The Complexity of the Homotopy Method, Equilibrium Selection, and Lemke-Howson Solutions

Paul W. Goldberg
Dept. of Computer Science
University of Liverpool,
Liverpool L69 3BX, UK.
Email: goldberg@liv.ac.uk

Christos H. Papadimitriou
Computer Science Division, Soda Hall,
University of California at Berkeley,
Berkeley, CA 94720, USA.
Email: christos@cs.berkeley.edu

Rahul Savani
Dept. of Computer Science
University of Liverpool,
Liverpool L69 3BX, UK.
Email: rsjs@liv.ac.uk

Abstract— We show that the widely used homotopy method for solving fixpoint problems, as well as the Harsanyi-Selten equilibrium selection process for games, are PSPACE-complete to implement. Extending our result for the Harsanyi-Selten process, we show that several other homotopy-based algorithms for finding equilibria of games are also PSPACE-complete to implement. A further application of our techniques yields the result that it is PSPACE-complete to compute any of the equilibria that could be found via the classical Lemke-Howson algorithm, a complexity-theoretic strengthening of the result in [24]. These results show that our techniques can be widely applied and suggest that the PSPACE-completeness of implementing homotopy methods is a general principle.

Keywords-game theory, computational complexity

1. Introduction

According to Roger Myerson [20], the 1950 publication of Nash's paper on equilibria was a watershed event not just for Game Theory, but for Economics in general. The new general equilibrium concept, and its established universality, was an impetus for understanding rationality in much more general economic contexts, and inspired the important price equilibrium results by Arrow and Debreu. Myerson argues convincingly in [20] that the concept of Nash equilibrium lies at the foundations of modern economic thought.

Seen from an algorithmic perspective, however, the Nash equilibrium suffers from two important problems: First, it is not clear how to find it efficiently (the same is true for the Arrow-Debreu variety for markets and prices). This shortcoming had already been identified by economists since the 1950s, and much effort has been devoted to algorithms for finding Nash equilibria, see [25, 19, 14] for examples from a very extensive literature. None of these algorithms came with polynomial-time guarantees, however, and the recent result [6, 4] establishing that the problem is PPAD-complete explains why. Of the many algorithmic approaches proposed by economists over the past 50 years for finding Nash equilibria, most have been shown by now to require exponential time in the worst case [24, 16]. One exception is an important algorithmic genre known as *homotopy methods* [7]; see [15] for a recent survey.

In topology, a *homotopy* is a continuous transformation from one function to another (as, for example, between two paths joining two points on a map). The homotopy method starts with a fixpoint problem that is easy to solve (say, a rotation of a disc around its center), and continuously transforms it into the problem in hand, by "pivoting" to new fixpoints along the way. A theorem by Browder [2] establishes the validity of this method in the limit, by showing the existence of a continuous path of fixpoints that joins two fixpoints of the initial and the final problems.

The second algorithmic obstacle for the Nash equilibrium concept is *multiplicity.* Games have multiple equilibria, and markets many price equilibria, and thus the corresponding equilibrium concepts are only *nondeterministic predictions* (oxymoron intended). In price equilibria, this multiplicity has been blamed for *economic crises:* The path guaranteed by Browder's theorem is non-monotonic, going back and forth in time. As a result, equilibria vanish at its folds, leaving the market in turmoil [1]. In games, a proposed remedy for multiplicity is the so-called focal point theory see, e.g., [18] p. 414, postulating that players implicitly coordinate their equilibrium choice by focusing on the most obvious, or mutually advantageous, equilibrium; repeated play and learning (see, e.g., [9]) can also be considered a remedy for multiplicity. In 1975, Harsanyi proposed the *tracing procedure* [10] for battling equilibrium multiplicity, a theory further explicated in his joint 1988 book "A General Theory of Equilibrium Selection in Games" with Selten [11] (Harsanyi and Selten shared in 1994 the Nobel prize with Nash). The tracing procedure asserts that players engaged in a game $\mathcal{G}$ play at first a simple game $\mathcal{G}_0$, in which their prior beliefs about the other players' behavior result in a dominant strategy. As time t progresses, and their priors are falsified by life, they play a more and more realistic game $\mathcal{G}_t = (1-t)\cdot \mathcal{G}_0 + t\cdot \mathcal{G}$, until, at time $t = 1$, they end up playing the intended game $\mathcal{G}$. They show that, for almost all games, tracing the equilibrium path of this process results in a unique equilibrium. Notice the parallel with the homotopy method; apparently the two were discovered independently.

