Electronic Colloquium on Computational Complexity, Report No. 57 (2024)

Computing a Fixed Point of Contraction Maps in Polynomial Queries

Xi Chen*
Columbia University
xichen@cs.columbia.edu

Yuhao Li†
Columbia University
yuhaoli@cs.columbia.edu

Mihalis Yannakakis‡
Columbia University
mihalis@cs.columbia.edu

Abstract

We give an algorithm for finding an ε-fixed point of a contraction map $f : [0,1]^k \mapsto [0,1]^k$ under the ℓ_∞-norm with query complexity $O(k^2 \log(1/\varepsilon))$.

*Supported by NSF grants IIS-1838154, CCF-2106429 and CCF-2107187.
†Supported by NSF grants IIS-1838154, CCF-2106429 and CCF-2107187.
‡Supported by NSF grants CCF-2107187 and CCF-2212233.

ISSN 1433-8092

1 Introduction

A map $f : \mathcal{M} \mapsto \mathcal{M}$ on a metric space $(\mathcal{M}, d)$ is called a *contraction* map (or a $(1-\gamma)$-contraction map) if there exists $\gamma \in (0, 1]$ such that $d(f(x), f(y)) \leq (1-\gamma) \cdot d(x, y)$ for all points $x, y \in \mathcal{M}$. In 1922, Banach [Ban22] proved a seminal fixed point theorem which states that every contraction map must have a unique fixed point, i.e., there is a unique $x \in \mathcal{M}$ that satisfies $f(x) = x$. Distinct from another renowned fixed point theorem by Brouwer, Banach's theorem not only guarantees the uniqueness of the fixed point but also provides a method for finding it: iteratively applying the map f starting from any initial point will always converge to the unique fixed point. Over the past century, Banach's fixed point theorem has found extensive applications in many fields. For example, in mathematics it can be used to prove theorems such as the Picard–Lindelöf (or Cauchy-Lipschitz) theorem on the existence and uniqueness of solutions to differential equations (see e.g. [CL55]), and the Nash embedding theorem [Nas56, Gün89]. In optimization and machine learning, it is used in the convergence and uniqueness analysis of value and policy iteration in Markov decision processes and reinforcement learning [Bel57, How60]. Indeed, as pointed out by Denardo [Den67], contraction mappings underlie many classical dynamic programming (DP) problems and sequential decision processes, including DP models of Bellman, Howard, Blackwell, Karlin and others.

A particularly important metric space to study the problem of finding a Banach's fixed point is the k-cube $[0, 1]^k$ with respect to the ℓ_∞-norm, since many important problems can be reduced to that of finding an ε-fixed point (i.e., $x \in [0, 1]^k$ satisfying $\|f(x)-x\|_\infty \leq \varepsilon$) in a $(1-\gamma)$-contraction map under the ℓ_∞-norm. Such problems arise from a variety of fields including stochastic analysis, optimization, verification, semantics, and game theory. For example, the classical dynamic programming models mentioned above (Markov decision processes etc.) involve contraction maps under the ℓ_∞-norm. Furthermore, the same holds for several well-known open problems that have been studied extensively and are currently not known to be in P. For instance, Condon's simple stochastic games (SSGs) [Con92] can be reduced to the problem of finding an ε-fixed point in a $(1-\gamma)$-contraction map over $[0, 1]^k$ under the ℓ_∞-norm. A similar reduction from [EY10] extends to an even broader class of games, namely, Shapley's stochastic games [Sha53], which lay the foundation of multi-agent reinforcement learning [Lit94]. The same holds also of course for other problems known to be subsumed by SSGs, like parity games, which are important in verification (see e.g. [EJ91, CJK+22]), and mean payoff games [ZP96]. Crucially, in all these reductions, both the approximation parameter ε and the contraction parameter γ are inversely exponential in the input size. Therefore, efficient algorithms in this context are those with a complexity upper bound that is polynomial in $k, \log(1/\varepsilon)$ and $\log(1/\gamma)$.

In this paper we consider general algorithms that access the contraction map in a black-box manner (as an oracle), and study the *query complexity* of finding an ε-fixed point of a $(1-\gamma)$-contraction map over the k-cube $[0, 1]^k$ under the ℓ_∞-norm (which we denote by $\textsc{Contraction}_\infty(\varepsilon, \gamma, k)$). An algorithm under this model is given k, ε, γ, and oracle access to an unknown $(1-\gamma)$-contraction map f over $[0, 1]^k$. In each round the algorithm can send a point $x \in [0, 1]^k$ to the oracle to reveal its value $f(x)$. The goal of the algorithm is to find an ε-fixed point with as few queries as possible.

Prior work. Despite much ongoing interest on this problem (e.g., [EY10, DP11, DTZ18, FGMS20, Hol21, FGHS23]), progress in understanding the query complexity of $\textsc{Contraction}_\infty(\varepsilon, \gamma, k)$ has been slow. Banach's value iteration method needs $\Omega((1/\gamma)\log(1/\varepsilon))$ iterations to converge to an ε-fixed point. For the special case of $k = 2$, [SS02] obtained an $O(\log(1/\varepsilon))$-query algorithm. Subsequently, [SS03] obtained an $O(\log^k(1/\varepsilon))$-query algorithm for general k by applying a non-trivial recursive binary search procedure across all k dimensions. (Recently [FGMS20] obtained similar upper bounds for all ℓ_p-norms with $2 < p < \infty$, though the complexity grows to infinity as $p \to \infty$.) Note, however, that all known upper bounds so far are exponential in either k or $\log(1/\gamma)$, and this is in sharp contrast with the ℓ_2-norm case, for which [STW93, HKS99] gave an algorithm with both query and time complexity polynomial in $k, \log(1/\varepsilon)$ and $\log(1/\gamma)$.

Our contribution. We obtain the first algorithm for $\textsc{Contraction}_\infty(\varepsilon, \gamma, k)$ with polynomial query complexity:

Theorem 1. *There is an $O(k^2 \log(1/\varepsilon))$-query algorithm for $\textsc{Contraction}_\infty(\varepsilon, \gamma, k)$.*

The observation below explains why our upper bound does not depend on γ:

Observation 1. *Let $f : [0,1]^k \mapsto [0,1]^k$ be a $(1-\gamma)$-contraction map under the ℓ_∞-norm. Consider the map $g : [0,1]^k \mapsto [0,1]^k$ defined as $g(x) := (1-\varepsilon/2)f(x)$. Clearly g is a $(1-\varepsilon/2)$-contraction. Let x be any point with $\|g(x) - x\|_\infty \le \varepsilon/2$. We have*

$$\varepsilon/2 \ge \|g(x) - x\|_\infty = \|(1-\varepsilon/2)f(x) - x\|_\infty \ge \|f(x) - x\|_\infty - \varepsilon/2$$

This gives a black-box reduction from $\textsc{Contraction}_\infty(\varepsilon, \gamma, k)$ to $\textsc{Contraction}_\infty(\varepsilon/2, \varepsilon/2, k)$, which is both query-efficient and time-efficient.

In Section 4, we give an $O(k^2 \log(1/\varepsilon))$-query algorithm for $\textsc{Contraction}_\infty(\varepsilon/2, \varepsilon/2, k)$, from which Theorem 1 follows. Indeed, note that Observation 1 holds even if f is a *non-expansive* map (i.e., f has Lipschitz constant 1: $\|f(x) - f(y)\|_\infty \le \|x - y\|_\infty$ for all $x, y \in [0,1]^k$). As a result, the same query upper bound applies to $\textsc{NonExp}_\infty(\varepsilon, k)$, the problem of finding an ε-fixed point in a non-expansive map over $[0,1]^k$ under the ℓ_∞-norm:

Corollary 1. *There is an $O(k^2 \log(1/\varepsilon))$-query algorithm for $\textsc{NonExp}_\infty(\varepsilon, k)$.*

Another corollary of Theorem 1 is about finding a *strong* ε-fixed point in a contraction map f. We say x is a strong ε-fixed point of f if $\|x - x^*\|_\infty \le \varepsilon$, where x^* is the unique fixed point of f. The following observation leads to Corollary 2, where $\textsc{StrContraction}_\infty(\varepsilon, \gamma, k)$ denotes the problem of finding a strong ε-fixed point:

Observation 2. *Let f be a $(1-\gamma)$-contraction map and x^* be its unique fixed point. Let x be any $(\varepsilon\gamma)$-fixed point of f, i.e., x satisfies $\|f(x) - x\|_\infty \le \varepsilon\gamma$. Then we have*

$$\|x - x^*\|_\infty \le \|x - f(x)\|_\infty + \|f(x) - x^*\|_\infty \le \varepsilon\gamma + (1-\gamma)\|x - x^*\|_\infty,$$

which implies $\|x - x^*\|_\infty \le \varepsilon$. This gives a black-box reduction from $\textsc{StrContraction}_\infty(\varepsilon, \gamma, k)$ to $\textsc{Contraction}_\infty(\varepsilon\gamma, \gamma, k)$, which is both query-efficient and time-efficient.

Corollary 2. *There is an $O(k^2 \log(1/(\varepsilon\gamma)))$-query algorithm for $\textsc{StrContraction}_\infty(\varepsilon, \gamma, k)$.*

In sharp contrast with Corollary 2, however, we show that it is impossible to strongly approximate an exact fixed point in a non-expansive map over $[0,1]^2$ under the ℓ_∞ norm.

