



ELSEVIER

Available online at www.sciencedirect.com

ScienceDirect

**Electronic Notes in
Theoretical Computer
Science**

Electronic Notes in Theoretical Computer Science 167 (2007) 95–115

www.elsevier.com/locate/entcs

On the Complexity of the Pancake Problem

Fuxiang Yu^{1,2}*Department of Computer Science
State University of New York
Stony Brook, NY, U.S.A.*

Abstract

We study the computational complexity of finding a line that bisects simultaneously two sets in the two-dimensional plane, called the *pancake problem*, using the oracle Turing machine model of Ko. We also study the basic problem of bisecting a set at a given direction. Our main results are: (1) the complexity of bisecting a nice (thick) polynomial-time approximable set at a given direction can be characterized by the counting class $\#P$; (2) the complexity of bisecting simultaneously two linearly separable nice (thick) polynomial-time approximable sets can be characterized by the counting class $\#P$; and (3) for either of these two problems, without the thickness condition and the linear separability condition (for the two-set case), it is arbitrarily hard to compute the bisector (even if it is unique).

Keywords: Computational complexity, the pancake problem, $\#P$, complexity theory of real functions.

1 Introduction

Given two Lebesgue measurable sets of arbitrary shapes in the two-dimensional plane \mathbb{R}^2 , there exists a line that simultaneously bisects them (i.e., the line cuts each set into two parts of equal area). This is the famous *Pancake Theorem*, or, the two-dimensional version of a more general *Ham Sandwich Theorem*. These theorems are related to two fundamental theorems, the *Brouwer Fixed Point Theorem* and the *Borsuk-Ulam Theorem*, in

¹ This material is based upon work supported by National Science Foundation under grant No. 0430124. The author is grateful to his advisor, Professor Ker-I Ko, for valuable discussions and his support.

² Email: fuxiang@cs.sunysb.edu

topology (see, e.g., Fulton [5]).

The Pancake Theorem leads to a computational problem, called the *pancake problem*, of finding the line, called the *bisector* (or *pancake/ham sandwich cut* in literature), that simultaneously bisects two given regions. In this paper, we study the computational complexity of the pancake problem. It has been well studied for the cases where the sets in question are polygons (see, e.g., [12,1]), or are composed of a finite number of points (in this case, the measure is the number of points instead of the area; see, e.g., [8,9,4]). Here, we consider the general cases where the sets may be subsets of the plane \mathbb{R}^2 of more complicated shapes.

In order to study the pancake problem for arbitrary subsets S_1 and S_2 of \mathbb{R}^2 , we need to fix a specific model for representing such sets and measure the computational complexity accordingly. In this paper, we apply the model of Turing-machine based complexity theory of real functions to this problem, and focus on the case where the sets S_1 and S_2 have polynomial-time representations, as introduced by Ko [7] and Chou and Ko [2]. That is, we assume that sets S_1 and S_2 are polynomial-time approximable, polynomial-time recognizable, or are bounded Jordan domains with polynomial-time computable boundaries, and investigate whether the corresponding bisectors are polynomial-time computable.

We also consider two related problems: (1) for a fixed angle $\alpha \in [0, \pi)$ (e.g., $\alpha = \pi/2$), computing the bisector L of a set S such that the angle from the positive x -axis to L is α ; and (2) computing two lines that are perpendicular to each other and divide a set S into four parts of equal area. Our main results use discrete complexity class $\#P$ and other related classes to characterize the complexity of the bisectors. They can be summarized as follows:

(1) These problems are all solvable if the sets in question are polynomial-time approximable and there exists a unique bisector (or a unique pair of lines for the problem of dividing a set into four parts). However, there are no fixed complexity bounds even if the sets have sufficient polynomial-time representations. For example, it can be arbitrarily hard to compute two lines that are perpendicular to each other and divide a polynomial-time approximable set S into four parts of equal area, even if S is also polynomial-time recognizable and convex, and has a polynomial-time computable Jordan curve as the boundary.

(2) The complexity of finding the vertical bisector L of a polynomial-time approximable set S has a lower bound of $P^{\#P_1[1]}$ and an upper bound of $P^{\#P}$, provided that S is *sufficiently thick* around L . Here $P^{\#P}$ is the class of languages A such that A is polynomial-time decidable with a $\#P$ function as an oracle, $\#P_1$ is the unary version of $\#P$, and the notation $[1]$ in $P^{\#P_1[1]}$

means that the oracle is only queried once.

(3) The complexity of finding the bisector L that bisects two P -approximable sets S_1 and S_2 has a lower bound of $P^{\#P_1[1]}$ and an upper bound of $P^{\#P}$, provided that S_1 and S_2 are linearly separable (i.e., S_1 and S_2 are on different sides of some line) and both S_1 and S_2 are *sufficiently thick* around their own bisectors at all angles.

We give, in Section 2, a brief summary of the basic computational model of the Turing machine-based complexity theory of real functions, as well as the exact definition of the above discrete complexity classes. For more careful discussion of the theory, see Ko [7] and Chou and Ko [2]; for the discrete complexity classes related to $\#P$, see, for example, Du and Ko [3] and Green et al. [6].

2 Preliminary

2.1 Basic notation

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 $\ell(w)$. We write $\langle w_1, w_2 \rangle$ to denote the pairing function on w_1 and w_2 . We write $||S||$ to denote the number of elements in a (finite) set S .

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 $\mathbb{D}_n = \{m/2^n : m \in \mathbb{Z}\}$. Each dyadic rational d has infinitely many binary representations with arbitrarily many trailing zeros. For each such representation s , we write $\ell(s)$ to denote its length. If the specific representation of a dyadic rational d is understood (often the shortest binary representation), then we write $\ell(d)$ to denote the length of this representation.

We use \mathbb{R} to denote the real line (and the class of real numbers) and \mathbb{R}^2 the 2-dimensional plane. We often use letters in math bold font, such as \mathbf{z} , or two letters in brackets, such as $\langle x, y \rangle$, to represent a point in \mathbb{R}^2 . For any point $\mathbf{z} \in \mathbb{R}^2$ and any set $S \subseteq \mathbb{R}^2$, we let $\text{dist}(\mathbf{z}, S)$ be the distance between \mathbf{z} and S ; that is, $\text{dist}(\mathbf{z}, S) = \inf\{|\mathbf{z} - \mathbf{z}'| : \mathbf{z}' \in S\}$, where $|\cdot|$ denotes the absolute value. For two sets $S_1, S_2 \subseteq \mathbb{R}^2$, we let $\text{dist}(S_1, S_2) = \inf\{|\mathbf{z}_1 - \mathbf{z}_2| : \mathbf{z}_1 \in S_1, \mathbf{z}_2 \in S_2\}$ be the distance between S_1 and S_2 . The complement of a set S is written as S^c , the closure of a set S is written as \overline{S} , and the boundary of a set S is written as ∂S .

We say a function $\phi : \mathbb{N} \rightarrow \mathbb{D}$ *binary converges* to (or *represents*) a real number x , or ϕ is a *Cauchy function* of x , if (i) for all $n \geq 0$, $\phi(n) \in \mathbb{D}_n$, and (ii) for all $n \geq 0$, $|\phi(n) - x| \leq 2^{-n}$. For any $x \in \mathbb{R}$, let CF_x denote the

set of functions that binary converge to x . Among functions CF_x there is a unique function $b_x : \mathbb{N} \rightarrow \mathbb{D}$ that satisfies the condition $x - 2^{-n} < b_x(n) \leq x$ for all $n \geq 0$. We call this function b_x the *standard Cauchy function* for x . We say two functions $\phi_x, \phi_y : \mathbb{N} \rightarrow \mathbb{D}$ binary converge to (or represent) a point $\mathbf{z} := \langle x, y \rangle \in \mathbb{R}^2$ if ϕ_x and ϕ_y binary converge to two real numbers x and y , respectively.

2.2 Complexity classes

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 counting class $\#P$, i.e., the class of functions that count the number of accepting paths of nondeterministic polynomial-time machines.

