Available online at www.sciencedirect.com

ScienceDirect

Electronic Notes in Theoretical Computer Science

Electronic Notes in Theoretical Computer Science 167 (2007) 425–447

www.elsevier.com/locate/entcs

Jordan Curves with Polynomial Inverse Moduli of Continuity

Ker-I Ko[a,1,2] Fuxiang Yu[a,1,3]

[a] *Department of Computer Science*
State University of New York
Stony Brook, NY, U.S.A.

Abstract

Computational complexity of two-dimensional domains whose boundaries are polynomial-time computable Jordan curves with polynomial inverse moduli of continuity is studied. It is shown that the membership problem of such a domain can be solved in P^{NP}, i.e., in polynomial time relative to an oracle in NP, in contrast to the higher upper bound P^{MP} for domains without the property of polynomial inverse modulus of continuity. On the other hand, the lower bound of UP for the membership problem still holds for domains with polynomial inverse moduli of continuity. It is also shown that the path problem of such a domain can be solved in *PSPACE*, matching its known lower bound, while no fixed upper bound was known for domains without this property.

1 Introduction

In computable analysis, we know that if a real, one-to-one function f from $[0,1]$ to $[0,1]$ is computable, then its inverse function f^{-1} is also computable. One can find an approximate value of $f^{-1}(y)$ by the binary search method.[4]

[1] This material is based upon work supported by National Science Foundation under grant No. 0430124.

[2] Email: keriko@cs.sunysb.edu

[3] Email: fuxiang@cs.sunysb.edu

[4] Although the problem of determining whether two given real numbers are equal is undecidable, we can use a modified binary search to avoid this issue. At each stage of the binary search, we can select two dyadic rationals d and e which are near each other, and

doi:10.1016/j.entcs.2006.08.022

However, there is, in general, no fixed complexity bound for f^{-1}, even if f itself is known to be polynomial-time computable (see Ko [9]). To be more precise, the complexity of the inverse function f^{-1} depends exactly on its modulus of continuity, or, the inverse modulus of continuity of f:

Proposition 1.1 *Let $f : [0,1] \to [0,1]$ be a one-to-one function computable in time $t(n)$. Assume that $m : \mathbb{N} \to \mathbb{N}$ is an inverse modulus of continuity of f; that is, m satisfies that $|x-y| \geq 2^{-n}$ implies $|f(x)-f(y)| \geq 2^{-m(n)}$. Then, f^{-1} is computable in time $O(n \cdot t(m(n)))$.*

This simple fact suggests that, when we study the complexity of roots or the inverse of a given function f, we first need to know the inverse modulus of continuity of f. This idea also applies to one-to-one functions f which define more complicated objects. For instance, Berg et al. [1] pointed out that, from a constructive point of view, when a one-to-one function $f : [0,1] \to \mathbb{R}^2$ defining a Jordan curve on the two-dimension plane is "given," the modulus of continuity of f^{-1} needs to be given too.

In this paper, we investigate the effect of the inverse modulus of continuity of a Jordan curve on the computational complexity of functions defined on two-dimensional domains. Chou and Ko [4,5,6] have studied the complexity issues of two-dimensional domains S whose boundaries are polynomial-time computable Jordan curves Γ. Among the problems studied in this investigation, results about the following two problems are incomplete. (We write $\mathrm{Int}(\Gamma)$ and $\mathrm{Ext}(\Gamma)$ to denote the interior and the exterior, respectively, of the curve Γ.)

Membership Problem: Given a point $\mathbf{z}$ on the two-dimensional plane and an integer $n > 0$, determine whether $\mathbf{z}$ locates in $\mathrm{Int}(\Gamma)$, or in $\mathrm{Ext}(\Gamma)$, or is close to the boundary Γ within a distance 2^{-n}.

Path Problem: Given two points $\mathbf{x}$ and $\mathbf{y}$ in $\mathrm{Int}(\Gamma)$ with distance at least 2^{-n} from the boundary Γ and an integer $m \geq n$, find a (shortest) path from $\mathbf{x}$ to $\mathbf{y}$ that lies entirely within $\mathrm{Int}(\Gamma)$, and has a distance at least 2^{-m} from the boundary Γ.

It was shown in Chou and Ko [4] that, for a polynomial-time computable Jordan curve Γ, the membership problem has an upper bound P^{MP} (or, $P^{MidbitP}$) and a lower bound UP. The gap between the upper and lower bounds is quite big. For the (shortest) path problem, Chou and Ko [5] showed an upper bound of $PSPACE$ and a lower bound of $\#P$ for polynomial-time computable Jordan curves, assuming that such a path exists. It is, however,

test whether $f(d) < y$, $f(d) > y$, $f(e) < y$ or $f(e) > y$. Since f is one-to-one, one of these four conditions must hold.

not clear whether this assumption holds for a given Jordan curve, even if it is known to be computable in polynomial time. Indeed, a polynomial-time computable Jordan curve could be very "complex." For instance, Ko [10] and Ko and Weihrauch [11] constructed polynomial-time computable Jordan curves which have Hausdorff dimension two (and hence are fractals) and whose interior measures are not even recursive real numbers.

In this paper, we restrict our attention to two-dimensional domains whose boundaries are polynomial-time computable Jordan curves that also have polynomial inverse moduli of continuity. With respect to such curves, we are able to show the following:

(1) For the complexity of the membership problem, the upper bound is reduced to $\Delta_2^P = P^{NP}$.

(2) For the path problem, it is shown that there is a polynomial q such that, for any two points $\mathbf{x}$ and $\mathbf{y}$ in $\mathrm{Int}(\Gamma)$ that have distance 2^{-n} away from the boundary Γ, there must be a path from $\mathbf{x}$ to $\mathbf{y}$ that lies in $\mathrm{Int}(\Gamma)$ and has distance $2^{-q(n)}$ away from the boundary Γ.

Result (2) above is particularly useful when we consider analytic functions defined on two-dimensional regions. For instance, combining it with the algorithm for the shortest-path problem of Chou and Ko [5], we show that the problem of analytic continuation has an exponential space upper bound.

On the other hand, for the lower bounds, we observe that the Jordan curves constructed for the UP lower bound of the membership problem (Chou and Ko [4]) and the $\#P$ lower bound for the path problem (Chou and Ko [5]) both have polynomial inverse moduli of continuity. These results seem to indicate that Jordan curves with polynomial inverse moduli of continuity are a natural class of objects, and by studying such curves, we can get more accurate characterization of the complexity of related problems.

Our basic computational model for real-valued functions and two-dimensional regions is the oracle Turing machine model. For the general theory of computable analysis based on the Turing machine model, see, for instance, Pour-El and Richards [14] and Weihrauch [17]. For the theory of computational complexity of real functions based on this computational model, see Ko [9]. The extension of this theory to include the computational complexity of two-dimensional regions has been presented in Chou and Ko [4]. Computational complexity of problems related to two-dimensional regions has been studied recently in several directions. Chou and Ko [4], Ko [10] and Ko and Weihrauch [11] studied the notion of polynomial-time computable two-dimensional regions. Rettinger and Weihrauch [16], Braverman [2], Rettinger [15] and Braverman and Yampolsky [3] studied the the computational complexity of Julia sets. Chou and Ko [5,6] studied the problem of finding paths in a two-dimensional domain. Ko and Yu [12] studied the problem of computing single-valued analytic branches of logarithm and square-root functions on a two-dimensional domain. Yu, Chou and Ko [18] studied the problem of computing the minimum-area circumscribed rectangles and squares of a Jordan curve. All these works (and many others) used Turing machines and oracle Turing machines as the basic model.

2 Definitions

This paper involves notions used in both discrete computation and continuous computation. The basic computational objects in discrete computation are integers and strings in $\{0,1\}^*$. The length of a string w is denoted $\ell(w)$.

The fundamental complexity classes we are interested in are the class P of sets accepted by deterministic polynomial-time Turing machines, and the class FP of functions (mapping strings to strings) computable by deterministic polynomial-time Turing machines. We will also use in this paper the following complexity classes (see, e.g., Du and Ko [7]):

$\#P$: the class of functions that count the number of accepting paths of nondeterministic polynomial-time machines.

UP: the class of sets A accepted by an unambiguous nondeterministic polynomial-time Turing machine M such that for all $x \in \{0,1\}^*$, $M(x)$ has at most one accepting computational path.

$PSPACE$: the class of sets accepted by polynomial-space Turing machines.

The basic computational objects in continuous computation are dyadic rationals $\mathbb{D} = \{m/2^n : m \in \mathbb{Z}, n \in \mathbb{N}\}$, and we denote $\mathbb{D}_n = \{m/2^n : m \in \mathbb{Z}\}$.

We use $\mathbb{R}$ to denote the class of real numbers and $\mathbb{R}^2$ the class of points on the two-dimensional plane. A point in $\mathbb{R}^2$ is denoted by a boldface font, such as $\mathbf{z}$, or a pair of real numbers, such as $\langle x, y\rangle$. For any point $\mathbf{z} \in \mathbb{R}^2$ and any set $S \subseteq \mathbb{R}^2$, we let $\mathrm{dist}(\mathbf{z}, S)$ be the distance between $\mathbf{z}$ and S; that is, $\mathrm{dist}(\mathbf{z}, S) = \inf\{|\mathbf{z} - \mathbf{z}'| : \mathbf{z}' \in S\}$, where $|\cdot|$ denotes the absolute value. The boundary of a set S is written as ∂S.

We also define a function $\delta(s,t)$, which can be viewed as the distance of two sets $\{s+m : m \in \mathbb{Z}\}$ and $\{t+n : n \in \mathbb{Z}\}$. Assume that $0 \le s < t \le 1$, we write $\delta(s,t) = \min\{t-s, 1+s-t\}$. If $\delta(s,t) < 1/2$ and $t-s = \delta(s,t)$, then we let $I(s,t)$ be the interval $[s,t]$; otherwise, if $\delta(s,t) < 1/2$ and $t-s > \delta(s,t)$ then let $I(s,t) = [0,s] \cup [t,1]$.