Theorem 2. *There is no deterministic or randomized algorithm which, when given oracle access to any non-expansive map $f : [0,1]^2 \mapsto [0,1]^2$ under the ℓ_∞-norm, computes in an expected bounded number of queries a point that is within distance $1/4$ of an exact fixed point of f.*

The problem $\textsc{Contraction}_\infty(\varepsilon, \gamma, k)$ is a *promise problem*, i.e., it is promised that the function f in the black-box (the oracle) is a $(1-\gamma)$-contraction. In the various relevant applications (stochastic games etc.), the corresponding function that is induced is by construction a contraction, thus it is appropriate in these cases to restrict attention to functions that satisfy the contraction promise.

For any promise problem, one can define a corresponding total search problem, where the black-box can be any function f on the domain and the problem is to compute either a solution or a violation of the promise. In our case, the corresponding total search problem, denoted $\text{T-}\textsc{Contraction}_\infty(\varepsilon, \gamma, k)$, is the problem of computing for a given function $f : [0,1]^k \mapsto [0,1]^k$ either an ε-fixed point or a violation of the contraction property, i.e. a pair of points $x, y \in [0,1]^k$ such that $\|f(x) - f(y)\|_\infty > (1-\gamma)\|x - y\|_\infty$. For any promise problem, the corresponding total search problem is clearly at least as hard as the promise problem. For some problems it can be strictly harder (and it may depend on the type of violation that is desired). However, we show that in our case the two versions have the same query complexity.

Theorem 3. *There is an $O(k^2 \log(1/\varepsilon))$-query algorithm for* $\text{T-}\textsc{Contraction}_\infty(\varepsilon, \gamma, k)$.

Similar results hold for $\textsc{NonExp}_\infty(\varepsilon, k)$ and $\textsc{StrContraction}_\infty(\varepsilon, \gamma, k)$: the total search versions have the same query complexity as the corresponding promise problems.

Remark. It is also important to note that while our algorithm in Theorem 1 is query efficient, it is not time efficient for the current version. The algorithm guarantees that within polynomial queries we can find a weak ε-fixed point, but each iteration requires a brute force procedure to *determine the next query point*. We will explain more details of techniques in Section 1.1.

Other Related Work. We have already mentioned the most relevant works addressing the query complexity of computing the fixed point of a contraction map. For continuous functions $f : [0,1]^k \mapsto [0,1]^k$ that have Lipschitz constant greater than 1 (i.e. are expansive), there are exponential lower bounds on the query complexity of computing a (weak) approximate fixed point [HPV89, CD08].

$\textsc{Contraction}_\infty(\varepsilon, \gamma, k)$ when considered in the white-box model[1] can be formulated as a total search problem so that it lies in the class TFNP. In fact, it is one of the motivating problems in [DP11] to define the class CLS for capturing problems that lie in both PLS [JPY88] and PPAD [Pap94]. Later, it is placed in UEOPL [FGMS20][2], a subclass of CLS to capture problems with a unique solution. It is not known that $\textsc{Contraction}_\infty(\varepsilon, \gamma, k)$ is complete for any TFNP class. Notably, to the best of our knowledge, for the known fixed point problems that are complete for some TFNP class, their query complexity in the black-box model is exponential. Examples of such well-known problems include PPAD-complete problems Brouwer and Sperner [Pap94, CD09], PPA-complete problems Borsuk–Ulam, Tucker [Pap94, ABB20] and MöbiusSperner [DEF+21], CLS-complete problems KKT [FGHS23] and MetricBanach[3] [DTZ18], and UEOPL-complete problem OPDC [FGMS20].

[1]The white-box model refers to the model where the function is explicitly given by a polynomial-size circuit, in contrast to the black-box model where the function can only be accessed via an oracle as we studied in this paper. When we talk about a computational problem under the white-box model, we measure the efficiency by the time complexity.

[2]Technically speaking, they show that UEOPL contains the problem of computing an exact fixed point of contraction maps specified by `LinearFIXP` arithmetic circuits, where the unique fixed point is guaranteed to be rational. For our more general version $\textsc{Contraction}_\infty(\varepsilon, \gamma, k)$, the unique fixed point of the underlying map may be irrational.

[3]MetricBanach refers to the problem of computing an approximate fixed point of a contraction map where the distance function d is also part of the input.

However, our results indicate that $\textsc{Contraction}_\infty(\varepsilon,\gamma,k)$ is dramatically different from all these fixed point problems above in terms of query complexity. Thus, we would like to interpret our results as evidence supporting that $\textsc{Contraction}_\infty(\varepsilon,\gamma,k)$ under white-box model might be computationally tractable. Ideally, if it is in FP, it would imply many breakthroughs in the fields of verification, semantics, learning theory, and game theory as we discussed before.

1.1 Sketch of the Main Algorithm

We give a high-level sketch of the main query algorithm for Theorem 1. We start by discretizing the search space. Let $g:[0,n]^k \mapsto [0,n]^k$ with $g(x) := n \cdot f(x/n)$ and $n := \lceil 16/(\gamma\varepsilon)\rceil$. It is easy to show that g remains a $(1-\gamma)$-contraction over $[0,n]^k$ and it suffices to find a $(16/\gamma)$-fixed point of g. Moreover, by rounding the unique fixed point x^* of g to an integer point, we know trivially that at least one integer point x in the grid $[0:n]^k$, where $[0:n] := \{0,1,\ldots,n\}$, satisfies $\|x-x^*\|_\infty \le 1$ and it is easy to show that any such point x must be a $(16/\gamma)$-fixed point. So our goal is to find a point $x \in [0:n]^k$ that satisfies $\|x-x^*\|_\infty \le 1$ query-efficiently.

To this end, we use $\texttt{Cand}^t$ to denote the set of $[0:n]^k$ that remains possible to be close to the unknown exact fixed point x^* of g. Starting with $\texttt{Cand}^0$ set to be the full grid $[0:n]^k$, the success of the algorithm relies on whether we can cut down the size of $\texttt{Cand}^t$ efficiently. For this purpose we prove a number of geometric lemmas in Section 3 to give a characterization of the exact fixed point x^*, which lead to the following primitive used by the algorithm repeatedly:

> Given $x \in [0,n]^k$, $i \in [k]$ and $\phi \in \{\pm 1\}$, we write $\mathcal{P}_i(x,\phi)$ to denote the set of points $y \in [0,n]^k$ such that $\phi \cdot (y_i - x_i) = \|y-x\|_\infty$, where $\mathcal{P}$ is a shorthand for pyramid. Then after querying a point $a \in [0,n]^k$, either a was found to be a $(16/\gamma)$-fixed point (in which case the algorithm is trivially done), or one can find $\phi_i \in \{\pm 1\}$ for each $i \in [k]$ such that no point in $\mathcal{P}_i(a,\phi_i)$ can be close (within ℓ_∞-distance 1) to x^* (in which case we can update $\texttt{Cand}^t$ by removing all points in $\cup_{i\in[k]}\mathcal{P}_i(a,\phi_i)$).

Given this, it suffices to show that for any set of points $T \subseteq [0:n]^k$ (as $\texttt{Cand}^t$), there exists a point a to be queried such that for any $\phi_i \in \{\pm 1\}$:

$$\left| T \cap \left(\bigcup_{i\in[k]} \mathcal{P}_i(a,\phi_i) \right) \right|$$

is large relative to $|T|$, which is equivalent (up to a factor of k) to showing that there exists a point a and $i \in [k]$ such that

$$\min\left(\left|T \cap \mathcal{P}_i(a,+1)\right|, \left|T \cap \mathcal{P}_i(a,-1)\right| \right) \tag{1}$$

is large relative to $|T|$. This unfortunately turns out to be not true (see Example 1). However, it turns out that such a point (which we will refer to as a *balanced point*) always exists if we replace the integer grid $[0:n]^k$ by the grid of odd-even points: $\texttt{OE}(n,k)$, where $y \in \texttt{OE}(n,k)$ iff $y \in [0:n]^k$ and its coordinates are either all odd or all even. To prove the existence of a balanced point, we construct an infinite sequence of continuous maps $\{f^t\}$ that can be viewed as relaxed versions of the search for a balanced point. Using Brouwer's fixed point theorem, every map f^t has a fixed point p^t and thus, by the Bolzano–Weierstrass theorem, there must be an infinite subsequence of $\{p^t\}$ that converges. Letting p^* be the point it converges to, we further round p^* to $q^* \in \texttt{OE}(n,k)$ and show that the latter is a balanced point in the grid. While we show such a point always exists, the brute-force search to find $q^* \in \texttt{OE}(n,k)$ is the reason why our algorithm is not time-efficient.

2 Preliminaries

Definition 1 (Contraction). *Let $0 < \gamma < 1$ and $(\mathcal{M}, d)$ be a metric space. A map $f : \mathcal{M} \mapsto \mathcal{M}$ is a $(1-\gamma)$-*contraction map *with respect to $(\mathcal{M}, d)$ if $d(f(x), f(y)) \le (1-\gamma) \cdot d(x, y)$ for all $x, y \in \mathcal{M}$.*

A map $f : \mathcal{M} \mapsto \mathcal{M}$ is said to be non-expansive *if $d(f(x), f(y)) \le d(x, y)$ for all $x, y \in \mathcal{M}$.*

Every non-expansive map has a fixed point, i.e., x^* with $f(x^*) = x^*$, and it is unique when f is a $(1-\gamma)$-contraction map for any $\gamma > 0$. In this paper, we study the query complexity of finding an ε-fixed point of a $(1-\gamma)$-contraction map f over the k-cube $[0, 1]^k$ with respect to the infinity norm:

Definition 2 ($\textsc{Contraction}_\infty(\varepsilon, \gamma, k)$). *In problem $\textsc{Contraction}_\infty(\varepsilon, \gamma, k)$, we are given oracle access to a $(1-\gamma)$-contraction map f over $[0, 1]^k$ with respect to the infinity norm, i.e., f satisfies*

$$\|f(x) - f(y)\|_\infty \le (1-\gamma) \cdot \|x - y\|_\infty, \quad \textit{for all } x, y \in [0, 1]^k$$

and the goal is to find an ε-fixed point of f, i.e., a point $x \in [0, 1]^k$ such that $\|f(x) - x\|_\infty \le \varepsilon$.