A set $T \subseteq \{0, 1\}^*$ is called a *tally set* if $T \subseteq \{0\}^*$. For a complexity class \mathcal{C} of sets of strings in $\{0, 1\}^*$, we let \mathcal{C}_1 denote the complexity class $\{A \in \mathcal{C} : A \subseteq \{0\}^*\}$. Similarly, for a complexity class \mathcal{FC} of functions from $\{0, 1\}^*$ to $\{0, 1\}^*$, we let \mathcal{FC}_1 denote the complexity class $\{F \in \mathcal{FC} : \text{the domain of } F \text{ is } \{0\}^*\}$. In this paper, we will use complexity classes FP_1 , P_1 and $\#P_1$.

The following properties of the complexity class $\#P$ are well known (see Du and Ko [3] and Green et al. [6]) and are useful in our constructions in Section 3.

Proposition 2.1 *Let $f : \{0, 1\}^* \rightarrow \mathbb{N}$ be a function. The following are equivalent:*

(a) $f \in \#P$.

(b) *There exist a polynomial-time computable predicate R and a polynomial p that for all inputs of length n , $f(x)$ is equal to the number of strings y of length $p(n)$ such that $R(x, y)$, i.e., $f(x) = ||\{y : \ell(y) = p(\ell(x)), R(x, y)\}||$.*

2.3 Computational model for continuous computation

To compute a real-valued function $f : \mathbb{R} \rightarrow \mathbb{R}$, 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 and two extra states: the query state and the answer state. The machine M makes a query to an oracle function $\phi : \mathbb{N} \rightarrow \mathbb{D}$ as follows: First, it writes an integer k on the query tape, then enters the query state and waits for the answer $\phi(k)$. The oracle ϕ then reads the input k , replaces the integer k on the query tape with $\phi(k)$, then places the machine M on the answer state. After the machine M enters the answer state, it continues like ordinary machines; in particular, it can read

$\phi(k)$ off the query tape. The action from the query state to the answer state 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 2.2 (a) A function $f : [0, 1] \rightarrow \mathbb{R}$ is said to be *computable* if there is an oracle Turing machine M that, on an oracle function $\phi : \mathbb{N} \rightarrow \mathbb{D}$ 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 $|d - f(x)| \leq 2^{-n}$.

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

When an oracle ϕ used by M is the standard Cauchy function b_d for a dyadic rational d , we can simulate the computation of M using d as an input instead of an oracle. To emphasize this fact, we write M^d instead of M^{b_d} to denote that the oracle is the standard Cauchy function of d .

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

Proposition 2.3 A function $f : [0, 1] \rightarrow \mathbb{R}$ is polynomial-time computable if and only if

(i) f has a polynomial modulus, and

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

The notions of computable and polynomial-time computable real functions can be extended naturally to functions $f : \mathbb{R} \rightarrow \mathbb{R}^2$ and functions $f : \mathbb{R}^2 \rightarrow \mathbb{R}^2$. For instance, the machine computing a function from \mathbb{R}^2 to \mathbb{R}^2 will use two oracles ϕ and ψ , representing two real numbers x and y , and will output two dyadic rationals e_1 and e_2 , such that $|\langle e_1, e_2 \rangle - f(\langle x, y \rangle)| \leq 2^{-n}$.

A Jordan curve (simple, closed curve) Γ in \mathbb{R}^2 is *polynomial-time computable* if there exists a polynomial-time computable function $f : [0, 1] \rightarrow \mathbb{R}^2$ such that the range of f is Γ , f is one-to-one on $[0, 1)$ and $f(0) = f(1)$. It is well known that the interior S of a Jordan curve Γ is a simply connected domain, which is called a Jordan domain. If Γ is polynomial-time computable, we say S is a polynomial-time Jordan domain.

Besides polynomial-time Jordan domains, we also consider the following two other kinds of subsets in \mathbb{R}^2 , which were introduced in Chou and Ko [2].

Given an oracle Turing machine M and a set $S \subseteq \mathbb{R}^2$, for $n \in \mathbb{N}$, we define an *error set* $E_n(M)$ as the set of all $\mathbf{z} \in \mathbb{R}^2$ having a Cauchy function representation $\langle \phi, \psi \rangle$ such that $M^{\phi, \psi}(n) \neq \chi_S(\mathbf{z})$, where $\chi_S(\mathbf{z})$ is the characteristic function defined on \mathbb{R}^2 such that $\chi_S(\mathbf{z}) = 1$ iff $\mathbf{z} \in S$, and μ^* is the outer Lebesgue measure.

Definition 2.4 (a) We say a set $S \subseteq \mathbb{R}^2$ is *polynomial-time approximable* (or simply *P-approximable*) if there exists a polynomial-time oracle Turing machine M such that for any input n , the error set $E_n(M)$ has size $\mu^*(E_n(M)) \leq 2^{-n}$.

(a) We say a set $S \subseteq \mathbb{R}^2$ is *polynomial-time recognizable* (or simply *P-recognizable*) if there exists a polynomial-time oracle Turing machine M such that $M^{\phi, \psi}(n) = \chi_S(\mathbf{z})$ whenever $\langle \phi, \psi \rangle$ represents a point \mathbf{z} whose distance to Γ_S is $> 2^{-n}$ (where Γ_S is the boundary of S); i.e., $E_n(M) \subseteq \{\mathbf{z} : \text{dist}(\mathbf{z}, \Gamma_S) \leq 2^{-n}\}$.

The above two concepts are not equivalent by Chou and Ko [2]: there exists a Jordan domain S that is *P-recognizable* but not *P-approximable*; the polynomial-time probabilistic class *BPP* collapses to *P* if all *P-approximable* sets are *P-recognizable*. On the other hand, while there exists a *P-recognizable* set S whose measure is not recursive, the measure of a *P-approximable* set is polynomial-time computable relative to an oracle in $\#P$. When we present negative results in this paper, we often construct sets that are both *P-approximable* and *P-recognizable* (*P-approximable/recognizable*, for short), and show the complexity of bisecting these sets is arbitrarily high.

Because the Lebesgue measure is consistent with *area* of the traditional sense (e.g., if S is a rectangle, $\mu(S)$ is the area of S), hereafter we use *area* and the Lebesgue measure interchangeably. Note that there exists a set $S \subseteq \mathbb{R}^2$ that is not measurable. However, in this paper we only consider measurable sets S , especially *P-approximable/recognizable* sets and polynomial-time Jordan domains.

3 Bisecting one set

From now on, unless specified otherwise, by “bisecting” a set we mean “dividing” a subset of \mathbb{R}^2 into two parts of equal area using a line, and by “bisector” of a set we mean a line that bisects the set. The Pancake Theorem states that two bounded sets have a common bisector. We first study the complexity of bisecting one set.

Let S be a bounded set in \mathbb{R}^2 . Fix an angle $\alpha \in [0, \pi)$. If the angle from the positive x -axis to a line L is α , we say that L is at angle α . It is easy

to see that there exists a bisector L of S at each angle α . It is possible that there is another bisector L' of S at the same angle α . In this case, the area of the part of S between L and L' is zero, and any line between L and L' that is parallel to L is a bisector of S at angle α . Then, the complexity of some bisector can be very high. We assume that this does not happen and assume that there exists a unique bisector of S at any fixed angle α .

Definition 3.1 Let S be a bounded set in \mathbb{R}^2 . A line L defined by $x = a$ divides S into two parts, with the left part denoted $S_{x \leq a}$ and the right part $S_{x \geq a}$. We define the *thickness* of S at L , denoted $thk_S(L)$, as the greatest lower limit of the area change of $S_{x \leq a}$ when moving L parallelly, that is,

$$thk_S(L) = \liminf_{\delta \rightarrow 0} \frac{area(S_{x \leq a+\delta}) - area(S_{x \leq a})}{\delta}.$$

For a line L at an angle $\alpha \in [0, \pi)$, we can also define $thk_S(L)$ in a similar way. More precisely, we rotate S and L about the origin by an angle $\pi/2 - \alpha$ to obtain S' and L' , respectively, and define $thk_S(L) = thk_{S'}(L')$.

It is obvious that for any line L at any angle $\alpha \in [0, \pi)$, $thk_S(L)$ is non-negative. If S is a convex Jordan domain, then $thk_S(L)$ is the length of the chord $L \cap S$. If L is a bisector of S at angle α such that $thk_S(L) > 0$, then L is the unique bisector of S at angle α . Note that it is possible that $thk_S(L) > 0$, but for any $\epsilon > 0$, there exists a line L' such that L' is parallel to L and $\text{dist}(L, L') < \epsilon$, but $thk_S(L') = 0$. We say S has a positive thickness around a line L if there exist two positive real numbers ϵ and δ , such that if L' is a line parallel to L with $\text{dist}(L, L') < \epsilon$, then $thk_S(L') > \delta$.