We say a function $\phi : \mathbb{N} \to \mathbb{D}$ *binary converges* to (or *represents*) a real number x, if (i) for all $n \ge 0$, $\phi(n) \in \mathbb{D}_n$, and (ii) for all $n \ge 0$, $|\phi(n) - x| \le 2^{-n}$.

For any $x \in \mathbb{R}$, there is a unique function $b_x : \mathbb{N} \to \mathbb{D}$ that binary converges to x and satisfies the condition $x - 2^{-n} < b_x(n) \le x$ for all $n \ge 0$. We call this function b_x the *standard Cauchy function* for x. We say two functions $\phi_x, \phi_y : \mathbb{N} \to \mathbb{D}$ binary converge to (or represent) a complex number $\langle x, y\rangle$ if ϕ_x and ϕ_y binary converge to two real numbers x and y, respectively.

To compute a real-valued function $f : \mathbb{R} \to \mathbb{R}$, we use oracle Turing machines as the computational model. An oracle Turing machine M is an ordinary Turing machine equipped with an extra query tape which it may use to make queries to an oracle function. M makes queries by writing the query on the query tape, and then reading the answer after the oracle replaces the query strings by an answer string. Each query made by M counts only one machine step. We use $M^\phi(n)$ to denote the output of machine M with regard to an input n and an oracle ϕ. We say an oracle Turing machine M *operates in polynomial time* if there exists a polynomial p such that for all inputs n and all oracles ϕ, $M^\phi(n)$ halts in time $p(n)$.

Definition 2.1 (*a*) A function $f : [0,1] \to \mathbb{R}$ is said to be *computable* if there is an oracle Turing machine M that, on an oracle function $\phi : \mathbb{N} \to \mathbb{D}$ that binary converges to a real number x and an input $n \in \mathbb{N}$, outputs a string $d \in \mathbb{D}_n$ such that $|d - f(x)| \le 2^{-n}$.

(*b*) A function $f : [0,1] \to \mathbb{R}$ is *polynomial-time computable* if it is computable by an oracle Turing machine that operates in polynomial time.

The following equivalent definition for polynomial-time computable real functions f is useful. We say a function $f : [0,1] \to \mathbb{R}$ has a *polynomial modulus* if there exists a polynomial p such that $|x - y| \le 2^{-p(n)}$ implies $|f(x) - f(y)| \le 2^{-n}$.

Proposition 2.2 *A function $f : [0,1] \to \mathbb{R}$ is polynomial-time computable if and only if*

(i) f has a polynomial modulus, and

(ii) There exists a Turing machine M and a polynomial p such that for any integer n and any $d \in \mathbb{D}_m \cap [0,1]$, $M(d,n)$ outputs, in time $p(m+n)$, a dyadic rational number e such that $|e - f(d)| \le 2^{-n}$.

We say a real function $f : [0,1] \to \mathbb{R}$ is computable in nondeterministic polynomial time if there are two polynomial functions p, q, and a deterministic oracle TM $\widetilde{M}$ such that for any oracle ϕ representing a real number $x \in [0,1]$ and any integer $n > 0$,

(i) $\widetilde{M}^\phi(n, w)$ halts in $p(n)$ moves for all binary strings $w \in \{0,1\}^{p(n)}$, and it may or may not output a dyadic rational;

(ii) For at least one string $w \in \{0,1\}^{p(n)}$, $\widetilde{M}^\phi(n, w)$ outputs a dyadic rational d;

(iii) If $\widetilde{M}^{\phi}(n, w)$ outputs a dyadic rational d, then $|d - f(x)| \leq 2^{-n}$.

We say a function $f : [0, 1] \to \mathbb{R}^2$ represents a Jordan curve if (i) f is one-to-one on $[0, 1)$ and $f(0) = f(1)$, and (ii) the image of $[0, 1]$ under f is the curve Γ. In this paper, we consider Jordan curves that are represented by polynomial-time computable functions f. Note that if f represents a Jordan curve and f is polynomial-time computable, then f has a polynomial modulus of continuity; that is, there is a polynomial p such that, for all $n > 0$, $\delta(s, t) \leq 2^{-p(n)}$ implies $f(s, t) \leq 2^{-n}$.

We say a function $m : \mathbb{N} \to \mathbb{N}$ is an inverse modulus of continuity of a function $f : [0, 1] \to \mathbb{R}^2$ if there is a polynomial function $p : \mathbb{N} \to \mathbb{N}$ and an integer n_0 such that the following holds for all $n > n_0$: For any two points $s, t \in [0, 1]$ $|f(s) - f(t)| > 2^{-p(n)}$ whenever $\delta(s, t) > 2^{-n}$. We say a Jordan curve Γ is polynomial-time computable and has a polynomial inverse modulus of continuity if there is a polynomial-time computable function f representing Γ which has a polynomial inverse modulus of continuity.

In general, when we deal with two-dimensional domains, there is a boundary issue. Consider, for instance, the membership problem: When a point $\mathbf{z}$ is very close to the boundary Γ, it requires more resources to decide whether it is inside or outside of Γ; moreover, it is undecidable whether $\mathbf{z}$ is on Γ. In other words, the function $f : \mathbb{R}^2 \to \{-1, 0, 1\}$ defined by

$$f(\mathbf{z}) = \begin{cases} -1, & \mathbf{z} \in Ext(\Gamma). \\ 0, & \mathbf{z} \in \Gamma. \\ 1, & \mathbf{z} \in Int(\Gamma). \end{cases}$$

is not computable since it is not continuous on the boundary Γ. Similarly, for the path problem, when a path is very close to the boundary Γ, it is relatively harder to decide whether the path is inside Γ or not.

To overcome this problem, we adopt a less restrictive model of computation and allow the oracle Turing machines that solve the problems to make errors near the boundary. More precisely, as stated in Section 1, for an input $n \in \mathbb{N}$, the membership problem of a point $\mathbf{z}$ needs to be solved only if the distance $\mathrm{dist}(\mathbf{z}, \Gamma)$ is no less than 2^{-n}, and the path problem of two points $\mathbf{x}$ and $\mathbf{y}$ needs to be solved only if $\mathrm{dist}(\mathbf{x}, \Gamma) > 2^{-n}$ and $\mathrm{dist}(\mathbf{y}, \Gamma) > 2^{-n}$. Similar treatments can be found in Chou and Ko [4] and Ko and Yu [12].

3 Inverse moduli and inverse functions

One-way functions are a fundamental concept in discrete complexity theory, with important applications in cryptography. Intuitively, a function $\phi : \{0,1\}^* \to \{0,1\}^*$ is *one-way* if it satisfies the following properties:

(i) ϕ is one-to-one,

(ii) ϕ is polynomial-time computable, and

(iii) ϕ^{-1} is not polynomial-time computable.

A function satisfying the above conditions, however, does not really capture the concept of one-way functions. For instance, for any integer $n > 0$, let $b(n)$ denote the binary representation of n. Then, the function $\phi(0^n) = b(n)$, and $\phi(w) = 0w$ if $w \notin \{0\}^*$, satisfies the above conditions (i)–(iii). We note, however, that he inverse ϕ^{-1} of this function is actually very "easy" to compute; it is not polynomial-time computable simply because its output is too long to be written down in polynomially many steps. The usual way to get around this issue is to require that a one-way function ϕ must be *polynomially honest*; that is, ϕ must satisfy an addition condition:

(iv) There is a polynomial q such that $q(\ell(\phi(w))) \geq \ell(w)$.

With this additional condition, the existence of one-way functions is no longer easy to prove. In fact, it is known that if $P = NP$ then there are no one-way functions. More precisely, one-way functions with properties (i)–(iv) exist if and only if $P = UP$ (see, e.g., Ko [9] and Du and Ko [7]).

The analogous notion of polynomial honestity for continuous functions is polynomial inverse modulus of continuity. Intuitively, it means that when two numbers s and t are far from each other, then their images $f(s)$ and $f(t)$ cannot be too close to each other. For instance, consider polynomial-time computable one-to-one functions f mapping $[0,1]$ to $[0,1]$. It is easy to see that f^{-1} is polynomial-time computable if and only if f has a polynomial inverse modulus of continuity (cf. Proposition 1.1).

For a function f defining a Jordan curve on the two-dimensional plane, it is not hard to show that if f is polynomial-time computable and has a polynomial inverse modulus of continuity, then the complexity of its inverse function f^{-1} is similar to that of discrete one-way functions.

Theorem 3.1 *Assume that $f : [0,1] \to \mathbb{R}^2$ represents a Jordan curve. Also assume that f has a polynomial inverse modulus of continuity. Then, there is a nondeterministic oracle TM M that computes the inverse function f^{-1} of f.*

Proof. Assume that p is a polynomial function that is both a time bound for f and an inverse modulus of continuity of f. To find an approximate value to $f^{-1}(\mathbf{z})$ within error 2^{-n}, M works as follows:

> Guess a dyadic point $d \in \mathbb{D}_{p(p(n)+1)}$. Then, simulate the computation of f to find an approximate point $\mathbf{w}$ to $f(d)$ with error $\leq 2^{-(p(n)+1)}$. Outputs d if $|\mathbf{w} - \mathbf{z}| \leq 2^{-(p(n)+1)}$.

Note that, if $f(e) = \mathbf{z}$ and $|f(d) - \mathbf{z}| \leq 2^{-p(n)}$, then $|d - e| \leq 2^{-n}$. Thus, the output of M is always correct. In addition, for any input $\mathbf{z} = f(e)$, if M guesses a dyadic point $d \in \mathbb{D}_{p(p(n)+1)}$ such that $|d - e| \leq 2^{-p(p(n)+1))}$, then we must have $|f(d) - \mathbf{z}| \leq 2^{-(p(n)+1)}$, and M will accept. That is, for any $\mathbf{z}$ in the range of f, M always has at least one computation path that accepts. It is obvious that M always halts in polynomial time. □

Corollary 3.2 *In the following, (a)⇒(b)⇒(c):*

(a) $P = NP$.

(b) For any polynomial-time computable function $f : [0, 1] \to \mathbb{R}^2$ that represents a Jordan curve Γ and has a polynomial inverse modulus of continuity, f^{-1} is polynomial-time computable.