Our results: This paper is a complexity-theoretic critique of the tracing procedure and the homotopy method: we show that finding the solutions they prescribe requires the power of PSPACE. In particular, finding the Brouwer fixpoint that would have been discovered by the homotopy method, for a simple starting function and an adversarial final one, is PSPACE-complete. The same is true, via standard reductions, for price equilibria. We also construct examples where the homotopy method not only will undergo an exponential number of pivots (this was expected since [16]), but will suffer an exponential number of *direction reversals.* As for the tracing procedure, we show that it is PSPACE-complete to find the Nash equilibrium selected by it, even in two-player games, and even if the initial game has dominant strategies obtained from priors, exactly as prescribed by Harsanyi and Selten. We extend this result to homotopy-based algorithms where the starting game depends on the final game and show that it is PSPACE-complete to implement the Herings-van den Elzen, Herings-Peeters, and van den Elzen-Talman algorithms for finding equilibria in games. Finally, it is particularly noteworthy that PSPACE-completeness prevails even for finding the solutions that would be returned by the classical Lemke-Howson algorithm, a simplex-like method that had long been considered an oasis of conceptual simplicity and (until [24]) of algorithmic hope in this field. This reinforces the "exponentially long paths" result of [24] with a new result which says that, subject only to the hardness of PSPACE, no short cuts to Lemke-Howson solutions are possible (for any of the different initial choices of the algorithm). Since it is known that the Lemke-Howson algorithm can be expressed as a homotopy [15], this result can also be seen as a powerful specialization of our first result.

The algorithms we consider solve problems in the complexity class PPAD, which is contained in TFNP, the class of all total function problems in NP. Another prominent complexity class contained in TFNP is PLS (for polynomial local search). Many common problems in PLS (e.g., local max cut and finding pure equilibria of congestion games) are complete under a so-called *tight* PLS-reduction, implying that the corresponding standard local search algorithm is exponential (for certain starting configurations and *any* choices of the local search algorithm). Furthermore, one can conclude that the computational problem of finding a local optimum reachable from a given starting configuration by local search is PSPACE-complete.

No such concept of tight reductions is known for PPAD, and our results can be seen as addressing this deficiency. Specifically, we show the PSPACE-completeness (and exponential worst-case behaviour) of a number of homotopy-based algorithms for finding equilibria. Our reductions start with the problem OTHER END OF THIS LINE (OEOTL), which is related to the problem END OF THE LINE used in the definition is PPAD, seeking not just any end of a path, but the other end of the particular path starting at the origin. OEOTL was known to be PSPACE-complete since [23], but this fact has so far remained unexploited for proving lower bounds for other problems.

Outline of the paper: In Section 2.1, we give an overview of the linear homotopy method as applied to Brouwer functions and games. In Section 2.2, we recall the PSPACE-complete problem OEOTL (OTHER END OF THIS LINE), which serves as the starting point for all our main reductions. In Section 3, we show that the linear homotopy method to compute a Brouwer fixpoint is PSPACE-complete, which is proved in Section 3.3. In Section 4, we establish the PSPACE-completeness of the linear tracing procedure for two-player strategic form games for a special starting game that is independent of the final game. These results are extended to starting games that depend on the final game in Section 5, where we show that it is PSPACE-complete to implement the Herings-van den Elzen, Herings-Peeters, and van den Elzen-Talman algorithms for computing equilibria of games. The techniques of [4, 6] are central to both Section 3 and Section 4 and are recalled and extended along the way. Finally, in Section 6, we show that it is PSPACE-complete to find any solution of a two-player game by the Lemke-Howson algorithm.

2. PRELIMINARIES

2.1. Homotopies

A *Brouwer function* $\mathcal{F}$ is a continuous function from a convex and compact domain D to itself; by Brouwer's fixpoint theorem there exists $x \in D$ such that $\mathcal{F}(x) = x$. A *homotopy* between two functions $\mathcal{F}_0 : X \longrightarrow Y$ and $\mathcal{F}_1 : X \longrightarrow Y$ (where X and Y are topological spaces) is a continuous function $H : [0,1] \times X \longrightarrow Y$ such that for all $x \in X$, $H(0,x) = \mathcal{F}_0(x)$ and $H(1,x) = \mathcal{F}_1(x)$. In this paper, we are interested in the special case where $X = Y = D$, for D a closed compact subset of Euclidean space, such as a cube. Thus, $\mathcal{F}_0$ and $\mathcal{F}_1$ are Brouwer functions on D. Given two continuous functions $\mathcal{F}_0, \mathcal{F}_1 : D \longrightarrow D$, the *linear homotopy* is given by the expression $H(t,x) = (1-t)\mathcal{F}_0(x) + t\mathcal{F}_1(x)$, and (if D is convex) results in a continuum of Brouwer functions $\mathcal{F}_t : D \longrightarrow D$ given by $\mathcal{F}_t = (1-t)\cdot \mathcal{F}_0 + t\cdot \mathcal{F}_1$ for $t \in [0,1]$.

Browder's fixpoint theorem [2] (not to be confused with Brouwer's fixpoint theorem) asserts that given a homotopy connecting $\mathcal{F}_0$ and $\mathcal{F}_1$, there is a path in $[0,1] \times D$ from some fixpoint of $\mathcal{F}_0$ to some fixpoint of $\mathcal{F}_1$, such that for every point (t,x) on that path, x is a fixpoint of $\mathcal{F}_t$. The homotopy method [7, 15] for finding a fixpoint of $\mathcal{F}_1$ selects $\mathcal{F}_0$ to have a unique and easy to find fixpoint, and essentially follows such a path. As noted in [15], we do not expect the path to be monotonic in t — indeed, we show in the full paper that an exponential number of direction reversals is possible.