We also write $\textsc{StrContraction}_\infty(\varepsilon, \gamma, k)$ to denote the problem with the same input but the goal is to find a strong ε-fixed point *of f, i.e., $x \in [0, 1]^k$ such that $\|x - x^*\|_\infty \le \varepsilon$, where x^* is the unique fixed point of f.*

We define similar problems for non-expansive maps over $[0, 1]^k$:

Definition 3 ($\textsc{NonExp}_\infty(\varepsilon, k)$). *In problem $\textsc{NonExp}_\infty(\varepsilon, k)$, we are given oracle access to a non-expansive map $f : [0, 1]^k \to [0, 1]^k$ with respect to the infinity norm, i.e., f satisfies*

$$\|f(x) - f(y)\|_\infty \le \|x - y\|_\infty, \quad \textit{for all } x, y \in [0, 1]^k$$

and the goal is to find an ε-fixed point of f.

We write $\textsc{StrNonExp}_\infty(\varepsilon, k)$ to denote the problem with the same input but the goal is to find a strong ε-fixed point *of f.*

Let $f : [0, 1]^k \to [0, 1]^k$ be a $(1-\gamma)$-contraction map. For convenience, our main algorithm for $\textsc{Contraction}_\infty(\varepsilon, \gamma, k)$ will work on $g : [0, n]^k \mapsto [0, n]^k$ with

$$n := \left\lceil \frac{16}{\gamma\varepsilon} \right\rceil \quad \text{and} \quad g(x) := n \cdot f(x/n), \quad \text{for all } x \in [0, n]^k.$$

We record the following simple property about g:

Lemma 1. *The map g constructed above is still a $(1-\gamma)$-contraction and finding an ε-fixed point of f reduces to finding a $(16/\gamma)$-fixed point of g.*

Proof. For any two points $x, y \in [0, n]^k$, we have

$$\|g(x) - g(y)\|_\infty = \|n \cdot f(x/n) - n \cdot f(y/n)\|_\infty \le n(1-\gamma) \cdot \|x/n - y/n\|_\infty = (1-\gamma)\|x - y\|_\infty.$$

Suppose we found a $(16/\gamma)$-fixed point a of g. We show that $x := a/n$ is an ε-fixed point of f:

$$\|f(x) - x\|_\infty = \|g(a)/n - a/n\|_\infty = \frac{1}{n} \cdot \|g(a) - a\|_\infty \le \frac{1}{n} \cdot \frac{16}{\gamma} \le \varepsilon.$$

This finishes the proof of the lemma. ∎

Notation. Given a positive integer m, we use $[m]$ to denote $\{1,\ldots,m\}$. For a real number $t \in \mathbb{R}$, we let $\mathrm{sgn}(t) = 1$ if $t > 0$, $\mathrm{sgn}(t) = -1$ if $t < 0$, and $\mathrm{sgn}(t) = 0$ if $t = 0$. Given positive integers n and k, we use $\mathtt{OE}(n,k)$ to denote the set of all integer points $x \in [0,n]^k$ such that *either* x_i is odd for all $i \in [k]$ *or* x_i is even for all $i \in [k]$. ($\mathtt{OE}$ is a shorthand for odd and even points.)

For a point $x \in \mathbb{R}^k$ (not necessarily in $[0,n]^k$), a coordinate $i \in [k]$, and a sign $\phi \in \{\pm 1\}$, we use $\mathcal{P}_i(x,\phi)$ to denote

$$\mathcal{P}_i(x,\phi) := \Big\{ y \in [0,n]^k : \phi \cdot (y_i - x_i) = \|y - x\|_\infty \Big\},$$

where $\mathcal{P}$ is a shorthand for pyramid.

Given a $(1-\gamma)$-contraction map g over $[0,n]^k$, we use $\mathtt{Fix}(g)$ to denote the unique fixed point of g. For any point $x \in [0,n]^k$, we use $\mathtt{Around}(x)$ to denote the set

$$\mathtt{Around}(x) := \big\{ y \in \mathtt{OE}(n,k) : \|x - y\|_\infty \le 1 \big\}.$$

We note that $\|g(y) - y\|_\infty \le 2$ for all $y \in \mathtt{Around}(\mathtt{Fix}(g))$ and thus, any point in $\mathtt{Around}(\mathtt{Fix}(g))$ is a desired $(16/\gamma)$-fixed point (given that $\gamma < 1$). To see this, letting $x = \mathtt{Fix}(g)$, we have

$$\|g(y) - y\|_\infty \le \|g(y) - g(x)\|_\infty + \|x - y\|_\infty \le 2, \quad \text{for any } y \in \mathtt{Around}(x).$$

3 Characterizing the Unique Fixed Point

Lemma 2. *Let $g : [0,n]^k \mapsto [0,n]^k$ be a $(1-\gamma)$-contraction map and let $a \in [0,n]^k$ be a point such that $\|g(a) - a\|_\infty > 16/\gamma$ and $s \in \{\pm 1, 0\}^k$ be the sign vector such that $s_i = \mathrm{sgn}(g(a)_i - a_i)$. Then*

$$\mathtt{Fix}(g) \in \bigcup_{i:s_i \neq 0} \mathcal{P}_i(a + 4s, s_i).$$

Proof. Let $c = a + 4s$ and $x^* = \mathtt{Fix}(g)$ be the unique fixed point.

First, we show that $\|x^* - a\|_\infty > 8/\gamma$. Otherwise, we have

$$\|g(a) - a\|_\infty \le \|g(a) - g(x^*)\|_\infty + \|x^* - a\|_\infty \le (1 - \gamma + 1) \cdot \|x^* - a\|_\infty \le \frac{16}{\gamma},$$

which contradicts the assumption that $\|g(a) - a\|_\infty > 16/\gamma$. Now it suffices to show that $g(x) \neq x$ for any point $x \in [0,n]^k$ that satisfies both

$$x \notin \bigcup_{i,s_i \neq 0} \mathcal{P}_i(c, s_i) \quad \text{and} \quad \|x - a\|_\infty > \frac{8}{\gamma}.$$

Let j be a dominating coordinate between x and c, i.e., a $j \in [k]$ such that $|x_j - c_j| = \|x - c\|_\infty$. We divide the proof into two parts.

Part 1: $s_j = 0$**.** Thus $g(a)_j = a_j$ and $c_j = a_j$. Assume without loss of generality that $x_j \ge c_j$; the case when $x_j \le c_j$ is symmetric. On the one hand, we have $x_j - c_j = \|x - c\|_\infty$, which gives

$$x_j = c_j + \|x - c\|_\infty. \tag{$\star$}$$

On the other hand, using that g is a $(1-\gamma)$-contraction and $g(a)_j = a_j$, we have

$$g(x)_j - a_j = g(x)_j - g(a)_j \le (1 - \gamma) \cdot \|x - a\|_\infty.$$

Equivalently, $g(x)_j \le a_j + \|x-a\|_\infty - \gamma\|x-a\|_\infty$. Combining with $\|x-a\|_\infty > 8/\gamma$, this implies

$$g(x)_j < a_j + \|x-a\|_\infty - 8. \tag{$\diamond$}$$

Putting ($\star$) and ($\diamond$) together and the facts that $\|c-a\|_\infty = 4$ and $c_j = a_j$, we have

$$x_j - g(x)_j > \|x-c\|_\infty - \|x-a\|_\infty + 8 \ge -\|c-a\|_\infty + 8 > 0,$$

which implies that $g(x)_j \ne x_j$ and $g(x) \ne x$.

Part 2: $s_j \ne 0$. Assume without loss of generality that $s_j = +1$; the case $s_j = -1$ is symmetric.

Since we are considering points not in $\bigcup_{i:s_i\ne 0} \mathcal{P}_i(c, s_i)$, it must be the case that $x \in \mathcal{P}_j(c, -s_j)$ and thus, $x_j \le c_j$. Since $\|x-a\|_\infty > 8/\gamma$, we have $x_j \le a_j$; otherwise $a_j \le x_j \le c_j$ and thus,

$$\|x-a\|_\infty \le \|x-c\|_\infty + \|a-c\|_\infty \le 8.$$

Given the $(1-\gamma)$-contraction of g, we have $g(a)_j - g(x)_j \le (1-\gamma)\cdot\|x-a\|_\infty$, which implies

$$g(x)_j \ge g(a)_j - \|x-a\|_\infty + \gamma\|x-a\|_\infty > g(a)_j - \|x-a\|_\infty + 8. \tag{2}$$

Next, we show an upper bound on $\|x-a\|_\infty$. Recall that $x \in \mathcal{P}_j(c, -s_j)$. Consider $y = x - 4s$. We have $y \in \mathcal{P}_j(a, -s_j)$. So

$$\|x-a\|_\infty \le \|x-y\|_\infty + \|y-a\|_\infty = 4 + (a_j - y_j) = 4 + (a_j - x_j + 4) = 8 + (a_j - x_j).$$

Given this and plugging the upper bound in Equation (2), we will get

$$g(x)_j > g(a)_j + x_j - a_j.$$

Recall that $s_j = +1$ implies that $g(a)_j > a_j$. So we have $g(x)_j > x_j$ and $g(x) \ne x$.

