and so the minimum must be attained infinitely often by a single x. This means there is some x_0 such that $\liminf\limits_{s\to\infty} g(x_0, s) \leq m$. Clearly we must in fact have $\liminf\limits_{s\to\infty} g(x_0, s) = m$. □

Lemma 2.7. *Let x be an active module and assume that the orbit is split at x infinitely often. Then $\mathcal{L}_e$ has a block of size strictly less than* $\max\{k, x\}$.

Proof. The orbit can be split at x only in cases 2.1(iii), 2.3(iii), 3.1(ii) and 3.2(ii) of the x-module. Suppose case 2.1(iii) happens infinitely often. Then infinitely often we have $d_r + d(x, y, s+1) < \max\{k, x\}$ for some new element $y < x$. However the value of d_r was earlier (at stage s') assigned under case 1.1 (or 2.2) where we discovered a new element y' such that $y' > x$ and $d_r = d(x, y', s')$. This means that there are infinitely many pairs of distinct elements of the form y, y' and $s' < s$ where $y < x < y'$ and where $d(x, y', s') + d(x, y, s) < \max\{k, x\}$. This means that the block containing x cannot have size $\max\{k, x\}$ or greater; otherwise after $\max\{k, x\}$ many elements around x are stable, the new elements y, y' must appear outside these elements, and so $d(x, y', s') + d(x, y, s)$ cannot possibly be $< \max\{k, x\}$. Thus, x is in an $\mathcal{L}_e$-block of size strictly less than $\max\{k, x\}$.

A similar argument holds for cases 2.3(iii), 3.1(ii) and 3.2(ii). □

Here we list a fact that is important, but easy to verify:

Fact 2.8. Suppose that there are only finitely many h_1-expansionary stages. Then the parameters h_1, Orb, $Cand$ and $Cand^*$ are all eventually stable.

Proof. Trivial by the construction. □

Lemma 2.9. *Suppose that $\lim\limits_{s\to\infty} g(x, s)$ does not exist and $\liminf\limits_{s\to\infty} g(x, s) = m < \infty$, then x is in a $\mathcal{L}_e$-block of size strictly less than m.*

Proof. Suppose that the x-module completes only finitely many cycles and is eventually stuck waiting at some step. Examining the x-module reveals that it can only be stuck mid-cycle in cases 1.3, 2.1(i), 2.3(i), 2.5 or 3.3. In case 2.5, g is eventually never redefined and so $\lim\limits_{s\to\infty} g(x, s)$ exists, contrary to our assumptions. In the first three cases, since $\lim\limits_{s\to\infty} g(x, s)$ does not exist, this means that $\lim\limits_{s\to\infty} h_1(s) = \infty$, which means that $\liminf\limits_{s\to\infty} g(x, s) = \infty$, again contrary to our assumptions. Suppose we get stuck in case 3.3, then it must be the case that we switch between $g(x) = 2h_1 + 1$ and $g(x) = \max\{k, x\} + 1$ infinitely often. As the x-module is never completed, there are only finitely many expansionary stages and by Fact 2.8, h_1 and Orb are eventually stable. There are only finitely many elements which can be labeled "old" (specifically, only those elements which are around and close to x at stage t can qualify). Of these finitely many elements, if any one, say z, is labeled "old", then the label remains forever on z (with only at most one exception when the module is split) and the inequality $d\,(Orb, z) > 16h_1$ is eventually forever satisfied (as the parameters h_1 and Orb are eventually stable). Thus we cannot possibly switch between $g(x) = 2h_1 + 1$ and $g(x) = \max\{k, x\} + 1$ infinitely often, a contradiction. Thus we may assume that the x-module completes infinitely many cycles.

Since infinitely many x-cycles are completed, there are infinitely many h_1-expansionary stages. If the orbit is split at x infinitely often, we apply Lemma 2.7

to conclude that x is in a block of size less than $\max\{k, x\} \leq m$. (Recall that $g(x,s) \geq \max\{k,x\}$ for every x and s). Therefore we assume that the orbit is split at x finitely often.

Whenever we finish an x-cycle at an expansionary stage, we must have last updated z_l or z_r for the x-module in cases 2.1(ii), 2.3(ii), 3.1(i) or 3.2(i). In any case, we must have defined $g(x,s) = d(x,y) + d(x,y')$ for some new elements $y < x < y'$. A reasoning similar to the one in the proof of Lemma 2.7 shows that the size of the block containing x must be $\leq \liminf_{y,y'} d(x,y) + d(x,y') - 1$. (The crucial point here is that as infinitely many x-cycles are completed, this $\liminf$ is taken over an infinite collection of pairs y, y').

Now our case assumption is that the orbit is split at x finitely often and that $\lim\limits_{s\to\infty} h_1(s) = \infty$. Hence there are only four possibilities for the definition of $g(x,s)$ at a stage s; either $g(x,s+1) = h_1(s) + d_l - 2$, or $g(x,s+1) = h_1(s) + d_r - 2$, or $2h_1(s)+1$ or $d_l + d_r$. Definitions of the first three kinds do not affect $\liminf_s g(x,s)$ since $\lim\limits_{x\to\infty} h_1(s) = \infty$. Since d_l and d_r must be attained by $d(x,y)$ and $d(x,y')$ for some y, y', this means that $m = \liminf_s g(x,s) = \liminf_{y,y'} d(x,y) + d(x,y')$. Hence the size of the block containing x has size $\leq m - 1 < m$. □

Lemma 2.10. *If there is some active x-module such that $\lim\limits_{s\to\infty} g(x,s) = m < \infty$ and $\liminf\limits_{s\to\infty} g(y,s) \geq m$ for all active y, then either $\mathcal{L}_e \not\cong \mathcal{L}$ or $\mathcal{L}_e$ has a block of size strictly larger than m.*

Proof. Fix x such that $\lim\limits_{s\to\infty} g(x,s) = m < \infty$. We first suppose that there are infinitely many h_1-expansionary stages; therefore, $\lim\limits_{s\to\infty} h_1(s) = \infty$. If infinitely many x-cycles are ended, then $g(x,s+1) = 2h_1(s) + 1$ for infinitely many s, contrary to our assumption that $\lim\limits_{s\to\infty} g(x,s) = m < \infty$. Therefore the x-module is eventually stuck in a final cycle. If it is stuck in case 1 then $g(x,s+1) = 2h_1(s)+1$ for almost all s, which is impossible. If it is stuck in case 2 then condition (ii) of §2.2.4 will never hold and so there cannot be infinitely many h_1-expansionary stages. If it is stuck in case 3 then at the next h_1-expansionary stage we will end the "final" x-cycle, which is impossible.

Therefore for the remainder of this proof we will assume that there are only finitely many h_1-expansionary stages, i.e. $\lim\limits_{s\to\infty} h_1(s)$ exists. We first prove two claims:

Claim 2.11. If there is some y_0 such that the y_0-module is forever stuck waiting in Case 2, then $\mathcal{L}_e$ has a block of size strictly larger than m.

Proof of claim. Fix such a y_0. Suppose the y_0-module is stuck waiting in case 2. Cases 2.2 and 2.4 can only apply finitely many times before we have to leave case 2. Cases 2.1(i) and 2.3(i) can only apply finitely often, because $\lim\limits_{s\to\infty} h_1(s)$ exists. Therefore we may assume that the y_0-module is forever waiting in case 2.5.

Without loss of generality, assume $d_l = -\infty$; we argue symmetrically if $d_r = +\infty$. Let t be the least stage after which the y_0-module is forever waiting in case 2.5. By §2.2.4 as $\langle y_0, -\rangle \in W_s$, we have $h_1(s) = h_1(t)$ for all $s > t$. Furthermore $g(y_0, s) = g(y_0, t+1)$ for all $s > t$, where at stage t we had defined $g(y_0, t+1) = h_1(t) + d_r - 2 \geq m$. Since case 2.1 does not hold after stage t, every element enumerated into $\mathcal{L}_e$ to the left of y_0 must have a distance of at least $h_1(s)$ to y_0.

Also as case 2.2 does not hold after stage t, every element enumerated into $\mathcal{L}_e$ to the right of y_0 must have a distance of at least d_r to y_0. As we assume that $\mathcal{L}_e$ has no greatest or least element, at stage t there must already be at least $h_1(t)-1$ many elements to the left and d_r-1 many elements to the right of y_0. These elements must be in the same block as y_0, hence, $\mathcal{L}_e$ has a block of size at least $(h_1(t)-1)+(d_r-1)+1>m$. □

Let s_0 be the final h_1-expansionary stage, then (by examining Fact 2.8) the values of h_1, Orb and $Cand$ are stable after s_0.

Claim 2.12. Suppose there is some y_0 such that after s_0, the y_0-module is forever stuck waiting in Case 3, and there is some old element z such that $d(Orb(s+1), z, s+1) \leq 16h_1(s_0)$ for almost all $s>s_0$. Then $\mathcal{L}_e \not\cong \mathcal{L}$ or $\mathcal{L}_e$ has a block of size strictly larger than m.

Proof of claim. Fix such a y_0. Suppose the y_0-module is stuck waiting in case 3. Note that cases 3.1 and 3.2 can apply only finitely many times, therefore we assume that the y_0-module forever waiting in case 3.3, say after stage t_0. Let t_1 be the stage where this final y_0-cycle first enters case 2; notice that after we begin case 2 of this final y_0-cycle, there cannot be any more h_1-expansionary stages (otherwise the "final" y_0-cycle has to end). Therefore, $t_0>t_1>s_0$. Let z be the old element in the statement of the claim.

There are two possibilities, either the final y_0-cycle is split before getting stuck in case 3.3, or it is never split. We first assume that the final y_0-cycle is never split. As the cycle is never split, z must already be labeled old at the point the module began getting stuck (at t_0). Since our assumption is that $d(Orb(s+1), z, s+1) \leq 16h_1(s_0)$ for almost all $s>s_0$, we must in fact have $d(Orb(s+1), z, s+1) \leq 16h_1(s_0)$ for all $s \geq t_0$. Together with the fact that z was already labeled old at t_0, this means that $g(y_0, s) = d_l + d_r$ for all $s > t_0$ where $\langle y_0, d_l, d_r, z_l, z_r\rangle \in W_{t_0}$ (and never gets redefined under case 3.3). Therefore $m \leq \liminf_{s\to\infty} g(y_0, s) = d_l + d_r$.

Without loss of generality assume that $z<y_0$. Let $y_1<y_0$ be the rightmost such element in $Orb(s_0)$ (an easier argument follows if y_1 does not exist). By Lemma 2.4 we have $d(y_0, y_1, s+1) > 2h_1(s_0)$ for all $s>s_0$. Since z is declared old, by definition, we have $d(z, y_0, t_1) \leq h_1(s_0)$, and so $d(z, y_1, t_1) > h_1(s_0)$. We wish to now argue that z and y_1 are in different $\mathcal{L}_e$-blocks; suppose not, then the block containing z and y_1 will have size at least $h_1(s_0)+2$. However from step 4 of the main module we see that $\liminf_{s\to\infty} g(s) \leq h_1(s_0)+1$ and by Lemma 2.5 $\mathcal{L}$ does not have a block of size larger than $h_1(s_0)+1$, while $\mathcal{L}_e$ does, hence $\mathcal{L}_e \not\cong \mathcal{L}$.

So we may assume that z and y_1 are in different blocks, in particular, that $d(z, y_1)$ is eventually $>16h_1(s_0)$. As $d(Orb(s+1), z, s+1) \leq 16h_1(s_0)$ for all large s, it must mean that $d(y_0, z, s+1) \leq 16h_1(s_0)$ for all large s. This means that that z and y_0 are in the same block. But checking the construction reveals that when z was first declared old we had $d(z, y_0, \text{stage where } z \text{ declared old}) \geq d_l+1$. Furthermore every new L_e-element to the right of y_0 must have distance $\geq d_r$ to y_0, as Case 3.1 no longer holds after t_0. This means that the block containing y_0 has size strictly greater than $d_l+d_r \geq m$.