We are often interested in *approximate* fixpoints[1]. If $\mathcal{F}$ is a Brouwer function, an ϵ-approximate fixpoint is a point x such that $|\mathcal{F}(x) - x| \leq \epsilon$ (we shall use the L_∞ metric throughout). It follows from Browder's theorem that, for any $\mathcal{F}_0, \mathcal{F}_1$, there is a finite sequence $x_0, x_{t_1}, \ldots, x_{t_k}, x_1$ of ϵ-approximate fixpoints of $\mathcal{F}_0, \mathcal{F}_{t_1}, \ldots, \mathcal{F}_{t_k}, \mathcal{F}_1$, for some k and $t_1, \ldots, t_k$, such

[1]A very interesting alternative consideration [8] focuses on exact fixpoints, resulting in higher complexity of the search problem; here we could also consider exact fixpoints and equilibria without much effect on our results, since we are dealing with PSPACE-completeness. It is known from [8] that this harder problem belongs to PSPACE.

that any two consecutive fixpoints in the sequence are at most ϵ apart.

We shall be interested in the following problem, which we call BROWDER FIXPOINT: Given two arithmetic circuits computing two functions $\mathcal{F}_0$ and $\mathcal{F}_1$ from $[0,1]^d$ to itself with Lipschitz constant ℓ, an $\epsilon > 0$, where $\mathcal{F}_0$ has a unique fixpoint x_0, find an ϵ-approximate fixpoint x_1 of $\mathcal{F}_1$ that is connected via a sequence of ϵ-approximate fixpoints to x_0. (To make this definition precise, we of course have to identify classes of functions from which $\mathcal{F}_0$ and $\mathcal{F}_1$ may be drawn.) Notice that the homotopy method for computing Brouwer fixpoints provides a solution to this problem.

Homotopies can be defined very similarly also for games. Given two games $\mathcal{G}_0, \mathcal{G}_1$ of the same type (number of players and strategies), we consider $\mathcal{G}_t = (1-t)\cdot \mathcal{G}_0 + t\cdot \mathcal{G}_1$, where it is the players' utilities that are interpolated. It is routine to extend this definition to more general classes of games, such as graphical games [17] (in which case, in addition to the players and strategies, the two graphs must be the same). Browder's theorem, via Nash's reduction, establishes that there is a path of approximate Nash equilibria here as well. The problem LINEAR TRACING is the following: Given two games $\mathcal{G}_0$ and $\mathcal{G}_1$, an $\epsilon > 0$, and a Nash equilibrium x_0 of $\mathcal{G}_0$, find an ϵ-approximate Nash equilibrium x_1 of $\mathcal{G}_1$ that is connected via a sequence of ϵ-approximate Nash equilibria to x_0.

It is easy to see that LINEAR TRACING is in PSPACE, and it can be checked that the algorithm of Herings and van den Elzen [13] achieves this. BROWDER FIXPOINT is also in PSPACE.

2.2. OTHER END OF THIS LINE

We consider directed graphs on 2^n vertices represented as n-bit vectors. The arcs are represented by two polynomial-size circuits S and P, each having n inputs and outputs, as follows. There is an arc from vertex v to w provided that $S(v) = w$ and $P(w) = v$. Notice that all vertices of the graph have both indegree and outdegree 0 or 1, that is, the graph consists of paths, cycles, and isolated vertices.

Definition 1: An (S,P)-graph with parameter n is a graph on $\{0,1\}^n$ specified by circuits S and P, as described above, subject to the constraint that vertex 0^n has no incoming arc but does have an outgoing arc.

The problem END OF THE LINE is the problem of finding a vertex of a given (S,P)-graph other than 0^n which has at most one incident arc. Note that this problem is in the class TFNP of total search problems in NP: there exists a solution that could be obtained by following the directed path that starts at 0^n, and any given solution may be efficiently checked for correctness. The class PPAD [23] is defined as all search problems polynomial-time reducible to END OF THE LINE. The problem OTHER END OF THIS LINE (which we will subsequently abbreviate to OEOTL) is the problem of finding the end of *the particular path* that starts at 0^n. In contrast with END OF THE LINE, a given solution to an instance of OEOTL has no obvious concise certificate that it is the correct endpoint, so while OEOTL is a total search problem, it is apparently not an NP total search problem. In fact, we have the following (Theorem 2 of [23]), which is the starting-point of our reductions.

Theorem 1: [23] OEOTL is PSPACE-complete.

We build on the ideas of [6, 4] (showing that unrestricted Nash equilibria can efficiently encode END OF THE LINE solutions) to show how equilibria defined by homotopies can efficiently encode OEOTL solutions.