(c) $P = UP \cap coUP$.

Proof. The part of (a) ⇒ (b) follows from Theorem 3.1.

(b) ⇒ (c): The proof is similar to the proof for the UP lower bound for the membership problem (see Theorem 7.3 of Chou and Ko [4]). We only give a sketch here.

Assume that $P \neq UP \cap coUP$ and $A \subseteq \{0, 1\}^*$ is in $UP \cap coUP - P$. Then, there exist two sets $B, C \in P$ and a polynomial function p such that, for any $s \in \{0, 1\}^*$ of length n,

$$s \in A \iff (\exists! t, \ell(t) = p(n))\, \langle s, t\rangle \in B,$$

$$s \notin A \iff (\exists! t, \ell(t) = p(n))\, \langle s, t\rangle \in C,$$

where $(\exists! t)$ means "there exists a unique t."

Now, we define, for each string $s \in \{0, 1\}^*$ of length $n > 0$, a dyadic rational a_s as follows: Let i_s be the integer whose n-bit binary expansion (with possible extra leading zeros) is equal to s, and define $a_s = 1 - 2^{-(n-1)} + i_s \cdot 2^{-2n}$. That is, $a_{0^n} = 1 - 2^{-n+1}$, and $a_{0^{n-1}1}, \ldots, a_{1^n}$ evenly divides the interval $[1 - 2^{-(n-1)}, 1 - 2^{-n}]$. Also let $c_s = a_s + 2^{-2n-1}$.

Next, for any $s, t \in \{0, 1\}^*$ with length $\ell(s) = n$ and $\ell(t) = p(n)$, we define $b_{s,t} = a_s + i_t \cdot 2^{-p(n)-2n}$. That is, $\{b_{s,t} : \ell(t) = p(n)\}$ evenly divides the interval $[a_s, a_s + 2^{-2n}]$.

Now, we define a function $f : [0,1] \to \mathbb{R}$ that satisfies the property of part (b): f is a piecewise linear function with the following breakpoints:

(1) $f(0) = f(1) = \langle 0, 1\rangle$, $f(1/2) = \langle 1, 1\rangle$, $f(3/4) = \langle 1, 0\rangle$, $f(7/8) = \langle 0, 0\rangle$;
(2) For $s, t \in \{0,1\}^*$ with $\ell(s) = n > 0$ and $\ell(t) = p(n)$, if $\langle s, t\rangle \in B \cup C$, then f has a bump with the breakpoints $f(b_{s,t}/2) = \langle b_{s,t}, 1\rangle$, $f((b_{s,t} + 2^{-p(n)-2n-1})/2) = \langle c_s, 1 + 2^{-2n}\rangle$, and $f((b_{s,t} + 2^{-p(n)-2n})/2) = \langle b_{s,t} + 2^{-p(n)-2n}, 1\rangle$.

Note that a string s is either in A or in $\overline{A}$, and so there is a unique t of length $p(\ell(s))$ such that $\langle s, t\rangle \in B \cup C$. That is, the Jordan curve Γ defined by f has exactly one bump in $[a_s/2, (a_s + 2^{-2n})/2]$, and so f is one-to-one and $\langle c_s, 1 + 2^{-2n}\rangle$ is always on Γ. Now, if we can compute $f^{-1}(\langle c_s, 1 + 2^{-2n}\rangle)$, then we can get t such that $\langle s, t\rangle \in B \cup C$, and we can use this string t to determine whether $s \in A$. Thus, $A \notin P$ implies that f^{-1} is not polynomial-time computable. □

We note that the above result is weaker than the analogous result in discrete complexity theory: one-way functions exists if and only if $P \neq UP$. In particular, our condition (a) of $P = NP$ appears much stronger than the condition $P = UP$. This seems to come from the nature of continuous computation: In the algorithm for M in the proof of Theorem 3.1, there may be *many* dyadic numbers d satisfying $|f(d) - \mathbf{z}| \leq 2^{-(p(n)+1)}$, and we cannot tell which one is the best approximation to e. It remains open whether the gap between conditions (a) and (c) of Corollary 3.2 can be narrowed.

4 Membership problem

We assume, in this section, that $f : [0,1] \to \mathbb{R}^2$ is a polynomial-time computable function representing a Jordan curve Γ, and has a polynomial inverse modulus of continuity. That is, we assume that there exists a polynomial function p and a constant $n_0 > 0$ such that (i) f is computable in time $p(n)$, and (ii) for $n \geq n_0$, $\delta(s - t) > 2^{-n}$ implies $|f(s) - f(t)| > 2^{-p(n)}$. Without loss of generality, we assume that $p(n) > 2n$ for all $n > 0$.

Recall that the membership problem about the curve Γ asks, for an oracle $\langle \phi_1, \phi_2\rangle$ representing a point $\mathbf{z} \in \mathbb{R}^2$ and an input integer $n > 0$, whether $\mathbf{z}$ is in $\mathrm{Int}(\Gamma)$. An algorithm solving the membership problem is required to give the correct answer if $\mathrm{dist}(\mathbf{z}, \Gamma) > 2^{-n}$.

In the following, we are going to present such an algorithm which runs in polynomial time relative to an oracle $A \in NP$. We first explain the basic idea of the algorithm.

First, we consider the curve Γ as a directed curve which begins at point $f(0)$ and ends at $f(1)$. In the following, by a *section* of Γ from a point $\mathbf{x}$ to a point $\mathbf{y}$, written as $\Gamma_{\mathbf{xy}}$, we mean either (i) the image of the interval $[a, b]$ under f, if $0 \le a < b \le 1$ and $\mathbf{x} = f(a)$, $\mathbf{y} = f(b)$, or (ii) the image of $[a, 1] \cup [0, b]$ under f if $0 \le b < a \le 1$, and $\mathbf{x} = f(a)$, $\mathbf{y} = f(b)$. We assume that the directed curve Γ goes around its interior points in the counterclockwise direction; that is, for any interior point $\mathbf{z}_0$, the winding number of $\mathbf{z}_0$ about the directed curve Γ is equal to 1. (The case of Γ going around its interior points clockwise is symmetric.)

Now, we consider a point $\mathbf{z}$ with $\text{dist}(\mathbf{z}, \Gamma) > 2^{-n}$, where $n > n_0$. Let $m = p(n) + 1$, and $q(n) = p(p(p(n) + 2)) + 5$. Let $\mathbf{x}$ be a point on Γ satisfying $|\mathbf{x} - \mathbf{z}| \le \text{dist}(\mathbf{z}, \Gamma) + 2^{-q(n)+2}$, and C the circle that centered at $\mathbf{x}$ with radius 2^{-m}. Assume that Γ does not lie entirely within C (otherwise, it is obvious that $\mathbf{z} \in \text{Ext}(\Gamma)$.) Let $\mathbf{x}_0$ and $\mathbf{x}_1$ be two points on $\Gamma \cap C$ such that $\mathbf{x}$ lies on the section $\Gamma_{\mathbf{x}_0\mathbf{x}_1}$, and the whole section $\Gamma_{\mathbf{x}_0\mathbf{x}_1}$ lies inside the circle C. Also let $\mathbf{y}$ be the intersection point of the line segment $\overline{\mathbf{zx}}$ and the circle C. (See Figure 1.)

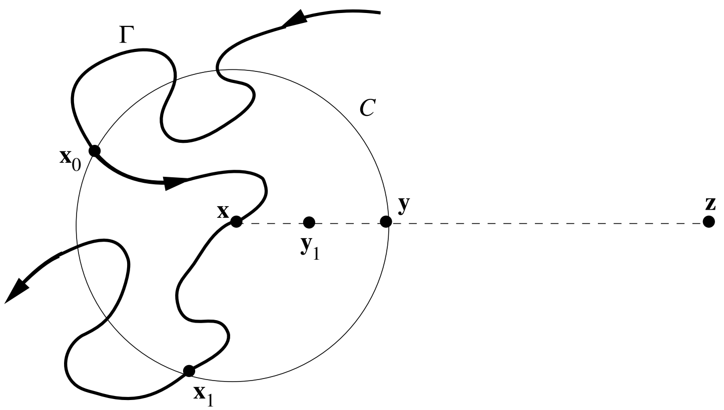

Fig. 1. Curve Γ enters circle C at $\mathbf{x}_0$ and leaves at $\mathbf{x}_1$.

We need to define the notion of the *orientation* of three points P, Q, R around a point X. For any three points P, Q, X on the plane such that Q does not lie on the halfline $\overrightarrow{XP}$, we let $\angle PXQ$ denote the angle from the halfline $\overrightarrow{XP}$ to the halfline $\overrightarrow{XQ}$, measured in the *counterclockwise* direction. (Thus, $\angle PXQ$ is always between 0 and 2π.) We say the orientation of the points P, Q, R around X is in the *counterclockwise* direction if $0 < \angle PXQ < \angle PXR < 2\pi$, and it is in the *clockwise* direction if $0 < \angle PXR < \angle PXQ < 2\pi$. (The orientation is undefined if $\angle PXQ$ or $\angle PXR$ is equal to 0, or if $\angle PXQ = \angle PXR > 0$.)

Lemma 4.1 *The point* $\mathbf{z}$ *lies in the interior of* Γ *if and only if the orientation* *of the three points* $\mathbf{x}_1$, $\mathbf{y}$ *and* $\mathbf{x}_0$ *around the point* $\mathbf{x}$ *is in the counterclockwise direction.*

Proof. Let $\mathbf{y}_1$ be the point on $\overline{\mathbf{x}\mathbf{z}}$ with distance $2^{-(m+1)}$ from $\mathbf{x}$. Notice that $|\mathbf{z}-\mathbf{x}| \leq \mathrm{dist}(\mathbf{z},\Gamma) + 2^{-q(n)+2} \leq \mathrm{dist}(\mathbf{z},\Gamma) + 2^{-(m+2)}$. Therefore, the curve Γ does not intersect the line segment $\overline{\mathbf{y}_1\mathbf{z}}$. It follows that $\mathbf{z} \in \mathrm{Int}(\Gamma)$ if and only if $\mathbf{y}_1 \in \mathrm{Int}(\Gamma)$.

