Electronic Notes in Theoretical Computer Science 202 (2008) 89–99 

[www.elsevier.com/locate/entcs](http://www.elsevier.com/locate/entcs)

Effective Symbolic Dynamics

Douglas Cenzer[1](#_bookmark0)*,*[2](#_bookmark0)

*Department of Mathematics University of Florida*

*Gainesville, FL 32611-8105, USA*

S. Ali Dashti [3](#_bookmark0)

*Department of Mathematics, University of Florida*

Jonathan L.F. King [4](#_bookmark0)

*Department of Mathematics, University of Florida*

**Abstract**

We investigate computable subshifts and the connection with effective symbolic dynamics. It is shown that a decidable Π0 class *P* is a subshift if and only if there is a computable function *F* mapping 2N to 2N such that *P* is the set of itineraries of elements of 2N. A Π0 subshift is constructed which has no computable element. We also consider the symbolic dynamics of maps on the unit interval.

1

1

*Keywords:* Computability, symbolic dynamics, Π0 Classes.

1

# Introduction

Computable analysis studies the effective content of theorems and constructions in analysis. In this paper, we study computable dynamical systems and symbolic dynamics associated with computable functions on the Cantor space 2N and the unit interval [0*,* 1]. The papers of Gregorczyk [[8](#_bookmark12)] and Lacombe [[12](#_bookmark19)] which initiated the study of computable analysis provide the starting point of our study since those papers provide careful definitions of computably closed sets of reals and computable

1 Research partially supported by NSF grants DMS 0532644, 0554841 and 0652732

2 [cenzer@ufl.edu](mailto:cenzer@ufl.edu)

3 [mashadeo@math.ufl.edu](mailto:mashadeo@math.ufl.edu)

4 [squash@math.ufl.edu](mailto:squash@math.ufl.edu)

1571-0661 © 2008 Elsevier B.V. Open access under [CC BY-NC-ND license.](http://creativecommons.org/licenses/by-nc-nd/3.0/)

doi:10.1016/j.entcs.2008.03.010

real functions. Computable real dynamical systems have been studied by Cenzer [[2](#_bookmark9)], where the Julia set of a computably continuous real function is shown to be a Π0 class and Ko [[11](#_bookmark20)], who examined fractal dimensions and Julia sets. The computabil- ity of complex dynamical systems have recently been investigated by Rettinger and Weihrauch [[15](#_bookmark22)], and by Braverman and Yampolsky [[1](#_bookmark10)]. Weihrauch [[16](#_bookmark23)] has provided a comprehensive foundation for computability theory on various spaces, including the space of compact sets and the space of continuous real functions. The com- putability of symbolic dynamical systems has recently been studied by Delvenne [[6](#_bookmark13),[7](#_bookmark14)].

1

Effectively closed sets (Π0 classes) occur naturally in the application of com-

1

putability to many areas of mathematics. See the recent surveys of Cenzer and Remmel [[3](#_bookmark11),[4](#_bookmark15)] for many examples. In particular, the computability of a closed set *K* in a computable metric space (*X, d*) may be defined in terms of the distance func-

tion *dK*, where *dK*(*x*) is the infimum of *{d*(*x, y*) : *y ∈ K}*. *K* is a Π0 class if and

1

only if *dK* is upper semi-computable and *K* is a *decidable* (or *computable*) closed set if *dK* is computable. One important example in Euclidean space is the set of zeroes of a computably continuous function. This leads easily to related examples such as the set of fixed points or the set of extrema of a computably continuous function. That is, for any continuous function *F* , it is easy to see that the set of zeroes of *F* , the set of fixed points of *F* , and the set of points where *F* attains an extremum, are all closed sets. For a computably continuous function *F* , the corresponding closed sets are all Π0 classes. In fact, Nerode and Huang [[9](#_bookmark17)] showed that any Π0

1 1

class of reals may be represented as the set of zeroes of a computably continuous function. Ko extended the Nerode-Huang results [[10](#_bookmark18)] to show that any Π0 class

1

may be represented as the set of zeroes of a polynomial time computable function.

Thus Π0 classes also appear naturally in the theory of polynomial time computable functions on the reals.

1

The outline of this paper is as follows. Section 2 contains definitions and prelimi- naries. In section 3, we construct a subsimilar Π0 class with no computable element. The symbolic dynamics of effective dynamical systems on the Cantor space 2N is studied in section 4. For any finite *k*, the *shift* function on *{*0*,* 1*,..., k}*N is defined by *σ*(*x*) = *y*, where *y*(*n*) = *x*(*n* + 1). A closed set *Q ⊆ {*0*,* 1*,..., k}*N is said to be a *subshift* if it is closed under the shift function. We will refer to a Π0 class which is

1

1

also a subshift as a *subsimilar* Π0 class.