If for a set S and an angle α , there are two bisectors L and L' , then $thk_S(L) = thk_S(L') = 0$. However, it is possible that there exists a unique bisector L of S at angle α even if $thk_S(L) = 0$. Below we show that the complexity of finding such a bisector can be arbitrarily hard, while a positive thickness around a bisector reduces the complexity (see Theorem 3.3).

Theorem 3.2 Let $b \in (0, 1)$ be a computable number. There exists a P -approximable/recognizable set $S \subseteq [0, 1]^2$ such that the line L defined by $x = b$ is the unique bisector of S at angle $\pi/2$; furthermore, $thk_S(L) = 0$ and for any line L' defined by $x = b'$, where $b' \in (0, 1) - \{b\}$, $thk_S(L') > 0$.

Proof. We construct a P -approximable/recognizable set S such that (1) the boundary of S is the union of line segment $\{(x, 0) : x \in [0, 1]\}$ and the image of a polynomial-time computable function f , and is of a finite length; and (2) the set S is “symmetric” with respect to the bisector.

We assume that b is not a dyadic since, if b is a dyadic, it is very easy

to construct a polygon whose vertical bisector is defined by $x = b$. Let ϕ be a computable Cauchy function in CF_b and let $t : \mathbb{N} \rightarrow \mathbb{N}$ be a computable function which bounds the runtime of the function ϕ . We assume that $t(k) \geq 2k$ for all $k \in \mathbb{N}$. We inductively define two sequences $\{\underline{d}_k\}$ and $\{\bar{d}_k\}$ of dyadic numbers in $[0, 1]$:

$$\underline{d}_1 = 0; \bar{d}_1 = 1;$$

$$\underline{d}_k = \max\{\underline{d}_{k-1}, \phi(k) - 2^{-k}\},$$

$$\bar{d}_k = \min\{\bar{d}_{k-1}, \phi(k) + 2^{-k}\}, \quad k \geq 2.$$

We observe that for any $k \geq 2$,

$$0 = \underline{d}_1 \leq \underline{d}_2 \leq \underline{d}_k \leq \underline{d}_{k+1} < b < \bar{d}_{k+1} \leq \bar{d}_k \leq \bar{d}_2 \leq \bar{d}_1 = 1,$$

and that \underline{d}_k and \bar{d}_k are computable in time $O(s(k))$, where $s(k) = \sum_{i=1}^k t(i) \geq k^2 + k$.

Let $I_1 \subseteq \mathbb{N}$ be a set of indices such that $i \in I_1$ if and only if $\underline{d}_{i+1} > \underline{d}_i$; in other words, $\{\underline{d}_i\}_{i \in I_1}$ is the largest strictly increasing subsequence of the increasing subsequence $\{\underline{d}_j\}_{j \in \mathbb{N}}$. Similarly, let $I_2 \subseteq \mathbb{N}$ be a set of indices such that $i \in I_2$ if and only if $\bar{d}_{i+1} < \bar{d}_i$. Note that since b is not a dyadic, both I_1 and I_2 contain infinitely many elements. Let $a_{m,n}$ be the n -th smallest number in I_m , where $m \in \{1, 2\}$ and $n \in \mathbb{N}$. For any $n \in \mathbb{N}$, let i denote $a_{1,n}$ and j denote $a_{2,n+1}$, $h_1 = \max(a_{1,n+1}, a_{2,n+1})$ and $h_2 = \max(a_{1,n+2}, a_{2,n+2})$. We note that $\underline{d}_j - \underline{d}_i \geq 2^{-j}$ since $\underline{d}_j - \underline{d}_i$ is a positive dyadic in \mathbb{D}_j . We call $[\underline{d}_i, \underline{d}_j]$ the n -th interval of the first kind. The part of set S between two lines $x = \underline{d}_i$ and $x = \underline{d}_j$ is a pentagon of area $2^{-(2s(h_1+1)+1)}$; more precisely, the five vertices of the pentagon are $\langle \underline{d}_i, 0 \rangle$, $\langle \underline{d}_i, 2^{-2s(h_1+1)} \rangle$, $\langle (\underline{d}_i + \underline{d}_j)/2, 2^{-2s(h_1+1)}/(\underline{d}_j - \underline{d}_i) \rangle$, $\langle (\underline{d}_j, 2^{-2s(h_2+1)})/2 \rangle$, and $\langle \underline{d}_j, 0 \rangle$; we denote this pentagon P_{i,j,h_1,h_2} . Similarly we define the other parts of S according to I_2 (e.g., we will have the n -th interval of the second kind).

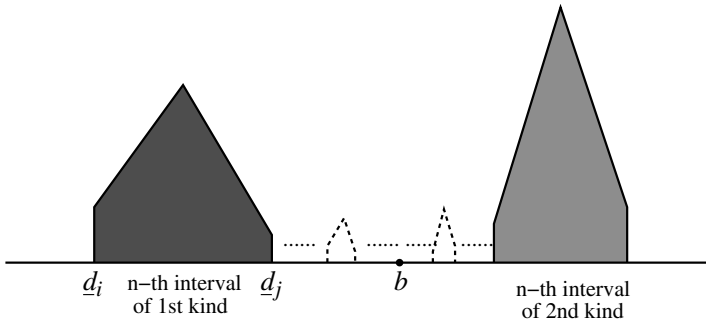


Fig. 1. The set S consisting of pentagons.

We can see that the line $x = b$ is the unique bisector of S at angle $\pi/2$,

since from the construction of S , for any $n \in \mathbb{N}$, the pentagon associated with the n -th interval of the first kind and that associated with the n -th interval of the second kind are of the same area. Note that the boundary of S is the union of the unit interval $[0, 1]$ on the x -axis and the set $\{\langle x, f(x) \rangle : x \in [0, 1]\}$, where the function f is easy to obtain from the construction of S above. We first show that f is polynomial-time computable by proving that f satisfies both conditions in Proposition 2.3. Indeed, it is easy to see that f has a linear modulus of continuity since, for each $n \in \mathbb{N}$, let $i = a_{1,n}$, $j = a_{1,n+1}$, $h_1 = \max(a_{1,n+1}, a_{2,n+1})$ and $h_2 = \max(a_{1,n+2}, a_{2,n+2})$, then we have $s(h_1+1) \geq h_1^2 + 1 \geq j$, $|\underline{d}_j - \underline{d}_i| \geq 2^{-j} \geq 2^{-s(h_1+1)}$ and $\frac{2^{-2s(h_1+1)}/(\underline{d}_j - \underline{d}_i)}{(\underline{d}_j - \underline{d}_i)/2} < 2$, which implies that the derivative (which exists almost everywhere) of f is between -2 and 2 ; furthermore, we can see that the boundary of S is of a finite length. Next we show that for any two integers $m, n \in \mathbb{N}$ and any dyadic $d \in \mathbb{D}_m \cap [0, 1]$, a dyadic e can be computed in time $O(m+n)$ such that $|e - f(d)| < 2^{-n}$ by the following algorithm:

- (1) Compute in time $O(n)$ an integer k such that $k = \max\{i : s(i) \leq n\}$ by simulating a machine M with time bound t that computes ϕ as follows: Let M compute one by one $\phi(1), \phi(2), \dots$, and halt after n moves, now k is the maximum number such that $\phi(k)$ is computed in this process. (Note that the simulation may have started to compute $\phi(k+1)$ but it is not finished since it is terminated after n moves. In this case $s(k) \leq n < s(k+1)$.)
- (2) Compute in time $O(n)$ the (multi)set $\{\underline{d}_i, \bar{d}_i : i \leq k\}$, $I'_1 := I_1 \cap \{1, 2, \dots, k-1\}$, and $I'_2 := I_2 \cap \{1, 2, \dots, k-1\}$. (Note that it may take more than n moves to decide whether $k \in I_1$ (or $k \in I_2$), but we do not need this result. This step can be combined with step (1).)
- (3) Compute in time $O(k^2)$ three integers $q = \min(|I'_1|, |I'_2|)$, $i_0 = a_{1,q}$ and $j_0 = a_{2,q}$. Check in time $O(m+k)$ whether $d \in [\underline{d}_{i_0}, \bar{d}_{j_0}]$, $d < \underline{d}_{i_0}$, or $d > \bar{d}_{j_0}$: if $d \in [\underline{d}_{i_0}, \bar{d}_{j_0}]$, let $e = 0$ and halt; if $d < \underline{d}_{i_0}$, go to step (4); if $d > \bar{d}_{j_0}$, go to step (5).
- (4) Compute in time $O(k^2 + m) = O(n + m)$ three integers $\ell < q$, $i = a_{1,\ell}$ and $j = a_{1,\ell+1}$ such that $d \in [\underline{d}_i, \underline{d}_j]$. There are two sub cases:
 - (4.1) $\ell = q-1$. Compute in time $O(k^2)$ an integer $h_1 = \max(a_{1,q}, a_{2,q})$. We have $h_2 := \max(a_{1,q+1}, a_{2,q+1}) > k$ (we do not compute h_2 now) and $s(h_2) > n$. The quadrangle P'_{i,j,h_1} with vertices $\langle \underline{d}_i, 0 \rangle$, $\langle \underline{d}_i, 2^{-2s(h_1+1)} \rangle$, $\langle (\underline{d}_i + \underline{d}_j)/2, 2^{-2s(h_1+1)}/(\underline{d}_j - \underline{d}_i) - 2^{-2s(h_1+1)}/2 \rangle$ and $\langle \underline{d}_j, 0 \rangle$ is an approximation to the pentagon P_{i,j,h_1,h_2} with error $\leq 2^{-2s(h_2+1)} < 2^{-2n}$. Do an interpolation in time $O(m+n+j+s(h_1+1)) = O(m+n)$ on

P'_{i,j,h_1} to obtain an approximation e to $f(d)$ such that $|e - f(d)| < 2^{-n}$ and halt.

(4.2) $\ell < q - 1$. Compute in time $O(k^2)$ two integers $h_1 = \max(a_{1,\ell+1}, a_{2,\ell+1})$ and $h_2 := \max(a_{1,\ell+2}, a_{2,\ell+2})$. (Now the pentagon P_{i,j,h_1,h_2} is exactly the part of S between lines $x = \underline{d}_i$ and $x = \underline{d}_j$.) Do an interpolation in time $O(m + n + j + s(h_1 + 1) + s(h_2 + 1)) = O(m + n)$ on P_{i,j,h_1,h_2} to obtain an approximation e to $f(d)$ such that $|e - f(d)| < 2^{-n}$ and halt.

(5) (It is similar to step (4) and is omitted.)

We can see that the above algorithm takes time $O(m + n)$ since each step takes time $O(m + n)$. Also from the algorithm, for the cases of $d < \underline{d}_{i_0}$ and $d > \bar{d}_{j_0}$, an approximation e to $f(d)$ with an error $\leq 2^{-n}$ is obtained. We need to show that if $d \in [\underline{d}_{i_0}, \bar{d}_{j_0}]$, $e = 0$ is an approximation to $f(d)$ with an error $< 2^{-n}$, that is, $|f(d)| < 2^{-n}$. Without loss of generality, assume that $d \in [\underline{d}_{i_0}, b)$, then there exists an integer $\ell \geq q$ such that $d \in [\underline{d}_i, \underline{d}_j]$, where $i = a_{1,\ell}$, $j = a_{1,\ell+1}$; that is, d is in the ℓ -th interval of the first kind. Let $h_1 = \max(a_{1,\ell+1}, a_{2,\ell+1})$ and $h_2 = \max(a_{1,\ell+2}, a_{2,\ell+2})$. From the definition of q , $h_2 > h_1 \geq k$. Then $f(d)$ is decided by the pentagon P_{i,j,h_1,h_2} . Since $h_1 \geq k$, $s(h_2 + 1) > s(h_1 + 1) \geq s(k + 1) > n$, and $|f((\underline{d}_i, \underline{d}_j)/2)| = 2^{-2s(h_1+1)}/(\underline{d}_j - \underline{d}_i) - (2^{-2s(h_1+1)} + 2^{-2s(h_2+1)})/2 < 2^{-n}$, we have $|f(d)| < 2^{-n}$, which completes the proof that f is polynomial-time computable.

In order to check whether a point $\langle x, y \rangle$ is in S , we check whether $0 < y < f(x)$. As f is polynomial-time computable, S is P -recognizable. Note that $f(b) = 0$, and the boundary of S is the union of two Jordan curves and of a finite length, it follows that S is also P -approximable (Chou and Ko [2]). It is also easy to check that $\text{thk}_S(L) = 0$ and for any line L' defined by $x = b'$, where $b' \in (0, 1) - \{b\}$, $\text{thk}_S(L') > 0$: f is continuous, thus for any line L_r defined by $x = r$, where $r \in (0, 1)$, $\text{thk}_S(L_r) = f(r)$; furthermore, $f(r) > 0$ iff $r \neq b$. \square

Theorem 3.2 is a negative result for the case where the thickness of the set S at a bisector is zero. Next we prove a positive result for the case where S has a positive thickness around a bisector.

Let S be a P -approximable set with a positive thickness around a bisector L_α at an angle α . Without loss of generality, we consider the case $\alpha = \pi/2$.

Theorem 3.3 *In the following, (a) \Rightarrow (b) \Rightarrow (c):*

(a) $FP = \#P$.

(b) *For any P -approximable set $S \subseteq [0, 1]^2$ that has a positive thickness $W > 0$ around the unique vertical bisector L defined by $x = b$, the real number b is polynomial-time computable.*

(c) $FP_1 = \#P_1$.

Proof. (a) \Rightarrow (b). We define a function $g : [0, 1] \rightarrow \mathbb{R}^2$ such that $g(t) = \text{area}(S_{x \leq t}) - \text{area}(S_{x > t})$, where $S_{x \leq t} = \{\langle x, y \rangle \in S : x \leq t\}$ and $S_{x > t} = S - S_{x \leq t}$. Then b is the root of $g(x)$. We will show that the function g is polynomial-time computable under the condition $FP = \#P$. The function g is increasing and furthermore, since S has a positive thickness around the vertical bisector L , we have $|\text{area}(S_{x \leq b+\delta}) - \text{area}(S_{x \leq b})| \geq |\delta| \cdot \text{thk}_S(L)/2$ when δ is sufficiently small, which means that g has a polynomial inverse modulus near b . According to Corollary 4.7 of Ko [7], the root b of $g(x)$ can be computed in polynomial time by binary search.

Now we show how to compute g in polynomial time. First, g has a linear modulus of continuity, because $S \subseteq [0, 1]^2$ and by the definition of g , $|g(t_1) - g(t_2)| \leq |t_1 - t_2|$ for $t_1, t_2 \in [0, 1]$. Then by Proposition 2.3, we only need to show that the values of g at dyadic points can be approximated in polynomial time. Since S is P -approximable, there exists a polynomial-time oracle Turing machine M such that for any input n , the error set $E_n(M)$ has size $\mu^*(E_n(M)) \leq 2^{-n}$. Let p be a polynomial function that bounds M . Without loss of generality, assume that $p(n) > n$.