We treat C as a directed curve going around the point $\mathbf{x}$ counterclockwise, and let Λ be the directed curve formed with $\Gamma_{\mathbf{x}_0\mathbf{x}_1}$ plus the arc of C from $\mathbf{x}_1$ to $\mathbf{x}_0$ (in the counterclockwise direction). Now we define $\mathbf{x}'$ to be the point in $\Gamma \cap \overline{\mathbf{x}\mathbf{y}}$ that is the closest to $\mathbf{y}$. Then, $|\mathbf{x}'-\mathbf{x}| \leq 2^{-q(n)+2}$.

Case 1. $\mathbf{x}' = \mathbf{x}$. This means that the Γ meets $\overline{\mathbf{x}\mathbf{y}_1}$ only at the point $\mathbf{x}$. Since Γ goes around its interior points counterclockwise, points to its "left-hand side" are the interior points of Γ. Thus, $\mathbf{y}_1 \in \mathrm{Int}(\Gamma)$ if and only if $\mathbf{y}_1$ lies to the left of $\Gamma_{\mathbf{x}_0\mathbf{x}_1}$, or equivalently, $\mathbf{y}_1 \in \mathrm{Int}(\Lambda)$. The last condition is equivalent to the condition that $\mathbf{y}$ lies on the arc of C from $\mathbf{x}_1$ to $\mathbf{x}_0$, or that the orientation of points $\mathbf{x}_1, \mathbf{y}, \mathbf{x}_0$ around point $\mathbf{x}$ is in the counterclockwise direction (cf. Figure 1).

Case 2. $\mathbf{x}' \neq \mathbf{x}$. Then, we claim that $\mathbf{x}'$ lies on $\Gamma_{\mathbf{x}_0\mathbf{x}_1}$. To see this, we let $\mathbf{x}_0 = f(t_0)$, $\mathbf{x}_1 = f(t_1)$, $\mathbf{x} = f(t)$, and $\mathbf{x}' = f(t')$. Assume, for the sake of contradiction, that $\mathbf{x}'$ is not on $\Gamma_{\mathbf{x}_0\mathbf{x}_1}$; that is, $t' \notin I(t_0, t_1)$. Then, we must have $\delta(t',t) \geq \min\{\delta(t_0,t), \delta(t_1,t)\}$. Since $|\mathbf{x}_0-\mathbf{x}| = |\mathbf{x}_1-\mathbf{x}| = 2^{-m}$, we know that both $\delta(t_0,t)$ and $\delta(t_1,t)$ are at least $2^{-p(m)}$. Thus, we get $\delta(t',t) \geq 2^{-p(m)}$, and by the assumption of the inverse modulus p, $|\mathbf{x}'-\mathbf{x}| \geq 2^{-p(p(p(n)+1))} > 2^{-q(n)+2}$. It follows that $|\mathbf{x}'-\mathbf{z}| < \mathrm{dist}(\mathbf{z},\Gamma)$ and leads to a contradiction. We conclude that $\mathbf{x}'$ must be on the section $\Gamma_{\mathbf{x}_0\mathbf{x}_1}$.

Now, we see that the line segment $\overline{\mathbf{x}'\mathbf{y}_1}$ does not meet Γ other than the point $\mathbf{x}'$, which lies on $\Gamma_{\mathbf{x}_0\mathbf{x}_1}$. So, the argument for Case 1 also applies to Case 2, and the lemma is proven. □

Based on Lemma 4.1, we can devise an algorithm to determine the membership of $\mathbf{z}$. In the following, let $m = p(n)+1$, $q(n) = p(p(p(n)+2))+5$, and $r(n) = p(q(n))$. Without loss of generality, we assume that $\mathbf{z} \in \mathbb{D}_{q(n)}{}^2$ (otherwise, we can find a dyadic rational point $\mathbf{z}' \in \mathbb{D}_{q(n)}{}^2$ with $|\mathbf{z}-\mathbf{z}'| \leq 2^{-q(n)}$ and solve the membership problem about $\mathbf{z}'$ instead).

(1) Find a dyadic rational $d \in \mathbb{D}_{q(n)}$ such that $|d - \mathrm{dist}(\mathbf{z},\Gamma)| \leq 2^{-q(n)}$.

(2) Find a dyadic rational $t \in \mathbb{D}_{r(n)} \cap [0,1]$, and a dyadic point $\mathbf{w} \in \mathbb{D}_{q(n)}{}^2$ such that $|\mathbf{w} - f(t)| \leq 2^{-q(n)}$ and $d - 2^{-q(n)+1} < |\mathbf{w}-\mathbf{z}| < d + 2^{-q(n)+1}$.

(3) Find a dyadic rational point $\mathbf{v} \in \mathbb{D}_{q(n)}{}^2$ such that $|\mathbf{v}-\mathbf{y}| < 2^{-q(n)}$, where $\mathbf{y}$ is the point in the line segment $\overline{\mathbf{z}f(t)}$ that has the distance $|\mathbf{y} - f(t)| = 2^{-m}$.

(4) Find two dyadic rationals $t_0, t_1 \in \mathbb{D}_{r(n)} \cap [0,1]$, and two dyadic points $\mathbf{w}_0, \mathbf{w}_1 \in {\mathbb{D}_{q(n)}}^2$, with $|\mathbf{w}_0 - f(t_0)| \le 2^{-q(n)}$ and $|\mathbf{w}_1 - f(t_1)| \le 2^{-q(n)}$, that satisfy
   (a) $\delta(t_0, t_1) \le 2^{-n+1}$ and $t \in I(t_0, t_1)$; i.e., either $(t_0 < t < t_1 < t_0 + 2^{-n+1})$ or $(t_0 < t_1 + 1 < t_0 + 2^{-n+1}$ and $t \in [0, t_1] \cup [t_0, 1])$;
   (b) $\Big||\mathbf{w}_0 - \mathbf{w}| - 2^{-m}\Big| \le 2^{-q(n)+2}$, and $\Big||\mathbf{w}_1 - \mathbf{w}| - 2^{-m}\Big| \le 2^{-q(n)+2}$; and
   (c) for any $u \in \mathbb{D}_{r(n)} \cap I(t_0, t_1)$, $|f(u) - \mathbf{w}| \le 2^{-m} + 2^{-q(n)+2}$.

(5) Determine the orientation of the three points $\mathbf{w}_1, \mathbf{v}$ and $\mathbf{w}_0$ around the point $\mathbf{w}$. Return the answer "$\mathbf{z} \in \mathrm{Int}(\Gamma)$" if and only if the orientation is in the counterclockwise direction.

Before we can prove that the above algorithm is correct, we need to verify that it is well-defined. First, for step (4), we need to verify that dyadic rationals t_0 and t_1 satisfying conditions (a)—(c) do exist. Let $\mathbf{x} = f(t)$, and C the circle centered at $\mathbf{x}$ with radius 2^{-m}. Then, from

$$\begin{aligned} |\mathbf{x} - \mathbf{z}| &\le |\mathbf{x} - \mathbf{w}| + |\mathbf{w} - \mathbf{z}| \le d + 2^{-q(n)} + 2^{-q(n)+1} \\ &\le \mathrm{dist}(\mathbf{z}, \Gamma) + 2^{-q(n)+2} \le \mathrm{dist}(\mathbf{z}, \Gamma) + 2^{-(p(n)+3)}, \end{aligned}$$

we see that the circle C and point $\mathbf{x}$ satisfy the condition of Lemma 4.1. Now let $\mathbf{x}_0$ and $\mathbf{x}_1$ be the points in $\Gamma \cap C$ as defined before Lemma 4.1; that is, $\mathbf{x}$ lies in $\Gamma_{\mathbf{x}_0\mathbf{x}_1}$, and the whole section $\Gamma_{\mathbf{x}_0\mathbf{x}_1}$ lies inside the circle C. Also let $f(s_0) = \mathbf{x}_0$ and $f(s_1) = \mathbf{x}_1$. Then, if we select t_0 and t_1 such that $|t_0 - s_0| \le 2^{-r(n)}$ and $|t_1 - s_1| \le 2^{-r(n)}$, it can be verified that the conditions (a)—(c) of step (4) are satisfied.

Next, for Step (5), we need to verify that the orientation of the three points $\mathbf{w}_1$, $\mathbf{v}$, $\mathbf{w}_0$ around $\mathbf{w}$ is well-defined; that is, the angles $\angle \mathbf{w}_1\mathbf{w}\mathbf{v}$ and $\angle \mathbf{w}_1\mathbf{w}\mathbf{w}_0$ are not equal to zero and are not equal to each other. From conditions of steps (3) and (4), we know that points $\mathbf{w}_1, \mathbf{v}, \mathbf{w}_0$ all lie within distance $2^{-q(n)+2}$ of the circle C. In addition, since $|\mathbf{x} - \mathbf{z}| \le \mathrm{dist}(\mathbf{z}, \Gamma) + 2^{-q(n)+2}$, we know that $\mathrm{dist}(\mathbf{y}, \Gamma) \ge 2^{-m} - 2^{-q(n)+2}$. It follows that both $|\mathbf{v} - \mathbf{w}_0|$ and $|\mathbf{v} - \mathbf{w}_1|$ are at least $2^{-m} - 2^{-q(n)+3} \ge 2^{-q(n)+4}$, and hence the angle $\angle \mathbf{w}_1\mathbf{w}\mathbf{v}$ is greater than $2^{-q(n)}$, and the difference between $\angle \mathbf{w}_1\mathbf{w}\mathbf{v}$ and $\angle \mathbf{w}_1\mathbf{w}\mathbf{w}_0$ is greater than $2^{-q(n)}$.