1

Fix a finite alphabet Σ, let *F* : ΣN *→* ΣN be a computable function and let a partition *{U*0*, U*1*,..., Uk}* of ΣN into clopen sets be given. The *itinerary* of a point *x ∈* ΣN is the sequence *It*(*x*) *∈ {*0*,* 1*,..., k}*N where

*It*(*x*)(*n*) = *i ⇐⇒ Fn*(*x*) *∈ Ui.*

Now let *IT* [*F* ] = *{It*(*x*) : *x ∈* ΣN*}*. We show that *IT* [*F* ] is a decidable subsimilar

0 class and that, for any decidable subsimilar Π0 class *Q ⊆* ΣN, there exists a

Π

1

1

computable *F* such that *Q* = *IT* [*F* ].

In section 5, we briefly examine symbolic dynamics for functions on the unit interval and pose some problems for future study. The class of unimodal maps and their interesting dynamical properties play an important role in many areas

of modern science and have undergone thorough investigation both experimentally and theoretically. (See for example [[5](#_bookmark16)].) One of the earliest papers to study the symbolic dynamics of unimodal and other mappings is [[14](#_bookmark21)].

# Preliminaries

We begin with some basic definitions. Let N = *{*0*,* 1*,* 2*, ...}* denote the set of natural numbers. For any set Σ, Σ*∗* denotes the set of finite strings of elements from Σ and ΣN denotes the set of countably infinite sequences from Σ. We write Σ*<n* for

*i<n* Σ*i*. For any set *A*, we let *card*(*A*) denote the cardinality of the set *A*.

For a string *w* = (*w*(0)*, w*(1)*,..., w*(*n −* 1)), *|w|* denotes the length *n* of *w*. The shift function on strings is defined by *σ*(*w*) = (*w*(1)*,..., w*(*n −* 1)). The empty string has length 0 and will be denoted by *λ*. A string of *n k*’s will be denoted

*kn*. For *m < |w|*, *w[m* is the string (*w*(0)*,..., w*(*m −* 1)). We say *w* is an *initial*

*segment* of *v* (written *w ≤ v*) if *w* = *v[m* for some m. Given two strings *v* and *w*, the *concatenation v-w* is defined by

*v-w* = (*v*(0)*, v*(1)*,..., v*(*m −* 1)*, w*(0)*, w*(1)*,..., w*(*n −* 1))*,*

where *|v|* = *m* and *|w|* = *n*. For *a ∈* Σ, we write *w-a* (or just *wa*) for *w-*(*a*) and we write *a-w* (or just *aw*) for (*a*)*-w*. For any *x ∈* Σ*ω* and any finite *n*, the *initial segment x[n* of *x* is (*x*(0)*,..., x*(*n −* 1)). For a string *w ∈* Σ*∗* and any *x ∈* ΣN, we write *w ≺ x* if *w* = *x[n* for some *n*. For any *w ∈* Σ*n* and any *x ∈* ΣN, we let *w-x* = (*w*(0)*,..., w*(*n −* 1)*, x*(0)*, x*(1)*,.. .*). The shift function on Σ*N* is defined by *σ*(*x*) = (*x*(1)*, x*(2)*,.. .*).

The topology on ΣN has a basis of *intervals*, which are clopen sets of the form

*J* [*w*] = *{x* : *w ≺ x}.*

A subset of ΣN is clopen if and only if it is a finite union of intervals.

A *tree T* over Σ*∗* is a set of finite strings from Σ*∗* which contains the empty string *λ* and which is closed under initial segments. We say that *w ∈ T* is an *immediate successor* of *v ∈ T* if *w* = *va* for some *a ∈* Σ. We will assume that Σ *⊆* N, so that *T ⊆* N*∗*. For any tree *T* , an *inﬁnite path* through *T* is a sequence (*x*(0)*, x*(1)*,.. .*) such that *x[n ∈ T* for all *n*. We let [*T* ] denote the set of infinite paths through *T* . It is well-known that a subset *Q* of ΣN is closed if and only if *Q* = [*T* ] for some

tree *T* . A subset *P* of NN is a Π0 *class* (or *effectively closed set* ) if *P* = [*T* ] for

1

some computable tree *T* . A node *w ∈ T* is *extendible* if there exists *x ∈* [*T* ] such

that *w ≺ x*. The set of extendible nodes forms a tree *TP* which is a co-c.e. subset of Σ*∗* but is not in general computable. *P* is said to be *decidable* (or *computable*) if *TP* is a computable set. A tree *T ⊆* Σ*∗* said to be *subsimilar* if for every *v* and *w*, *v-w ∈ T* implies *w ∈ T* . The closed set *P* is subsimilar (or a subshift) if *TP* is subsimilar. Equivalently *P* is a subshift if it is closed under the shift function.