3. THE HOMOTOPY METHOD FOR BROUWER FIXPOINTS

In this section we give detailed definitions of classes of fixpoint and approximate fixpoint computation problems. In Section 3.1, we review the definition of Brouwer-mapping functions —and related concepts— from Chen et al. [4], here applied to a three dimensional domain. In Section 3.2, we review the techniques of [6, 4] for implementing Brouwer-mapping functions as arithmetic circuits. In Section 3.3, we prove Theorem 3, the main result of Section 3, in which we establish the PSPACE-completeness of a linear homotopy for finding a fixpoint of a Brouwer function. $n \in \mathbb{N}$ will denote a complexity parameter of problem instances. We define a sequence $\mathcal{F}_0^{(n)}$ of "basic Brouwer functions" having unique known fixpoints. For each n we define a class of Brouwer functions whose members encode (S,P)-graphs on $\{0,1\}^n$. The homotopy of Equation (1) defines a class of functions $\mathcal{F}_t$, $t \in [0,1]$, that interpolate between $\mathcal{F}_0$ and $\mathcal{F}_1$ and specifies a particular fixpoint of $\mathcal{F}_1$. We will show that from that fixpoint, we can efficiently recover a solution to OEOTL for the graph encoded by $\mathcal{F}_1$.

3.1. Definitions and notation

Notation 1: Let K be the unit 3-D cube $[0,1]^3$. For $n \in \mathbb{N}$ let $\mathcal{K}^{(n)}$ denote a partition of K into 2^{3n} "cubelets", $\mathcal{K}^{(n)} = \{K_{ijk} \ : \ 0 \leq i,j,k \leq 2^n - 1\}$; K_{ijk} is an axis-aligned cube of length 2^{-n} whose vertex closest to the origin has coordinates $2^{-n}(i,j,k)$.

We define a *Brouwer-mapping circuit* in a similar way to the definition in [4], here specialized to the case of 3 dimensions. We also introduce some variations of the definition, as follows:

Definition 2: A *Brouwer-mapping circuit (bmc)* is a directed boolean circuit with $3n$ input nodes and 2 output nodes. Note that any bmc B has an associated *Brouwer-mapping function (bmf)* $f_B : \mathcal{K}^{(n)} \longrightarrow \{0,1,2,3\}$ that maps any cubelet K_{ijk} to one of the four colors $\{0,1,2,3\}$. We require the colors of all exterior cubelets to be predetermined as follows. For $i = 0$, $f_B(K_{ijk}) = 1$. For $j = 0$, $i > 0$, $f_B(K_{ijk}) = 2$. For $k = 0$, $i,j > 0$, $f_B(K_{ijk}) = 3$. All other exterior cubelets are mapped to 0.

The *basic bmf* $f_0^{(n)} : \mathcal{K}^{(n)} \longrightarrow \{0,1,2,3\}$ has the additional property that all internal cubelets get mapped to 0. Notice that $f_0^{(n)}$ is computable by a bmc of size polynomial in n.

A *DGP-style bmf* is one that is derived from an (S, P)-graph in the manner of [6], and so is computable with a bmc of size polynomial in the size of circuits S and P. (Proposition 1 notes the relevant property of DGP-style bmf's.)

A *partial bmf* f is defined with respect to a set $\mathcal{S} \subseteq \mathcal{K}^{(n)}$; f assigns a color to elements of $\mathcal{S}$ but f may be undefined on non-elements of $\mathcal{S}$.

Proposition 1: The following problem is PSPACE-complete. Given a Brouwer-mapping circuit B, find a point in K that is a vertex of 4 cubelets mapped to all 4 colors by the associated bmf f_B, and which is connected to the origin via cubelets having colors other than 0.

This is a total search problem: the topological intuition is that there is a line that is adjacent to the colors $\{1, 2, 3\}$ and has one end at $2^{-n}(1, 1, 0)$. The other end must be inside K and adjacent to color 0, since no other exterior point is adjacent to the colors $\{1, 2, 3\}$.

The proof of Proposition 1 in the full paper, applies the reduction of [6] from END OF THE LINE to the version of the problem where a *panchromatic vertex* is sought that is adjacent to all colors. The resulting $\{1, 2, 3\}$-colored line has a structure that faithfully simulates the arcs of the (S, P)-graph from which is was derived. Panchromatic vertices correspond to END OF THE LINE solutions, and are linked-up with the $\{1, 2, 3\}$-colored line, whose structure corresponds to the END OF THE LINE graph, and whose orientation arises from the clockwise order of $\{1, 2, 3\}$ around it.

3.2. Implementing Brouwer-mapping functions as arithmetic circuits

We review a class of functions used to establish PPAD-completeness of graphical and strategic-form games. Recall that K denotes the 3-dimensional unit cube; we consider continuous functions $\mathcal{F} : K \longrightarrow K$ having the following structure. Each function is an arithmetic circuit composed of nodes, with each node taking inputs from up to 2 other nodes, and producing an output, for example, the sum of its inputs. All values are constrained to $[0, 1]$, so a node that adds its inputs would output 1 if their sum is greater than 1. Identify 3 nodes as "input nodes" and another 3 as "output nodes", so if $\mathcal{F}$ is a continuous function from K to K, it has a Brouwer fixpoint.

Definition 3: A *linear arithmetic circuit* is an arithmetic circuit that computes a function from K to K, represented by a directed graph whose nodes are "gates" that perform certain basic arithmetic operations on their inputs as follows. Each gate takes as input 0, 1 or 2 real values in $[0, 1]$ and outputs a single real value in $[0, 1]$, where the output of a gate may be the sum/difference/max/min of two inputs, or a constant multiple of a single input, or no input and constant output. (An output value is set to 1 if for example two inputs that sum to more than 1 are input to a "sum" gate.) We also allow "comparator gates" in which the output of such a gate evaluates to 1 (respectively, 0) if its first input is greater (respectively, less) than the second input, and may take any value if they are equal.