Furthermore, we note that

$$
\begin{aligned}
|f(t_0)-\mathbf{x}| &\geq |\mathbf{w}_0-\mathbf{w}|-|f(t_0)-\mathbf{w}_0|-|\mathbf{w}-\mathbf{x}| \\
&\geq 2^{-m}-2^{-q(n)+2}-2^{-q(n)}-2^{-q(n)} \\
&\geq 2^{-m}-2^{-q(n)+3} \geq 2^{-(p(n)+2)}.
\end{aligned}
$$

From this, we get $|t_0-t_1| \geq |t_0-t| \geq 2^{-p(p(n)+2)}$ and, by the assumption about the inverse modulus p, $|f(t_0)-f(t_1)| \geq 2^{p(p(p(n)+2))} = 2^{-q(n)+5}$, and $|\mathbf{w}_0-\mathbf{w}_1| \geq 2^{-q(n)+4}$. It follows that the angle between $\angle \mathbf{w}_1\mathbf{w}\mathbf{w}_0$ is greater than $2^{-q(n)}$. This shows that the orientation of the points $\mathbf{w}_1$, $\mathbf{v}$, $\mathbf{w}_0$ around $\mathbf{w}$ is well-defined and, in fact, can be determined in time polynomial in n.

Correctness. To prove that the algorithm always outputs the correct answer, we need, by Lemma 4.1, to verify that the orientation of points $\mathbf{x}_1, \mathbf{y}, \mathbf{x}_0$ around $\mathbf{x}$ must be the same as that of the points $\mathbf{w}_1, \mathbf{v}, \mathbf{w}_0$ around $\mathbf{w}$. We note that, although $\mathbf{w}_0$ is intended to be an approximation to the point $\mathbf{x}_0$, the conditions (a)—(c) of step (4) do not guarantee this property. (In fact, the points $\mathbf{x}_0$ and $\mathbf{x}_1$ are, in general, like roots of a polynomial-time computable function, hard to compute from Γ and C.) Nevertheless, from the above analysis, we know that all points $\mathbf{x}_0, \mathbf{x}_1, \mathbf{w}_0$ and $\mathbf{w}_1$ lie within $2^{-q(n)+2}$ of the circle C. In other words, let C_1 be the circle centered at $\mathbf{x}$ with radius $2^{-m}+2^{-q(n)+2}$, C_0 be the circle centered at $\mathbf{x}$ with radius $2^{-m}-2^{-q(n)+2}$, and C' be the circle centered at $\mathbf{z}$ with radius $|\mathbf{z}-\mathbf{x}|-2^{-q(n)+2}$. Then, these points all lie between C_1 and C_0 but outside C', because $\mathrm{dist}(\mathbf{z}, \Gamma) \geq |\mathbf{z}-\mathbf{x}|-2^{-q(n)+2}$ (see Figure 2). Now, let Λ_0 be the section of Γ between $f(t_0)$ and $\mathbf{x}_0$ (i.e., Λ_0 is equal to $\Gamma_{\mathbf{x}_0 f(t_0)}$ if $t_0 \in I(s_0, s_1)$, or equal to $\Gamma_{f(t_0)\mathbf{x}_0}$ if $t_0 \notin I(s_0, s_1)$), and Λ_1 be the section of Γ between $f(t_1)$ and $\mathbf{x}_1$ (i.e., Λ_1 is equal to $\Gamma_{f(t_1)\mathbf{x}_1}$ if $t_1 \in I(s_0, s_1)$, or equal to $\Gamma_{\mathbf{x}_1 f(t_1)}$ if $t_1 \notin I(s_0, s_1)$). Then we claim that for any points $\mathbf{v}_0 \in \Lambda_0$, $\mathbf{v}_1 \in \Lambda_1$, the distances $|\mathbf{v}_0-\mathbf{v}_1|$, $|\mathbf{v}_0-\mathbf{x}|$ and $|\mathbf{v}_1-\mathbf{x}|$ are all greater than $2^{-q(n)+5}$. To see this, assume that $\mathbf{v}_0 = f(u_0)$ and $\mathbf{v}_1 = f(u_1)$. Then, from the fact that $|f(t_0)-\mathbf{x}|$ and $|f(s_0)-\mathbf{x}|$ are greater than $2^{-m}-2^{-q(n)+2} \geq 2^{-(p(n)+2)}$, we see that both $|s_0-t|$ and $|t_0-t|$ are greater than $2^{-p(p(n)+2)}$. It follows that $|u_0-t| \geq 2^{-p(p(n)+2)}$ since $u_0 \in I(s_0, t_0)$. By the assumption of the inverse modulus p, we get $|\mathbf{v}_0-\mathbf{x}| \geq 2^{-p(p(p(n)+2))} = 2^{-q(n)+5}$. Similarly, we have $|\mathbf{v}_1-\mathbf{x}| \geq 2^{-q(n)+5}$. Also, $|u_0-u_1| \geq |u_0-t| \geq 2^{-p(p(n)+2)}$, and we also have $|\mathbf{v}_1-\mathbf{v}_1| \geq 2^{-q(n)+5}$.

Now, we show that the orientation of points $\mathbf{x}_1, \mathbf{y}, \mathbf{x}_0$ around $\mathbf{x}$ and the orientation of $f(t_1), \mathbf{y}, f(t_0)$ around $\mathbf{x}$ must be identical. First, we observe that halflines $\overrightarrow{\mathbf{x}\mathbf{x}_0}$ and $\overrightarrow{\mathbf{x}\mathbf{x}_1}$ divide the area between circles C_0, C_1 and outside

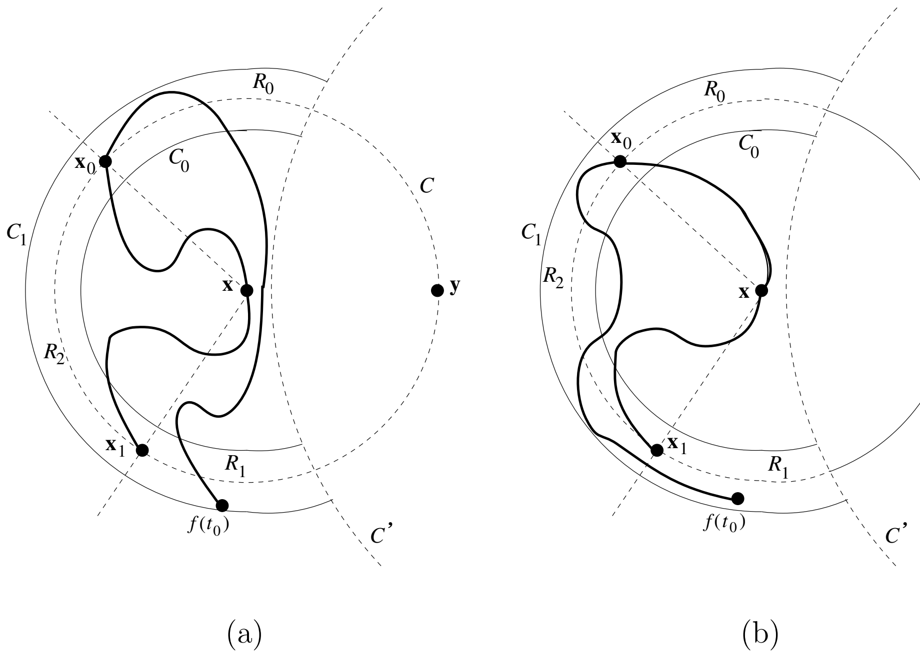

Fig. 2. (a) Λ_0 passes near $\mathbf{x}$; (b) Λ_0 passes near $\mathbf{x}_1$.

C' into three parts. We let R_0 be the part between $\overrightarrow{\mathbf{x}\mathbf{x}_0}$ and C', R_1 be the part between $\overrightarrow{\mathbf{x}\mathbf{x}_1}$ and C', and R_2 the part between the two halflines $\overrightarrow{\mathbf{x}\mathbf{x}_0}$ and $\overrightarrow{\mathbf{x}\mathbf{x}_1}$. Now, if the orientation of points $\mathbf{x}_1, \mathbf{y}, \mathbf{x}_0$ around $\mathbf{x}$ and the orientation of $\mathbf{w}_1, \mathbf{v}, \mathbf{w}_0$ around $\mathbf{w}$ are not the same, then one of the following cases must occur:

Case 1. $\mathbf{w}_0 \in R_1$. Then, $f(t_0)$ is in R_1 or within distance of $2^{-q(n)}$ of the halfline $\overrightarrow{\mathbf{x}\mathbf{x}_1}$. We note that Λ_0 does not intersect $\Gamma_{\mathbf{x}_0\mathbf{x}_1}$ except at $\mathbf{x}_0$. Therefore, in order to move from $\mathbf{x}_0$ in region R_0 to $f(t_0)$ in region R_1, Λ_0 must either pass through the narrow area between $\mathbf{x}$ and C' (as shown in Figure 2(a)) or pass through the area between $\mathbf{x}_1$ and C_1 (as shown in Figure 2(b)). In either way, it contradicts the above claim.

Case 2. $\mathbf{w}_1 \in R_0$. We can show, similar to Case 1, that Λ_1 must pass through the area between $\mathbf{x}$ and C' or pass through the area between $\mathbf{x}_0$ and C_1, contradicting the claim.

Case 3. $\mathbf{w}_0, \mathbf{w}_1 \in R_2$. In this case, both $\mathbf{w}_0$ and $\mathbf{w}_1$ are in R_2, but $\mathbf{w}_0$ is closer than $\mathbf{w}_1$ to $\mathbf{x}_1$. Thus, either (i) Λ_0 or Λ_1 passes through the narrow area between $\mathbf{x}$ and C' (i.e., Λ_0 and Λ_1 intersects line segment $\overline{\mathbf{xy}}$), or (ii) neither Λ_0 nor Λ_1 intersects line segment $\overline{\mathbf{xy}}$. The case (i) is similar to Cases 1 and 2 (see Figure 3(a)), and contradicts the claim. In the case (ii), since both $\mathbf{w}_0$ and $\mathbf{w}_1$ are in region R_2, Λ_0 and Λ_1 must pass through each other within R_2 at some points (see Figure 3(b)). Again, this case contradicts the claim.