Now we assume the second possibility, where the final y_0-cycle is split before t_0. Since z is an old element for which $d(Orb(s+1), z, s+1) \leq 16h_1(s_0)$ for almost all $s>s_0$, case 3.3 will ensure that $m \leq \liminf_{s\to\infty} g(y_0, s) = \max\{k, y_0\}+1$. However as z is eventually labeled old we must have, by definition, $d(z, y_0, t_1) \leq h_1(s_0)$. Define y_1 as above, and we see that we also have $d(z, y_1, t_1) > h_1(s_0)$. The same argument above shows that z and y_1 are in different blocks (unless already $\mathcal{L}_e \ncong \mathcal{L}$). In particular, as above we can conclude that z and y_0 are in the same block. But as we are in the case where y_0 is split and z is old, this means that $d(z, y_0) > \max\{k, y_0\} + 1 \geq m$. Hence, the block containing y_0 has size strictly greater than m. □

Now back to the proof of Lemma 2.10. There are three possible reasons why there are no more expansionary stages after s_0: (1) There is some active y_0-module which fails condition (ii) in §2.2.4. (2) $w_s \downarrow = y_1$ is eventually always defined, but $d(y_1, Orb(s), s+1) \leq 16h_1(s_0)$ for all $s > s_0$. (3) $h_2(s+1) \leq h_1(s_0)$ for all $s > s_0$. We show below that in each case either $\mathcal{L}_e \ncong \mathcal{L}$ or $\mathcal{L}_e$ has a block of size strictly larger than m.

(1) There is some active y_0-module which fails condition (ii): Consider an active y_0 which fails condition (ii). We cannot complete infinitely many y_0-cycles because there are only finitely many expansionary stages. Therefore there is a final y_0-cycle. It cannot be stuck in case 1, as it fails condition (ii). If the y_0-module is stuck in case 2 we apply Claim 2.11. If the y_0-module is stuck in case 3, then as it fails condition (ii) we apply Claim 2.12.

(2) $d(y_1, Orb(s), s+1) \leq 16h_1(s_0)$ for almost all s: Assume that for almost every $s > s_0$ we have $w_s \downarrow = y_1$ and $d(y_1, Orb(s_0), s+1) \leq 16h_1(s_0)$. Suppose that $w_s \downarrow = y_1$ received its stable definition $w_s = y_1$ at some stage after s_0. At that point we must have $d(y_1, Orb(s_0)) > h_1(s_0)$. Furthermore we have $d(y_1, Orb(s_0), s+1) \leq 16h_1(s_0)$ for every $s > s_0$. This implies that for some $y_2 \in Orb(s_0)$ we have $h_1(s_0) < d(y_1, y_2, s+1) \leq 16h_1(s_0)$ for every $s > s_0$, and thus $\mathcal{L}_e$ contains a block of size larger than $h_1(s_0)+1$. However by step 4 of the main module, $\liminf_{s\to\infty} g(s) \leq h_1(s_0) + 1$ which by Lemma 2.5 implies that $\mathcal{L}_e \ncong \mathcal{L}$.

(3) $h_2(s+1) \leq h_1(s_0)$ for all $s > s_0$: As Orb, $Cand$ and h_1 are stable after s_0, it is obvious that $Cand^*$ is also eventually stable after s_0. Thus we have some $x \in Orb(s_0) \cup \{-\infty, \infty\}$ and some $y \in Orb(s_0) \cup Cand^*(s+1) \cup \{-\infty, \infty\}$ such that $x \neq y$ and $d(x, y, s+1) \leq 16h_1(s_0)$ for almost all s, and thus x and y are in the same block. However by Lemma 2.4, $d(x, y, s) > 2h_1(s_0)$ and thus $\mathcal{L}_e$ contains a block of size larger than $2h_1(s_0)+1$. As $\liminf_{s\to\infty} g(s) \leq h_1(s_0)+1$ we have that $\mathcal{L}_e \ncong \mathcal{L}$ by Lemma 2.5.

This ends the proof of Lemma 2.10. □

Lemma 2.13. *If $\liminf_{s\to\infty} g(s) = \infty$, then $\mathcal{L} \cong \zeta \cdot \eta$.*

Proof. For every m there is a stage s_0 such that for all $s \geq s_0$ we have $g(s) \geq m$. We always pad each $I(q)$ up to $g(s) \geq m$ many elements, and we always trim $I(q)$ symmetrically, it follows that the block $I(q)$ has at least m many elements. Each element x enumerated into $\mathcal{L}$ is permanently in some block $I(q)$; this follows from the same argument as in Lemma 2.5. □

Lemma 2.14. *If $\mathcal{L} \cong \mathcal{L}_e$, then $\mathcal{L} \cong \zeta \cdot \eta$.*

Proof. If $\mathcal{L} \not\cong \zeta \cdot \eta$ then by Lemmas 2.5 and 2.13, $\liminf_{s\to\infty} g(s) = m < \infty$ and $\mathcal{L} \cong m \cdot \eta$. Suppose that $\lim_{s\to\infty} h_1(s) + 1 > m$, then we can apply Lemma 2.6 to get some x such that $\liminf_{s\to\infty} g(x,s) = m$. By Lemmas 2.9 and 2.10 we get an $\mathcal{L}_e$-block of size different from m, or $\mathcal{L}_e \not\cong \mathcal{L}$. In either case, $\mathcal{L}_e \not\cong \mathcal{L}$.

Now suppose that $\lim_{s\to\infty} h_1(s)+1 = m$. In particular, there are only finitely many h_1-expansionary stages. Then $h_1, Orb, Cand$ and $Cand^*$ are all eventually stable. If no x-module eventually becomes active, then the reason there are only finitely many h_1-expansionary stages must be due to conditions (i) or (iii) of §2.2.4 failing to hold, and by the same argument as in Lemma 2.10 (note items (2) and (3) in the proof of Lemma 2.10), we conclude that $\mathcal{L}_e \not\cong \mathcal{L}$. So we may assume that some x-module is eventually active. In that case each active x-module gets stuck at a final x-cycle. It is easy to check that as $\lim_{s\to\infty} h_1(s)$ exists, we also have $\lim_{s\to\infty} g(x,s)$ exists. Since this conclusion holds for any active module, we apply Lemma 2.10 to an active x-module with the smallest $\lim_{s\to\infty} g(x,s)$. Thus either $\mathcal{L}_e \not\cong \mathcal{L}$ or $\mathcal{L}_e$ has a block of size larger than $\lim_{s\to\infty} g(x,s) \geq \liminf_{s\to\infty} g(s)$. In any case we have $\mathcal{L}_e \not\cong \mathcal{L}$. □

Lemma 2.15. *If $\liminf_{s\to\infty} g(s) = \infty$ then every $x \neq y \in Orb$ are in different blocks of $\mathcal{L}_e$.*

Proof. This follows directly from Lemma 2.4. □

Lemma 2.16. *If $\mathcal{L} \cong \mathcal{L}_e$ then Orb_m has order type ζ for every m.*

Proof. Since $\mathcal{L} \cong \mathcal{L}_e$, by Lemmas 2.5 and 2.14 we know that $\liminf_{s\to\infty} g(s) = \infty$. Since $g(s) \leq h_1(s) + 1$, this means that there are infinitely many h_1-expansionary stages.

By Lemma 2.2, $d(z,z') > 15h_1$ for every distinct $z, z' \in Orb$ just immediately after each time h_1 is incremented. The next action performed by the main module is the second extension of orbits, which means that there are infinitely many stages where the second extension of orbits is able to define w_s. Hence, every orbit is non-empty. (Notice that x_m^0 is never removed from Orb_m, so the condition $Orb_m \neq \emptyset$ is equivalent to $x_m^0 \downarrow$).

Now fix m and we know that x_m^0 exists. We shall now argue that Orb_m has order type ζ. By §2.2.5, $rng(l) = \omega$ and so there are infinitely many h_1-expansionary stages s such that $l(s) = \langle m, n\rangle$ for some n, where we attend to Orb_m. At each stage we extend Orb_m, we grow Orb_m from the outside, and (eventually) in both directions. Also Lemma 2.3 ensures we are always able to find an element to put in Orb_m. Thus Orb_m will have order type ζ unless the splitting module (§2.2.3) causes Orb_m to have a greatest or a least element.

Now we assume that Orb_m has a greatest or a least element x. By Lemma 2.7 the orbit can only be split at x finitely many times since we assume that $\mathcal{L}_e \cong \mathcal{L} \cong \zeta \cdot \eta$. Suppose that x is the greatest element of Orb_m. Let $x = x_m^j$. As x_m^0 is always in Orb_m, we must have $j \geq 0$. Each time $x_m^{j+1} \uparrow$ we must extend Orb_m on the right and find a new value for x_m^{j+1}, unless $x_m^{-j'}$ is the least element of Orb_m for some $j' < j$, in which case we consider $x_m^{-j'}$ instead. But as x_m^j is the largest element of Orb_m, we have to split the module at $x = x_m^j$. This means that we will split the module at x infinitely often, which is impossible, by what we just observed at the beginning of this paragraph.

A similar argument holds if x is the least element of Orb_m. □

The following is an easy, but very crucial fact about the construction. Informally, it says that whenever a new element enters $\mathcal{L}_e$ between z and some $x \in Orb$ which we currently think are in the same block, then the construction will force z to be in a different block from x, unless z is already too close to x.

Lemma 2.17. *Suppose $x \in Orb$ and z is an element such that $d(x,z) < h_1(s)$ and the x-module is active at s, where s is an expansionary stage. Suppose that a new element is later enumerated between x and z. Then at the next expansionary stage (if there is one), either z is labeled old or $d(x,z) \leq \max\{k,x\}+1$. In the latter case the x-module must be split.*

Proof. We check §2.2.2. Let t be the first stage after s where the x-module enters case 2; then obviously $z \in \mathcal{L}_e[t]$ and $d(z,x,t+1) \leq h_1(t)$. At the next expansionary stage, if the x-module is split and $d(x,z) > \max\{k,x\}+1$, then z is old. Otherwise if $d(x,z) \leq \max\{k,x\}+1$ then we are in the second alternative in the statement of the lemma. So we can suppose that the x-module is not split before the next expansionary stage. Consider the first time an element is enumerated between x and z. It is easy to see that this must cause an element between z and x (possibly z itself) to be defined as z_l or z_r. At the next expansionary stage, z is old. □

The rest of the proof will be devoted to showing that f is a strongly nontrivial Π^0_1 automorphism. We will do this over several lemmas.

Lemma 2.18. *If $\mathcal{L} \cong \mathcal{L}_e$ then for every block $B \subset L_e$ there is an $x \in Orb$ such that $x \in B$.*

Proof. Suppose that x is the $\mathbb{N}$-least number such that the block containing x has no elements from Orb. As we assume that $\mathcal{L} \cong \mathcal{L}_e$, by Lemmas 2.5 and 2.14 we know that $\liminf_{s\to\infty} g(s) = \infty$. Hence we must have $\lim_{s\to\infty} h_1(s) = \infty$. Then there is a stage s_0 such that $h_1(s) > \max\{x,k\}$ for all $s \geq s_0$.

We first claim that $w_s \downarrow \to w_s \geq x$ for every $s > s_0$. Suppose not. Then at the next expansionary stage (after the counterexample stage) we will put $w_s < x$ into Orb (or something even $\mathbb{N}$-smaller), and we may assume s_0 is large enough so this does not happen (notice that Orb is a c.e. set).