Notation 2: Let $\alpha = 2^{-2n}$. Let $\delta_1 = (\alpha, 0, 0)$, $\delta_2 = (0, \alpha, 0)$, $\delta_3 = (0, 0, \alpha)$, $\delta_0 = (-\alpha, -\alpha, -\alpha)$.

Definition 4: We shall say that a Brouwer-mapping function f is *implemented by an arithmetic circuit C* if whenever $f(K_{ijk}) = c$, then $C(x) - x = \delta_c$ when x is at the center of K_{ijk}. For x not at a center, $C(x) - x$ should be a convex combination of values of $C(z) - z$ for cubelet centers z within L_∞ distance 2^{-n} of x. Given $\mathcal{F} : K \longrightarrow K$ computed by such a C, we shall similarly say that $\mathcal{F}$ *implements* f.

Observation 1: If $\mathcal{F}$ implements f, then any fixpoints of $\mathcal{F}$ must lie within distance 2^{-n} of panchromatic vertices of f, and vice versa.

Theorem 2: A Brouwer-mapping function having complexity parameter n can be implemented using a linear arithmetic circuit having $poly(n)$ gates, that computes a continuous function.

The proof gives a new technique to implement any Brouwer-mapping function f as a *continuous* function $\mathcal{F}$ that uses a linear arithmetic circuit. This is in contrast with the corresponding techniques of [4, 6] that used a sampling-based approach in order to smooth the transition between distinct cubelets. The sampling-based approach results in discontinuous functions, where Browder's theorem would not be applicable (although it could still be applied to a continuous approximation). The technique only works in constant dimension; if can be extended to higher dimension using the "snake-embeddings" of [4].

Proof: Let $f : \mathcal{K}^{(n)} \longrightarrow \{0, 1, 2, 3\}$ be a Brouwer-mapping function. We construct a continuous Brouwer function $\mathcal{F} : K \longrightarrow K$ computed by a linear arithmetic circuit C as follows.

For x at the center of cubelet K_{ijk}, set $\mathcal{F}(x) - x = \delta_c$ where $c = f(K_{ijk})$. For x a vertex of cubelets $\mathcal{K}_x \subset \mathcal{K}^{(n)}$, set $\mathcal{F}(x) - x$ to be the average of $\mathcal{F}(z) - z$ for all points z at the centers of members of $\mathcal{K}_x$. The relevant points z can be obtained using a polynomial-sized piece of circuitry.

Let $\mathcal{S}$ be a simplicial decomposition of the unit cube consisting of 12 simplices that share a vertex at the center of the cube, and all other vertices are vertices of the cube. Let $\mathcal{S}_{ijk}$ be the simplicial decomposition of cubelet K_{ijk} obtained by scaling $\mathcal{S}$ down to K_{ijk}. Applied to all cubelets in $\mathcal{K}^{(n)}$ this results in a highly regular decomposition $\mathcal{S}^{(n)}$ of K into 12.2^{3n} simplices.

For any $x \in K$, $F(x)$ is obtained by linearly interpolating between the vertices of the simplex in $\mathcal{S}^{(n)}$ that contains x. Clearly $\mathcal{F}$ is continuous.

The result follows from the following claim:

Proposition 2: $\mathcal{F}$ as defined above, may be computed by a linear arithmetic circuit of size polynomial in n.

Proof: If x is not a vertex of $\mathcal{S}^{(n)}$, the circuit can determine the vertices of a simplex $S_x \in \mathcal{S}^{(n)}$ that contains x. There may be more than one such simplex, in which case *it does not matter which is chosen*.

The circuit has 12 cases to consider, depending on the orientation of S_x. Each case can be handled in the same general manner, by subtracting some vertex v of S_x from x, and multiplying $(x - v)$ by some constants (the coefficients of the linear function that interpolated between the vertices of S_x). Note that we never need to multiply two computed quantities together, multiplication only ever takes place between a computed quantity and a constant, as required for a linear arithmetic circuit. ■

■

3.3. *The* PSPACE *reduction to linear arithmetic circuits*

In this subsection, we establish the PSPACE-completeness of the problem BROWDER FIXPOINT, mentioned in the Introduction, which can now be made precise as follows. We use two bmfs f_0 and f_1, where f_0 is the *basic bmf* of Definition 2, and f_1 shall be a DGP-style bmf that encodes an instance of END OF THE LINE as constructed in [6]. Let $\mathcal{F}_0$ and $\mathcal{F}_1$ be implementations of f_0 and f_1 using linear arithmetic circuits as described in the proof of Theorem 2. For $\mathcal{F}: K \longrightarrow K$ let $\mathcal{F}^{(i)}$ denote the i-th component of $\mathcal{F}$. For $i = 1, 2, 3$ let