This finishes the proof of Lemma 2. ◻

Lemma 3. *Let $b \in \mathbb{R}^k$ and $s \in \{\pm 1, 0\}^k$ such that $s \ne 0^k$. Then every $x \in \bigcup_{i:s_i\ne 0} \mathcal{P}_i(b+2s, s_i)$ must have* $\texttt{Around}(x) \subseteq \bigcup_{i:s_i\ne 0} \mathcal{P}_i(b, s_i)$.

Proof. Let i^* be such that $s_{i^*} \ne 0$ and $x \in \mathcal{P}_{i^*}(b+2s, s_{i^*})$. Assume without loss of generality that $s_{i^*} = +1$. We have $x_{i^*} - (b_{i^*} + 2) \ge |x_i - (b_i + 2s_i)|$ for every $i \in [k]$.

Fix an arbitrary $y \in \texttt{Around}(x)$. We must have $y_{i^*} - b_{i^*} \ge 0$, which follows from

$$0 \le x_{i^*} - (b_{i^*} + 2) \le y_{i^*} + 1 - (b_{i^*} + 2)$$

and thus, $y_{i^*} \ge b_{i^*} + 1$. Let

$$j \in \operatorname*{arg\,max}_{i\in[k]} \big\{ s_i(y_i - b_i) \mid s_i \ne 0 \big\}.$$

Since $s_{i^*} = +1$, we have

$$s_j(y_j - b_j) \ge y_{i^*} - b_{i^*} \ge 0.$$

Our goal is to show $y \in \mathcal{P}_j(b, s_j)$ and it suffices for us to show $|y_j - b_j| \ge |y_i - b_i|$ for all $i \in [k]$.

Let's consider first an arbitrary $i \in [k]$ with $s_i = 0$. Recall that $x_{i^*} - (b_{i^*} + 2) \ge |x_i - b_i|$. In particular, this implies $|x_{i^*} - b_{i^*}| - 2 \ge |x_i - b_i|$. Putting everything together, we have

$$|y_j - b_j| \ge |y_{i^*} - b_{i^*}| \ge |x_{i^*} - b_{i^*}| - 1 \ge |x_i - b_i| + 1 \ge |y_i - b_i|.$$

Finally let's consider an $i \in [k]$ such that $s_i \ne 0$. If $s_i(y_i - b_i) > 0$, by the definition of how we picked j, we have $|y_j - b_j| \ge |y_i - b_i|$. If $s_i(y_i - b_i) < 0$, then we have

$$|y_i - b_i| = |y_i - (b_i + 2s_i)| - 2 \le |x_i - (b_i + 2s_i)| - 1 \le x_{i^*} - b_{i^*} - 3 \le y_{i^*} - b_{i^*} - 2 < |y_j - b_j|.$$

This finishes the proof of the lemma. ◻

Lemma 4. *Let $a \in [0, n]^k$ and $s \in \{\pm 1, 0\}^k$ such that $s \neq 0^k$. Then for every $j \in [k]$, there exists $\phi \in \{\pm 1\}$ such that*

$$\mathcal{P}_j(a, \phi) \cap \left(\bigcup_{i: s_i \neq 0} \mathcal{P}_i(a + 2s, s_i) \right) = \emptyset.$$

Proof. First we note that

$$\left(\bigcup_{i: s_i \neq 0} \mathcal{P}_i(a, -s_i) \right) \cap \left(\bigcup_{i: s_i \neq 0} \mathcal{P}_i(a + 2s, s_i) \right) = \emptyset.$$

This implies that

$$\mathcal{P}_j(a, \phi) \cap \left(\bigcup_{i: s_i \neq 0} \mathcal{P}_i(a + 2s, s_i) \right) = \emptyset$$

for all j with $s_j \neq 0$ by setting $\phi = -s_j$.

Now consider a j with $s_j = 0$. Under this case, we show in fact that

$$\mathcal{P}_j(a, \phi) \cap \left(\bigcup_{i: s_i \neq 0} \mathcal{P}_i(a + 2s, s_i) \right) = \emptyset$$

for both $\phi \in \{\pm 1\}$. Consider any point $x \in \bigcup_{i: s_i \neq 0} \mathcal{P}_i(a + 2s, s_i)$ and we show that $x \notin \mathcal{P}_j(a, -1)$ and $x \notin \mathcal{P}_j(a, +1)$. Let $b = a + 2s$. As $x \in \bigcup_{i: s_i \neq 0} \mathcal{P}_i(b, s_i)$, there exists i^* with $s_{i^*} \neq 0$ such that

$$s_{i^*}(x_{i^*} - b_{i^*}) = \|x - b\|_\infty \geq |x_j - b_j|.$$

Note also that $|x_{i^*} - a_{i^*}| = s_{i^*}(x_{i^*} - b_{i^*} + 2s_{i^*}) = s_{i^*}(x_{i^*} - b_{i^*}) + 2$ and $b_j = a_j$, so we have

$$|x_{i^*} - a_{i^*}| = \|x - b\|_\infty + 2 > |x_j - b_j| = |x_j - a_j|.$$

Thus $x \notin \mathcal{P}_j(a, -1)$ and $x \notin \mathcal{P}_j(a, +1)$. This finishes the proof of the lemma. □

4 The Algorithm

We prove Theorem 1 in this section. Our algorithm for $\textsc{Contraction}_\infty(\varepsilon, \gamma, k)$ is described in Algorithm 1. Given oracle access to a $(1 - \gamma)$-contraction map $g : [0, n]^k \to [0, n]^k$, we show that it can find a $(16/\gamma)$-fixed point of g within

$$O\left(k^2 \log\left(\frac{1}{\varepsilon\gamma} \right) \right)$$

many queries. This is sufficient given Lemma 1.

The analysis of Algorithm 1 uses the following theorem which we prove in the next section. In particular, it guarantees the existence of the point a^t to be queried in round t that satisfies (3). We will call a point q^* with the property stated in Theorem 4 below a *balanced* point for T.

Theorem 4. *For any $T \subseteq \mathtt{OE}(n, k)$, there exist a point $q^* \in \mathtt{OE}(n, k)$ and an $i^* \in [k]$ such that*

$$\left| \mathcal{P}_{i^*}(q^*, \phi) \cap T \right| \geq \frac{1}{2k} \cdot |T|, \quad \textit{for both } \phi \in \{\pm 1\}.$$

Algorithm 1 Query Algorithm for $\textsc{Contraction}_\infty(\varepsilon, \gamma, k)$

Let $\texttt{Cand}^0 \leftarrow \texttt{OE}(n, k)$

for $t = 1, 2, \ldots$ **do**

Find and **query** an $a^t \in \texttt{OE}(n, k)$ such that a^t is a balanced point of $\texttt{Cand}^{t-1}$ on some $j \in [k]$:

$$\left|\mathcal{P}_j(a^t, +1) \cap \texttt{Cand}^{t-1}\right|, \left|\mathcal{P}_j(a^t, -1) \cap \texttt{Cand}^{t-1}\right| \geq \frac{1}{2k} \cdot \left|\texttt{Cand}^{t-1}\right| \tag{3}$$

if $\|g(a^t) - a^t\|_\infty \leq 16/\gamma$ **then return** a^t as a $(16/\gamma)$-fixed point of g

Let $s \in \{\pm 1, 0\}^k$ be such that $s_i = \operatorname{sgn}(g(a^t)_i - a^t_i)$ for all $i \in [k]$, $b^t \leftarrow a^t + 2s$, and

$$\texttt{Cand}^t \leftarrow \texttt{Cand}^{t-1} \cap \left(\bigcup_{i : s_i \neq 0} \mathcal{P}_i(b^t, s_i) \right). \tag{4}$$

At a high level, Algorithm 1 maintains a subset of grid points $\texttt{OE}(n, k)$ as candidate solutions, which is denoted by $\texttt{Cand}^t$ after round t. We show the following invariants:

Lemma 5. *For every round $t \geq 1$, either the point a^t queried is a $(16/\gamma)$-fixed point of g (and the algorithm terminates), or we have both $\texttt{Around}(\texttt{Fix}(g)) \subseteq \texttt{Cand}^t$ and*

$$\left|\texttt{Cand}^t\right| \leq \left(1 - \frac{1}{2k}\right) \cdot \left|\texttt{Cand}^{t-1}\right|. \tag{5}$$

Proof of Lemma 5. We start with the proof of (5). Suppose that a^t satisfies (3) with $j \in [k]$. Then by Lemma 4, there must exist a sign $\phi \in \{\pm 1\}$ such that

$$\mathcal{P}_j(a^t, \phi) \cap \left(\bigcup_{i : s_i \neq 0} \mathcal{P}_i(b^t, s_i) \right) = \emptyset.$$

The inequality (5) follows directly from (3).

Next we prove by induction that $\texttt{Around}(\texttt{Fix}(g)) \subseteq \texttt{Cand}^t$ for every t before the round that the algorithm terminates. The basis is trivial given that $\texttt{Cand}^0$ is set to be $\texttt{OE}(n, k)$. For round $t \geq 1$, we assume that $\|g(a^t) - a^t\|_\infty > 16/\gamma$; otherwise a solution is found and the algorithm terminates.

Let $b^t = a^t + 2s$ and $c^t = b^t + 2s$. By Lemma 2, we know that

$$\texttt{Fix}(g) \in \left(\bigcup_{i : s_i \neq 0} \mathcal{P}_i(c^t, s_i) \right).$$

Since b^t is defined as $c^t - 2s$, by Lemma 3, we know that

$$\texttt{Around}(\texttt{Fix}(g)) \subseteq \bigcup_{i : s_i \neq 0} \mathcal{P}_i(b^t, s_i).$$

We finish the proof by using the inductive hypothesis $\texttt{Around}(\texttt{Fix}(g)) \subseteq \texttt{Cand}^{t-1}$ and (4). $\square$

Given Lemma 5 and that $|\texttt{Cand}^0| \le n^k$, within at most

$$O\left(k\log(n^k)\right) = O\left(k^2 \log\left(\frac{1}{\varepsilon\gamma}\right)\right)$$

many rounds, one of the points a^t queried by the algorithm must be a $(16/\gamma)$-fixed point of g. This finishes the proof of Theorem 1.