Now if there is a stage $s > s_0$ such that $d(x, Orb(s), s+1) > h_1(s)$, then at the beginning of stage s we will set $w_{s+1} = x$. (Notice that the second extension of orbits is always done at the beginning of a stage). Thus x will be added to Orb at the next expansionary stage after that.

Thus we may assume that at (the beginning of) every stage $s > s_0$, we have $d(x, Orb(s), s+1) \leq h_1(s)$. We fix some (in fact, any) y and some expansionary stage $s_1 > s_0$ such that $y \in Orb(s_1)$ and $d(x,y,s_1+1) \leq h_1(s_1)$. We claim that $d(x,y,s+1) \leq h_1(s)$ at the beginning of every $s > s_1$. Suppose instead we have $d(x,y,s+1) > h_1(s)$ at the beginning of some least stage $s > s_1$. Note that $s-1$ cannot be an expansionary stage as $d(x,y,s) \leq h_1(s-1)$ and we assume that at every stage only at most one element is enumerated into $\mathcal{L}_e$. (Note that this is unaffected by any assumptions on speeding up the enumeration of $\mathcal{L}_e$; in speeding up we only ask for evidence that $\mathcal{L}_e$ has no greatest or least element and so the distance between two points are unaffected by the speedup). Thus $h_1(s-1) = h_1(s)$ and $Orb(s-1) = Orb(s)$. Furthermore as $d(x,y,s) \leq h_1(s-1)$, by Lemma 2.4 we can see that $d(x, y', s) > h_1(s-1)$ for every $y' \neq x, y$, $y' \in Orb(s-1)$. This means that at stage s, we must have $d(x, Orb(s), s+1) > h_1(s)$, a contradiction. Thus $d(x, y, s+1) \leq h_1(s)$ at the beginning of every $s > s_1$.

Let $s_2 > s_1$ be large enough such that the y-module is active, the orbit is never again split at y (by Lemmas 2.7 and 2.14), $d(x, y, s_2+1) \leq h_1(s_2)$ and some new element p is enumerated into $\mathcal{L}_e$ between x and y where $d(p, y) < h_1(s_2)$ (at the beginning of s_2). This stage exists because we assumed that no element in the same block as x is in Orb. By Lemma 2.17 we have $d(x, y) > 16h_1$ at the next expansionary stage, contrary to our assumption that $d(x, y) \leq h_1$. □

Lemma 2.19. *For every $x \in Orb$ there is m such that $x \in Orb_m$ stably.*

Proof. Suppose that $x \in Orb_m(s)$ and $x \in Orb_n(s+1)$, then $x = x^i_m$ at stage s and $x = x^j_n$ at stage $s+1$. Examining the steps in §2.2.3 shows that $|j| < |i|$. Notice that x^0_m will never leave Orb_m. □

Lemma 2.20. *At every stage s and for every $x \in Cand(s)$, there is at most one y such that $y \in Orb(s)$ and $d(x, y) \leq 2h_1(s)$.*

Proof. We proceed by induction on s; note that $Cand$ is a c.e. set. Initially, $Cand$ is empty. If an action in the construction does not increase $Cand$, Orb or h_1, then it will not cause a problem. Let's first consider an action which adds elements to $Cand$. This is done in §2.2.4(ii), but the condition for performing this action is that $d(Orb, z) > 16h_1$ for every z added to $Cand$. If an element y is added to Orb then $d(y, y') > 16h_1$ for every existing $y' \in Orb$ if this is done in §2.2.4(iii), and $d(y, x) > 2h_1$ for every $x \in Cand$ if y is added in §2.2.5. In either case the statement still holds after the action.

Finally consider the action §2.2.4 which increases h_1. Apply Lemma 2.2 to see that $d(y, y') > 15h_1$ for every pair of distinct $y, y' \in Orb$ immediately after the action, which means of course that no $x \in Cand$ can be close to two elements of Orb. □

Lemma 2.21. *At every stage s and for every $x \in Cand(s)$, $y \in Orb(s)$ such that $d(x, y) \leq 2h_1(s)$, we have $y = x^0_m$ for some m such that y is added to Orb at some stage $t \leq s$ where $d(x, y) \leq 2h_1$ holds at every point between t and s.*

Proof. We fix x and suppose that x is added to $Cand$ under step 2.2.4(ii) at stage s_0. Just before this step we had $d(Orb, x) > 16h_1$. Suppose after this step we have $d(x, y) \leq 2h_1$ for some $y \in Orb$. If y was also added at this step, then y was defined to be x^0_m for some $Orb_m = \emptyset$, so the statement holds after this step. If y was not added at this step, then $d(x, y) > 16h_1$ before the step, so after incrementing h_1 we have $d(x, y) > 15h_1$, which is impossible.

Now we consider some step at s performed after the initial step at s_0, and assume the statement holds just before performing the step. We want to argue that the statement still holds after performing the step at s. If the step at s did not increase Orb or h_1, then the inductive step is trivial. Suppose the action at s enumerated y into Orb. If this action was done under §2.2.5 then note that $d(y, Cand) > 2h_1$, so y cannot be close to x. Suppose y was enumerated by §2.2.4(iii), then as above, y was defined to be x^0_m for some $Orb_m = \emptyset$, so the statement holds after this step.

Now we assume that the step at s increased h_1. Since we assumed that this is not the initial step, $s > s_0$, thus x was already in $Cand$ before taking this step at s. If $x \in Cand^*$ before this step then condition §2.2.4(i) implies that

$d(x, Orb) \geq h_2 > 16h_1$ just before the increment, and thus the argument in the first paragraph above can be applied. Thus we assume that $x \notin Cand^*$ before this step. As $x \in Cand$, this means that $d(x, y_0) \leq 2h_1$ for some $y_0 \in Orb$ just before the step, and by induction hypothesis, $y_0 = x_m^0$ is of the desired type. But after the increment to h_1, due to Lemma 2.20, there cannot be $y_1 \neq y_0$ such that $d(x, y_1) \leq 2h_1$ and $y_1 \in Orb$. Thus the statement holds after the step. □

The following lemma will help us show that whenever we wish to make a definition of $f(x)$ under §2.2.7 we will not be blocked from doing so.

Lemma 2.22. *Suppose that s is an expansionary stage and there are some i, m, j, j' such that $-h_1(s+1) < j, j' < h_1(s+1)$ such that $\left(x_m^{i,j}[s], x_m^{i+1,j'}[s]\right)$ is already in G. Then both $x_m^i[s]$ and $x_m^{i+1}[s]$ are in Orb at the previous expansionary stage. Suppose $t < s$ is the previous stage where we acted for $x_m^i[s]$ and between t and s we did not split the $x_m^i[s]$-module nor the $x_m^{i+1}[s]$-module. Then $j' \neq q + j$, where*

$$q = \begin{cases} d\left(f(x_m^i), x_m^{i+1}\right)[t], & \text{if } d\left(f(x_m^i), x_m^{i+1}\right)[t] < h_1(s+1), \\ 0, & \text{if } d\left(f(x_m^i), x_m^{i+1}\right)[t] \geq h_1(s+1). \end{cases}$$

Proof. Let $a = x_m^{i,j}[s]$ and $b = x_m^{i+1,j'}[s]$ and let $s_0 < s$ be an expansionary stage where we had enumerated (a, b) into G. Then $a = x_{m_0}^{i_0,j_0}[s_0]$ and $b = x_{m_0'}^{i_0',j_0'}[s_0]$ for some $i_0, i_0', j_0, j_0', m_0$ and m_0' such that $|j_0|, |j_0'| < h_1(s_0+1)$. Note that the $x_{m_0}^{i_0}$-module and the $x_{m_0'}^{i_0'}$-module have to be active at s_0.

If $x_{m_0}^{i_0}[s_0] \neq x_m^i[s]$ then this is a pair of distinct elements of Orb. By Lemma 2.4, $d\left(a, x_{m_0}^{i_0}[s_0]\right) \geq h_1$ between s_0 and s, and since $h_1 > \max\left\{k, x_{m_0}^{i_0}[s_0]\right\} + 1$, we can apply Lemma 2.17 to conclude that $a \in Cand(s)$. By Lemma 2.21 $i = 0$ and $x_m^0[s]$ must be added to Orb after stage s_0; this last fact follows by applying Lemma 2.2 to see that at each expansionary stage, $d\left(x_m^0, x_{m_0}^{i_0}\right) > 15h_1$. Similarly, if $x_{m_0'}^{i_0'}[s_0] \neq x_m^{i+1}[s]$, then the same argument above shows that $i+1 = 0$ and $x_m^0[s]$ is added to Orb after s_0.

Now we claim that we must have $x_{m_0}^{i_0}[s_0] = x_m^i[s]$ and $x_{m_0'}^{i_0'}[s_0] = x_m^{i+1}[s]$. Suppose $x_{m_0}^{i_0}[s_0] \neq x_m^i[s]$. Then the preceding paragraph tells us that $i = 0$ and x_m^0 is added to Orb after s_0. In other words, x_m^0 is first defined after s_0. This also means that the element $x_m^1[s]$ is added to Orb after s_0, which means that $x_{m_0'}^{i_0'}[s_0] \neq x_m^{i+1}[s]$. By the preceding paragraph, $i+1 = 0$, contradicting the fact that $i = 0$. A similar argument applies if $x_{m_0'}^{i_0'}[s_0] \neq x_m^{i+1}[s]$. Let us call $a^* = x_{m_0}^{i_0}[s_0] = x_m^i[s]$ and $b^* = x_{m_0'}^{i_0'}[s_0] = x_m^{i+1}[s]$.

It is clear that $m_0 = m_0'$, otherwise $a^* = x_{m_0}^{i_0}[s_0]$ and $b^* = x_{m_0'}^{i_0'}[s_0]$ are in different orbits at s_0, but have to end up in the same orbit at the later stage s, which is impossible. Thus $m_0 = m_0'$. In this case it is also easy to see that we must have $i_0' = i_0 + 1$, because a^* and b^* have to end up as successive elements of the same orbit at the later stage s.

Summarizing, we are now able to assume that $a^* = x_m^i[s] = x_{m_0}^{i_0}[s_0]$ and $b^* = x_m^{i+1}[s] = x_{m_0'}^{i_0'}[s_0] = x_{m_0}^{i_0+1}[s_0]$. Let $t < s$ be the largest stage where we were able to act for a^*. We first make a couple of observations.

(i) Clearly t exists and in fact, $s_0 \leq t$.

(ii) At stage t, both $a^* = x_m^i[s]$ and $b^* = x_m^{i+1}[s]$ are already in Orb, and are successive elements of the same orbit, although their Orb-indices might change.

(iii) If $i_0 \geq 0$ then for every stage $\hat{s}$ between s_0 and s, if $a^* = x_{\hat{m}}^{\hat{i}}[\hat{s}]$ then $\hat{i} \geq 0$. Similarly if $i_0 + 1 \leq 0$ then for every stage $\hat{s}$ between s_0 and s, if $b^* = x_{\hat{m}}^{\hat{i}}[\hat{s}]$ then $\hat{i} \leq 0$.

(iv) For every stage $\hat{s}$ between s_0 and s, $d(a, a^*, \hat{s}) < h_1(\hat{s})$ and $d(b, b^*, \hat{s}) < h_1(\hat{s})$. This follows by the previous part of the proof of the current lemma.