For $n \in \mathbb{N}$ and $t \in \mathbb{D}_n \cap [0, 1]$, consider the following two sets:

$$\begin{aligned} A(n, t) &= \{\langle d_1, d_2 \rangle \in \mathbb{D}_{p(n)}^2 : d_1 < t, M^{d_1, d_2}(n) = 1\}; \\ B(n, t) &= (\mathbb{D}_{p(n)} \cap [0, t)) \times (D_{p(n)} \cap [0, 1]) - A(n, t). \end{aligned}$$

For $x, y, d \in \mathbb{R}$ with $d > 0$, let $N(\langle x, y \rangle; d)$ denote the set $\{\langle x', y' \rangle \in \mathbb{R}^2 : |x' - x| < d, |y' - y| < d\}$. Suppose that a dyadic point $\mathbf{d} = \langle d_1, d_2 \rangle$ is in $A(n, t)$, then for any $\mathbf{z} \in N(\mathbf{d}; 2^{-(p(n)+1)})$, since \mathbf{z} has an oracle representation $\langle \phi, \psi \rangle$ such that $\phi(i) = b_{d_1}(i)$ and $\psi(i) = b_{d_2}(i)$ for all $i \leq p(n)$, the computation $M^{\phi, \psi}(n)$ works exactly the same as that of $M^{d_1, d_2}(n)$, and hence it outputs 1. That means that either $\mathbf{z} \in S_{x \leq t}$ or $\mathbf{z} \in E_n(M)$; in other words, $N(\mathbf{d}; 2^{-(p(n)+1)}) \subseteq S_{x \leq t} \cup E_n(M)$. Similarly, if $\mathbf{d} \in B(n, t)$, then $N(\mathbf{d}; 2^{-(p(n)+1)}) \subseteq S_{x \geq t} \cup E_n(M)$. Note that the small squares $N(\mathbf{d}; 2^{-(p(n)+1)})$, where $\mathbf{d} \in (\mathbb{D}_{p(n)} \cap [0, t)) \times (D_{p(n)} \cap [0, 1])$, do not overlap each other and the union of them cover the rectangle $[-2^{-(p(n)+1)}, t - 2^{-(p(n)+1)}] \times [-2^{-(p(n)+1)}, 1 + 2^{-(p(n)+1)}]$ (except a finite number of line segments whose area is zero). Now it is easy to see that $2^{-2p(n)} \cdot ||A(n, t)||$ is close to the area of $S_{x \leq t}$, with an error bounded by $\mu^*(E_n(M)) + 4 \cdot 2^{-(p(n)+1)} < 2^{-(n-1)}$. We define a function $G : \mathbb{N} \times (\mathbb{D} \cap [0, 1]) \rightarrow \mathbb{N}$ such that $G(n, t) = ||A(n, t')||$ for all $(n, t) \in \mathbb{N} \times (\mathbb{D} \cap [0, 1])$, where $t' = \max\{x \in \mathbb{D}_n : x \leq t\}$. Note that $A(n, t')$ is the number of elements $\langle d_1, d_2 \rangle \in \mathbb{D}_{p(n)}^2$ that satisfy the polynomial-time

computable predicate $d_1 < t' \wedge M^{d_1, d_2}(n) = 1$, thus according to Proposition 2.1, G is in $\#P$. Therefore, if $\#P = FP$, then the area of $S_{x \leq t}$ can be computed in polynomial time. The same conclusion holds for $S_{x > t}$. Therefore, g is polynomial time computable if $\#P = FP$.

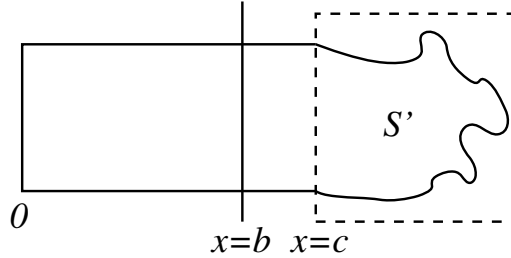


Fig. 2. The domain S for (b) \Rightarrow (c) of Theorem 3.3.

(b) \Rightarrow (c). Chou and Ko [2] constructed a P -approximable set $S' \subseteq [0, 1]^2$ such that the area of S' is polynomial time computable if and only if $\#P_1 = FP_1$. We construct a domain S such that the left part of S is a large rectangle $[0, c] \times [0, 1]$, and the right part of S is a copy of S' (see Figure 2). Then, because the problem of computing b is equivalent to decide the area of S' (i.e., $\text{area}(S') = 2b - c$), the theorem is proved. \square

In other words, Theorem 3.3 shows that the complexity of computing the bisector at a given angle of a P -approximable bounded set with a positive thickness around the bisector is between $P^{\#P}$ and $P^{\#P_1}$ (more precisely, $P^{\#P_1[1]}$, since to compute an approximation to the area of S' , the $\#P_1$ oracle will only be queried once; for details, see Chou and Ko [2]). Whether $P^{\#P} = P^{\#P_1}$ is still an open question in discrete complexity theory (see, e.g., Ogihara et al. [10]).

We present a pure mathematical result below that any Jordan domain has a positive thickness around any bisector; furthermore, the thickness is always greater than a positive constant (i.e., bounded below).

Lemma 3.4 *Let S be a Jordan domain and Γ the boundary of S . Then there exists a real number $\Delta > 0$, such that for any line L that divides S into two parts S_1 and S_2 with $|\text{area}(S_1) - \text{area}(S_2)| < \text{area}(S)/3$, there is a point Q in the intersection $L \cap S$ of L and S such that $\text{dist}(Q, \Gamma) > \Delta$. It follows that S has a positive thickness around all bisectors.*

Proof. Let $f : [0, 1] \rightarrow \mathbb{R}^2$ be a continuous function such that f is 1-1 on $[0, 1)$, $f(0) = f(1)$ and the image of f is Γ . Let $\{t\}$ denote the distance of real number t and integer points (e.g., $\{0.6\} = 0.4$, $\{1\} = 0$). Let $K > 0$ be a real number such that $\pi K^2 < \text{area}(S)/3$ and $K < \text{area}(S)/(6R)$, where R is the

radius of a disk that covers S . Then there exists a real number $H > 0$ such that for any $t_1, t_2 \in [0, 1]$, $\{t_1 - t_2\} < H \Rightarrow |f(t_1) - f(t_2)| < K$.

Let $A = \{(t_1, t_2) \in [0, 1]^2 : t_1 < t_2, |f(t_1) - f(t_2)| \geq 3K\}$. It is clear that A is closed. From the assumption of $K < \text{area}(S)/(6R)$, A is nonempty, for otherwise, all Γ is contained in a rectangle of dimensions $3K \times 2R < \text{area}(S)$, which is a contradiction. Now we define a function g on A as follows. Let $(t_1, t_2) \in A$. If γ is a path from $f(t_1)$ to $f(t_2)$ such that $\gamma \subseteq \overline{S}$, we let γ_K denote the portion of γ whose distance to $f(t_1)$ and $f(t_2)$ is no less than K , i.e., $\gamma_K = \{Q \in \gamma : \text{dist}(Q, f(t_i)) \geq K, i = 1, 2\}$. From the definition of A , γ_K is nonempty. $g(t_1, t_2)$ is the least upper bound of the distances $\text{dist}(\gamma_K, \Gamma)$ over all paths $\gamma \subseteq \overline{S}$ from $f(t_1)$ to $f(t_2)$. Roughly speaking, there is a tube that connects $f(t_1)$ and $f(t_2)$ such that the tube is of width $\geq 2g(t_1, t_2)$ in the middle. We have $g(t_1, t_2) > 0$ since there exists a path $\gamma \subseteq \overline{S}$ from $f(t_1)$ to $f(t_2)$ such that γ_K is strictly inside Γ , which implies that $g(t_1, t_2) \geq \text{dist}(\gamma_K, \Gamma) > 0$.

It is an interesting problem to prove that g is continuous. For any real number $\epsilon > 0$, there exists a real number $\delta > 0$ such that for any $t_1, t_2 \in [0, 1]$, $\{t_1 - t_2\} < \delta \Rightarrow |f(t_1) - f(t_2)| < \epsilon/2$. For any two pairs $(t_1, t_2), (t'_1, t'_2) \in A$ such that $|t_i - t'_i| < \delta$ for $i = 1, 2$, let γ be a path from $f(t_1)$ to $f(t_2)$ such that $\text{dist}(\gamma_K, \Gamma) \geq g(t_1, t_2) - \epsilon/2$. Assume that $t_1 < t'_1$ and $t_2 < t'_2$. Let γ' be the path consisting of γ , $f([t_1, t'_1])$ and $f([t_2, t'_2])$ (removing any repeated portion). Then $g(t'_1, t'_2) \geq \text{dist}(\gamma'_K, \Gamma) \geq \text{dist}(\gamma_K, \Gamma) - \max(|f(t_1) - f(t'_1)|, |f(t_2) - f(t'_2)|) > (g(t_1, t_2) - \epsilon/2) - \epsilon/2 = g(t_1, t_2) - \epsilon$. Symmetrically, $g(t_1, t_2) > g(t'_1, t'_2) - \epsilon$. Therefore, $|g(t_1, t_2) - g(t'_1, t'_2)| < \epsilon$. Thus, g is continuous.

Since g is continuous on a bounded closed set, g assumes its minimum at some points. That is, there exists a real number $\Delta > 0$ such that $g(t_1, t_2) \geq 2\Delta$ for any $(t_1, t_2) \in A$.