$$\begin{aligned} \bar{\mathcal{F}}_t^{(i)} &= (\mathcal{F}_0^{(i)} - t) + (\mathcal{F}_1^{(i)} - (1-t)) \\ \mathcal{F}_t^{(i)} &= \max(\min(\mathcal{F}_0^{(i)}, \mathcal{F}_1^{(i)}), \bar{\mathcal{F}}_t^{(i)}) \end{aligned} \tag{1}$$

where in (1), the outputs of operators $+$ and $-$ are restricted to lie in $[0,1]$ (so, rounding to 0 or 1 if needed). $\mathcal{F}_t$ interpolates continuously between $\mathcal{F}_0$ and $\mathcal{F}_1$ and is constructed from them using elements of the linear arithmetic circuits of Definition 3 (which is useful later; the natural alternative $\mathcal{F}_t = t\mathcal{F}_0 + (1-t)\mathcal{F}_1$ does not have this property.)

Observation 2: For all $t \in [0,1]$, $\mathcal{F}_t^{(i)}$ is Lipschitz continuous, with Lipschitz value $< 2.2^{-n}$.

$\mathcal{F}_0$ has a unique fixpoint close to $2^{-n}(1,1,1)$. $\mathcal{F}_0$ is a "basic Brouwer function" which forms the starting-point of homotopies we consider. Hence Observation 2 and Browder's fixpoint theorem implicitly define a corresponding fixpoint of $\mathcal{F}_1$.

Define an *approximate fixpoint* of $\mathcal{F}: K \longrightarrow K$ to be a point $x \in K$ with $|\mathcal{F}(x) - x| \leq \alpha/5$ (recall $\alpha = 2^{-2n}$).

Theorem 3: It is PSPACE-complete to find, within accuracy 2^{-n}, the coordinates of the fixpoint of $\mathcal{F}_1$ that corresponds to the homotopy of (1). It is also PSPACE-complete to find the coordinates of an approximate fixpoint of $\mathcal{F}_1$ that would be obtained by following a sequence of approximate fixpoints of $\mathcal{F}_t$ in which consecutive points are within distance α of each other.

Proof: We reduce from the problem defined in Proposition 1 as follows. Let B be a Brouwer-mapping circuit derived from OEOTL-instance (S,P) using Proposition 1 and let $f_B : \mathcal{K}^{(n)} \longrightarrow \{0,1,2,3\}$ be the function computed by B. Let $\mathcal{F}_1 : K \longrightarrow K$ be the function computed by a linear arithmetic circuit that implements f_B, and $\mathcal{F}_0$ be computed by a circuit that implements the basic bmf f_0 (where both implementations apply Theorem 2). $\mathcal{F}_t$ is given by (1).

Let P be a connected subset of $K \times [0,1]$ such that for any $(x,t) \in P$, x is a fixpoint of $\mathcal{F}_t$, and P contains $x_0 \in (K,0)$ and $x_1 \in (K,1)$. Browder's fixpoint theorem (with Observation 2) assures us that such a P exists. We claim that x_1 is within distance 2^{-n} of the unique solution to B of the problem specified in Proposition 1 (and hence, given x_1 we can easily construct this solution).

Suppose otherwise. For x_1 to be a fixpoint (even an approximate one) of $\mathcal{F}_1$, by Observation 1 it must be within distance 2^{-n} of a panchromatic vertex v of f_B. But now, v is not connected to the origin via non-zero cubelets of f_B. By connectivity of P, there must exist $(x,t) \in P$ such that x lies within a cubelet K_x where $f_B(K_x) = 0$.

We may assume further that x is at least 2^{-n} distant from any non-zero cubelet of f_B. This follows provided we assume that connected components of non-zero cubelets of f_B are separated from each other by a layer of 0-colored cubelets of thickness at least 3. This may be safely assumed by increasing n by a factor of 3 and subdividing the cubelets. We note that

1) each entry of vector $\mathcal{F}_0(x) - x$ is $< -\alpha/5$, and
2) each entry of $\mathcal{F}_1(x) - x$ is $< -\alpha/5$.

It follows that for $t \in [0,1]$, each entry of $f_t(x) - x$ is less than $-\alpha/5$, since coordinatewise, $f_0 \leq f_t \leq f_B$. That means that x cannot be an approximate fixpoint of any f_t, contradicting the assumption as required.

Since x is at least 2^{-n} distant from any non-zero cubelet of f_B, it is also at least 2^{-n} distant from any non-zero cubelet of f_0, since for any cubelet K_{ijk}, $f_B(K_{ijk}) = 0 \implies f_0(K_{ijk}) = 0$. The implementation of any bmf f as a function $\mathcal{F}$ computed by a linear arithmetic circuit, as referred to in Theorem 2, ensures that $\mathcal{F}(x) - x$ is a convex combination of vectors $\mathcal{F}(z) - z$ for cubelet centers z in the vicinity of x, and since all those cubelet centers are colored 0, we have that the entries of $\mathcal{F}(x)$ are all less than $-\alpha/5$, as required. ■

4. THE LINEAR TRACING PROCEDURE

We now turn to games and Nash equilibrium. Let $\mathcal{G}$ denote an $n \times n$ game that we wish to solve, assumed to be chosen by an adversary. $\mathcal{G}_0$ is a game with a unique "obvious" solution. In $\mathcal{G}_0$ each player receives payoff 1 for his first action, and payoff 0 for all others, regardless of what the other player does.