A function *F* : ΣN *→* ΣN is computable (or *computably continuous*) if there exists a computable approximating function *f* : Σ*∗ →* Σ*∗* such that, for all *x ∈* ΣN and all *v, w ∈* Σ*∗*:

* 1. *v ≺ w −→ f* (*v*) *≤ f* (*w*).
  2. (*∀m*)(*∃n*)(*∀v ∈* Σ*n*)*|f* (*v*)*|≥ m*.
  3. *F* (*x*) = *n f* (*x[n*).

Note that (iii) implies (ii) by compactness.

A function *F* : [0*,* 1] *→* [0*,* 1] is computable if there exists a computable ap- proximating sequence *⟨fn⟩* of functions *fn* : *D → D* (where *D* is the set of dyadic rationals in [0*,* 1]) and a computable modulus function *M* : N *→* N such that for all *n ∈* N, all *d ∈ D* and all *x ∈* [0*,* 1],

*|x − d| <* 2*−M*(*n*) *−→ |F* (*x*) *− fn*(*d*)*| <* 2*−n.*

# Undecidable subshifts

In this section, we construct a subsimilar Π0 class with no computable element. We will give the construction in 2N, but it can be generalized to ΣN for any finite Σ. Now every decidable Π0 class has a computable element (in fact, the leftmost path is computable). Hence we have an undecidable subsimilar Π0 class.

1

1

1

Let us say that a string *v* is a *factor* of a string *w* if there exist *w*1 and *w*2 such that *w* = *w*1*-v-w*2. For any set *S* of strings, we may define a closed set *PS*, where *x ∈ PS* if and only if, for all *n* and all *w ∈ S*, *w* is not a factor of *x[n*. If the set *PS* is nonempty, then *S* is said to be *avoidable*. For this section, we restrict ourselves to Σ = *{*0*,* 1*}*

**Lemma 3.1** *Given any sequence x*0*, x*1*,... of elements of* 2N*, there is a nonempty subshift containing no xi.*

**Proof.** Define the sequence *l*0*, l*1*,...* by

*ln* = 3(2*n*(*n*+3))*.*

This will imply that *ln*+1 = 22*n*+4*ln*. Now let *wn* = *xn[*2*ln* for each *n* and define subshift *P* to consist of all *x* which do not contain any *wn* as a factor. Clearly *xi ∈/ P* for all *i*. It remains to show that *P* is nonempty, that is, *{wn* : *n ∈* N*}* is avoidable.

It is important to notice that given any word *w* of length 2*k*, it has at most *k* +1 distinct factors of length *k*. Since there are 2*k* words of length *k*, for *k* large enough so that 2*k > k* + 1, there are words of length *k* that do not appear as a factor of

*w*. With this in mind, we construct recursively two sequences of words *< An >n∈*N

and *< Bn >n∈*N such that, for all *n*:

* 1. *|An|* = *|Bn|* = *ln*;
  2. *An /*= *Bn*;
  3. *An* and *Bn* are not factors of *wn*; this is possible for *n* = 0 since *|w*0*|* = 6 so *w*0

has at most 4 distinct factors of length 3.

* 1. *An*+1 and *Bn*+1 are taken from *{An, Bn}∗*, have *An* as a prefix, and have length *m* = 22*n*+4 = *ln*+1*/ln*. This is possible since there are 2*m−*1 such words, but there are at most *ln*+1+1 factors of length *ln*+1 in *wn*+1 and 2*m−*1 ≥ *ln*+1+1+2.

To check this, note that *l*

*n*+1

+ 3 = 3(2*n*2+3*n*) + 3 *≤* 2*n*2+3*n*+2 *≤* 2*m−*1, since

*m −* 1 = 22*n*+4 *−* 1 *≥* 22*n*+3 *≥ n*2 + 3*n* + 2 for all *n*.

Now let *x* = *limnAn*. This exists since each *An ≺ An*+1. We claim that *x ∈ P* . Suppose by way of contradiction that some *wn* is a factor of *x*. We can view *x* as an infinite concatenation of blocks length *ln*, where each block is either *An* or *Bn*. Since *wn* has length 2 *ln*, it must completely contain one of the blocks, which would imply that either *An* or *Bn* is a factor of *wn*. This contradiction shows that *x ∈ P* .

We need to improve this lemma in two ways. First, we may be avoiding a set of words *wnk* of length 2*lnk* for a subset *{nk* : *k ∈* N*} ⊂* N. Second, we need an effective version of the construction to obtain a Π0 class.