Now we prove the lemma. From the assumption of K , there exist $t_1, t_2 \in [0, 1]$ such that $f(t_1)$ and $f(t_2)$ are on different sides of L and are the furthest points away from L among all points of Γ on two sides of L , respectively. Then $\text{dist}(f(t_i), L) > K$ for $i = 1, 2$ and $|f(t_1) - f(t_2)| > 3K$. Assume $t_1 < t_2$, then $(t_1, t_2) \in A$ and $g(t_1, t_2) \geq 2\Delta$. Then there exists a path $\gamma \subseteq \overline{S}$ from $f(t_1)$ to $f(t_2)$ such that $\text{dist}(\gamma_K, \Gamma) \geq g(t_1, t_2) - \Delta \geq 2\Delta - \Delta = \Delta$. The curve γ_K must intersect L , since $f(t_1)$ and $f(t_2)$ are on different sides of L and $\text{dist}(f(t_i), L) > K$ for $i = 1, 2$. Pick any point Q in $\gamma_K \cap L$, we have $\text{dist}(Q, \Gamma) \geq \Delta$. \square

Remark: In the proof above, we can show further that $g(t_1, t_2) = \text{dist}(\gamma_K, \Gamma)$ for some path $\gamma \subseteq \overline{S}$ from $f(t_1)$ to $f(t_2)$, but this requires the *Arzela-Ascoli Theorem* in real analysis (see, e.g., Rudin [11]), which complicates the proof

unnecessarily.

Note that by Lemma 3.4, any Jordan domain has a thickness W around any bisector such that W is greater than a positive constant 2Δ . Then we have the following corollary from Theorem 3.3.

Corollary 3.5 *In the following, $(a) \Rightarrow (b) \Rightarrow (c)$:*

(a) $FP = \#P$.

(b) *For any P -approximable Jordan domain S , the bisector of S at any polynomial-time computable angle $\alpha \in [0, \pi)$ is polynomial-time computable.*³

(c) $FP_1 = \#P_1$.

There is also another version of bisecting a single set S : the bisector is through a fixed point Q , instead of forming a fixed angle with the positive x -axis. If the fixed point Q is away from the set S in the sense that Q and S are on different sides of a line (i.e., Q and S are *linearly separable*), and S has a positive thickness around a bisector L of S that goes through Q , then L is the unique such bisector. Note that here thickness around a line L is more naturally defined as the rate of area change over angle change. More precisely, suppose Q and S are separated by a horizontal line and Q is below S . For an angle $\alpha \in [0, \pi)$, let L_α be the half line through Q at angle α , and $S_{angle < \alpha}$ be the part of S on the right of L_α . Now the thickness of S at line L_α (or we may say *the thickness of S at angle α with respect to Q*) is defined as

$$\liminf_{\delta \rightarrow 0} \frac{\text{area}(S_{angle < \alpha + \delta}) - \text{area}(S_{angle < \alpha})}{\delta},$$

and similarly we can define positive thickness around a line (or angle). We remark that this definition is consistent with the previous one in the sense that if S has a positive thickness around a line L_α by the previous definition, Definition 3.1, then by the current definition S still has a positive thickness around angle α with respect to a point Q on L_α that is linearly separable from S . (However, the exact thickness values under these two definitions are not the same.)

Theorem 3.6 *In the following, $(a) \Rightarrow (b) \Rightarrow (c)$:*

(a) $FP = \#P$.

(b) *Let S be a P -approximable set such that the origin O and S are linearly separated by a horizontal line with O below S . If a line L_α ($\alpha \in [0, \pi)$) through O is a bisector of S and S has a positive thickness around angle α with*

³ Note that the bisector is defined by $y = \tan(\alpha)x + b$ if $\alpha \neq \pi/2$ and $x = b$ if $\alpha = \pi/2$, for some real number b , and the problem is to compute b in polynomial time.

respect to O , then L_α is the unique such bisector of S and α is polynomial-time computable.

(c) $FP_1 = \#P_1$.

Proof. The proof is similar to that of Theorem 3.3 and thus is omitted. Note that for the part (b) \Rightarrow (c), we still use the domain S shown in Figure 2, and we can make the bisector L_α only intersect the left part of S , i.e., the rectangle, then the problem of finding L_α is equivalent to decide the area of S . \square

On the other hand, if the fixed point Q can be inside S , there is no fixed complexity bound for finding the bisector of S through Q .

Theorem 3.7 *Suppose $\alpha \in [0, \pi)$ is a computable number. Then there exists a P -approximable/recognizable convex Jordan domain S , such that the unique bisector of S through the origin O is at angle α .*

Proof. The idea of the construction of S is similar to that in the proof of Theorem 3.2. The boundary of S is the image of a function $f : [0, 2\pi] \rightarrow \mathbb{R}^2$ such that for any $\beta \in [0, 2\pi]$, $f(\beta) = g(\beta)\langle \cos(\beta), \sin(\beta) \rangle$, where $g : [0, 2\pi] \rightarrow \mathbb{R}$ is a function whose values are always positive. It suffices to just describe the boundary of S in order to explain what f is.

Without loss of generality, we assume that $\alpha \in (1/4, 1/2)$. We follow the notations of the proof of Theorem 3.2, and define two sequences $\{\underline{d}_n\}$ and $\{\bar{d}_n\}$ that binary converge to α from below and upper, function $s(n)$, sets I_1 and I_2 , the intervals of the first and second kinds, and let $a_{m,n}$ be the n -th smallest number in I_m , where $m \in \{1, 2\}$.

For any $n \in \mathbb{N}$, let $i = a_{1,n+1}$ and $j = a_{2,n+1}$ and $h = \max(a_{1,n+1}, a_{2,n+1})$. We have $\underline{d}_j - \underline{d}_i \geq 2^{-s(h)}$. Then on the n -th interval $[\underline{d}_i, \underline{d}_j]$ of the first kind, the image of f is identical to the part of the unit circle at the sector between angles \underline{d}_i and \underline{d}_j , except that there is a bump on $[\underline{d}_i, \underline{d}_i + 2^{-s(h)}]$; more precisely, the image of f on $[\underline{d}_i, \underline{d}_i + 2^{-s(h)}]$ is the union of two tangents: that is, on $[\underline{d}_i, \underline{d}_i + 2^{-s(h)}]$, the image of f is a piecewise linear curve with breakpoints $\langle \cos(\underline{d}_i), \sin(\underline{d}_i) \rangle$, $\langle \cos(\underline{d}_i + 2^{-(s(h)+1)}), \sin(\underline{d}_i + 2^{-(s(h)+1)}) \rangle / \cos(2^{-(s(h)+1)})$ and $\langle \cos(\underline{d}_i + 2^{-s(h)}), \sin(\underline{d}_i + 2^{-s(h)}) \rangle$. Similarly, we define f on the intervals of the second kind. The image of f on $[0, 2\pi] - [1/4, 1/2]$ is identical to the unit circle on the same domain.

Figure 3 illustrates the domain S : the lines OA , OB and OC are at angles α , $1/2$, $1/4$ respectively; the bumps on different sides of OA can be paired up such that the two bumps in each pair are of the same area. We can check that S is a P -approximable/recognizable Jordan domain similar to the proof of Theorem 3.2. Also, S is convex, since its boundary ∂S consists of circular

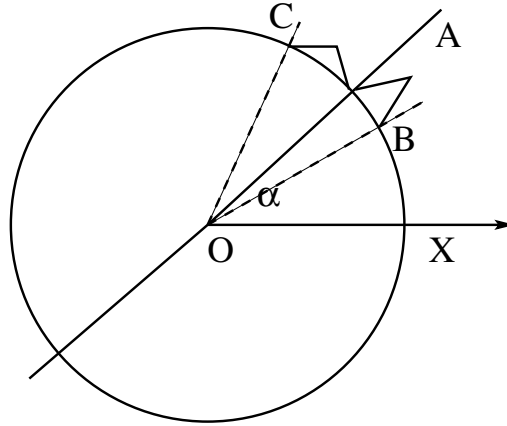


Fig. 3. The domain S for (b) Theorem 3.7.

arcs and tangents. (The domain S does not look convex in the figure because we want to emphasize the bumps in a small region.) As shown in the figure, the area of bumps in the sector AOB is the same as the area of bumps in the sector AOC , so OA is a bisector of S that goes through the origin O . There are no other such bisectors because for any line OD other than OA , on one side of OD , there are only a finite number of bumps, while on the other side, there are infinitely many bumps. \square

4 Bisecting two sets simultaneously

Mathematically, the Pancake Theorem follows the *intermediate value theorem* and the *Borsuk-Ulam Theorem* (see, e.g., Fulton [5]). For the sake of self-containness, we show how the intermediate value theorem implies the Borsuk-Ulam Theorem and how the Borsuk-Ulam Theorem implies the Pancake Theorem on the two-dimensional plane (see, e.g., Fulton [5]).