5 Existence of Balanced Point

We prove Theorem 4 in this section. Before that, we first illustrate why the structure of $\texttt{OE}(n,k)$ is necessary to guarantee the existence of a balanced point. In particular, for the standard grid $[n]^k$, we give an example of a set $T \subseteq [n]^k$ such that no point in $[n]^k$ can be a balanced point.

Example 1. *Let $p = (n/2, \ldots, n/2)$ if n is odd and let $p = (\frac{n+1}{2}, \cdots, \frac{n+1}{2})$ if n is even. Note that $p \notin [n]^k$. We construct the following $T \subseteq [n]^k$ such that p is the only choice of a balanced point:*

$$T = \left\{x \in [n]^k : |x_i - p_i| = \|x - p\|_\infty \text{ for all } i \in [k]\right\}.$$

Precisely, it is easy to verify that for any point $p' \neq p$ and any coordinate $i \in [k]$, we have that

$$\left|\mathcal{P}_{i^*}(p', \phi) \cap T\right| \le O\left(|T|/2^k\right), \quad \text{for some } \phi \in \{\pm 1\}.$$

However, since $p \notin [n]^k$, we conclude that there is no balanced point for T in $[n]^k$.

We restate Theorem 4:

Theorem 4. *For any $T \subseteq \texttt{OE}(n,k)$, there exist a point $q^* \in \texttt{OE}(n,k)$ and an $i^* \in [k]$ such that*

$$\left|\mathcal{P}_{i^*}(q^*, \phi) \cap T\right| \ge \frac{1}{2k} \cdot |T|, \quad \text{for both } \phi \in \{\pm 1\}.$$

Proof. For each positive integer $t \ge 4$ we let

$$S^t := \cup_{x \in T} B(x, 1/t) \subset [-1/4, n + 1/4]^k,$$

where $B(x, 1/t)$ denotes the ℓ_2-ball of radius $1/t$ centered at x. We write $\mathbf{vol}(S^t)$ to denote the volume of S^t and $\mathbf{vol}(S^t \cap \mathcal{P})$ to denote the volume of the intersection of S^t and some pyramid $\mathcal{P}$.

We apply Brouwer's fixed point to prove the existence of a balanced (real) point for the balls:

Lemma 6. *For every integer $t \ge 4$, there exist $p^* \in [-1/4, n+1/4]^k$ and a coordinate $i^* \in [k]$ such that $\mathbf{vol}(\mathcal{P}_{i^*}(p^*, +1) \cap S^t) = \mathbf{vol}(\mathcal{P}_{i^*}(p^*, -1) \cap S^t) \ge \mathbf{vol}(S^t)/2k$.*

Proof. We define a continuous map $f : [-1/4, n+1/4]^k \mapsto [-1/4, n+1/4]^k$ and apply Brouwer's fixed point theorem on f to find a fixed point p^* of f, and show that p^* satisfies the property above.

We define f as follows: For every $p \in [-1/4, n+1/4]^k$ and $i \in [k]$, let

$$f_i(p) := p_i + \frac{\mathbf{vol}\big(\mathcal{P}_i(p, +1) \cap S^t\big) - \mathbf{vol}\big(\mathcal{P}_i(p, -1) \cap S^t\big)}{(n + 0.5)^{k-1}}.$$

It is clear that f is continuous. To see that it is from $[-1/4, n+1/4]^k$ to itself, we note that

$$0 \le \frac{\mathbf{vol}\big(\mathcal{P}_i(p,+1)\cap S^t\big)}{(n+0.5)^{k-1}} \le n+\frac{1}{4}-p_i \quad \text{and} \quad 0 \le \frac{\mathbf{vol}\big(\mathcal{P}_i(p,-1)\cap S^t\big)}{(n+0.5)^{k-1}} \le p_i+\frac{1}{4}.$$

As a result, one can apply Brouwer's fixed point theorem on f to conclude that there exists a point $p^* \in [-1/4, n+1/4]^k$ such that $f(p^*) = p^*$, which implies that

$$\mathbf{vol}\big(\mathcal{P}_i(p^*,+1)\cap S^t\big) = \mathbf{vol}\big(\mathcal{P}_i(p^*,-1)\cap S^t\big)$$

for all $i \in [k]$. On the other hand, we have

$$\sum_{i\in[k]} \Big(\mathbf{vol}\big(\mathcal{P}_i(p^*,+1)\cap S^t\big) + \mathbf{vol}\big(\mathcal{P}_i(p^*,-1)\cap S^t\big)\Big) = \mathbf{vol}(S^t).$$

Therefore there must be an $i^* \in [k]$ such that

$$\mathbf{vol}\big(\mathcal{P}_{i^*}(p^*,+1)\cap S^t\big) = \mathbf{vol}\big(\mathcal{P}_{i^*}(p^*,-1)\cap S^t\big) \ge \frac{\mathbf{vol}(S^t)}{2k}.$$

This finishes the proof of the lemma. $\square$

By Lemma 6, we have that for every t, there exist $p^t \in [-1/4, n+1/4]^k$ and $i_t \in [k]$ such that

$$\mathbf{vol}\big(\mathcal{P}_{i_t}(p^t,+1)\cap S^t\big) = \mathbf{vol}\big(\mathcal{P}_{i_t}(p^t,-1)\cap S^t\big) \ge \frac{\mathbf{vol}(S^t)}{2k}.$$

Given that there are only k choices for i_t, there exists an $i^* \in [k]$ such that $\{p^t\}$ has an infinite subsequence $\{p^{t_\ell}\}_{\ell\ge 1}$ with $t_1 < t_2 < \cdots$ such that

$$\mathbf{vol}\big(\mathcal{P}_{i^*}(p^{t_\ell},+1)\cap S^{t_\ell}\big) = \mathbf{vol}\big(\mathcal{P}_{i^*}(p^{t_\ell},-1)\cap S^{t_\ell}\big) \ge \frac{\mathbf{vol}(S^{t_\ell})}{2k}$$

for all $\ell \ge 1$. Given that $[-1/4, n+1/4]^k$ is compact, $\{p^{t_\ell}\}$ has an infinite subsequence that converges to a point $p^* \in [-1/4, n+1/4]^k$. For convenience, we still refer to the subsequence as $\{p^{t_\ell}\}_{\ell\ge 1}$.

In Lemma 7, we show that both $\mathcal{P}_{i^*}(p^*,+1)\cap T$ and $\mathcal{P}_{i^*}(p^*,-1)\cap T$ are at least $|T|/2k$. After this, in Lemma 8, we show how to round p^* to $q^* \in \mathtt{OE}(n,k)$ while making sure that

$$\mathcal{P}_{i^*}(p^*,+1)\cap T \subseteq \mathcal{P}_{i^*}(q^*,+1)\cap T \quad \text{and} \quad \mathcal{P}_{i^*}(p^*,-1)\cap T \subseteq \mathcal{P}_{i^*}(q^*,-1)\cap T.$$

Our goal then follows by combining these two lemmas.

Lemma 7. *We have*

$$\min\left\{\frac{|\mathcal{P}_{i^*}(p^*,+1)\cap T|}{|T|}, \frac{|\mathcal{P}_{i^*}(p^*,-1)\cap T|}{|T|}\right\} \ge \frac{1}{2k}.$$

Proof. We write A to denote the following (potentially empty) set of positive real numbers defined using p^*: $a \in (0,1)$ is in A if there are $i \ne j \in [k]$ such that either

1. $p^*_i + p^*_j$ is an integer plus a; or

2. $p_i^* + p_j^*$ is an integer minus a; or

3. $p_i^* - p_j^*$ is an integer plus a; or

4. $p_i^* - p_j^*$ is an integer minus a.

Consider the easier case when A is empty, i.e. $p_i^* + p_j^*$ and $p_i^* - p_j^*$ are integers for all $i \neq j \in [k]$. Let ℓ be a sufficiently large integer such that $1/t_\ell \leq 0.1$ and $\|p^{t_\ell} - p^*\|_\infty \leq 0.1$. We show that

$$\frac{|\mathcal{P}_{i^*}(p^*, +1) \cap T|}{|T|} \geq \frac{\mathbf{vol}(\mathcal{P}_{i^*}(p^{t_\ell}, +1) \cap S^{t_\ell})}{\mathbf{vol}(S^{t_\ell})} \geq \frac{1}{2k}$$

and the same inequality holds for the -1 side. For this purpose it suffices to show that every point $x \in T$ satisfies that

$$\mathcal{P}_{i^*}(p^{t_\ell}, +1) \cap B(x, 1/t_\ell) \neq \emptyset \quad \Longrightarrow \quad x \in \mathcal{P}_{i^*}(p^*, +1).$$

Let's prove the contrapositive so take any $x \notin \mathcal{P}_{i^*}(p^*, +1)$. There exists a $j \neq i^*$ such that either

$$x_{i^*} - p_{i^*}^* < x_j - p_j^* \quad \text{or} \quad x_{i^*} - p_{i^*}^* < p_j^* - x_j$$

For the first case we have $x_{i^*} - x_j < p_{i^*}^* - p_j^*$. Since both sides are integers we have

$$x_{i^*} - x_j \leq p_{i^*}^* - p_j^* - 1 \tag{6}$$

so intuitively x is far from $\mathcal{P}_{i^*}(p^*, +1)$. From this we can conclude that $B(x, 1/t_\ell) \cap \mathcal{P}_{i^*}(p^{t_\ell}, +1)$ is empty. To see this is the case, for any $y \in B(x, 1/t_\ell)$, it follows from $\|x - y\|_\infty \leq \|x - y\|_2 \leq 0.1$ and $\|p^* - p^{t_\ell}\|_\infty \leq 0.1$ and (6) that

$$y_{i^*} - y_j \leq p_{i^*}^{t_\ell} - p_j^{t_\ell} - (1 - 0.4) < p_{i^*}^{t_\ell} - p_j^{t_\ell}$$

and thus, $y \notin \mathcal{P}_{i^*}(p^{t_\ell}, +1)$. The other case follows from a similar argument.