1

**Theorem 3.2** *There is a recursive sequence of natural numbers l*0*, l*1*,... such that if for any subsequence < lnk >k∈*N *and any set S* = *{vk* : *k ∈* N*} of words such that |vk|* = 2*lnk , S is avoidable. Furthermore, if φ is a partial computable function such that φ*(*nk*) = *vk, then there is a nonempty subsimilar* Π0 *class P such that no*

1

*element of P contains any factor vk.*

**Proof.** For the first part, simply let *wnk* = *vk* and choose arbitrary words *wi* of length 2*li* for *i ∈/ {nk* : *k ∈* N*}* and apply the lemma.

For the second part, we have

*x ∈ P ⇐⇒* (*∀n*)(*∀k*)[*vk* is not a factor of *x[n*]

**Theorem 3.3** *There is a nonempty subsimilar* Π0 *class P with no computable el-*

1

*ement.*

**Proof.** Let the sequence *< ln >* be given as in Lemma [3.1](#_bookmark1). Let *φ*0*, φ*1*, ..., φe,...* be an enumeration of partial computable functions.

Now define the partial recursive function *φ* so that *φ*(*k*) = *φk[*2*lk*, if *φk*(*i*) *↓* for all *i <* 2*lk*, and *φ*(*k*) is undefined otherwise.

By Theorem [3.2](#_bookmark2), there is a nonempty subsimilar Π0 class *P* such that no element

1

of *P* has any word *φ*(*k*) as a factor. Now let *y* be any computable element of 2N. Then *y* = *φk* for some *k* such that *φk* is a total function. Thus *φ*(*k*) = *φk[k* is defined and is not a factor of any *x ∈ P* and hence certainly *φk ∈/ P* .

# Symbolic Dynamics for Functions on ΣN

Fix a finite alphabet Σ = *{*0*,* 1*,..., k}*, let *F* : ΣN *→* ΣN be a computable function and let a partition *{U*0*, U*1*,..., Uk}* of ΣN into clopen sets be given. The *itinerary* of a point *x ∈* ΣN is the sequence *It*(*x*) *∈ {*0*,* 1*,..., k}*N where

*It*(*x*)(*n*) = *i ⇐⇒ Fn*(*x*) *∈ Ui.*

Now let *IT* [*F* ] = *{It*(*x*) : *x ∈* ΣN*}*. We observe that *IT* [*F* ] is a subshift. That is, suppose *y* = *It*(*x*) *∈ IT* [*F* ]. Then *σ*(*y*) = *It*(*F* (*x*)), so that *σ*(*y*) *∈ IT* [*F* ] as well. The function *It* is continuous and hence *IT* [*F* ] is a closed set, as seen by the proof of the following lemma.

**Lemma 4.1** *The function from* ΣN *→ {*0*,* 1*,..., k}*N *mapping x to I*(*x*) *is com- putable.*

**Proof.** Given clopen sets *U*0*,..., Uk*, there exists a finite *j* and a finite subset *W* of

*{*0*,* 1*}j* such that each *Ui* is a finite union of intervals *J* [*w*] for some set of *w ∈ W* . Thus one can determine from *y[j* the unique *i* for which *y ∈ Ui*. Given *x ∈* ΣN, let *y* = *I*(*x*). To compute *y*(*n*), it suffices to find the first *j* values of *Fn*(*x*), which can be computed uniformly from *x* and *n*.

**Theorem 4.2** *Let F* : ΣN *to* ΣN *be computable and let {U*0*, U*1*,..., Uk} be a parti- tion of* ΣN *into clopen sets. Then*

1. *For any computable x ∈* ΣN*, the itinerary It*(*x*) *is computable.*
2. *The set IT* [*F* ] *of itineraries is a decidable, subsimilar* Π0 *class.*

1

**Proof.** Part (a) follows from the well-known result that computable functions map computable points to computable points and (b) follows from the fact that the image of a decidable Π0 class under a computable function is a decidable Π0 class.

1 1

See [[3](#_bookmark11),[4](#_bookmark15)].

Next we prove the converse. Note that *F* 0(*x*) = *x* for all *x ∈* ΣN and therefore *IT* [*F* ] meets every *Ui*. Note that if *Q* is a subshift and *Q* does not meet *J* [*i*], then *Q ⊆ {*0*,* 1*,...,i −* 1*,i* + 1*,..., k}*N.