Let C be the unit circle with its center at the origin. For $\alpha \in [0, 2\pi]$, we use \mathbf{z}_α to denote a point $\langle \cos \alpha, \sin \alpha \rangle$. The Borsuk-Ulam Theorem on the two-dimensional plane states that any continuous function $f : C \rightarrow \mathbb{R}$ must map a pair of antipodal points \mathbf{z}_{α_0} and $\mathbf{z}_{\alpha_0+\pi}$ to the same value. To see this, let $g : C \rightarrow \mathbb{R}$ be the function defined by $g(\mathbf{z}_\alpha) = f(\mathbf{z}_\alpha) - f(\mathbf{z}_{\alpha+\pi})$. Then g is a continuous function satisfying $g(\mathbf{z}_0) = -g(\mathbf{z}_\pi)$. From the intermediate value theorem, there must exist a number $\alpha_0 \in [0, \pi]$ such that $g(\mathbf{z}_{\alpha_0}) = 0$, i.e., $f(\mathbf{z}_{\alpha_0}) = f(\mathbf{z}_{\alpha_0+\pi})$.

Now we show how the Borsuk-Ulam Theorem implies the Pancake Theorem. Let S and S' be two Lebesgue measurable sets. We can assume that S and S' are both inside the unit circle: if S and S' are unbounded, we can consider the problem on bounded approximations to S and S' and then take

a limit. Now we define a function $f : C \rightarrow \mathbb{R}$ as follows: for any $\mathbf{z}_\alpha \in C$, let $L_{\alpha+\pi/2}$ be the “middle” bisector of S at angle $\alpha + \pi/2$ (we make $L_{\alpha+\pi/2}$ in the middle among all possible bisectors at the same angle, so that for the antipodal point $\mathbf{z}_{\alpha+\pi}$ of \mathbf{z}_α , the same bisector $L_{\alpha+\pi/2}$ will be chosen), which divides S' into two parts S'_1 and S'_2 with S'_1 closer to \mathbf{z}_α , and $f(\mathbf{z}_\alpha)$ is the area of S'_1 . Assuming that f is continuous,⁴ then from the Borsuk-Ulam theorem, there exists a number $\alpha_0 \in [0, \pi]$ such that $f(\mathbf{z}_{\alpha_0}) = f(\mathbf{z}_{\alpha_0+\pi})$, which implies that $L_{\alpha_0+\pi/2}$ bisects S' too.

Next we present some results on the computability and complexity of the intermediate value theorem and the Borsuk-Ulam Theorem.

Theorem 4.1 (a) *Let $f : [0, 1] \rightarrow \mathbb{R}$ be a computable function such that $f(0)f(1) < 0$. Then there exists a computable number $r \in (0, 1)$ such that $f(r) = 0$.*

(b) (Ko [7, Theorem 4.4]) *For any computable number $r \in (0, 1)$, there exists a polynomial-time computable function $f : [0, 1] \rightarrow \mathbb{R}$ such that (1) f is strictly increasing; (2) $f(0)f(1) < 0$; and (3) $f(r) = 0$.*

Part (a) of Theorem 4.1 provides a positive result that every computable function that changes sign has a computable zero, while part (b) shows a negative result that it is arbitrarily hard to compute it even if there is only one zero and the function is polynomial-time computable. As pointed out in Ko [7], if the function f has a polynomial inverse modulus, the unique zero of f is polynomial-time computable.

In the following, we prove a negative result about the complexity of the Borsuk-Ulam Theorem.

Theorem 4.2 *Let $\alpha \in [0, \pi)$ be a computable number. Then there exists a polynomial-time computable function $f : C \rightarrow \mathbb{R}$ such that $(\mathbf{z}_\alpha, \mathbf{z}_{\alpha+\pi})$ is the only pair of antipodal points that satisfies $f(\mathbf{z}_\alpha) = f(\mathbf{z}_{\alpha+\pi})$.*

Proof (Ideas) Without loss of generality, suppose $\alpha \in (\pi/3, 2\pi/3)$. We can construct a function $g : [0, 2\pi] \rightarrow \mathbb{R}$ such that

(1) g is strictly increasing on $[0, \pi]$ with $g(\alpha) = 1$.

(2) g is piecewise-linear on $[\pi, 2\pi]$ with $g(4\pi/3) = 1$, $g(5\pi/3) = 1$ and $g(2\pi) = g(0)$.

The construction of (1) follows that of part (b) of Theorem 4.1. It is obvious that $g(\alpha) = g(\alpha + \pi) (= 1)$. To prove that $(\alpha, \alpha + \pi)$ is the only such pair, we note that for $\beta \in [0, \pi/3]$, $g(\beta) < 1$ and $g(\beta + \pi) \geq 1$; for

⁴ This is not obvious to be true, but we can assume that S and S' are of “good” shapes so that f is continuous, then prove the Pancake Theorem and take a limit for the general case.

$\beta \in [2\pi/3, \pi]$, $g(\beta) > 1$ and $g(\beta + \pi) \leq 1$; for $\beta \in [\pi/3, 2\pi/3]$, $g(\beta + \pi) = 1$ but $g(\beta) \neq 1$ if $\beta \neq \alpha$.

Let $f(\mathbf{z}_\beta) = g(\beta)$ for any $\beta \in [0, 2\pi]$. From the above discussion, it is clear that $(\mathbf{z}_\alpha, \mathbf{z}_{\alpha+\pi})$ is the only pair of antipodal points that satisfies $f(\mathbf{z}_\alpha) = f(\mathbf{z}_{\alpha+\pi})$. \square

Now we consider the Pancake Theorem. Again the computability problem has an affirmative answer and we omit its proof. Our results below show that in general it is arbitrarily hard to compute the common bisector of two P -approximable sets even if (1) each of these sets has a positive thickness around any bisector; and (2) there is exactly one common bisector. However, if the two sets are linearly separable, the complexity of the problem is characterized by counting classes $\#P$ and $\#P_1$.

Note that we only need to compute the angle α from the positive x -axis to the common bisector L , since once α is known, the common bisector can be computed.

Theorem 4.3 *Let $\alpha \in [0, \pi)$ be a computable number. Then there exist two convex P -approximable/recognizable Jordan domains S and S' that have only one common bisector L , and the angle from the positive x -axis to L is α .*

Proof. The domain S is the same as in Theorem 3.7, and S' is the unit disk. \square

There is another version of the Pancake theorem, which states that for any bounded set $S \subseteq \mathbb{R}^2$, there exist two lines that are perpendicular to each other and divide S into four parts of equal area.

Theorem 4.4 *Let $\alpha \in [0, \pi)$ be a computable number. Then there exists a convex P -approximable/recognizable Jordan domain S whose boundary is polynomial-time computable, such that the following properties are satisfied: (1) there exist exactly two lines L_1 and L_2 that are perpendicular to each other and divide S into four parts of equal area; (2) the angle from the positive x -axis to L_1 is α ; and (3) both lines L_1 and L_2 pass through the origin.*

Proof. The domain S is similar to the one constructed in the proof of Theorem 3.7, except that we remove some parts from S in intervals $[\alpha + 3\pi/2, 2\alpha + 3\pi/2 - 1/4]$ and $[2\alpha + \pi/2 - 1/2, \alpha + \pi/2]$, with the area in each removed part equals what is added to S (compared to the unit disk) in the interval $[1/4, \alpha]$ (and in the interval $[\alpha, 1/2]$). Then the area of S is the same as the unit disk, that is, $area(S) = \pi$. See Figure 4(a) for an illustration of S . It is easy to check that the two lines L_1 and L_2 satisfying conditions (2) and (3) divide S into four parts of equal area.