Now we consider the general case when A is not empty and let $\alpha > 0$ be the smallest value in A; note that $\alpha \leq 1/2$. In this case we let ℓ be a sufficiently large integer such that $1/t_\ell \leq 0.1\alpha$ and $\|p^{t_\ell} - p^*\|_\infty \leq 0.1\alpha$. Similarly it suffices to show that every point $x \in T$ satisfies that

$$\mathcal{P}_{i^*}(p^{t_\ell}, +1) \cap B(x, 1/t_\ell) \neq \emptyset \quad \Longrightarrow \quad x \in \mathcal{P}_{i^*}(p^*, +1).$$

Let's prove the contrapositive so take any $x \notin \mathcal{P}_{i^*}(p^*, +1)$. There exists a $j \neq i^*$ such that either

$$x_{i^*} - p_{i^*}^* < x_j - p_j^* \quad \text{or} \quad x_{i^*} - p_{i^*}^* < p_j^* - x_j$$

For the first case we have $x_{i^*} - x_j < p_{i^*}^* - p_j^*$. Since $p_{i^*}^* - p_j^*$ is either an integer or an integer $\pm$ something that is between α and $1 - \alpha$, we have

$$x_{i^*} - x_j \leq p_{i^*}^* - p_j^* - \alpha.$$

The rest of the proof is similar. $\square$

Given p^*, we round it to an integer point $q^* \in \mathtt{OE}(n, k)$ as follows. First let $q_{i^*}^* \in [0, n]$ be an integer such that $|p_{i^*}^* - q_{i^*}^*| \leq 1/2$ (note that $q_{i^*}^*$ may not be unique but we can break ties arbitrarily). It is clear that $q_{i^*}^* \in \{0, \ldots, n\}$. Assume without loss of generality that $q_{i^*}^*$ is even (so we need to set q_j^* to be even for every other j, in order to have $q^* \in \mathtt{OE}(n, k)$). Then for each $j \neq i^*$:

1. $q_j^* = 0$ if $p_j^* \in [-1/4, 0)$;
2. q_j^* is set to be the even number in $\{n-1, n\}$ if $p_j^* \in (n, n+1/4]$; and
3. Otherwise, set q_j^* to be an even number in $\{0, \ldots, n\}$ such that $|p_j^* - q_j^*| \le 1$ (again breaking ties arbitrarily).

Note that we have $q^* \in \mathtt{OE}(n,k)$ and it satisfies $|p_j^* - q_j^*| \le 5/4$ for all $j \neq i^*$.

Lemma 8. *The point q^* satisfies $q^* \in \mathtt{OE}(n,k)$ and*

$$\mathcal{P}_{i^*}(p^*, +1) \cap T \subseteq \mathcal{P}_{i^*}(q^*, +1) \cap T \quad \textit{and} \quad \mathcal{P}_{i^*}(p^*, -1) \cap T \subseteq \mathcal{P}_{i^*}(q^*, -1) \cap T.$$

Proof. Let's prove the first part since the other part is symmetric.

Let $x \in \mathtt{OE}(n,k)$ be a point in $\mathcal{P}_{i^*}(p^*, +1) \cap T$. So for every $j \neq i^*$ we have

$$x_{i^*} - p_{i^*}^* \ge x_j - p_j^* \quad \text{and} \quad x_{i^*} - p_{i^*}^* \ge p_j^* - x_j,$$

or equivalently

$$x_{i^*} - x_j \ge p_{i^*}^* - p_j^* \quad \text{and} \quad x_{i^*} + x_j \ge p_{i^*}^* + p_j^*.$$

It suffices to show that

$$x_{i^*} - x_j \ge q_{i^*}^* - q_j^* \quad \text{and} \quad x_{i^*} + x_j \ge q_j^* + q_{i^*}^*$$

To see the first part, we have

$$x_{i^*} - x_j \ge p_{i^*}^* - p_j^* \ge (q_{i^*}^* - 1/2) - (q_j^* + 5/4) = (q_{i^*}^* - q_j^*) - 7/4.$$

But given that $x \in \mathtt{OE}(n,k)$ and $q^* \in \mathtt{OE}(n,k)$, both $x_{i^*} - x_j$ and $q_{i^*}^* - q_j^*$ are even numbers and thus, the inequality above implies $x_{i^*} - x_j \ge q_{i^*}^* - q_j^*$. The other part can be proved similarly. □

This finishes the proof of Theorem 4. □

Remark on Theorem 4. We note that the (possibly off-grid point) p^* in the proof already satisfies the desired property and our algorithm can proceed by querying p^*. However, p^* as defined here is the limit of fixed points found in an infinite sequence of maps. In contrast, Lemma 8 shows that, after rounding, a grid balanced point always exists, which can be found by brute-force enumeration.

6 Impossibility of Strong Approximation under Non-expansion

We consider functions f on the plane with bounded domain and range, e.g. the unit square, that are non-expansive under the ℓ_∞ metric. We will show the following impossibility result.

Theorem 2. *There is no deterministic or randomized algorithm which, when given oracle access to any non-expansive map $f : [0,1]^2 \mapsto [0,1]^2$ under the ℓ_∞-norm, computes in an expected bounded number of queries a point that is within distance $1/4$ of an exact fixed point of f.*

In the proof it will be more convenient to use as the domain a square that is tilted by 45°. We call a rectangle whose sides are at 45° and −45°, a *diamond*. Let D be the diamond whose vertices are the midpoints of the sides of the unit square. Any function g over D can be extended to a function g' over the unit square, by defining for every point $p \in [0,1]^2$ the value of the function as $g'(p) = g(\pi(p))$, where $\pi(p)$ is the projection of p onto D. Clearly, for any two points $p, q \in [0,1]^2$, $\|\pi(p) - \pi(q)\|_\infty \le \|p - q\|_\infty$, hence if the function g over D is non-expansive, then so is the function g' over $[0,1]^2$. Furthermore, the fixed points of g' are exactly the fixed points of g.

We will prove the statement of the theorem for the domain D. The claim then follows for the unit square. To see this, restrict attention to the non-expansive functions g' on the unit square that are extensions of functions g on the diamond D. If we have an algorithm for the unit square, then we can use the algorithm also for the diamond D: when the algorithm queries a point $p \in [0,1]^2$ then we query instead its projection $\pi(q) \in D$. If the algorithm outputs at the end a point that is close to a fixed point of g', then its projection on D is a valid output for g.

For any $\delta \in (0, 1/2)$ and any point s on the SW or NE side of the diamond D that is at least at Euclidean distance δ from the vertices of D, we will define a non-expansive function $f_{\delta,s}$ with unique fixed point s. The function is defined as follows. Draw the line l_0 through s at 45° and let t be the point of intersection with the opposite side of D. Let l_1 and l_2 be the two lines parallel to l_0 that are left and right of l_0 respectively at Euclidean distance δ, and let D_0 be the strip of D that is strictly between the lines l_1 and l_2. Let $D' = D \setminus D_0$. Every point $p \in D'$ is mapped by $f_{\delta,s}$ to the point that is at Euclidean distance δ towards the line l_0; i.e., if $p = (p_1, p_2)$ is left and above l_0 then $f_{\delta,s}(p) = (p_1 + \delta/\sqrt{2}, p_2 - \delta/\sqrt{2})$, and if p is right and below l_0 then $f_{\delta,s}(p) = (p_1 - \delta/\sqrt{2}, p_2 + \delta/\sqrt{2})$.

For a point p in D_0 we define $f_{\delta,s}(p)$ as follows. Let p' be the projection of p onto the line l_0. Then $f_{\delta,s}(p)$ is the point on l_0 that is at Euclidean distance $(\delta - |pp'|) \cdot |p's|$ from p' in the direction of s, where $|pp'|, |p's|$ are the (Euclidean) lengths of the segments pp' and $p's$. Thus for example, if $p = s$ then $p' = s$ and $f_{\delta,s}(s) = s$. If $p = t$ then $p' = t$ and $(\delta - |pp'|) \cdot |p's| = \delta/\sqrt{2}$, so t moves along l_0 distance $\delta/\sqrt{2}$ towards s. Note that if p is on line l_1 or l_2 (i.e. on the boundary sides between D_0 and D'), then $(\delta - |pp'|) \cdot |p's| = 0$, since $|pp'| = \delta$, thus p is mapped to p' whether we treat p as a member of D' or as a member of D_0. It follows that $f_{\delta,s}(p)$ is continuous over D.

As we noted above, s is a fixed point of $f_{\delta,s}(p)$. We claim that it is the only fixed point. Clearly, any fixed point p must be in D_0 and must lie on the line l_0, thus $p = p'$. It must satisfy also $(\delta - |pp'|) \cdot |p's| = 0$, hence $|p's| = 0$, and thus, $p = s$.