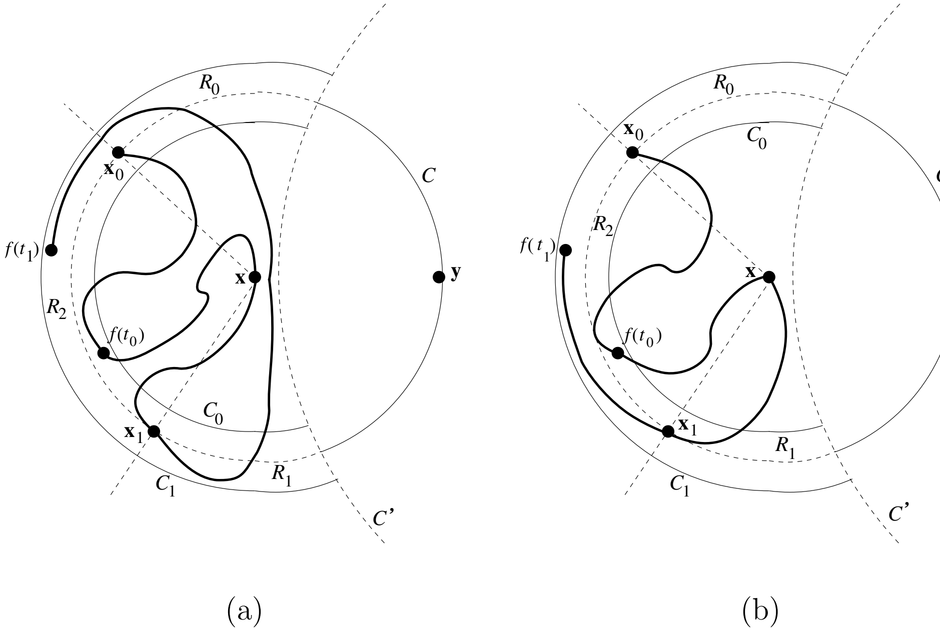

Fig. 3. (a) Λ_1 passes near $\mathbf{x}$; (b) Λ_0 and Λ_1 pass near each other.

Thus, we conclude that the orientation of points $\mathbf{x}_1, \mathbf{y}, \mathbf{x}_0$ around $\mathbf{x}$ and the orientation of points $\mathbf{w}_1, \mathbf{v}, \mathbf{w}_0$ around $\mathbf{w}$ must be identical, and the algorithm always outputs the correct answer.

Complexity. It is shown in Chou and Ko [4] that the distance function $\delta(\mathbf{z}) = \text{dist}(\mathbf{z}, \Gamma)$ is computable in polynomial time relative to an oracle in NP. Thus, Steps (1) and (2) are computable in polynomial time relative to an oracle in NP. Also, it is clear that Step (3) is computable in polynomial time.

For step (4), we note that for any given t_0 and t_1, condition (a) can be verified in polynomial time. Furthermore, conditions (b) and (c) can be verified in polynomial time with the help of an oracle in NP. In fact, let M be the oracle Turing machine that computes the function f, and let $M^u(k)$ denote the dyadic rational in $\mathbb{D}_k$ output by M using the standard Cauchy function b_u as the oracle. Then, define X to be the set of all dyadic rationals $s \in \mathbb{D}_{r(n)}$ such that (i) $\delta(s,t) \le 2^{-n+1}$, and (ii) $(\exists u \in \mathbb{D}_{r(n)})\, [u \in I(s,t), |M^u(q(n)) - \mathbf{w}| > 2^{-(p(n)+1)} + 2^{-q(n)+2}\}]$. It is clear that $X \in NP$. We can now use a binary search to find the dyadic rational t_0 not in X that has the least $\delta(t, t_0)$. We note that such a point t_0 must also satisfy the condition (b) that $\left| |M^{t_0}(q(n)) - \mathbf{w}| - 2^{-(p(n)+1)} \right| \le 2^{-q(n)+2}$, and we can assign $\mathbf{w}_0 := M^{t_0}(q(n))$. Similarly, we can use a binary search to find t_1 and $\mathbf{w}_1$ using a similar oracle set in NP.

Finally, for Step (5), we already proved that the orientation of the three points $\mathbf{w}_1, \mathbf{v}, \mathbf{w}_0$ around $\mathbf{w}$ is well defined, with the angles $\angle \mathbf{w}_1\mathbf{w}\mathbf{v}$, $\angle \mathbf{v}\mathbf{w}\mathbf{w}_0$ and $\angle \mathbf{w}_0\mathbf{w}\mathbf{w}_1$ all greater than $2^{-q(n)}$. Thus, the orientation can be determined in polynomial time by calculating these angles correct within $2^{-q(n)-1}$ and compare their values.

The above analysis completes the proof of the following theorem:

Theorem 4.2 *Assume that $f : [0,1] \to \mathbb{R}^2$ is a polynomial-time representation of a Jordan curve Γ and that it has a polynomial inverse modulus of continuity. Then, the membership problem about Γ is solvable in polynomial time relative to an oracle in NP.*

Corollary 4.3 *Assume that $P = NP$. Then, a two-dimensional region, whose boundary is a polynomial-time computable Jordan curve with a polynomial inverse modulus of continuity, is P-recognizable.*

We note that, in contrast to Theorem 4.2, it is not hard to verify that the curve constructed for the UP lower bound for the membership problem, as given in Theorem 7.3 of Chou and Ko [4], actually has a polynomial inverse modulus. Thus, we have the following stronger results about the membership problem than that of Chou and Ko [4].

Corollary 4.4 *In the following, (a)⇒(b)⇒(c):*

(a) $P = NP$.

(b) For any Jordan curve Γ that has a polynomial-time representation with a polynomial inverse of modulus, the membership problem of Γ is solvable in polynomial time.

(c) $P = UP$.

5 Path problem

Recall that the path problem asks for a path in the given domain S that connects two given points and has a distance 2^{-m} from the boundary Γ of S. However, it is in general undecidable whether such a path exists, even if Γ is polynomial-time computable. In this section, we show that if Γ has a polynomial-time representation which also has a polynomial inverse modulus, then a path with distance $2^{-q(n)}$ away from Γ, for some polynomial q, must exist between two points which have distance 2^{-n} away from Γ.

Theorem 5.1 *Assume that Γ is a Jordan curve represented by a polynomial-time computable function f which has a polynomial inverse modulus. Then, there exist a polynomial function q and a constant n_0 satisfying the following conditions: For any integer $n > n_0$ and any two points $\mathbf{z}_0, \mathbf{z}_1$ in $\mathrm{Int}(\Gamma)$ with*

$\mathrm{dist}(\mathbf{z}_0, \Gamma) > 2^{-n}$ and $\mathrm{dist}(\mathbf{z}_1, \Gamma) > 2^{-n}$, there is a path from $\mathbf{z}_0$ to $\mathbf{z}_1$ that lies in $\mathrm{Int}(\Gamma)$ with distance at least $2^{-q(n)}$ from the curve Γ.

Proof. Assume that p is a polynomial function that bounds the runtime of function f and the inverse modulus of f. Let $q(n) = p(p(n)) + 2$. As in the proof of Theorem 4.2, we regard Γ as a directed curve.

First, we note that if $|\mathbf{z}_0 - \mathbf{z}_1| \leq 2^{-n}$, then it is obvious that the line segment $\overline{\mathbf{z}_0\mathbf{z}_1}$ has distance at least $2^{-(n+1)} \geq 2^{-q(n)}$ from Γ, and the required condition is satisfied. Therefore, we may assume that $|\mathbf{z}_0 - \mathbf{z}_1| > 2^{-n}$.

We first draw a line L_0 passing point $\mathbf{z}_0$ that is perpendicular to the line segment $\overline{\mathbf{z}_0\mathbf{z}_1}$. Let $\mathbf{v}_0$ and $\mathbf{w}_0$ be the intersection points of L_0 and Γ such that $\overline{\mathbf{v}_0\mathbf{w}_0}$ lies entirely in $\Gamma \cup \mathrm{Int}(\Gamma)$ and contains point $\mathbf{z}_0$. Then $\overline{\mathbf{v}_0\mathbf{w}_0}$ divides $\mathrm{Int}(\Gamma)$ into two parts, with $\mathbf{z}_1$ belonging to one part. Move L_0 toward the part that does not contain $\mathbf{z}_1$ for distance $2^{-q(n)+2}$. Let $\mathbf{x}_0$ and $\mathbf{y}_0$ be the intersection of this line L'_0 and Γ such that $\overline{\mathbf{x}_0\mathbf{y}_0}$ lies entirely in $\Gamma \cup \mathrm{Int}(\Gamma)$ and $\mathrm{dist}(\mathbf{z}_0, \overline{\mathbf{x}_0\mathbf{y}_0}) = 2^{-q(n)+2}$. Define lines L_1, L'_1 and points $\mathbf{x}_1$ and $\mathbf{y}_1$ around point $\mathbf{z}_1$ in a similar way. We arrange the names of these four points $\mathbf{x}_0, \mathbf{y}_0, \mathbf{x}_1, \mathbf{y}_1$ in such a way that the directed curve Γ goes through these points in the order of $\mathbf{x}_0$, $\mathbf{x}_1$, $\mathbf{y}_1$, $\mathbf{y}_0$, and back to $\mathbf{x}_0$ (We note that it is not possible for Γ to go in the order of $\mathbf{x}_0$, $\mathbf{x}_1$, $\mathbf{y}_0$, $\mathbf{y}_1$, since such a curve must leave either $\mathbf{z}_0$ or $\mathbf{z}_1$ in the exterior of Γ.) We also re-arrange the names of $\mathbf{v}_0, \mathbf{w}_0, \mathbf{v}_1$ and $\mathbf{w}_1$ so that $\mathbf{v}_0$, $\mathbf{x}_0$ are on the same side of $\mathbf{z}_0$, and $\mathbf{w}_0$, $\mathbf{y}_0$ are on the other side of $\mathbf{z}_0$; and that $\mathbf{v}_1$, $\mathbf{x}_1$ are on the same side of $\mathbf{z}_1$, and $\mathbf{w}_1$, $\mathbf{y}_1$ are on the other side.