**Theorem 4.3** *Let* Σ = *{*0*,* 1*,..., k} be a ﬁnite alphabet and let Q ⊆* ΣN *be a decidable, subsimilar* Π0 *class which meets J* [*i*] *for all i. Then there exists a partition*

1

*{U*0*,..., Uk} and a computable F* : ΣN *→* ΣN *such that Q* = *IT* [*F* ]*.*

**Proof.** We will use the partition given by *Ui* = *J* [*i*]. Since *Q* is decidable, we can define a function *G* : ΣN *→ Q* such that *G*(*x*) = *x* for all *x ∈ Q*. Let *Q* = [*T* ] where *T* is a computable tree without dead ends. The approximating function *g* for *G* is defined as follows. For any *w ∈ {*0*,* 1*,..., k}n*, find the longest initial segment *v* such that *v ∈ T* and let *g*(*v*) be the lexicographically least (or leftmost) extension of *v* which is in *T ∩ {*0*,* 1*,..., k}n*; this exists since *T* has no dead ends. Now let *F* (*x*) = *σ*(*G*(*x*)). We claim that *IT* (*F* ) = *Q*.

For any *x ∈ Q*, we have *F* (*x*) = *σ*(*x*) and *σ*(*x*) *∈ Q*, since *Q* is a subshift. Hence *Fn*(*x*) = *σn*(*x*), so that *Fn*(*x*)(0) = *x*(*n*), so that *Fn*(*x*) belongs to the set *Ux*(*n*). Thus the itinerary *I*(*x*) = *x*. This shows that *Q ⊆ IT* [*F* ].

Next consider any *x ∈* ΣN. We will show by induction that *Fn*(*x*) = *σn*(*G*(*x*)) for all *n >* 0. For *n* = 1, this is the definition. Then

*Fn*+1(*x*) = *σ*(*G*(*Fn*(*x*))) = *σ*(*G*(*σn*(*G*(*x*))))*,*

by induction. But *G*(*x*) *∈ Q*, so that *σn*(*G*(*x*)) *∈ Q* by subsimilarity and therefore *G*(*σn*(*G*(*x*))) = *σn*(*G*(*x*)) and finally *Fn*+1(*x*) = *σn*+1(*G*(*x*)), as desired. It follows that for *n >* 0, *It*(*x*)(*n*) = *G*(*x*)(*n*). But for *n* = 0, the assumption that *Q* meets *J* [*x*(0)] implies that *G*(*x*)(0) = *x*(0) and hence *It*(*x*)(0) = *x*(0) = *G*(*x*)(0) as well. Therefore *It*(*x*) *∈ Q* as desired.

# Unimodal Maps

In this section, we consider symbolic dynamics for mappings on the unit interval, which is much more complicated. We recall some definitions and facts of [[14](#_bookmark21)].

**Definition 5.1** A function *F* : [0*,* 1] *→* [0*,* 1] is a unimodal map with critical point

*c* if

* 1. *F* (*c*) is the unique absolute maximum of *F* ;
  2. *F* is strictly increasing over the interval [0*, c*) and is strictly decreasing over (*c,* 1].
  3. F(0)=0=F(1).

The value of the critical point is not essential for this discussion, so for simplicity

*c* is taken to be 1 . Given any *x ∈* [0*,* 1] the itinerary of *x* under *F* , *It*(*x*), is defined

2

as follows:

⎧⎪ 1 *Fi*(*x*) *>* 1 *,*

*It*(*x*)*i* =

2

2

*C Fi*(*x*) = 1 *,*

⎪⎨

⎪

⎩ 0 *Fi*(*x*) *<* 1

2

Hence the space of itineraries, *IT* [*F* ], of the elements of the interval [0*,* 1] is a symbolic subspace of *X* = *{*0*,* 1*, C}ω*. In contrast to the symbolic dynamics of ΣN, the function taking *x* to *It*(*x*) need not be continuous and the set of itineraries need not be a closed set. We are also interested in the subset of itineraries in 2N, that is,

*I*[*F* ] = 2N *∩ {It*(*x*) : *x ∈* [0*,F* (*c*)]*}.*

The most important itinerary of a unimodal map *F* is its *kneading sequence*, *KS*(*F* ), which is *It*(*F* ( 1 )). There is a connection between the kneading sequence and the set *I*[*F* ] by way of the following well-known linear ordering on *{*0*,* 1*, C}∗*. Here *C* represents the critical point 1 .

2

2

A word *w ∈ {*0*,* 1*, C}∗* is said to be *even* (respectively, *odd* ) if it has an even (odd) number of ones. The ordering is defined as follows.

* 0 *< C <* 1
* For any *w*1 and *w*2,

1. If *w*1 *≤ w*2, then *w*1 *≤ w*2 and vice versa.
2. Otherwise, let *u* be the largest common prefix of *w*1 and *w*2 and let *|u|* = *m*. If *u* is even, then *w*1 *< w*2 if and only if *w*1(*m*) *< w*2(*m*) and if *u* is odd, then *w*1 *< w*2 if and only if *w*1(*m*) *> w*2(*m*).