(v) Either $d(a, a^*, s_0) = d(a, a^*, s)$ and $d(b, b^*, s_0) = d(b, b^*, s)$, or else $j' \neq q+j$. We now show this. Suppose $i_0 \geq 0$. If some new element enters $\mathcal{L}_e$ between a and a^*, by Lemma 2.17, then either the a^* module is split, or $d(a, a^*)$ is increased large. By item (iv) above, the second alternative is impossible. However, if the a^* module is split, as $i_0 \geq 0$, by item (iii) above, we must put a^* and b^* into different orbits, and they cannot end up in the same orbit at s. Therefore, $d(a, a^*, s_0) = d(a, a^*, s)$ and thus $j_0 = j$.

Now we suppose that some new element enters $\mathcal{L}_e$ between b and b^*. Similarly, we can conclude that the b^*-module must be split, and in fact $d(b, b^*) < \max\{k, b^*\} + 1$. This must take place before t, as the b^*-module is assumed not to split between t and s. Repeating this each time a new element enters b and b^*, we see that by the time we get to stage s, we must still have $d(b, b^*, s) < \max\{k, b^*\} + 1$.

Consider the final time the b^*-module is split before t. At the next time we manage to act for a^* (which has to be at t or before, call it $t' \leq t$), we must be in case 2 of §2.2.7, and we would define $f(a^*, t')$ such that $2h_1(s_0) < d\left(f(a^*), b^*\right)[t'] < h_1(t')$. By item (iii) above, a^* and b^* are always successive elements of the same orbit, and so at every expansionary stage after t', the value of q is always non-zero. Thus value of q stays constant between t' and s, and therefore, the value of q at s is larger than $2h_1(s_0) > |j_0| + (\max\{k, b^*\} + 1) > |j| + d(b, b^*, s) = |j| + |j'|$. Thus, $j' \neq q + j$.

Finally, if $i_0 < 0$ we argue similarly. We can easily show that $d(b, b^*, s_0) = d(b, b^*, s)$, because splitting the b^*-module causes a^* and b^* to be put into different orbits. Now to argue that $d(a, a^*, s_0) = d(a, a^*, s)$, we observe that if the a^*-module is split then q will be redefined large, and proceed similarly to above.

Now we want to conclude the proof of the lemma. Let $u_0 < u_1$ be two consecutive stages where we had acted for a^*, such that $s_0 \leq u_0 < u_1 \leq t$. (If $s_0 = t$ then we immediately get the conclusion at the end of this paragraph). At u_0 since we had acted for a^* and since $u_0 \geq s_0$, we had to be in case 1 or case 2, which means that we were able to define $f(a^*)$ such that $d\left(f(a^*), b^*\right)[u_0] < h_1(u_0)$. If case 1 applies at u_1, then at stage u_1, we must have evaluated $q[u_1] = d\left(f(a^*), b^*\right)[u_0]$ and thus we would have kept $d\left(f(a^*), b^*\right)[u_0] = d\left(f(a^*), b^*\right)[u_1]$. If case 2 applies we would have redefined $f(a^*)$ such that $d\left(f(a^*), b^*\right)[u_1] > 2h_1(u_0+1) > d(a, a^*, u_0) + d(b, b^*, u_0) = d(a, a^*, s) + d(b, b^*, s) = |j| + |j'|$, due to item (v) above. This means that at stage t, either $d\left(f(a^*), b^*\right)[s_0] = d\left(f(a^*), b^*\right)[t]$, or $d\left(f(a^*), b^*\right)[t] > |j| + |j'|$. We also have $q = d\left(f(a^*), b^*\right)[t]$, since at stage t we had acted for a^*. If the second alternative holds, then $q = d\left(f(a^*), b^*\right)[t] > |j| + |j'|$ and so $j' \neq q + j$. If the first alternative holds, then $q = d\left(f(a^*), b^*\right)[s_0]$, and at stage s_0

when we enumerated (a,b) into G, the condition for doing so implies that $j_0' \neq d\left(f(a^*), b^*\right)[s_0] + j_0$. Again by item (v) above, we conclude that $j' \neq q + j$. □

Lemma 2.23. *If $\mathcal{L} \cong \mathcal{L}_e$ then for each m and i there are infinitely many stages where we are able to act for x_m^i in §2.2.7.*

Proof. First of all, observe that when considering some $x_m^i[s] \in Orb$ in §2.2.7 and $i < 0$, if it is the first time $x_m^i[s]$ is put in Orb, we will be able to act for it; for this we apply the first part of Lemma 2.22. Similarly if $i \geq 0$ and if $x_m^{i+1}[s]$ is first put into Orb, we will be able to act for $x_m^i[s]$. Therefore, the only way for any x_m^i to be stuck is under case 1 or case 2.

As we assume that $\mathcal{L} \cong \mathcal{L}_e$ we have $\lim_{s\to\infty} g(s) = \lim_{s\to\infty} h_1(s) = \infty$. Fix m and i. By Lemma 2.16, x_m^i and x_m^{i+1} are eventually defined and stable, and as $\lim_{s\to\infty} h_1(s) = \infty$, there are infinitely many expansionary stages and the x_m^i-module and x_m^{i+1}-module are eventually active. Let a^* and b^* be the final values of x_m^i and x_m^{i+1} respectively. If we get stuck after $x_m^i = a^*$ and x_m^{i+1} have received their final values, then we have to be stuck in either case 1 or case 2, and in particular, the final stage t where we were able to act for a^* exists.

Suppose after t we never split the a^*-module or the b^*-module. Then we are stuck in case 1 after t. We apply Lemma 2.22 to see that this case is impossible.

Suppose after t we split one of the two modules. It is straightforward to see that we are eventually stuck in case 2 (note that b^* must already be in $Orb(t)$ and be the Orb-successor of a^* at stage t). But as $\lim_{s\to\infty} h_1(s+1) = \infty$, we fix some s and some q such that $2h_1(t+1) < q < h_1(s+1)$ and argue that we must be able to act at stage s with q. Suppose not. Fix some j such that $-h_1(s+1) < j < h_1(s+1) - q$ and $\left(x_m^{i,j}[s], x_m^{i+1,q+j}[s]\right) \in G$. Let $a = x_m^{i,j}[s]$ and $b = x_m^{i+1,q+j}[s]$, and let s_0 be the stage where we enumerated (a,b) into G. Following the proof of the first part of Lemma 2.22, we can fix i_0, j_0, j_0' and m_0 such that $a = x_{m_0}^{i_0,j_0}[s_0]$ and $b = x_{m_0}^{i_0+1,j_0'}[s_0]$, and $|j_0|, |j_0'| < h_1(s_0+1)$. We also have $a^* = x_{m_0}^{i_0}[s_0]$ and $b^* = x_{m_0}^{i_0+1}[s_0]$. Thus, $s_0 \leq t$.

By Lemmas 2.17 and 2.21, either $d(a, a^*, s) = d(a, a^*, s_0)$ or $d(a, a^*, s) < \max\{k, a^*\} + 1$. (We argued similarly in Lemma 2.22). Since $d(a, a^*, s_0) < h_1(s_0 + 1)$, this means that $d(a, a^*, s) < h_1(t+1)$. The same holds for b and b^*. Thus, we have $|j| = d(a, a^*, s) < h_1(t+1)$ and $|q + j| = d(b, b^*, s) < h_1(t+1)$, which means that $q < 2h_1(t+1)$, contradicting the choice of q. □

Lemma 2.24. *If $\mathcal{L} \cong \mathcal{L}_e$ then $f(x) = \lim_{s\to\infty} f(x,s)$ is a strongly nontrivial automorphism of $\mathcal{L}_e$.*

Proof. In this lemma we do not worry about the complexity of f; this is taken care of in Lemma 2.25. Since we assume that $\mathcal{L} \cong \mathcal{L}_e$ we have $\lim_{s\to\infty} g(s) = \lim_{s\to\infty} h_1(s) = \infty$. Thus for each $x \in Orb$, the x-module eventually becomes active. By Lemma 2.19, there is m and i such that $x = x_m^i$ eventually. By Lemma 2.16, x_m^{i+1} is eventually defined and stable. Eventually the cycle cannot be split at x_m^i or x_m^{i+1}. By Lemma 2.23 we get to act infinitely often for $x = x_m^i$. Since we never split the cycle at x_m^i or x_m^{i+1}, we are always in case 1 of §2.2.7, and thus the value of $q = d\left(f(x), x_m^{i+1}\right)$ is eventually stable. Let q be the final value of $q = d\left(f(x), x_m^{i+1}\right)$. Thus $\lim_{s\to\infty} f(x,s) = x_m^{i+1,q}$ exists, as the order type of each $\mathcal{L}_e$-block is ζ.

We write $x \sim y$ to denote that x and y are in the same $\mathcal{L}_e$-block. Now fix an arbitrary element x and by Lemma 2.18 there is some $y \in Orb$ such that $x \sim y$.

Let $y_1 \in Orb$ and $p > 0$ be such that $\lim_{s\to\infty} f(y, s) \sim y_1$ with a distance of p away from y_1. Without loss of generality, assume that $x > y$. Since $\mathcal{L}_e$ also has order type $\zeta \cdot \eta$, the block containing y_1 has order type ζ, and so we can certainly find some $x_1 > y_1$ such that $d(x_1, y_1) = d(x, y) + p$, and hence $\lim_{s\to\infty} f(x, s) = x_1$. This shows that $f(x) = \lim_{s\to\infty} f(x, s)$ exists for every $x \in \mathcal{L}_e$, and if $x \in \mathcal{L}_e$ and $y \in Orb$ are in the same block, then $f(x) \sim f(y)$, and $x < y$ if and only if $f(x) < f(y)$. This obviously generalizes to any pair of elements $x, y \in \mathcal{L}_e$ in the same block.

The orbits satisfy the order-preserving condition, because the construction ensures that at every stage s, $Orb_m(s)$ and $Orb_n(s)$ are consistent for all m, n and s. Now we claim that for any pair of elements x, y, if $x < y$ then $f(x) < f(y)$. If $x \sim y$ then we have verified this above, so we assume that x and y are in different blocks. By Lemma 2.18 there are $x_1, y_1 \in Orb$ such that $x_1 \sim x$ and $y_1 \sim y$. Obviously $x_1 \neq y_1$ as $x \nsim y$. Since $x < y$ we must have $x_1 < y_1$. By Lemma 2.19, there are m and n such that $x_1 \in Orb_m$ and $y_1 \in Orb_n$.

If $n = m$ then as $x_1 < y_1$, there is $t > 0$ such that $f^t(x_1) \sim y_1$, and consequently, $f^{t+1}(x_1) \sim f(y_1)$ hence $f(x_1) < f(y_1)$. As $f(x) \sim f(x_1)$ and $f(y) \sim f(y_1)$ and $f(x_1) \nsim f(y_1)$, we have $f(x) < f(y)$. So, we suppose that $n \neq m$, and in particular, $z_0 \nsim z_1$ for any pair of distinct elements from $\{x_1, y_1, f(x_1), f(y_1)\}$. If $f(x_1) < y_1$ then we have $f(x_1) < y_1 < f(y_1)$, and since they are all in different blocks, this means that $f(x) < f(y)$. Finally we assume that $f(x_1) > y_1$. Then by the consistency of Orb_m and Orb_n we have $x_1 < y_1 < f(x_1) < f(y_1)$, and thus we again have $f(x) < f(y)$.