We let S denote the part of $\mathrm{Int}(\Gamma)$ that is enclosed between $\overline{\mathbf{x}_0\mathbf{y}_0}$ and $\overline{\mathbf{x}_1\mathbf{y}_1}$. In other words, if we let Λ_0 be the section of Γ from $\mathbf{x}_0$ to $\mathbf{x}_1$, and Λ_1 be the section of Γ from $\mathbf{y}_1$ to $\mathbf{y}_0$, then S is the domain whose boundary is $\overline{\mathbf{y}_0\mathbf{x}_0}$, Λ_0, $\overline{\mathbf{x}_1\mathbf{y}_1}$, plus Λ_1. Let this boundary of S be Γ_1. (See Figure 4).

Claim 1. $\mathrm{dist}(\Lambda_0, \Lambda_1) > 2^{-q(n)+2}$.

Proof. Let s_0, t_0, s_1, and t_1 be real numbers in $[0, 1]$ such that $f(s_0) = \mathbf{x}_0$, $f(t_0) = \mathbf{y}_0$, $f(s_1) = \mathbf{x}_1$, and $f(t_1) = \mathbf{y}_1$. Without loss of generality, assume that $0 \leq s_0 < s_1 < t_1 < t_0 < 1$. Since $\mathbf{x}_0$ and $\mathbf{y}_0$ lie on the two sides of $\mathbf{z}_0$ with distance at least 2^{-n} from $\mathbf{z}_0$, we must have $|\mathbf{x}_0 - \mathbf{y}_0| > 2^{-n+1} - 2^{-q(n)+3} > 2^{-n}$. Similarly, we also have $|\mathbf{x}_1 - \mathbf{y}_1| > 2^{-n}$. From the assumption that f is computable in time $p(n)$, we have that $\delta(s_0, t_0) > 2^{-p(n)}$ and $\delta(s_1, t_1) > 2^{-p(n)}$. It follows that, for any $s \in [s_0, s_1]$ and $t \in [t_1, t_0]$, $\delta(s, t) > 2^{-p(n)}$. Then, by the property of the inverse modulus p, we see that $\mathrm{dist}(\Lambda_0, \Lambda_1) > 2^{-p(p(n))} = 2^{-q(n)+2}$. □

Claim 2. $\mathbf{v}_0 \in \Lambda_0$ and $\mathbf{w}_0 \in \Lambda_1$.

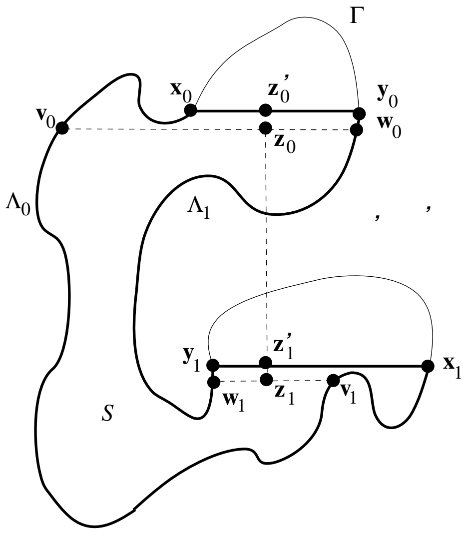

Fig. 4. Domain S.

Proof. According to the definition of line L_0', it is on the side of L_0 that does not contain $\mathbf{z}_1$. Therefore, when we trace the curve Γ from $\mathbf{x}_0$ to $\mathbf{x}_1$, it must first meet $\mathbf{v}_0$ before $\mathbf{x}_1$. Thus, $\mathbf{v}_0 \in \Lambda_0$. Similarly, if we trace Γ backward from $\mathbf{y}_0$ toward $\mathbf{y}_1$, then we must meet $\mathbf{w}_0$ before $\mathbf{y}_1$, and so $\mathbf{w}_0 \in \Lambda_1$. □

Let $\mathbf{z}_0'$ be the point on $\overline{\mathbf{x}_0\mathbf{y}_0}$ that has $|\mathbf{z}_0 - \mathbf{z}_0'| = 2^{-q(n)+2}$, and $\mathbf{z}_1'$ the point on $\overline{\mathbf{x}_1\mathbf{y}_1}$ that has $|\mathbf{z}_1 - \mathbf{z}_1'| = 2^{-q(n)+2}$.

Claim 3. (a) Let R_0 be the rectangle formed by the line segments $\overline{\mathbf{z}_0\mathbf{z}_0'}$ and $\overline{\mathbf{z}_0'\mathbf{y}_0}$. Then, $\Lambda_0 \cap R_0 = \emptyset$.

(b) Let R_1 be the rectangle formed by the line segments $\overline{\mathbf{z}_1\mathbf{z}_1'}$ and $\overline{\mathbf{z}_1'\mathbf{x}_1}$. Then, $\Lambda_1 \cap R_1 = \emptyset$.

(c) Let R_2 be the rectangle formed by the line segments $\overline{\mathbf{z}_1\mathbf{z}_1'}$ and $\overline{\mathbf{z}_1'\mathbf{y}_1}$. Then, $\Lambda_0 \cap R_2 = \emptyset$.

Proof. (a) For the sake of contradiction, assume that $\Lambda_0 \cap R_1 \neq \emptyset$. It is clear that Λ_0 does not run through the line segment $\overline{\mathbf{z}_0'\mathbf{y}_0}$. In addition, from Claim 1, we know that $\mathrm{dist}(\Lambda_0, \mathbf{y}_0) > 2^{-q(n)+2}$. Therefore, Λ_0 must meet the halfline $\overrightarrow{\mathbf{z}_0\mathbf{w}_0}$ at some point $\mathbf{w}$. From Claim 2, we know that this intersection point $\mathbf{w}$ must lie on $\overrightarrow{\mathbf{z}_0\mathbf{w}_0}$ beyond the point $\mathbf{w}_0$. Furthermore, since $\mathrm{dist}(\Lambda_0, \Lambda_1) > 2^{-q(n)+2}$, $\mathbf{w}$ must be at least $2^{-q(n)+2}$ beyond $\mathbf{w}_0$, and $\mathbf{y}_0$ must be at least $2^{-q(n)+3}$ beyond the point $\mathbf{w}_0$ in L_0' (see Figure 5). Now, consider the section of Λ_1 from $\mathbf{w}_0$ to $\mathbf{y}_0$. Since Λ_1 cannot pass through $\overline{\mathbf{z}_0'\mathbf{y}_0}$, it must go from $\mathbf{w}_0$ to $\mathbf{y}_0$ passing through the area between $\mathbf{w}$ and $\overline{\mathbf{z}_0'\mathbf{y}_0}$. However, this means that $\mathrm{dist}(\mathbf{w}, \Lambda_1) \leq 2^{-q(n)+2}$, contradicting Claim 1.

The proofs for parts (b) and (c) are similar. □

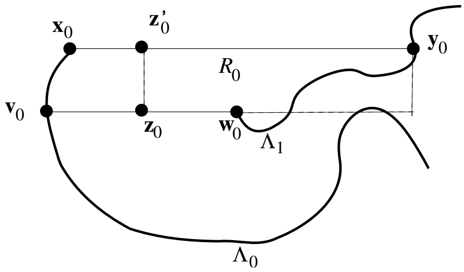

Fig. 5. Domain S.

Now, define $T = \{\mathbf{z} \in S : \mathrm{dist}(\mathbf{z}, \Gamma_1) > 2^{-q(n)}\}$. Then, $\mathbf{z}_0 \in T$. Let C be the connected component of T that contains $\mathbf{z}_0$. We claim that $\mathbf{z}_1$ belongs to the same component C and, hence, there is a path in T connecting $\mathbf{z}_0$ and $\mathbf{z}_1$, and the theorem follows.

To see this, let $\mathbf{z}_0''$ be the point on $\overline{\mathbf{z}_0\mathbf{z}_0'}$ with $|\mathbf{z}_0' - \mathbf{z}_0''| = 2^{-q(n)}$. Then, $\mathbf{z}_0''$ is on the boundary of C, and the boundary of C around $\mathbf{z}_0''$ is a line segment parallel to L_0. We follow this line segment in the direction toward $\mathbf{y}_0$. Then, by Claim 3(a), it must reach a point $\mathbf{z}_2$ that is near Λ_1 within distance $2^{-q(n)}$. (This line segment must get near $\mathbf{y}_0 \in \Lambda_1$ within distance $2^{-q(n)}$ if not any other point in Λ_1.) Then, the boundary of C continuous on "parallel" to the curve Λ_1. By Claim 1, it must then meet a point $\mathbf{z}_3$ that is near the line segment $\overline{\mathbf{x}_1\mathbf{y}_1}$ within distance $2^{-q(n)}$. (Note that the boundary of C does not necessarily go continuously parallel to Λ_1. It may "jump" from one section of Λ_1 to another section of Λ_1. However, it will be always within distance $2^{-q(n)}$ from some point in Λ_1 before it gets near the line segment $\overline{\mathbf{x}_1\mathbf{y}_1}$.) By Claim 3(b), we know that this point $\mathbf{z}_3$ has distance at least $2^{-q(n)+2}$ away from the line segment $\overline{\mathbf{z}_1'\mathbf{x}_1}$. Thus, $\mathbf{z}_3$ must be close to a point $\mathbf{u} \in \overline{\mathbf{z}_1'\mathbf{y}_1}$. Now, by Claim 3(c), the boundary of C must contain the line segment $\overline{\mathbf{u}\mathbf{z}_1''}$. Therefore, $\mathbf{z}_1''$ is on the boundary of C, and it is obvious that $\mathbf{z}_1$ is in the component C. This completes the claim and the theorem. □

From Corollary 4.3 of Chou and Ko [5], we get the following result about the path problem:

Corollary 5.2 *Assume that* Γ *is a Jordan curve represented by a polynomial-time computable function which has a polynomial inverse modulus of continuity. Then, there exist an oracle Turing machine* M*, two polynomial functions* $p(n), r(n)$ *and a constant* n_0 *such that the following holds:*

> *For any oracles* $\phi_1, \phi_2, \psi_1, \psi_2$ *representing, respectively, two points* $\mathbf{z}_0, \mathbf{z}_1$ *in* $\mathrm{Int}(\Gamma)$ *and any input* $n \geq n_0$*, if* $\mathrm{dist}(\mathbf{z}_0, \Gamma) > 2^{-n}$ *and* $\mathrm{dist}(\mathbf{z}_0, \Gamma) > 2^{-n}$*, then* M *finds, using* $p(n)$ *cells of storage space, a path* π *between* $\mathbf{z}_0$ *and* $\mathbf{z}_1$ *that lies within* $\mathrm{Int}(\Gamma)$ *with distance* $\mathrm{dist}(\pi, \Gamma) \geq 2^{-r(n)}$.