In the problem LINEAR TRACING the solution consists of the Nash equilibrium of $\mathcal{G}$ that is connected to the unique equilibrium (s_0^r, s_0^c) of $\mathcal{G}_0$ via equilibria of convex combinations $(1-t)\mathcal{G}_0 + t\mathcal{G}$. We can also define an approximate version of this problem, where instances include an additional parameter ϵ, and we seek an ϵ-Nash equilibrium that is connected to the solution of $\mathcal{G}_0$ via a sequence of ϵ-approximate solutions of $\mathcal{G}_t$. For the two-player case we assume $\epsilon = 0$. For more than 2 players, we need a positive ϵ to ensure that solutions can be written down as rational numbers.

Theorem 4: LINEAR TRACING is PSPACE-complete for 2-person games.

The same result then holds for strategic-form games with more than 2 players. It holds for a value of ϵ that is exponentially small; we could again use the ideas of [4] to obtain a version where ϵ is inverse polynomial.

4.1. *Brief overview of the proof ideas*

The following is a brief overview of the rest of this Section 4. Membership of PSPACE can be deduced from [13]. The reduction from the PSPACE-complete discrete Brouwer fixpoint problem of the previous section, applies the idea from [6] of going via *graphical games* to normal-form games. We derive a type of graphical game in which a specific player (denoted v_{switch}) acts as a switch, allowing the remaining players to simulate either the basic Brouwer-mapping function, or one associated with an instance of the search for a discrete Brouwer fixpoint. v_{switch} governs this behavior via his choice of either one of two alternative strategies, and we show that a *continuous* path of equilibria from one choice to the other, results in an equilibrium that ultimately represents a solution to OEOTL. The graphical game is then encoded as a 2-player game such that the linear-tracing procedure corresponds to this continuous path of equilibria in the graphical game.

4.2. *Graphical Games*

In a graphical game [17], each player is a vertex of a graph, and his payoffs depend on his own and his neighbors' actions. For a low-degree graph, this is one way that games having many players may be represented concisely. A homotopy between two graphical games $\mathcal{GG}_0$ and $\mathcal{GG}_1$ would require that these games have the same underlying graph, so that they differ only in their numerical payoffs. In the graphical games considered here, each player has just 2 actions and 3 neighbors. The main result of this section is

Proposition 3: Consider graphical games that contain a special player v_{switch} whose payoffs are constant (unaffected by his own actions or the other players'). The following problem is PSPACE-complete: find a Nash equilibrium of the game where v_{switch} plays 1, that is topologically connected to a Nash equilibrium in which v_{switch} plays 0, via a path of Nash equilibria in which v_{switch} plays mixed strategies.

Let $\mathcal{F}_0$ and $\mathcal{F}_1$ be functions computed by linear arithmetic circuits that implement Brouwer-mapping functions f_0 and f_1, where f_0 is the "basic bmf" of Definition 2, and f_1 is a DGP-style bmf that encodes some instance of END OF THE LINE.

Notation 3: In a graphical game in which all players have 2 pure strategies denoted 0 and 1, given a mixed-strategy profile for the players we let $\mathbf{p}[v]$ denote the probability that player v plays 1.

Definition 5: Given a bmf f, we construct an associated graphical game $\mathcal{GG}_f$ as follows. $\mathcal{GG}_f$ has 3 special players (v_x, v_y, v_z) whose strategies $(\mathbf{p}[v_x], \mathbf{p}[v_y], \mathbf{p}[v_z])$ represent a point in K. If f is implemented by $\mathcal{F} : K \longrightarrow K$ we use gadgets of [6] to simulate the nodes in the arithmetic circuit that computes $\mathcal{F}$ (each node of the circuit has an additional associated player in $\mathcal{GG}_f$). The game can pay them to adjust $(\mathbf{p}[v_x], \mathbf{p}[v_y], \mathbf{p}[v_z])$ in the direction $\mathcal{F}(\mathbf{p}[v_x], \mathbf{p}[v_y], \mathbf{p}[v_z]) - (\mathbf{p}[v_x], \mathbf{p}[v_y], \mathbf{p}[v_z])$. Then the players (v_x, v_y, v_z) are incentivized to play $\mathcal{F}(\mathbf{p}[v_x], \mathbf{p}[v_y], \mathbf{p}[v_z])$. Consequently a Nash equilibrium of $\mathcal{GG}_f$ corresponds to a fixpoint of $\mathcal{F}$. Moreover, an ϵ-Nash equilibrium corresponds to a $\mathrm{poly}(\epsilon)$-approximate fixpoint of $\mathcal{F}$. We call $\mathcal{GG}_f$ a *linear graphical game* since we only allow players whose payoffs cause them to simulate the gates of linear arithmetic circuits.