The fact that f is surjective follows easily from the fact that the order type of each $\mathcal{L}_e$-block is ζ, and the order type of each orbit is ζ. Notice also that $f(x)$ is well-defined, as there is at most one $z \in Orb$ such that $d(x, z, s) < h_1(s)$ at every s, by Lemma 2.4. The fact that f is strongly nontrivial follows from Lemma 2.15. Thus we have verified that f is a strongly nontrivial automorphism of $\mathcal{L}_e$. □

Lemma 2.25. *If $\mathcal{L} \cong \mathcal{L}_e$ then f has a Π^0_1 graph.*

Proof. Let $f(x) = \lim_{s\to\infty} f(x, s)$ for each $x \in \mathcal{L}_e$. We show that $f(x) = y$ if and only if $(x, y) \notin G$. Suppose $f(x) = y$. Then fix a stage s_0 such that we define $f(x, s_0) = y$ under §2.2.7, and for every stage after s_0, whenever we make a definition of $f(x, s)$ it is always equal to y. Suppose that $(x, y) \in G$, and $x \sim x^*$ where $x^* \in Orb$. Then for a large enough stage $s > s_0$ we will see $(x, y) \in G[s]$ and act for x^* under §2.2.7 (which is guaranteed by Lemma 2.23). We also assume $h_1(s+1)$ is large enough so that $d(x, x^*, s) < h_1(s+1) - q$, where q is the parameter corresponding to $f(x^*)$. At this point we will define $f(x, s) = y$, but as $(x, y) \in G[s]$, the instructions in case 1 of §2.2.7 will prevent us from doing so, contradicting the fact that we will act for x^* at s.

Now suppose $f(x) \neq y$. Let $x \sim x^*$ and $y \sim y^*$ where $x^*, y^* \in Orb$. By Lemma 2.23 we act for x^* infinitely often under case 1. Eventually when s and $h_1(s+1)$ are large enough, we will put (x, y) in G. □

This ends the proof of a single requirement. In the next section, we will handle all requirements by performing the construction in this section uniformly, so we will now give some remarks about the effectiveness of the construction for a single requirement.

An important observation is that the index for $\mathcal{L}$ can be effectively computed from an index for $\mathcal{L}_e$ and the parameter k, and does not depend, for example, on whether $\mathcal{L}_e \cong \mathcal{L}$. The order type of $\mathcal{L}$ will of course depend on whether or not $\mathcal{L}_e \cong \mathcal{L}$, but not the index of $\mathcal{L}$.

In fact, if we examine the construction in this section, we will see that §2.2.1 uses only g and does not refer to the other parameters $W_s, h_1, Orb, Cand, w_s, f$ and G defined in the rest of the sections §2.2.2 to §2.2.7. Similarly, sections §2.2.2 to §2.2.7 refer only to $\mathcal{L}_e$ and defines the parameters $g, W_s, h_1, Orb, Cand, w_s, f$ using only $\mathcal{L}_e$. Therefore, we could view the construction as consisting of two independent parts. The first part produces the parameters $g, W_s, h_1, Orb, Cand, w_s, f$ effectively from an index for $\mathcal{L}_e$. The second part produces the computable linear order $\mathcal{L}$ effectively from g.

Formally, the construction in this section produces total computable functions μ_0 and μ_1 with the following properties. If g is (an index for) a total computable function then $\mathcal{M}_{\mu_1(g)}$ is a computable linear ordering such that:

(i) If $\liminf\limits_{s\to\infty} g(s) = m < \infty$ then $\mathcal{M}_{\mu_1(g)} \cong m \cdot \eta$.

(ii) If $\liminf\limits_{s\to\infty} g(s) = \infty$ then $\mathcal{M}_{\mu_1(g)} \cong \zeta \cdot \eta$.

This follows from Lemmas 2.5 and 2.13.

Furthermore, for any e and k, $g_{\mu_0(e,k)}$ is a total computable function with the following properties:

(i) $g_{\mu_0(e,k)}(s) \geq k$ for every s.

(ii) If $\liminf\limits_{s\to\infty} g_{\mu_0(e,k)}(s) < \infty$ then $\mathcal{M}_{\mu_1\left(g_{\mu_0(e,k)}\right)} \not\cong \mathcal{L}_e$.

(iii) If $\mathcal{M}_{\mu_1\left(g_{\mu_0(e,k)}\right)} \cong \mathcal{L}_e$ then $\mathcal{L}_e$ has a strongly nontrivial Π^0_1-automorphism.

The first item follows from the fact that $g(x,s)$ is either undefined, or $g(x,s) \geq \max\{k,x\}$ for every x and s. The second item follows from Lemma 2.14, and the third item follows from Lemmas 2.24 and 2.25.

During the construction we had made some assumptions about speeding up the enumeration of $\mathcal{L}_e$ to search for confirmation that $\mathcal{L}_e$ has no greatest or least element. It could be that $\mathcal{L}_e$ does have a greatest or a least element, or could even be finite. (For the sake of uniformity, we must explain what we do in these cases). In this case the construction waits forever for the evidence it needs but never finds, and we will never take another nontrivial step in the construction, and never update the control parameters g and h_1. In this case $\liminf\limits_{s\to\infty} g_{\mu_0(e,k)}(s)$ will end up being finite, and obviously $\mathcal{M}_{\mu_1\left(g_{\mu_0}\right)} \not\cong \mathcal{L}_e$. Thus, the properties above still hold even if $\mathcal{L}_e$ is finite or has a greatest or a least element.

2.4. **Handling all requirements.** In this section we will complete the proof of the first main theorem and construct a computable linear order $\mathcal{L}_{\texttt{final}}$ satisfying all requirements. $\mathcal{L}_{\texttt{final}}$ will be of the form

$$1 + S_1 + 1 + \mathcal{M}_1 + 1 + S_2 + 1 + \mathcal{M}_2 + 1 + S_3 + \cdots + S_e + 1 + \mathcal{M}_e + 1 + S_{e+1} + \cdots,$$

where $S_e \cong (e+1)\cdot\eta$ and $\mathcal{M}_e$ will be built using μ_0 and μ_1 from the previous section. We will require that the blocks in $\mathcal{M}_e$ have size at least $e+3$, which we can ensure by the parameter k. Obviously, the intervals $1 + S_e + 1$ serve as separators. As these separators are "static", their locations can be found in any copy $\mathcal{L}_e$ of $\mathcal{L}_{\texttt{final}}$ in a Σ^0_3 way. This will allow us to run the basic construction and guess for the corresponding interval in $\mathcal{L}_e$.

We will satisfy the requirement corresponding to $\mathcal{L}_e$ inside the interval $\mathcal{M}_e$ of $\mathcal{L}_{\texttt{final}}$. However, if $\mathcal{L}_e$ is a copy of $\mathcal{L}_{\texttt{final}}$ then it will also contain copies of $\mathcal{M}_k$ for $k \neq e$. In order for us to meet the requirement we shall need to guess for the interval corresponding to $1 + \mathcal{M}_e + 1$ inside $\mathcal{L}_e$. Given any $\mathcal{L}_e$ which is isomorphic to $\mathcal{L}_{\texttt{final}}$, let $a_0^e <_{\mathcal{L}_e} a_1^e <_{\mathcal{L}_e} \cdots$ be exactly all the elements of block size 1 in $\mathcal{L}_e$. Obviously, any isomorphism between $\mathcal{L}_{\texttt{final}}$ and $\mathcal{L}_e$ has to fix this sequence, and consequently, $\left(a_{2i+1}^e, a_{2i+2}^e\right)_{\mathcal{L}_e} \cong \mathcal{M}_{i+1}$ and $\left(a_{2i}^e, a_{2i+1}^e\right)_{\mathcal{L}_e} \cong \mathcal{S}_{i+1}$ for every $i \geq 0$. For convenience, for each e such that $\mathcal{L}_e \cong \mathcal{L}_{\texttt{final}}$, we denote $\left(a_{2i+1}^e, a_{2i+2}^e\right)_{\mathcal{L}_e}$ by $\hat{\mathcal{M}}_{i+1}^e$ and $\left(a_{2i}^e, a_{2i+1}^e\right)_{\mathcal{L}_e}$ by $\hat{\mathcal{S}}_{i+1}^e$. Note that if φ is any isomorphism from $\mathcal{L}_{\texttt{final}}$ onto $\mathcal{L}_e$ then $\varphi\left(\mathcal{M}_{i+1}\right) = \hat{\mathcal{M}}_{i+1}^e$ and $\varphi\left(\mathcal{S}_{i+1}\right) = \hat{\mathcal{S}}_{i+1}^e$.

Lemma 2.26. *Suppose that $\mathcal{L}_{\texttt{final}} \cong \mathcal{L}_e$. Hence the sequence $a_e^1 <_{\mathcal{L}_e} a_e^2 <_{\mathcal{L}_e} \cdots$ exists. Then for any $b_0 <_{\mathcal{L}_e} b_1 \in \mathcal{L}_e$, $b_0 = a_{2e-1}^e$ and $b_1 = a_{2e}^e$ if and only if and only if $\left|[b_0]_{\mathcal{L}_e}\right| = \left|[b_1]_{\mathcal{L}_e}\right| = 1$ and there is a sequence $x_1 <_{\mathcal{L}_e} x_2 <_{\mathcal{L}_e} \cdots <_{\mathcal{L}_e} x_e <_{\mathcal{L}_e} b_0 <_{\mathcal{L}_e} b_1 <_{\mathcal{L}_e} x_{e+1}$ such that $\left|[x_i]_{\mathcal{L}_e}\right| = i+1$ for all $i = 1, \cdots, e+1$.*

Proof. We prove the nontrivial direction. Suppose that $\left|[b_0]_{\mathcal{L}_e}\right| = \left|[b_1]_{\mathcal{L}_e}\right| = 1$ and there is a sequence $x_1 <_{\mathcal{L}_e} x_2 <_{\mathcal{L}_e} \cdots <_{\mathcal{L}_e} x_e <_{\mathcal{L}_e} b_0 <_{\mathcal{L}_e} b_1 <_{\mathcal{L}_e} x_{e+1}$ such that $\left|[x_i]_{\mathcal{L}_e}\right| = i+1$.

Recall that for every i if $x \in \hat{\mathcal{M}}_i^e$ we had required that $|[x]_{\mathcal{L}_e}| > i+2$. Consequently, $x_{e+1} \notin \hat{\mathcal{M}}_i^e$ for $i \geq e$.

First of all, observe that $x_1 \in \hat{\mathcal{S}}_1^e$, because $|[x_1]_{\mathcal{L}_e}| = 2$ and this is the only interval which can contain a block of size 2. Similarly, $|[x_2]_{\mathcal{L}_e}| = 3$ and so $x_2 \in \hat{\mathcal{S}}_2^e$ because this is the only interval possible. Next $|[x_3]_{\mathcal{L}_e}| = 4$, and the only interval to the right of x_2 which can contain a block of size 4 is $\hat{\mathcal{S}}_3^e$. Continuing this way, we see that $x_i \in \hat{\mathcal{S}}_i^e$ for $i = 1, 2, \cdots, e, e+1$. Since the only pair of elements between x_e and x_{e+1} with a block size of 1 are a_{2e-1}^e and a_{2e}^e, we conclude that $b_0 = a_{2e-1}^e$ and $b_1 = a_{2e}^e$. □

It is easy to see that the condition in Lemma 2.26 "$\left|[b_0]_{\mathcal{L}_e}\right| = \left|[b_1]_{\mathcal{L}_e}\right| = 1$ and there is a sequence $x_1 <_{\mathcal{L}_e} x_2 <_{\mathcal{L}_e} \cdots <_{\mathcal{L}_e} x_e <_{\mathcal{L}_e} b_0 <_{\mathcal{L}_e} b_1 <_{\mathcal{L}_e} x_{e+1}$ such that $\left|[x_i]_{\mathcal{L}_e}\right| = i+1$ for all $i = 1, \cdots, e+1$" is Σ_3^0. Therefore, fix a Π_2^0 predicate R such that $\exists w R\left(e, b_0, b_1, w\right)$ if and only if the condition above holds inside $\mathcal{L}_e$. Since R is a Π_2^0 predicate, fix a computable approximation of it such that $R\left(e, b_0, b_1, w\right)$ holds if and only if $R\left(e, b_0, b_1, w\right)[s] = 1$ for infinitely many s. Now let $\texttt{wt}_e$ be a function ($\texttt{wt}$ for "witness") such that $\texttt{wt}_e(s) = \min\left\{\langle b_0, b_1, w\rangle \mid R(e, b_0, b_1, w)[s] = 1\right\}$. If this is not defined we retain the previous value of $\texttt{wt}_e$. Now obviously for any e, if $\mathcal{L}_{\texttt{final}} \cong \mathcal{L}_e$ then $\liminf_{s\to\infty} \texttt{wt}_e(s)$ exists and outputs $\langle a_{2e-1}^e, a_{2e}^e, w\rangle$ for some w.