Theorem 6.1 of Chou and Ko [5] showed a $\#P$ lower bound for the shortest path problem. More precisely, it states that, if $FP \neq \#P$, then there exists a domain S which has a polynomial-time computable boundary with respect to which the shortest path problem is not solvable in polynomial time. We observe that the boundary Γ of S construction in this proof actually has a polynomial inverse modulus of continuity. This means that the $\#P$ lower bound also holds for domains whose boundaries have polynomial-time representations with polynomial inverse moduli.

Corollary 5.3 *In the following, (a)⇒(b)⇒(c):*

(a) $P = PSPACE$.

(b) For any Jordan curve Γ which has a polynomial-time representation with a polynomial inverse modulus, the path problem (with $m = q(n)$ for some polynomial q) is solvable in polynomial time.

(c) $FP = \#P$.

The computation of analytic continuation of a function defined on a two-dimensional domain is an important problem in computational complex analysis. A simple approach, originated by Weierstrass (see, e.g., Henrici [8]), is to compute the analytic continuation through a path from a source point to the target point. This approach requires the fast computation of a path inside the domain with a reasonable distance from the boundary of the domain. Our result of Theorem 5.1 gives an exponential space upper bound for this approach.

Theorem 5.4 *Assume that S is a simply connected domain on the two-dimensional plane whose boundary Γ is represented by a polynomial-time computable function which has a polynomial modulus of continuity, and that $\mathbf{z}_0$ is a point in $\mathrm{Int}(\Gamma)$ with $\mathrm{dist}(\mathbf{z}_0, \Gamma) = 2^{-n_0}$. Also assume that function g is analytic on S, and that the power series of g at $\mathbf{z}_0 \in \mathrm{Int}(\Gamma)$ is polynomial-time computable. Then, there exists a polynomial $q(n)$ such that for any integer $n \geq n_0$ and any point $\mathbf{z} \in \mathrm{Int}(\Gamma)$ with $\mathrm{dist}(\mathbf{z}, \Gamma) \geq 2^{-n}$, the power series of G at $\mathbf{z}$ is computable using at most $2^{q(n+k)}$ cells.*

Proof. Our proof is based on the analysis in Section 3.6 of Henrici [8], and we only present a simple sketch here.

From Corollary 5.2, we can find, in polynomial space a path π connecting $\mathbf{z}_0$ and $\mathbf{z}$ with $\mathrm{dist}(\pi, \Gamma) \geq 2^{-p(n)}$ for some polynomial p. Then, we select points $\mathbf{z}_0, \mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_m$, where $m = 2^{O(q(n))}$, such that (i) $|\mathbf{z}_{k-1} - \mathbf{z}_k|$ are all equal and less than $h = 2^{-q(n)}$ for $k = 1, \ldots, m-1$, (ii) $\mathbf{z}_m = \mathbf{z}$ and $|\mathbf{z}_{m-1} - \mathbf{z}_m| \leq h$, and (iii) the radius of convergence of the power series of g at point $\mathbf{z}_k$ is greater than or equal to h. Let $\sum_{n=0}^{\infty} a_n(\mathbf{z}_k)\mathbf{w}^n$ be the power series of g at point $\mathbf{z}_k$, for $k = 0, 1, \ldots, m$.

From Henrici's analysis, we know that there exists a constant c such that an approximation to $a_n(\mathbf{z}_k)$ can be computed from the first $n \cdot 2^{cm}$ terms of the power series of g at $\mathbf{z}_0$ so that the error is bounded by 2^{-n}. To be more precise, define a double sequence $\{a_n^{(k)}\}$, for $k = 0, 1, \ldots, m$ and $n = 1, 2, \ldots$, recursively as follows:

$$a_n^{(k)} := \sum_{i=n}^{n2^c} \binom{i}{n} a_i^{(k-1)} h^{i-n}, \tag{1}$$

for $k = 1, 2, \ldots, m$, and $n = 1, 2, \ldots$. Then, it satisfies the condition that $|a_n^{(m)} - a_n(\mathbf{z}_m)| \leq 2^{-(n+1)}$.

In the above formula (1), it is assumed that all arithmetic operations are calculated with the exact real-number arithmetics. It is not hard to verify, though, that if we carry out all calculations over dyadic rationals of 2^{dn} bits for some constant $d > 0$, the extra error is bounded by $2^{-(n+1)}$. Thus, this procedure works in our finite-precision model.

Finally, we note that the value of $a_i^{(k)}$ can be computed recursively using formula (1). This recursive computation can be viewed as a computation tree of height m and width 2^{cm}, with each node $a_i^{(k)}$ at level k having $i2^c$ children. It is easy to see that this computation tree can be traversed in the depth-first order using only storage space $O(m)$ (at any time of the traversal, the machine only needs to store one node at each level). At each node of the tree, the local computation can be done using only $2^{O(q(n))}$ cells of storage. Therefore, the total storage space required is only $m \cdot 2^{O(q(n))} = 2^{O(q(n))}$. □

6 Conclusion

In the previous sections, we studied the notion of polynomial inverse modulus of two-dimension Jordan curves. We presented evidence that the property of polynomial inverse modulus helps to reduce the complexity of some important problems about two-dimensional domains, including the inverse function problem, the membership problem, the path problem, and the analytic continuation problem. In addition, we observe that most Jordan curves constructed, in the earlier studies of two-dimensional domains, for the lower bound results for these problems satisfy this property of polynomial inverse modulus of continuity. This includes the constructions for the lower bounds for the membership problem and the path problem, as well as the fractals constructed for the nonrecursiveness result for the area problem (see Ko [10] and Ko and Weihrauch [11]).

From these studies, we found that the notion of polynomial-time computable Jordan curves is probably too general, and some of these curves might be too complicated to understand its complexity issues. On the other hand, polynomial-time computable Jordan curves with polynomial inverse moduli are a rich, natural subclass, with respect to which we are able to get more accurate characterization of the complexity of some fundamental problems related to two-dimensional domains. It suggests that this notion is worth further study to get better understanding of the mathematical and algorithmic properties of two-dimensional domains.

References

[1] Berg, G., W. Julian, B. Mines and F. Richman, *The constructive Jordan curve theorem*, Rocky Mountain J. Math. **5**, 225–234, 1975.

[2] Braverman, M., *Hyperbolic Julia sets are poly-time computable*, Proceedings of the 6th Workshop on Computability and Complexity in Analysis, Vol. 120, Electron. Notes Theoret. Comput. Sci., Amsterdam, 17–30, 2005.

[3] Braverman, M. and M. Yampolsky, *Non-computable Julia sets*, J. Amer. Math. Soc. **19**(3), 551–578, 2006.

[4] Chou, A. W. and K.-I. Ko, *Computational Complexity of Two-Dimensional Regions*, SIAM. J. Comput. **24**, 923–947, 1995.

[5] Chou, A. W. and K.-I. Ko, *On the complexity of finding paths in a two-dimensional domain I: Shortest paths*, Math. Logic Quart. **50**(6), 551–572, 2004.

[6] Chou, A. W. and K.-I. Ko, *On the complexity of finding paths in a two-dimensional domain II: Piecewise straight-line paths*, Proceedings of the 6th Workshop on Computability and Complexity in Analysis, Vol. 120, Electron. Notes Theoret. Comput. Sci., 45–57, 2005.

[7] Du, D.-Z. and K.-I. Ko, "Theory of Computational Complexity", John Wiley & Sons, New York, 2000.

[8] Henrici, P., "Applied and Computational Complex Analysis", Vol. 1-3. John Wiley & Sons, New York, 1974.

[9] Ko, K.-I., "Complexity Theory of Real Functions", Birkhäuser, Boston, 1991.

[10] Ko, K.-I., *A polynomial-time computable curve whose interior has a nonrecursive measure*, Theor. Comput. Sci. **145**, 241–270, 1995.

[11] Ko, K.-I. and K. Weihrauch, *On the measure of two-dimensional regions with polynomial-time computable boundaries*, Proceedings of 11th IEEE Conference on Computational Complexity, 150–159, 1996.

[12] Ko, K.-I. and F. Yu, *On the complexity of computing the logarithm and square root functions on a complex domain*, Proceedings of 11th International Computing and Combinatorics Conference (COCOON), Vol. 3595, Lecture Notes in Computer Science, Springer, 349–358, 2005.

[13] Müller, N. T., *Constructive aspects of analytic functions*, Proceedings of Workshop on Computability and Complexity in Analysis, Vol. 190, Informatik Berichte, FernUniversität Hagen, 105–114, 1995.

[14] Pour-El, M. B. and J. I. Richards, "Computability in Analysis and Physics", Perspectives in Mathematical Logic, Springer, Berlin, 1989.

[15] Rettinger, R., *A fast algorithm for Julia sets of hyperbolic rational functions*, Proceedings of 6th Workshop on Computability and Complexity in Analysis, Vol. 120, Electron. Notes Theoret. Comput. Sci., 145–157, 2005.

[16] Rettinger, R. and K. Weihrauch, *The computational complexity of some Julia sets*, Proceedings of the 35th ACM Symposium on Theory of Computing, pp. 177–185, 2003.

[17] Weihrauch, K., "Computable Analysis", Springer, Berlin, 2000.

[18] Yu, F., A. Chou and K.-I. Ko, *On the complexity of finding circumscribed rectangles for a two-dimensional domain*, Proceedings of 2nd International Conference on Computability and Complexity in Analysis, Vol. 326, Informatik Berichte, FernUniversität in Hagen, 341–355, 2005.