This ordering can be extended to *{*0*,* 1*, C}*N just using clause (ii). For finite words, this is clearly a computable linear ordering.

A finite word *w* is called *shift-maximal* if *σi*(*w*) *≤ w* for all *i* with 1 *≤ i ≤ |w|−*1.

Similarly an infinite word *x* is shift-maximal iff *σi*(*x*) *≤ x* for all *i*.

It is well-known that the kneading sequence *KS*(*F* ) for a unimodal map *F* is shift-maximal [[14](#_bookmark21)].

Given a shift-maximal *x ∈ {*0*,* 1*, C}*N, *y ∈* 2N is said to be *admissible* with respect to *x* if *σi*(*y*) *≤ x* for all *i*. Let *Adm*(*x*) be the set of all admissible sequences with respect to *x*.

**Theorem 5.2** *For any computable shift-maximal sequence x ∈ {*0*,* 1*, C}*N*, Adm*(*x*)

*is a decidable, subsimilar* Π0 *class.*

1

**Proof.** It is immediate from the definition that *Adm*(*x*) is a subshift. For the effectiveness, we have *Adm*(*x*) = [*T* ], where *w ∈ T ⇐⇒* (*∀i < |w|*)*σiw ≤ x[|w|*. [*T* ] is decidable since, for any *w ∈ T* , either *w-*0 or *w-*1 in *T* .

**Theorem 5.3** *Given any decidable, subsimilar* Π0 *subclass Q of the cantor space, there exists a shift maximal sequence, x ∈* 2*ω, such that Q ⊆ Adm*(*x*)*.*

1

**Proof.** Let *T* be a computable tree without dead ends such that *Q* = [*T* ] and such that *w ∈ T* implies *σ*(*w*) *∈ T* . Let *x*(0) = 1, since clearly (1) is shift-maximal. Suppose we have defined the shift-maximal word *s* = *x[n ∈ T* such that for all *w ∈ T ∩ {*0*,* 1*}≤n*, *w ≤ x[n*. Let *w ∈ T* be the maximal word of length *n* + 1. We claim that *s ≺ w*. To see this, let *w* = *v-i* for *v ∈ T* and suppose by way of contradiction that *s /*= *v*. Since *T* has no dead ends, *s-j ∈ T* for some *j*. Since *s* is shift-maximal, *v < s*, so that *w* = *v-i < s-j*, contradicting the assumption that *w* is maximal in *T* . Now let *x*(*n*) = *w*(*n*) so that *x[n* +1 = *w*. To see that *w* is shift-maximal, let *u* = *σi*(*w*) for some *i*. Then *u ∈ T* since *Q* is subsimilar and thus *u ≤ w* by maximality of *w* in *T≤n*+1. Proceeding in this fashion, we construct *x ∈ T* such that *x[n* is shift-maximal for all *n*, and hence *x* is shift-maximal. Also *x* is maximal in *Q*, so that, for any *y ∈ Q*, *y ≤ x*.

It is not the case that every decidable, subsimilar Π0 class *Q equals Adm*(*x*) for the maximal element *x* of *Q*. For example, if *x* = 10*ω* , then *Adm*(*x*) = 2N. However, for *S* = *{*111*}* and *Q* = *PS*, we have *x ∈ Q* and thus *x* is the maximal element of *Q*, but certainly *Q /*= *Adm*(*x*) = 2N.

1

We return to the analysis of unimodal maps. The connection between the knead- ing sequence and the itineraries is given by the following [[17](#_bookmark24)]. For a continuous function *F* , let *Adm*(*F* ) denote *Adm*(*KS*(*F* )).

**Proposition 5.4** *For any unimodal map F* : [0*,* 1] *→* [0*,* 1] *with kneading sequence*

*KS*(*F* )*, the set Adm*(*F* ) *is the closure of the set I*[*F* ]*.*

Next we consider computable unimodal maps.

**Theorem 5.5** *For any computable unimodal map F* : [0*,* 1] *→* [0*,* 1] *and any com- putable real x ∈* [0*,* 1]*, It(x) is a computable sequence. In particular, the kneading sequence KS*(*F* ) *is computable.*

**Proof.** Note that, for any computable *x*, *L*(*x*) = *{n* : *Fn*(*x*) *<* 1 *}* and *R*(*x*) =

2

*{n* : *Fn*(*x*) *>* 1 *}* are both c. e. sets. Suppose first that *Fn*(*x*) */*= 1 for any *n*.

2 2

Then *L*(*x*) and *R*(*x*) are complements and hence both sets are computable. Then

*I*(*x*)(*n*) = 0 *⇐⇒ n ∈ L*(*x*), so that *I*(*x*) is computable.