Fix an e. We now describe how to define $\mathcal{M}_e$. Our description will be effective in e and so we can use this to define the computable $\mathcal{L}_{\texttt{final}}$. For each possible witness $\sigma = \langle b_0, b_1, w\rangle$, we let $\nu\left(\langle b_0, b_1, w\rangle\right)$ be an index for the subordering of $\mathcal{L}_e$ restricted to the interval (b_0, b_1).

Now we use $\texttt{wt}_e$ to define the different values of k which we will later use to form $\mathcal{M}_e$. Each possible witness σ will get a parameter $k_\sigma(s)$. At the beginning, let $k_\sigma(0) = \langle\sigma\rangle + e + 3$. This ensures that the size of each block in $\mathcal{M}_e$ is at least $e+3$. At stage $s > 0$, and for each σ with $\langle\sigma\rangle > \texttt{wt}_e(s)$, we increase the value of $k_\sigma(s)$. Finally we define the function $\tilde{g}$ by the following. Let $\tilde{g}(s) = g_{\mu_0(\nu(\langle\sigma\rangle), k_\sigma(s))}(t+1)$, where $\sigma = \texttt{wt}_e(s)$, where t was the previous value of the input for $g_{\mu_0(\nu(\langle\sigma\rangle), k_\sigma(s))}$ we used to define $\tilde{g}$. Now we let $\mathcal{M}_e = \mathcal{M}_{\mu_1(\tilde{g})}$. Now $\tilde{g}(s) = g_{\mu_0(\nu(\langle\sigma\rangle),k_\sigma(s))}(t+1) \geq k_\sigma(s)$ by the first item in the list of properties of μ_0. But $k_\sigma \geq e+3$, which means that the size of each block in $\mathcal{M}_e$ is at least $e+3$, as required for Lemma 2.26 to be applied.

Lemma 2.27. *If $\mathcal{L}_{\mathtt{final}} \cong \mathcal{L}_e$ then $\mathcal{L}_e$ has a strongly nontrivial Π^0_1 automorphism.*

Proof. Since $\mathcal{L}_{\mathtt{final}} \cong \mathcal{L}_e$, then by our previous observation, $\liminf_{s\to\infty} \mathtt{wt}_e(s)$ exists and outputs $\left\langle a^e_{2e-1}, a^e_{2e}, w\right\rangle$ for some w. Let $\langle\sigma\rangle = \liminf_{s\to\infty} \mathtt{wt}_e(s)$. Then clearly $k = \lim_{s\to\infty} k_\sigma(s)$ exists. Then $\nu(\langle\sigma\rangle)$ is an index for a computable linear ordering isomorphic to $\mathcal{M}_e$.

Now we claim that $\liminf_{s\to\infty} \tilde{g}(s) = \liminf_{s\to\infty} g_{\mu_0(\nu(\langle\sigma\rangle),k)}(s)$. Consider s_0 large enough such that $k_\sigma(s) = k$ and $\mathtt{wt}_e(s) \geq \langle\sigma\rangle$ for every $s \geq s_0$. Let τ be such that $\langle\tau\rangle$ is the least with $\langle\tau\rangle > \langle\sigma\rangle$. Then as there are infinitely many stages such that $\mathtt{wt}_e(s) = \langle\sigma\rangle$, we know that $\lim_{s\to\infty} k_\tau(s) = \infty$.

Let T_0 be the set of s such that $s \geq s_0$ and $\mathtt{wt}_e(s) \neq \langle\sigma\rangle$. Then for every $s \in T_0$, we see that $\mathtt{wt}_e(s) \geq \langle\tau\rangle$ and so $k_{\mathtt{wt}_e(s)}(s) \geq k_\tau(s)$. By the first property of μ_0, we know that $\tilde{g}(s) \geq k_{\mathtt{wt}_e(s)}(s)$, and if $s \in T_0$ then this quantity is $\geq k_\tau(s)$. Thus, this tells us that $\liminf_{s\in T_0} \tilde{g}(s) = \infty$. Therefore, this tells us that $\liminf_{s\to\infty} \tilde{g}(s) = \liminf_{s\notin T_0} \tilde{g}(s) = \liminf_{s\to\infty} g_{\mu_0(\nu(\langle\sigma\rangle),k)}(s)$, as required.

This means that $\mathcal{M}_{\mu_1(\tilde{g})} \cong \mathcal{M}_{\mu_1\left(g_{\mu_0(\nu(\langle\sigma\rangle),k)}\right)}$, since the two functions have the same $\liminf$. Now since $\nu(\langle\sigma\rangle)$ codes a linear ordering isomorphic to $\mathcal{M}_e = \mathcal{M}_{\mu_1(\tilde{g})} \cong \mathcal{M}_{\mu_1\left(g_{\mu_0(\nu(\langle\sigma\rangle),k)}\right)}$, by the third item in the list of properties of μ_0, we see that $\mathcal{L}_e \upharpoonright \left(a^e_{2e-1}, a^e_{2e}\right)$ has a strongly nontrival Π^0_1 automorphism. This clearly extends to a strongly nontrivial Π^0_1 automorphism of $\mathcal{L}_e$ by fixing all points outside the interval $\left(a^e_{2e-1}, a^e_{2e}\right)$. □

3. The proof of the second main theorem

3.1. **Intuition and the organization of the construction.** As before we use the following notations: $\{L_e\}_{e\in\omega}$ is a family of c.e. subsets of $\mathbb{Q}$, $\mathcal{L}_e = \langle L_e, <_{\mathbb{Q}}\rangle$ and $L_{e,s}$ is an enumeration of L_e at stage s. We also fix a sequence of total computable functions $\{\varphi_e\}_{e\in\omega}$ such that if F is any Δ^0_2 function, then there exists an e such that for every x, $\lim_{s\to\infty} \varphi_e(x,s)$ exists and is equal $F(x)$.

To prove the theorem we shall satisfy the following requirements.

$$\begin{aligned} R_e : \text{If } F_e(x) := \lim_{s\to\infty} \varphi_e(x,s) &\text{ exists for every } x, \\ \text{and } F_e : L \to L_e &\text{ is an isomorphism, then } \exists f : L_e \to L_e \\ &\text{such that } f \text{ is a strongly nontrivial } \Pi^0_1 \text{ automorphism.} \end{aligned}$$

We shall ensure that L is strongly η-like with block size of at most 4. As in the proof of the first main theorem, there are no interactions between the different requirements. We will first describe how to satisfy each requirement in isolation. As the requirements are satisfied in a uniform way, to meet all requirements we simply put all the constructions together. However, unlike in the previous theorem where we had to guess in a Σ^0_3 way the corresponding interval of $\mathcal{L}_e$ where we are satisfying the requirement corresponding to $\mathcal{L}_e$, in this theorem we will not need to do so. This is because the requirement R_e assumes that $F_e : L \to L_e$ is an isomorphism, and therefore will tell us which part of $\mathcal{L}_e$ to look at when meeting R_e, even though $\mathcal{L}$ contains lots of other intervals devoted to other requirements.

As in the previous construction, we will construct $\mathcal{L} = \sum_{q\in\mathbb{Q}} I(q)$ to be a computable linear order. Each $I(q)$ will eventually have size $2, 3$ or 4, depending on what $\mathcal{L}_e$ does on $rng(F_e)$. Again we approximate $I(q)$ by a computable sequence $\{I_s(q)\}_{s\in\omega}$ and take $I(q) = \lim_{s\to\infty} I_s(q)$. Every interval $I_s(q)$ has a special pair of elements $U_1(q)$ and $U_2(q)$ which gives the block $I(q)$ its identity. This pair of elements is enumerated into $I(q)$ the very first time it becomes non-empty, and will never leave $I(q)$. All other elements of $I_s(q)$ may leave the interval but cannot rejoin. Hence, every $I(q) = \lim_{s\to\infty} I_s(q)$ has at least two elements.

Let $h(q) = q + 1$. (Actually, any nontrivial computable automorphism of $\mathbb{Q}$ will do). We call a rational number q *even* if $\lfloor q \rfloor$ is even, and *odd* if $\lfloor q \rfloor$ is odd. We let $Orb_i = \{n + p_i \mid n \in \mathbb{Z}\}$ where p_i is the i-th rational number in the interval $[0, 1)$. Obviously, $\{Orb_i\}_{i\in\omega}$ satisfies all of the orbit conditions in §2.1.2 and is meant to be the corresponding notion of Orb in the proof of the first main theorem. The only difference is that the orbits in this proof is a computable set and is fixed right at the start. Let $\hat{I}_s(q) = \{\varphi_e(x, s) \mid x \in I_s(q)\}$. Our automorphism f obviously intended to map $\hat{I}_s(q)$ to $\hat{I}_s(q + 1)$. However, f will not obviously be computable because the elements of $\hat{I}_s(q)$ will change. Our goal is to enumerate a c.e. set G ensure that G is the $Cograph(f)$.

We will also build the computable function $g(q, s)$, which represents the current size of $I_s(q)$ we wish to have. However, unlike the proof of the first main theorem, we shall not need the function h_1; the control of when to allow for an expansionary stage is much simpler in the current proof.

The way we will build $I_s(q)$ is the following. The first time the construction looks at $I_s(q)$ we will pick the two special elements $U_1(q)$ and $U_2(q)$ and enumerate them into $I_s(q)$. Then depending on whether q is even or odd, we will add additional elements to the left of $U_1(q)$ or to the right of $U_2(q)$. For example, if q is even and $g(q, s) = 4$ then we let $I_s(q)$ be the set $x_0 < x_1 < U_1(q) < U_2(q)$. Non-special elements are always added to the left of $U_1(q)$. If q is odd and $g(q, s) = 3$ for example, then we let $I_s(q)$ be $U_1(q) < U_2(q) < x_0$. Non-special elements are always added to the right of $U_2(q)$.

At every stage s we will grow $\mathcal{L}$ by updating each $I_s(q)$ according to $g(q, s)$, and growing a new $I_s(q')$. Let s^- be the largest expansionary stage before s. Then s may or may not be an expansionary stage depending on whether $\mathcal{L}_e$ has recovered on its diagram at stage s^-. If s is not an expansionary stage, then we simply grow $\mathcal{L}$ and do nothing else. If s is an expansionary stage then on top of growing $\mathcal{L}$ we shall also need to take various other actions, such as updating f. Details are in the formal construction.

In Figure 5, the special elements of $I_s(q-1), I_s(q)$ and $I_s(q+1)$ are denoted by $\bullet$, while the non-special elements are denoted by $\circ$. f is always defined in a way that matches up the elements of $\varphi_e(I_s(q))$ with the corresponding elements of $\varphi_e(I_s(q + 1))$. Note that special elements will be mapped to non-special ones, and vice versa. This design of "alternating special elements" is important for our strategy to work.