We will show now that $f_{\delta,s}$ is a non-expansive function, i.e. that $\|f_{\delta,s}(p) - f_{\delta,s}(q)\|_\infty \le \|p - q\|_\infty$ for all $p, q \in D$. We show first that it suffices to check pairs p, q that are *diagonal* to each other, i.e. such that the line connecting them is at 45° or −45°. Note that such points have the property that the L_∞ distance is tight in both coordinates, $\|p - q\|_\infty = |p_1 - q_1| = |p_2 - q_2|$.

Lemma 9. *If a function f on the diamond D satisfies $\|f(p) - f(q)\|_\infty \le \|p - q\|_\infty$ for all diagonal pairs of points p, q, then f is non-expansive.*

Proof. Let x, y be any two points that are not diagonal. Consider the diamond with opposite vertices x, y, i.e. draw the lines through x, y at 45° and −45° and considered the rectangle enclosed by them. Let z, w be the other two vertices of this diamond. Suppose without loss of generality that $\|x - y\|_\infty = x_1 - y_1 > |x_2 - y_2|$. Then $x_1 > z_1 > y_1$, and similarly for w. We have $\|x - y\|_\infty = x_1 - y_1 = (x_1 - z_1) + (z_1 - y_1) = \|x - z\|_\infty + \|z - y\|_\infty$. Since f is non-expansive on diagonal pairs, $\|f(x) - f(z)\|_\infty \le \|x - z\|_\infty$ and $\|f(z) - f(y)\|_\infty \le \|z - y\|_\infty$. Therefore, $\|f(x) - f(y)\|_\infty \le \|f(x) - f(z)\|_\infty + \|f(z) - f(y)\|_\infty \le \|x - z\|_\infty + \|z - y\|_\infty = \|x - y\|_\infty$. $\square$

Remark. The lemma can be shown to hold more generally in any dimension. That is, if $f : [0,1]^k \mapsto$

$[0,1]^k$ has the property that $\|f(p)-f(q)\|_\infty \le \|p-q\|_\infty$ for all diagonal pairs of points p,q (i.e. such that $|p_i - q_i| = \|p-q\|_\infty$ for all $i \in [k]$), then f is non-expansive.

Lemma 10. *The function $f_{\delta,s}$ is non-expansive.*

Proof. The function $f_{\delta,s}$ was defined according to which region of the domain D a point lies in. There are three regions: the part of D' left of l_1, the middle region D_0, and the part of D' right of l_2. It suffices to check the non-expansiveness for diagonal pairs of points p,q that lie in the same region. If p,q are both in the region left of l_1, or if they are both right of l_2, then from the definition we have $\|f(p)-f(q)\|_\infty = \|p-q\|_\infty$.

So suppose p,q are both in D_0. Assume first that the line pq has angle $45°$, i.e. pq is parallel to the line l_0. Then $\|p-q\|_\infty = |pq|/\sqrt{2}$. Let p', q' be the projections of p,q on l_0, and let $p" = f_{\delta,s}(p)$, $q" = f_{\delta,s}(q)$. Then $|p'p"| = (\delta - |pp'|)\cdot|p's|$, $|q'q"| = (\delta - |qq'|)\cdot|q's|$. Since pq is parallel to l_0, $(\delta - |pp'|) = (\delta - |qq'|)$ and $|pq| = |p'q'|$. Assume without loss of generality that $|p's| > |q's|$. Then $|p"q"| = |p'q'| - (\delta - |pp'|)(|p's| - |q's|) \le |p'q'| = |pq|$. Since $\|p-q\|_\infty = |pq|/\sqrt{2}$ and $\|f(p)-f(q)\|_\infty = |p"q"|/\sqrt{2}$, it follows that $\|f(p)-f(q)\|_\infty \le \|p-q\|_\infty$.

Assume now that the line pq has angle $-45°$, i.e., pq is perpendicular to l_0. Again $\|p-q\|_\infty = |pq|/\sqrt{2}$. Now p and q have the same projection $p' = q'$ on l_0. Let $p" = f_{\delta,s}(p)$, $q" = f_{\delta,s}(q)$. We have $|p'p"| = (\delta - |pp'|)\cdot|p's|$, and $|q'q"| = |p'q"| = (\delta - |qp'|)\cdot|p's|$. Therefore, $|p"q"| = |(|pp'| - |qp'|)|\cdot|p's|$. If p,q are on the same side of l_0 then $|(|pp'| - |qp'|)| = |pq|$. If p,q are on opposite sides of l_0 then $|(|pp'| - |qp'|)| < |pq|$. In either case, we have $|p"q"| \le |pq|\cdot|p's| < |pq|$, since $|p's| \le |st| = 1/\sqrt{2}$. Again, since $\|p-q\|_\infty = |pq|/\sqrt{2}$ and $\|f(p)-f(q)\|_\infty = |p"q"|/\sqrt{2}$, it follows that $\|f(p)-f(q)\|_\infty \le \|p-q\|_\infty$. ☐

We are ready now to prove the theorem. Intuitively, if the given function is $f_{\delta,s}$ for some s on the NE or SW side of D and some small δ, then for an algorithm (deterministic or randomized) to find a point that is within L_∞ distance $1/4$ of s, it must ask a query within the central region D_0 around s, because otherwise it cannot know whether the fixed point s is on the NE or the SW side of D_0.

Proof of Theorem 2. Recall that binary search is an optimal algorithm for searching for an unknown item in a sorted array A, both among deterministic and randomized algorithms. If the array has size N, then any randomized comparison-based algorithm requires expected time at least $\log N - 1$ to look up an item in the array whose location is not known.

Suppose there is a (randomized) algorithm B that computes a point that is within $1/4$ of a fixed point of a non-expansive function f over the domain D within a finite expected number n of queries (the expectation is over the random choices of the algorithm). We will show how to solve faster the array search problem. Partition the diamond D into $N = 2^{2n}$ strips by drawing $N-1$ parallel lines at $45°$, spaced at distance $1/(N\sqrt{2})$ from each other, between the NW and SE side of D. Let $S_1, \dots, S_N$ be the N strips. Fix a $\delta < 1/(N2\sqrt{2})$. For each $x \in [N]$, let s_x be the point on the SW side of S_x at Euclidean distance δ from the S vertex of S_x, and t_x the point on the NE side of S_x at Euclidean distance δ from the N vertex. Note that $\|s_x - t_x\|_\infty > 1/2$. Let $\mathcal{F}$ be the family of non-expansive functions $\{f_{\delta,s_x}, f_{\delta,t_x} \,|x \in [N]\}$.

Consider the execution of algorithm B for a function $f \in \mathcal{F}$. Note that the central region D_0 for the functions f_{δ,s_x} and f_{δ,t_x} is contained in the strip S_x. If B queries a point p in another strip S_j, the answer $f(p)$ only conveys the information whether $j < x$ or $j > x$. If some execution of B returns a point q without ever having queried any point in S_x, then all the answers in the execution are consistent with both f_{δ,s_x} and f_{δ,t_x}. Since $\|s_x - t_x\|_\infty > 1/2$, either $\|q - s_x\|_\infty > 1/4$

or $\|q - t_x\|_\infty > 1/4$. Therefore, a correct algorithm B cannot terminate before querying a point in the strip S_x that contains the fixed point of the function.

We can map now the algorithm B to an algorithm B' for the problem of searching for an item in a sorted array A of size N. A choice of an index x in the array A corresponds to a choice of the strip S_x that contains the fixed point of the function $f \in \mathcal{F}$, i.e. choosing one of f_{δ,s_x}, f_{δ,t_x}. Since B terminates in expected number n of queries, it asks within n steps a query within the strip S_x of the fixed point, hence the expected time of the algorithm B' is at most $n = \log N/2$, a contradiction. □

7 Promise Problem versus Total Search Version

The problem $\text{Contraction}_\infty(\varepsilon, \gamma, k)$ is a *promise problem*, where we want to compute an ε-fixed point of a given function f with promise that f is a $(1-\gamma)$-contraction. For a promise problem, one can define its total search version by asking to find a desired solution as in the promise problem, or a short violation certificate indicating that the given function doesn't satisfy the promise. The total search version of $\text{Contraction}_\infty(\varepsilon, \gamma, k)$, denoted $\text{T-Contraction}_\infty(\varepsilon, \gamma, k)$ is naturally defined as the following search problem.

Definition 4 (Total search version $\text{T-Contraction}_\infty(\varepsilon, \gamma, k)$)**.** *Given a function $f : [0,1]^k \mapsto [0,1]^k$, find one of the following:*

- *a point $x \in [0,1]^k$ such that $\|f(x) - x\|_\infty \le \varepsilon$;*
- *two points $x, y \in [0,1]^k$ such that $\|f(x) - f(y)\|_\infty > (1-\gamma)\|x - y\|_\infty$.*

In the black-box setting, the function f is given by an oracle access.

Our theorem in this section shows $\text{T-Contraction}_\infty(\varepsilon, \gamma, k)$ admits the same query bounds as $\text{Contraction}_\infty(\varepsilon, \gamma, k)$.

Theorem 3. *There is an $O(k^2 \log(1/\varepsilon))$-query algorithm for* $\text{T-Contraction}_\infty(\varepsilon, \gamma, k)$.

Theorem 3 follows from Lemma 11 below.

Lemma 11. *Let $\{q^1, \cdots, q^m\}$ be a set of points in $[0,1]^k$ and $\{a^1, \cdots, a^m\}$ be the corresponding answers from the black-box oracle. There is a $(1-\gamma)$-contraction f that is consistent with all the answers if and only if there is no pair t_1, t_2 such that $\|a^{t_1} - a^{t_2}\|_\infty > (1-\gamma)\|q^{t_1} - q^{t_2}\|_\infty$.*

Proof. If there is some pair t_1, t_2 such that $\|a^{t_1} - a^{t_2}\|_\infty > (1-\gamma)\|q^{t_1} - q^{t_2}\|_\infty$, then obviously there is no $(1-\gamma)$ contraction that is consistent with the answers.