For the other case, we first consider the kneading sequence. If *Fn*( 1 ) = 1

2 2

for some *n*, then *KS*(*F* ) is periodic and certainly computable. Thus *KS*(*F* ) is computable in any case. Now for arbitrary *x ∈* [0*,* 1] such that *Fn*(*x*) = 1 for some *n*, then *I*(*x*) = *I*(*x*)*[n* + 1*-KS*(*F* ) and is therefore computable since *KS*(*F* ) is computable.

2

We have the following corollary to Theorems [5.2](#_bookmark3) and [5.5](#_bookmark4).

**Corollary 5.6** *For any computable unimodal map F* : [0*,* 1] *→* [0*,* 1]*, Adm*(*F* ) *is a decidable, subsimilar* Π0 *class.*

1

The remaining goal is to find a converse to this result, that is, given a decidable subsimilar Π0 class, to find a computable unimodal map *F* with *Adm*(*F* ) = *Q*. We will make some progress in this direction.

1

For the rest of the section, we confine the discussion of unimodal maps to the quadratic maps *Fμ*(*x*) = *μx*(1 *− x*). These form a so-called *full family*, so that, by [[17](#_bookmark24)], we have

**Proposition 5.7** *For any μ*0 *< μ*1 *and for every shift-maximal y ∈ {*0*,* 1*, C}*N*, there exists a parameter μ ∈* [*μ*0*, μ*1] *such that KS*(*Fμ*) = *y.*

For *μ >* 3, the unimodal map *Fμ* = *μx*(1 *− x*) has *Fμ*( 1 ) *>* 1 , so that there exist

2

points *x*0 *∈* (0*,* 1 ) and *x*1 *∈* ( 1 *,* 1) such that *Fμ*(*x*0) = 1

2

= *Fμ*(*x*1) and therefore

2 2 2

0*C ≺ I*(*x*0) and 1*C ≺ I*(*x*1). In general, *Fμ* may have *k*th order inverses of 1 for

2

all *k*.

It is clear that the surjective quadratic map *F*4 has 2*k k*-th order inverses of 1 for all *k* and hence for any *w ∈ {*0*,* 1*}∗*, there exists *x* such that *wC ≺ I*(*x*); we will say that this *x* is the *coordinate* of the path *wC*. For this map, we have the following.

2

**Lemma 5.8** *Let F* (*x*) = 4*x*(1*−x*) *and let It*(*x*) *be the itinerary of x under F. Then*

*It is one-to-one, I*[*F* ] = 2N*, and the inverse of It, restricted to* 2N*, is computable.*

For the surjective quadratic map *F* = *F*4, we say that *wC* is a *legal inverse path (l.i.p.)* if the coordinate *r ∈* [0*,* 1] of the path is the greatest numerical value of any point on the path, that is, if *Fn*(*r*) *< r* for *n ≤ |w|*. Metropolis et al [[14](#_bookmark21)] provides the following crucial fact.

**Proposition 5.9** *There is a one-to-one correspondence between the set of periodic kneading sequences of the full family of the quadratic maps and the the set of legal inverse paths of F*4*. In particular,* (*wC*)*ω is a periodic kneading sequence for some* *μ if and only if wC is a legal inverse path for F*4*. Moreover, if xw is the coordinate of wC and μw has kneading sequence* (*wC*)*ω, then in general xv < xw if and only if μv < μw.*

It follows from Proposition [5.9](#_bookmark6) that if *< wnC >* is a sequence of legal inverse paths such that the corresponding sequence *< xwn >n<ω* converges, then the se- quence *< μwn >n<ω* also converges. We also need the following corollary.

**Corollary 5.10** *For any ﬁnite string w, if wC is shift-maximal, then wC is a legal inverse path for F*4 *and there exists a μ such that* (*wC*)*ω* = *KS*(*Fμ.*

**Proof.** Suppose that *wC* is shift-maximal. Then (*wC*)*ω* is also shift-maximal, so that by Proposition [5.7](#_bookmark5) (*wC*)*ω* = *KS*(*Fμ*) or some *μ*. It follows by Proposition [5.9](#_bookmark6) that *wC* is a legal inverse path for *F*4.