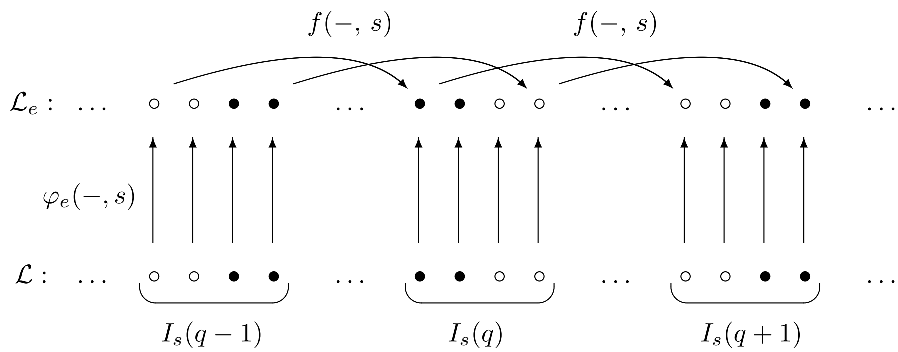

FIGURE 5. A diagram of mappings

We briefly describe the basic strategy at work here. The reader familiar with the workings of the first main theorem will have no trouble here. The basic strategy here is similar to the first main theorem, and much simpler. Refer to Figure 5. Let's call the elements of $I_s(q)$ (from left to right) $x_0 < x_1 < x_2 < x_3$, and the elements of $\varphi_e\left(I_s(q)\right)$ $y_0 < y_1 < y_2 < y_3$. Currently, $\varphi_e(x_i, s) = y_i$ and x_0, x_1 are the special elements of $I(q)$. Let's denote the corresponding elements over in the next block $I_s(q+1)$ the same, with x_i replaced by x_i' and y_i replaced by y_i'. So we have declared $f(y_i, s) = y_i'$ and also declared that $f(y_i) \neq y_j'$ for $j \neq i$. As with the previous proof, the difficulty of the construction comes down to enforcing these restrictions on f.

Suppose a new element y shows up between y_2 and y_3. This does not pose an immediate threat to f, because y is new and we can readily declare $f(y) = y_3'$, and since y_3 is now bumped out of the block $\varphi_e(I_s(q))$, it will be easy to redefine $f(y_3)$.

Now suppose y shows up between y_1 and y_2. Now the redefinition of y_2 is in danger, because according to the current configuration, we have to map y_2 to y_3', but we have already forbidden this definition. Therefore, we simply reduce the block size of $I(q)$ down to 3 and put more elements between x_2 and x_3. Provided that $f(\varphi_e(x_0)) = y_0$ and $f(\varphi_e(x_1)) = y_1$ and $f(\varphi_e(x_2)) = y$ do not change, we must force y_2 to leave the block, avoiding the need to redefine $f(y_2) = y_3'$.

Finally, suppose y shows up between y_0 and y_1. Then φ_e must change on a special element of $I(q)$. Everytime this happens, we will reduce the block size of *both* $I(q)$ and $I(q+1)$ down to 2. (For symmetrical reasons, we also reduce the size of $I(q-1)$, but this is irrelevant in this example). This will cause y_0' and y_1' to be replaced by fresh elements (never seen before in $\mathcal{L}_e$) which we can then take to be the new values of $f(\varphi_e(x_0))$ and $f\left(\varphi_e(x_1)\right)$.

3.2. **The formal construction for a single requirement.** In this section we fix a requirement R_e and describe the formal construction to meet R_e. We fix the associated $\mathcal{L}_e$. As in the proof of the first main theorem, the construction consists of several modules and are controlled by the main module. Since we fix e in this section we will not mention e when referring to the parameters of the construction.

3.2.1. *Construction of the linear order $\mathcal{L}$.* At stage $s = 0$, $I_0(q) = \emptyset$ for all $q \in \mathbb{Q}$. At stage $s+1$, we do the following. For all $q \in \mathbb{Q}$ such that the Gödel number of q is $\leq s$, we modify $I_{s+1}(q)$, according to the following two cases.

Case 1. $I_s(q)$ **is not empty:** Our actions for the block $I(q)$ will obviously depend on the current value of $g(q, s+1)$.

Subcase $g(q, s+1) = |I_s(q)|$**:** Then $|I_s(q)|$ is just right, and we set $I_{s+1}(q) = I_s(q)$.

Subcase $g(q, s+1) > |I_s(q)|$**:** Then we shall need to add elements to $I_s(q)$. Pick one or two new elements, x_1 or $x_1 < x_2$, depending on the value of $g(q, s+1) - |I_s(q)|$. If q is even we add x_1 and(or) x_2 immediately to the left of all elements of $I_s(q)$. If q is odd we add immediately to the right. After adding the new element(s) we have $I_{s+1}(q) = g(q, s+1)$, consisting of consecutive elements of $\mathcal{L}$.

Subcase $g(q, s+1) < |I_s(q)|$**:** Then we shall need to remove one or two elements depending on the value of $|I_s(q)| - g(q, s+1)$. If q is even we remove the least (or least two) elements of $I_s(q)$, and if q is odd we remove the greatest (or greatest two) elements of $I_s(q)$. After removing the new elements(s) we have $I_{s+1}(q) = g(q, s+1)$, and notice that this does not remove $U_1(q)$ or $U_2(q)$. The elements of $I_s(q) \setminus I_{s+1}(q)$ which are removed are of course removed from $I(q)$ but not from $\mathcal{L}$; we declare these elements *free*.

Case 2. $I_s(q)$ **is empty:** We would like to find a suitable pair of elements for $U_1(q)$ and $U_2(q)$. Search for $q_1, q_2 \in \mathbb{Q}$ such that $q_1 <_{\mathbb{Q}} q <_{\mathbb{Q}} q_2$, $I_s(q_1)$ and $I_s(q_2)$ are not empty and there is no q_3 such that $q_1 <_{\mathbb{Q}} q_3 <_{\mathbb{Q}} q_2$ and $I_s(q_3)$ is not empty. (Obviously, if q_1 or q_2 cannot be found we take it to be a point at infinity). Find the least x (in the standard order on $\mathbb{N}$) such that x *is free* and lies strictly between $I_s(q_1)$ and $I_s(q_2)$. If there is no such x then we add a new *free* element x between $I_s(q_1)$ and $I_s(q_2)$. Now add a new free element $x' > x$ as the successor of x. Define $U_1(q) = x$ and $U_2(q) = x'$. Populate the rest of $I_{s+1}(q)$ according to the instructions in Case 1.

Clearly, $|I(q)| = \liminf_{s\to\infty} |I_s(q)| = \liminf_{s\to\infty} g(q, s)$. Therefore, $\mathcal{L} = \sum_{q\in\mathbb{Q}} \liminf_{s\to\infty} g(q, s)$. It is strongly η-like with no rational subinterval because we will ensure that $2 \leq g(q, s) \leq 4$ for every q, s. Hence, $\mathcal{L}$ is of the desired type.

3.2.2. *The definition of an expansionary stage.* We call $s = 0$ an expansionary stage. For $s > 0$ we let s^- be the largest expansionary stage less than s. First define the set $Ind(s)$ to be all numbers of the form $q + n$ where $q \in \mathbb{Q}$ such that $0 \leq q < 1$ and the Gödel number of q is $< s$, and $n \in \mathbb{Z}$ such that $|n| < s$. We declare s to be an expansionary stage if and only if the following conditions hold.

- φ_e is currently order preserving on the set $\cup\{I_s(q) \mid q \in Ind(s^-)\}$.
- For each $q \in Ind(s^-)$, $\varphi_e\,(I_s(q), s)$ is an interval of $\mathcal{L}_e[s]$. That is, no other elements of $\mathcal{L}_e[s]$ lies strictly between two elements of $\varphi_e\,(I_s(q), s)$.
- For each $q \in Ind(s^-)$, $|I_s(q)| = 4$.

3.2.3. *The q-module.* The q-module is only active at a stage s if $q \in Ind(s^-)$. Otherwise, it is inactive, and doesn't do anything at stage s. Suppose the q-module is active at the current stage s. We call $U_1(q)$ and $U_2(q)$ the *special elements* of

$I(q)$, and the element $x \in I_s(q)$ the *secondary element* of $I(q)$ if x is the immediate predecessor of $U_1(q)$ if q is even, and the immediate successor of $U_2(q)$ if q is odd.

- **Case 1:** $\varphi_e(x,s) \neq \varphi_e(x,s-1)$ for some special element x of $I(q)$. Then we *request for* $g(q-1,s) = g(q,s) = g(q+1,s) = 2$.
- **Case 2:** The secondary element x of $I(q)$ exists and $\varphi_e(x,s) \neq \varphi_e(x,s-1)$. Then we request for $g(q-1,s) = g(q,s) = g(q+1,s) = 3$.
- **Case 3:** Otherwise. Then we check if the two elements immediately to the left of $\varphi_e(I_s(q-1))$, $\varphi_e(I_s(q))$ and $\varphi_e(I_s(q+1))$ and the two elements immediately to the right of $\varphi_e(I_s(q-1))$, $\varphi_e(I_s(q))$ and $\varphi_e(I_s(q+1))$ are enumerated after s^-. If all of these elements are new, we request for $g(q-1,s) = g(q,s) = g(q+1,s) = 4$. Otherwise we make the same requests as the previous stage.

3.2.4. *The definition of f.* We update f on $\varphi_e(I(q))$ only if s is an expansionary stage, $g(q-1,s) = g(q,s) = g(q+1,s) = 4$, $|I_s(q-1)| = |I_s(q)| = |I_s(q+1)|$ and $q \in Ind(s^-)$. Otherwise, as usual, we retain the previous value of $f(y)$ if there are no requests to update $f(y,s)$.

For each $x \in I_s(q)$ we define $f(\varphi_e(x,s),s)$ to be the corresponding element in $\varphi_e(I_s(q+1))$. That is, send the smallest element of $\varphi_e(I_s(q))$ to the smallest element in $\varphi_e(I_s(q+1))$, and so on.

For each $y \in \varphi_e(I_s(q))$ we also enumerate (y,z) into G_s for every $z \neq f(y,s)$ and $z \in \cup\{\varphi_e(I_s(q)) \mid q \in Ind(s^-)\}$.

3.2.5. *The main module for R_e.* At every stage we update $\varphi_e(x,s)$ at the beginning of stage s. The main module at stage $s > 0$ consists of the following steps.

1. Do step s of each active q-module (§3.2.3).
2. For each active q, define $g(q,s)$ to be the smallest requested value in the previous step. If q is not yet active, define $g(q,s) = 4$.
3. Do step s of the $\mathcal{L}$-construction (§3.2.1).
4. Do step s of the definition of f (§3.2.4).

3.3. **The formal verification for a single requirement.** First of all, notice that $g(q,s)$ is either $2, 3$ or 4 for every q and s. Furthermore, every x enumerated into $\mathcal{L}$ is eventually inside $I(q)$ for some fixed q. Since §3.2.1 is taken at every stage, it is clear that $\mathcal{L} = \sum_{q\in\mathbb{Q}} \liminf_{s\to\infty} g(q,s)$. Hence, $\mathcal{L}$ is of the desired type.

For the rest of the verification, we assume that $F_e = \lim \varphi_e(-,s)$ is total and is an isomorphism from $\mathcal{L} \to \mathcal{L}_e$.