Now suppose that no such pair exists. We define a function $f : [0,1]^k \mapsto [0,1]^k$ as follows: For every point $x \in [0,1]^k$ and coordinate $i \in [k]$, we let $f(x)_i = \min_{t \in [m]} \left\{(1-\gamma)\|x - q^t\|_\infty + a_i^t\right\}$; if the minimal value of this set is larger than 1, then we set $f(x)_i = 1$.

We show first that f is consistent with all the query answers, i.e. $f(q^j) = a^j$ for all $j \in [m]$. Since the query points satisfy the contraction property, we have $\|a^j - a^t\|_\infty \le (1-\gamma)\|q^j - q^t\|_\infty$ for all $t \neq j$. Therefore, for every coordinate $i \in [k]$, $a_i^j \le (1-\gamma)\|q^j - q^t\|_\infty + a_i^t$. Hence, $f(q^j)_i = \min_{t \in [m]} \left\{(1-\gamma)\|q^j - q^t\|_\infty + a_i^t\right\} = a_i^j$. Thus, $f(q^j) = a^j$.

We show now that the function f constructed above is a $(1-\gamma)$-contraction. Consider any two points $x, y \in [0,1]^k$ and a coordinate $i \in [k]$. Suppose without loss of generality that $f(y)_i \le f(x)_i$.

If $f(y)_i = 1$, then also $f(x)_i = 1$ and $|f(x)_i - f(y)_i| = 0 \leq (1-\gamma)\|x-y\|_\infty$. So suppose $f(y)_i = (1-\gamma)\|y-q^t\|_\infty + a_i^t$ for some $t \in [m]$. By the triangle inequality, $\|x-q^t\|_\infty \leq \|x-y\|_\infty + \|y-q^t\|_\infty$. Hence $f(x)_i \leq (1-\gamma)\|x-q^t\|_\infty + a_i^t \leq (1-\gamma)(\|x-y\|_\infty + \|y-q^t\|_\infty) + a_i^t = (1-\gamma)\|x-y\|_\infty + f(y)_i$. Therefore, $0 \leq f(x)_i - f(y)_i \leq (1-\gamma)\|x-y\|_\infty$. Thus, $\|f(x)-f(y)\|_\infty \leq (1-\gamma)\|x-y\|_\infty$. $\square$

It follows from Lemma 11 that we can use any algorithm that can solve the promise problem Contraction$_\infty(\varepsilon, \gamma, k)$ to solve the total search version T-Contraction$_\infty(\varepsilon, \gamma, k)$ within the same number of queries: If all pairs among the queries generated satisfy the contraction property, then there is a contraction that is consistent with all the queries, hence the algorithm will find an approximate fixed point within the same number of queries as in the promise version. If on the other hand there is a pair of queries that violate the contraction property, then the algorithm can return the pair and terminate. Theorem 3 follows.

8 Conclusions

We gave an algorithm for finding an ε-fixed point of a contraction (or non-expansive) map $f : [0,1]^k \mapsto [0,1]^k$ under the ℓ_∞ norm in polynomial query complexity. Contraction maps under the ℓ_∞ norm are especially important because several longstanding open problems from various fields can be cast in this framework. The main open question is whether our algorithm can be implemented to run also with polynomial time complexity, or alternatively if there is another general-purpose (black-box) algorithm for contraction maps that finds an approximate fixed point in polynomial time. Resolving positively this question would have tremendous implications.

Another natural open question is whether similar polynomial query bounds can be obtained for contraction maps under the ℓ_1-norm or norms ℓ_p with $p > 2$. Although ℓ_∞ seems to arise more in applications, understanding better contractions under other norms would also be useful.

Acknowledgements

We thank STOC anonymous reviewers for their feedbacks on improving the presentation.

References

[ABB20] James Aisenberg, Maria Luisa Bonet, and Sam Buss. 2-d Tucker is PPA-complete. *Journal of Computer and System Sciences*, 108:92–103, 2020. 1

[Ban22] Stefan Banach. Sur les opérations dans les ensembles abstraits et leur application aux équations intégrales. *Fundamenta mathematicae*, 3(1):133–181, 1922. 1

[Bel57] Richard Bellman. A Markovian decision process. *Journal of mathematics and mechanics*, pages 679–684, 1957. 1

[CD08] Xi Chen and Xiaotie Deng. Matching algorithmic bounds for finding a brouwer fixed point. *Journal of the ACM (JACM)*, 55(3):1–26, 2008. 1

[CD09] Xi Chen and Xiaotie Deng. On the complexity of 2D discrete fixed point problem. *Theoretical Computer Science*, 410(44):4448–4456, 2009. 1

[CJK+22] Cristian S. Calude, Sanjay Jain, Bakhadyr Khoussainov, Wei Li, and Frank Stephan. Deciding parity games in quasi-polynomial time. *SIAM J. Comput.*, 51(2):17–152, 2022. 1

[CL55] E. A. Coddington and N. Levinson. *Theory of Ordinary Differential Equations*. McGraw Hill, 1955. 1

[Con92] Anne Condon. The complexity of stochastic games. *Information and Computation*, 96(2):203–224, 1992. 1

[DEF+21] Xiaotie Deng, Jack R Edmonds, Zhe Feng, Zhengyang Liu, Qi Qi, and Zeying Xu. Understanding PPA-completeness. *Journal of Computer and System Sciences*, 115:146–168, 2021. 1

[Den67] Eic V. Denardo. Contraction mappings underlying dynamic programming. *SIAM Review*, 9(2):165–177, 1967. 1

[DP11] Constantinos Daskalakis and Christos H. Papadimitriou. Continuous local search. In Dana Randall, editor, *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2011, San Francisco, California, USA, January 23-25, 2011*, pages 790–804. SIAM, 2011. 1, 1

[DTZ18] Constantinos Daskalakis, Christos Tzamos, and Manolis Zampetakis. A converse to Banach's fixed point theorem and its CLS-completeness. In Ilias Diakonikolas, David Kempe, and Monika Henzinger, editors, *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Los Angeles, CA, USA, June 25-29, 2018*, pages 44–50. ACM, 2018. 1, 1

[EJ91] E. A. Emerson and C. Jutla. Tree automata, μ-calculus and determinacy. In *Proceedings IEEE Symp. on Foundations of Computer Science*, pages 368–377, 1991. 1

[EY10] Kousha Etessami and Mihalis Yannakakis. On the complexity of Nash equilibria and other fixed points. *SIAM J. Comput.*, 39(6):2531–2597, 2010. 1

[FGHS23] John Fearnley, Paul Goldberg, Alexandros Hollender, and Rahul Savani. The complexity of gradient descent: CLS = PPAD $\cap$ PLS. *J. ACM*, 70(1):7:1–7:74, 2023. 1, 1

[FGMS20] John Fearnley, Spencer Gordon, Ruta Mehta, and Rahul Savani. Unique end of potential line. *J. Comput. Syst. Sci.*, 114:1–35, 2020. 1, 1

[Gün89] Matthias Günther. Zum einbettungssatz von J. Nash. *Mathematische Nachrichten*, 144(1):165–187, 1989. 1

[HKS99] Z. Huang, Leonid G. Khachiyan, and Christopher (Krzysztof) Sikorski. Approximating fixed points of weakly contracting mappings. *J. Complex.*, 15(2):200–213, 1999. 1

[Hol21] Alexandros Hollender. *Structural results for total search complexity classes with applications to game theory and optimisation*. PhD thesis, University of Oxford, UK, 2021. 1

[How60] Ronald A Howard. *Dynamic programming and Markov processes*. John Wiley, 1960. 1

[HPV89] Michael D. Hirsch, Christos H. Papadimitriou, and Stephen A. Vavasis. Exponential lower bounds for finding Brouwer fixpoints. *J. Complex.*, 5(4):379–416, 1989. 1

[JPY88] David S. Johnson, Christos H. Papadimitriou, and Mihalis Yannakakis. How easy is local search? *J. Comput. Syst. Sci.*, 37(1):79–100, 1988. 1

[Lit94] Michael L Littman. Markov games as a framework for multi-agent reinforcement learning. In *Machine learning proceedings 1994*, pages 157–163. Elsevier, 1994. 1

[Nas56] John Nash. The imbedding problem for Riemannian manifolds. *Annals of mathematics*, 63(1):20–63, 1956. 1

[Pap94] Christos H Papadimitriou. On the complexity of the parity argument and other inefficient proofs of existence. *Journal of Computer and system Sciences*, 48(3):498–532, 1994. 1

[Sha53] L. Shapley. Stochastic games. *Proc. Nat. Acad. Sci.*, 39(10):1095–1100, 1953. 1

[SS02] Spencer D. Shellman and Kris Sikorski. A two-dimensional bisection envelope algorithm for fixed points. *J. Complex.*, 18(2):641–659, 2002. 1

[SS03] Spencer D. Shellman and Christopher (Krzysztof) Sikorski. A recursive algorithm for the infinity-norm fixed point problem. *J. Complex.*, 19(6):799–834, 2003. 1

[STW93] Christopher (Krzysztof) Sikorski, Chey-Woei Tsay, and Henryk Wozniakowski. An ellipsoid algorithm for the computation of fixed points. *J. Complex.*, 9(1):181–200, 1993. 1

[ZP96] Uri Zwick and Mike Paterson. The complexity of mean payoff games on graphs. *Theor. Comput. Sci.*, 158(1&2):343–359, 1996. 1

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il