We next give a condition for a finite word *w ∈ {*0*,* 1*}∗* which will imply that *wC* is shift-maximal if *w* is shift-maximal. Suppose that *w* is shift-maximal but *wC* is not shift-maximal. Then some *σi*(*wC*) *> wC*; let *σi*(*wc*) = *uC* and let *v* = *w[i*, so that *w* = *vu*. Then *wC < uC*. But we know that *u < w*, which implies that *uC < wC unless u* is a prefix of *w*. With this in mind, we say that for finite words *u* and *w*, *u /*= *w* is a *proper preﬁx-suffix* (in short *PS*) of *w*, if there exists words *t* and *v* such that *w* = *uv* and *w* = *tu*. Call a *PS u* of *w trivial* if *|u|* = 1; in this case *wC* will be shift-maximal. Note that given any shift-maximal sequence *x ∈* 2N, *x[n* is a shift-maximal word for all *n*, but it might have *PS* factors. Call *x ∈* 2N *strongly shift-maximal* if there is an increasing function *f* : N *→* N such that, for all *n*, *x[f* (*n*) has at most a trivial proper *PS*. For example, fix any *m* and let

*x* = 10*m*110*m*1110*m ...*

Then for each *n*, 10*m*11 *...* 0*m*1*nC* is shift-maximal. The argument above has proved the following.

**Lemma 5.11** *If the ﬁnite word w has at most a trivial proper PS and is shift- maximal, then wC is shift-maximal.*

**Theorem 5.12** *Given any (computable) strong shift-maximal sequence x ∈* 2N*, there is a (computable) unimodal map F with kneading sequence x.*

**Proof.** Let *x ∈* 2N be a strong shift-maximal sequence. Then by Lemma [5.11](#_bookmark8), we can find a subsequence *< wn >n<ω* of the initial segments of *x* such that *wnC* is shift-maximal word for each *n*. It follows from Proposition [5.7](#_bookmark5) and Corollary [5.10](#_bookmark7) that each *wnC* is a legal inverse path and hence there exist *μn* such that *Fμn* has kneading sequence (*wnC*)*ω* . Since *limnwnC* = *x*, it follows from Proposition [5.9](#_bookmark6) that *limnμn* = *μ* exists and that *KS*(*Fμ*) = *x*.

If *x* is computable, then we can compute the sequence *w**n* (since testing to see if *wnC* is shift-maximal is computable) and then compute *μn* from *wn*. By Proposition [5.7](#_bookmark5), we may assume that *μn*+2 is between *μn* and *μn*+1, so that the limit *μ* is also computable.

# References

1. Braverman, M. and M. Yampolski, *Non-computable Julia sets*, J. Amer. Math. Soc. **19** (2006), 551–578.
2. Cenzer, D. *Effective dynamics*, in “Logical Methods in honor of Anil Nerode’s Sixtieth Birthday” (J. Crossley, J. Remmel, R. Shore and M. Sweedler, eds.), Birkhauser (1993), 162–177.
3. Cenzer, D. and J.B. Remmel, Π0 *classes in mathematics*, in “Recursive Mathematics” (Y. Ersov, S. Goncharov, A. Nerode and J.B. Remmel, eds.), North-Holland Studies in Logic and Found. Math. **138** (1998), 623–821.

1

1. Cenzer, D. and J.B. Remmel, “Effectively closed sets”, Association for Symbolic Logic Lecture Notes, to appear.
2. Collet, P. and J.P. Eckmann, “Iterated Maps On The Interval As Dynamical Systems”, Birkhuser (1980).
3. Delvenne, J.-C. “Dynamics, Information and Computation”, Ph. D. thesis, Universite Catholique De Louvain (2005).
4. Delvenne, J.-C., P. Kurka, and V. Blondel, *Decidability and Universality in Symbolic Dynamical* *Systems*, Fund. Informaticae **74** (2006), 463–490.
5. Grzegorczyk, A., *On the definitions of computable real continuous functions*, Fund. Math. **44** (1957), 61–71.
6. Huang, W. and A. Nerode, *Applications of pure recursion theory to recursive analysis*, Acta Sinica **28**

(1985), 625–636.

1. Ko, K.,“Complexity Theory of Real Functions”, Birkhauser, (1991).
2. Ko, K., *On the computability of fractal dimensions and Julia sets*, Ann. Pure Appl. Logic **93** (1998), 195–216.
3. LaCombe, D., *Extension de la notion de fonction recursive aux d’une ou plusieurs variables reelles*

Comptes Rendus Acad. Sci. Paris **241** (1955), 2478–2480.

1. Lind, D. and B. Marcus, “An Introduction to Symbolic Dynamics and Coding”, Cambridge University Press, (1995).
2. Metropolis, N., M. L. Stein, and P. R. Stein, *On Finite Limit Sets for Transformation of the Unit Interval*, Journal of Combinatorial Theory (A) **15** (1973), 25–44.
3. Rettinger, R. and K. Weihrauch, *The computational complexity of some Julia sets*, in “Proc. 35th ACM Symposium on Theory of Computing (San Diego, June 2003)” (M.X. Goemans, ed.), ACM Press, New York (2003), 177–185.
4. Weihrauch, K., “Computable Analysis”, Springer (2000).
5. Xie, H., “Grammatical Complexity And One-Dimensional Dynamical Systems”, World Scientific (1996).