Lemma 3.1. *For any q, $g(q,s) = 4$ for almost all s.*

Proof. If q is never active, then $g(q,s) = 4$ for all s. We first assume that there is some q such that $g(q,s) = 2$ for infinitely many s. This means that for some $q' = q-1, q$ or $q+1$, the q'-module will infinitely often request for $g(q,s) = 2$ under case 1 or 3. As the special elements of $I(q')$ are fixed, and $\varphi_e(x,s)$ must eventually stop changing on these special elements, this means that eventually the q'-module is stuck in case 3. That means that from some point on, $I_s(q'-1), I_s(q')$ and $I_s(q'+1)$ will consist of only its special elements. This means that $\varphi_e(I_s(q'-1))$, $\varphi_e(I_s(q'))$ and $\varphi_e(I_s(q'+1))$ are all eventually stable. By the construction of $\mathcal{L}$, $I_s(q'-1), I_s(q')$ and $I_s(q'+1)$ are maximal blocks of $\mathcal{L}$, which means that in order to get stuck in case 3, F_e cannot be surjective. Therefore, we conclude that for every q, $g(q,s) > 2$ eventually.

Now fix some q with infinitely many s such that $g(q,s) = 3$. Again let the q'-module be infinitely often responsible for requesting $g(q,s) = 3$ under case 2 or 3. Since $g(q',s)$ is eventually ≥ 3, this means that the secondary element of $I(q')$ is eventually stable and φ_e is also stable on it. Hence the q'-module is eventually stuck in case 3. Since $g(q'-1,s), g(q',s)$ and $g(q'+1,s)$ are eventually ≥ 3, this means that $I(q'-1), I(q')$ and $I(q'+1)$ are eventually stable with exactly three elements each (two special, one secondary). The same argument as above produces a contradiction. Hence we conclude that for any q, $g(q,s) = 4$ for almost all s. $\square$

Therefore, under the additional assumptions that $\mathcal{L} \cong \mathcal{L}_e$ via F_e, we in fact have $\mathcal{L} \cong 4 \cdot \eta$, and that for every q, $I_s(q)$ is eventually stable with four elements. Thus, it can be easily seen that there are infinitely many expansionary stages in the construction.

Lemma 3.2. *$f(y) = \lim_{s\to\infty} f(y,s)$ is a nontrivial automorphism of $\mathcal{L}_e$.*

Proof. Fix $y \in \mathcal{L}_e$. As F_e is bijective, there is some unique x such that $F_e(x) = y$, which means that $\varphi_e(x,s) = y$ for almost all s. But x has to eventually be in some $I(q)$ for some fixed q, and since there are infinitely many expansionary stages, q is eventually active. Hence $f(y,s)$ must be defined for a large enough s. Since $I_s(q+1)$ and $\varphi_e\left(I_s(q+1)\right)$ are eventually stable, $\lim_{s\to\infty} f(y,s)$ is also eventually stable. Thus, the function $f(y) = \lim_{s\to\infty} f(y,s)$ is total.

It is easy to see that f is order-preserving. If $y_1 < y_2$ then fix the corresponding q_1 and q_2 such that $y_i \in \varphi_e\left(I(q_i)\right)$. Since φ_e is order-preserving, this means that $q_1 \leq q_2$. If $q_1 = q_2$ then obviously $f(y_1) < f(y_2)$. If $q_1 < q_2$ then $q_1 + 1 < q_2 + 1$ and so $\varphi_e(I(q_1+1))$ lies to the left of $\varphi_e(I(q_2+1))$ and thus $f(y_1) < f(y_2)$. The surjectivity of f follows similarly. $\square$

Lemma 3.3. *f has a Π^0_1 graph.*

Proof. We need to show that $f(y) = z$ if and only if $(y,z) \notin G$. If $f(y) \neq z$ then at a suitably large expansionary stage s after y and z are both in the set $\cup\{\varphi_e\left(I_s(q)\right) \mid q \in Ind(s^-)\}$ and $f(y,s) \neq z$, we will enumerate (y,z) into G.

Now assume for a contradiction that there exists some $(y,z) \in G$ such that $f(y) = z$. Fix x_0, x_1, q_0 and q_1 such that $\varphi_e(x_0,s) = y$, $\varphi_e(x_1,s) = z$, $x_0 \in I_s(q_0)$ and $x_1 \in I_s(q_1)$ for almost all s. Notice that as $f(y) = z$, we must have $q_1 = q_0 + 1$. Since y and z occupy the same position in $\varphi_e\left(I(q_0)\right)$ and $\varphi_e\left(I(q_0+1)\right)$, it must be that either x_0 is a special element of $I(q_0)$, or x_1 is a special element of $I(q_0+1)$. Without loss of generality, assume that x_0 is a special element of $I(q_0)$. Thus x_1 is a non-special element of $I(q_0+1)$.

Let s_0 be the first time (y,z) is enumerated into G. In particular, s_0 is an expansionary stage and both $y, z \in \mathcal{L}_e[s_0]$. After s_0, we cannot have a change in $\varphi_e(x)$ where x is a special element of $I(q_0+1)$. This is because otherwise, we would set $g(q_0+1,s) = 2$ and thus at every expansionary stage after s_0, $\varphi(v,s)$ has to be a new element not in $\mathcal{L}_e[s_0]$ for each non-special element v of $I(q_0+1)$. In particular, this contradicts the property of z.

Therefore, after s_0, we also see that we cannot have a change in $\varphi_e(x_0)$. Otherwise, as x_0 is a special element of $I(q_0)$, this would cause $g(q_0+1,s) = 2$. Since

$\varphi_e(U_1(q_0+1))$ and $\varphi_e(U_2(q_0+1))$ are stable at s_0, this also means that at every expansionary stage after s_0, $\varphi(v,s)$ has to be a new element not in $\mathcal{L}_e[s_0]$ for each non-special element v of $I(q_0+1)$, contradicting the property of z.

Thus we conclude that at s_0, $\varphi_e(x_0)$, $\varphi_e(U_1(q_0+1))$ and $\varphi_e(U_2(q_0+1))$ are all stable. In particular, at s_0, $\varphi_e(x_0,s_0)=y$. At stage s_0, since we enumerated (y,z) into G, there are some $q'\in Ind(s_0^-)$ and $x'\in I_{s_0}(q')$ such that $z=\varphi_e(x',s_0)$. Obviously, $q'=q_0+1$, because otherwise as s_0 is an expansionary stage, we have $|I_{s_0}(q_0+1)|=4$ and so there are at least two elements between z and $\varphi_e(U_1(q_0+1))$ and $\varphi_e(U_2(q_0+1))$. Since the latter two values are already stable at s_0, z cannot possibly end up in $\varphi_e(I(q_0+1))$ later.

Thus we see that at s_0, $z=\varphi_e(x',s_0)$ for some $x'\in I_{s_0}(q_0+1)$. However, one of the conditions for enumerating (y,z) into G is that $f(y,s_0)\neq z$. Therefore, the only possibility is that at s_0, x' is the secondary element of $I(q_0+1)$. (Otherwise if x' is not the secondary element of $I(q_0+1)$ then z is not adjacent to a special element of $\varphi(I(q_0+1))$ and of course cannot later become adjacent). Now after s_0, we must have $g(q_0+1,s)\geq 3$, otherwise we would require for at least two new elements to show up between the special elements of $\varphi_e(I(q_0+1))$ and z, which contradicts the property of z. Therefore, x' stays forever as the secondary element of $I(q_0+1)$. This means that if $\varphi_e(x',s)$ does not change after s_0, then $f(y,s)$ cannot be equal to z later on. However, if $\varphi_e(x',s)$ does change after s_0 but *not* $\varphi_e(U_1(q_0+1))$ or $\varphi_e(U_2(q_0+1))$, then exactly one new element must appear between the special elements of $\varphi_e(I(q_0+1))$ and z. This causes $g(q_0+1,s)=3$ and thus before the next expansionary stage we would require further elements between $\varphi_e(I(q_0+1))$ and z, which again contradicts the property of z. □

3.4. **Handling all requirements.** The previous section is effective in the sense that given any pair $\mathcal{L}_e$ and F_e, we are able to produce $\mathcal{L}$ with the desired properties. Let's give this output $\mathcal{L}$ a different name, say $\mathcal{M}_e$. We can take our final linear order

$$\mathcal{L}_{\texttt{final}} \cong 1+2\cdot\eta+1+\mathcal{M}_1+1+2\cdot\eta+1+\mathcal{M}_2+1+2\cdot\eta+1+\cdots$$

where in each interval $\mathcal{M}_e$ we run the basic construction and play against the pair $\mathcal{L}_e$ and $F_e\restriction\mathcal{M}_e$. Unlike the proof of the first main theorem, we do not need to worry about the subinterval of $\mathcal{L}_e$ corresponding to $\mathcal{M}_e$, because $F_e\restriction\mathcal{M}_e$ will automatically pick it out for us, if F_e is to be trusted. Thus we will be able to build a nontrivial Π^0_1 automorphism of $\mathcal{L}_e\restriction F_e(\mathcal{M}_e)$. This obviously extends to an automorphism of $\mathcal{L}_e$ by taking the identity on the outside.

References

1. R. G. Downey. Computability Theory and Linear Orderings, in: *Handbook of Recursive Mathematics Vol. 2., Elsevier, Amsterdam*, (1998), pp. 823–976.
2. R. G. Downey, A. M. Kach, D. Turetsky. Limitwise monotonic functions and their applications, *Proceedings of the 11th Asian Logic Conference*, (2012), pp. 59–85.
3. R. G. Downey, M. F. Moses. On Choice Sets and Strongly Non-Trivial Self-Embeddings of Recursive Linear Orders, *Mathematical Logic Quarterly*, **35** (1989), pp. 237–246.
4. B. Dushnik, E. W. Miller. Partially Ordered Sets, *American Journal of Mathematics*, **63** (1941), pp. 600–610.
5. A. N. Frolov, M. V. Zubkov. Increasing η-Representable Degrees, *Mathematical Logic Quarterly*, **55** (2009), pp. 633–636.
6. A. N. Frolov, M. V. Zubkov. Limitwise Monotonic Functions Relative to the Kleene's Ordinal Notation System, *Lobachevskii Journal of Mathematics*, **35** (2014), pp. 295–301.

7. K. Harris. η-Represetation of Sets and Degrees, *Journal of Symbolic Logic*, **73** (2008), pp. 1097–1121.
8. C. Harris, K. Lee, S. B. Cooper. Automorphisms of η-Like Computable Linear Orderings and Kierstead's Conjecture, *Mathematical Logic Quarterly*, **62(6)** (2016), pp. 481–506.
9. A. M. Kach, D. Turetsky. Limitwise Monotonic Fuctions, Sets and Degrees on Computable Domaine, *The Journal of Symbolic Logic*, **75** (2010), pp. 131–154.
10. H. A. Kierstead. On Π^0_1-Automorphisms of Recursive Linear Orders, *Journal of Symbolic Logic*, **52** (1987), pp. 681–688.
11. J. Rosenstein. Linear Orderings, Academic Press, New York, (1982).
12. S. T. Schwarz. Quotient Lattices, Index Sets, and Recursive Linear Orderings, Ph.D. Thesis: University of Chicago, Department of Mathematics, (1982).
13. S. T. Schwarz. Recursive Automorphisms of Recursive Linear Orderings, *Annals of Pure and Applied Logic*, **26** (1984), pp. 69–73.
14. G. Wu, M. Zubkov. The Kierstead's Conjecture and limitwise monotonic functions. *Annals of Pure and Applied Logic*, **169(6)** (2018), pp. 467–486.
15. M. V. Zubkov. Sufficient conditions for the existence of $0'$-limitwise monotonic functions for computable η-like linear orders. *Siberian Mathematical Journal*, **58(1)** (2017), pp. 80–90.

Division of Mathematical Sciences, School of Physical and Mathematical Sciences, Nanyang Technological University, 21 Nanyang Link, Singapore 637371

N.I. Lobachevsky Institute of Mathematics and Mechanics, Kazan Federal University, Russia, Kazan, Kremlevskaya 18, 420008