It remains to show that L_1 and L_2 are the only pair of lines that can do so. Suppose that two lines L'_1 and L'_2 are perpendicular to each other and divide

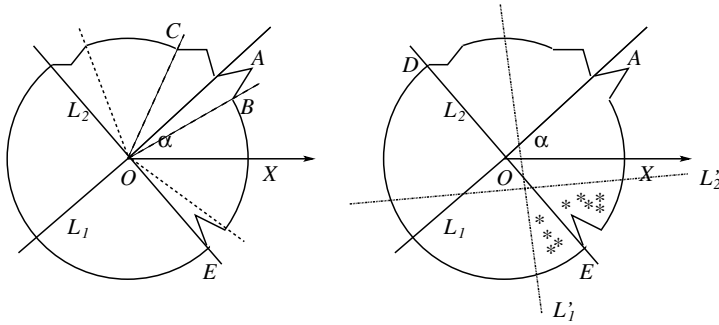


Fig. 4. (a) The domain S for Theorem 4.4, (b) Divisions with two lines.

S into four parts of equal area $area(S)/4 = \pi/4$, as in Figure 4(b). There are two cases.

Case (a): Both L_1' and L_2' contain the origin O . It is easy to check that $\{L_1, L_2\} = \{L_1', L_2'\}$.

Case (b): Not the above case. Without loss of generality, assume that L_1' does not contain the origin O . L_1' must intersect the arc from D through A to E of the boundary of S , for otherwise L_1' divides S into two parts, with the area of one of them less than $area(S)/2 = \pi/2$, since it is smaller than a half unit disk. Without loss of generality, assume that L_1' intersects the arc from D to A (see Figure 4(b)). Then D and O are on the same side of L_1' , for otherwise L_1' divides S into two parts, with the area of the part containing D less than $area(S)/2$. Then D and O must be on the same side of L_2' , for otherwise the part of S divided by L_1' and L_2' that contains O is larger than $1/4$ of a unit disk. Now the part of S divided by L_1' and L_2' that contains E (i.e., the part marked with *'s in Figure 4(b)) is of area $< area(S)/4$, which is a contradiction. \square

We point out that if we just want to get an approximate answer (i.e., for an integer $n > 0$, find a pair of two lines that are perpendicular to each other and divide S into four parts with the area difference between any two parts less than 2^{-n}), the complexity can be characterized by $NP^{\#P}$ (since we can guess two lines and then verify that the area differences are less than 2^{-n}). However, Theorem 4.4 shows that these pairs of lines may not converge rapidly to the exact pair of lines that are perpendicular to each other and divides S into four parts of equal area. An interesting question is whether there exists a natural condition, like the positive thickness condition for the problem of bisecting one set, that helps reduce the complexity.

Recall that we say two sets S and S' are linearly separable if they are on different sides of some line. If the two sets S and S' are linearly separable and each of them has a positive thickness around all its bisectors, there exists

exactly one common bisector of S and S' .

Theorem 4.5 *In the following, $(a) \Rightarrow (b) \Rightarrow (c)$:*

(a) $FP = \#P$.

(b) *For any two linearly separable bounded P -approximable sets S and S' with positive thickness around all bisectors, the unique line L that bisects simultaneously the two sets is polynomial-time computable.*

(c) $FP_1 = \#P_1$.

Proof. The spirits of the proof are quite similar to the proof of Theorem 3.3.

For the proof of $(a) \Rightarrow (b)$, suppose that S and S' are separated by a vertical line L (see Figure 5). For any line L' that intersects L , we define its direction as from a point on L' lying on the left of L to a point on L' lying on the right of L . Then we can say the angles of such lines with respect to the positive x -axis are from $-\pi/2$ to $\pi/2$.

We define a function $g : (-\pi/2, \pi/2) \rightarrow \mathbb{R}$ as follows: for any $\alpha \in (-\pi/2, \pi/2)$, there exists uniquely a line L_α that bisects S at angle α . L_α will divide S' into two parts S_1 and S_2 , with S_1 on the right of L_α (recall that L_α has a direction). Let $g(\alpha) = \text{area}(S_1) - \text{area}(S_2)$. Function g has only one root α_0 , with L_{α_0} being the unique common bisector of S and S' . Next we will show that under condition (a), α_0 is polynomial-time computable, which further implies that L_{α_0} is polynomial-time computable by Theorem 3.3.

First $g(\alpha)$ has a linear inverse modulus when α is close to the root α_0 of g : as in Figure 5, for two angles $\alpha, \beta \in (-\pi/2, \pi/2)$ with $\alpha < \beta$, S' and the intersection of L_α and L_β must be on different sides of L , which implies that $g(\alpha) < g(\beta)$; more precisely, $g(\beta) - g(\alpha)$ is twice the area of the intersection of S' and a sector region of angle $\beta - \alpha$, and from the thickness condition of S' , $g(\beta) - g(\alpha) > m(\beta - \alpha)$ for some $m > 0$, provided that α and β are sufficiently close to α_0 . Therefore, by Corollary 4.7 of Ko [7], the root α_0 of g is polynomial-time computable if g is polynomial-time computable. According to Theorem 3.3, g is polynomial-time computable under condition (a), thus the proof is completed.

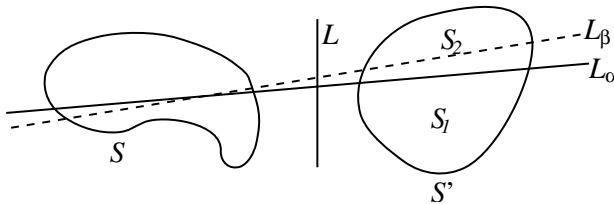


Fig. 5. The domains S and S' for $(a) \Rightarrow (b)$ of Theorem 4.5.

For the proof of $(b) \Rightarrow (c)$, we let S be a domain similar to the one in the proof of $(b) \Rightarrow (c)$ of Theorem 3.3 and S' a disk lying below S . Then the

common bisector of S and S' must go through the center of S' . The proof is completed by following $(b) \Rightarrow (c)$ of Theorem 3.6. \square

References

- [1] Abbott, Timothy, Erik D. Demaine, Martin L. Demaine, Daniel Kane, Stefan Langerman, Jelani Nelson and Vincent Yeung, *Dynamic ham-sandwich cuts of convex polygons in the plane*, In Proceedings of Canadian Conference on Computational Geometry (CCCG'05), pages 61–64, 2005.
- [2] Chou, Arthur, and Ker-I Ko, *Computational complexity of two-dimensional regions*, SIAM Journal on Computing, 24:923–947, 1995.
- [3] Du, Ding-Zhu, and Ker-I Ko, “Theory of Computational Complexity”, John Wiley & Sons, New York, 2000.
- [4] Edelsbrunner, H., and R. Waupotitsch, *Computing a ham-sandwich cut in two dimensions*, J. Symb. Comput., 2(2):171–178, 1986.
- [5] Fulton, William, “Algebraic Topology: A First Course”, Number 153 in Graduate Texts in Mathematics, Springer, 1997.
- [6] Green, F., J. Köbler, K. Regan, T. Schwentick, and J. Torán, *The Power of the Middle Bit of a $\#P$ Function*, Journal of Computer and System Sciences, 50:456–467, 1995.
- [7] Ko, Ker-I, “Complexity Theory of Real Functions”, Birkhäuser, Boston, 1991.
- [8] Lo, Chi-Yuan, Jirí Matousek, and William L. Steiger, *Ham-sandwich cuts in \mathbb{R}^d* , In Proceedings of Annual ACM Symposium on Theory of Computing (STOC 1992), pages 539–545. ACM, 1992.
- [9] Lo, Chi-Yuan, Jirí Matousek, and William L. Steiger, *Algorithms for ham-sandwich cuts*, Discrete & Computational Geometry, 11:433–452, 1994.
- [10] Ogihara, Mitsunori, Thomas Thierauf, Seinosuke Toda, and Osamu Watanabe, On closure properties of $\#P$ in the context of $PF \circ \#P$. J. Comput. Syst. Sci., 53(2):171–179, 1996.
- [11] Rudin, Walter, “Principles of Mathematical Analysis”, McGraw-Hill, 1976.
- [12] Stojmenovic, Ivan, *Bisections and ham-sandwich cuts of convex polygons and polyhedra*, Inf. Process. Lett., 38(1):15–21, 1991.