A game of the above kind is said to *simulate* f. We say further that a game $\mathcal{GG}$ simulates a partial bmf on a subset S of cubelets, if for any $K \in S$, when $(\mathbf{p}[v_x], \mathbf{p}[v_y], \mathbf{p}[v_z])$ lie at the center of K the players (v_x, v_y, v_z) are incentivized to play $(\mathbf{p}[v_x], \mathbf{p}[v_y], \mathbf{p}[v_z]) + \delta_c$, where $c = f(K)$.

Lemma 1: Given any linear graphical game $\mathcal{GG}_1$ that simulates a Brouwer-mapping function f_1, we can efficiently construct a new game $\mathcal{GG}^+$ having a player v_{switch} whose behavior can either cause $\mathcal{GG}$ to simulate f_1 (if v_{switch} plays 1) or cause $\mathcal{GG}$ to simulate f_0 if instead v_{switch} plays 0.

v_{switch} shall serve as a "switch", in allowing the game to switch between simulating f_0 and f_1 (using an additional 3 players (v_x^+, v_y^+, v_z^+) whose strategies represent a point in K) according to whether v_{switch} plays 0 or 1. Of course, v_{switch} has a key role in the associated two-player game.

Proof: For $i \in \{0,1\}$, let $\mathcal{GG}_i$ be a graphical game constructed from f_i according to Definitions 2, 4, 5. $\mathcal{GG}_i$ has 3 players/vertices whose mixed strategies, as represented by the probabilities that they play 1, represent a point in K. Denote these players (v_x^i, v_y^i, v_z^i).

Construct a "combined" game $\mathcal{GG}^+$ as follows. $\mathcal{GG}^+$ contains all the players in $\mathcal{GG}_0$ and $\mathcal{GG}_1$ together with a new player v_{switch}, where v_{switch} has the same fixed payoff for playing either 0 or 1. We add 3 players (v_x^+, v_y^+, v_z^+) whose mixed strategies represent a point in K, and players $(\bar{v}_x^+, \bar{v}_y^+, \bar{v}_z^+)$, whose behavior is governed by

$$\begin{aligned}\mathbf{p}[\bar{v}_x^+] &= (\mathbf{p}[v_x^0] - \mathbf{p}[v_{switch}]) + (\mathbf{p}[v_x^1] - (1 - \mathbf{p}[v_{switch}])) \\ \mathbf{p}[v_x^+] &= \max(\mathbf{p}[\bar{v}_x^+], \min(\mathbf{p}[v_x^0], \mathbf{p}[v_x^1]))\end{aligned} \quad (2)$$

(and similar expressions for v_y^+ and v_z^+) where the parentheses in the above expression are important since the outputs of the operators $+$ and $-$ are truncated to lie in $[0,1]$.

Players from $\mathcal{GG}_0$ and $\mathcal{GG}_1$ that take input from nodes v_i^0 or v_i^1 respectively, are then modified to take that input from v_i^+ instead. This completes the construction. ■

Proof: of Proposition 3: We reduce from the circuit homotopy of Theorem 3. Let $\{\mathcal{F}_t : t \in [0,1]\}$ be an instance of this circuit homotopy. Construct $\mathcal{G}_1$ from $\mathcal{F}_1$ as per Definition 5. Construct $\mathcal{GG}^+$ as in Lemma 1, and we make the following observation.

Observation 3: Suppose that in $\mathcal{GG}^+$ we have $\mathbf{p}[v_{switch}] = t \in (0,1)$. The resulting game $\mathcal{GG}_t^+$ simulates a partial Brouwer-mapping function f_t which is implemented by a Brouwer function $\mathcal{F}_t$ that is (pointwise) a convex combination of $\mathcal{F}_0$ and $\mathcal{F}_1$ and is defined on the subset of cubelets where $f_0 = f_1$. Given a homotopy path of Nash equilibria of $\mathcal{GG}^+$ that start at the unique equilibrium of $\mathcal{GG}^+$ that satisfies $\mathbf{p}[v_{switch}] = 0$ and ends at an equilibrium of $\mathcal{GG}^+$ in which $\mathbf{p}[v_{switch}] = 1$, there is a corresponding homotopy path from the fixpoint of $\mathcal{F}_0$ and a fixpoint of $\mathcal{F}_1$ (noting that (2) is essentially the same as (1)). That concludes the proof of Proposition 3. ■

The following version of Lemma 1 is useful in the construction for Lemke-Howson solutions, later on.

Corollary 1: Given any linear graphical game $\mathcal{GG}_1$ that simulates a Brouwer-mapping function f_1, we can efficiently construct a new game $\mathcal{GG}^+$ having 2 players v_{switch} and v'_{switch} whose behavior can either cause $\mathcal{GG}^+$ to simulate f_1 (if both v_{switch}, v'_{switch} play 1) or cause $\mathcal{GG}^+$ to simulate f_0 if instead either or both play 0.

4.3. *From graphical to two-player strategic-form games*

In this subsection we prove the following theorem, from which Theorem 4 follows.

Theorem 5: It is PSPACE-hard to compute the Nash equilibrium of a given 2-player normal-form game $\mathcal{G}_1$, that is obtained via the linear homotopy that starts from $\mathcal{G}_0$, a version of $\mathcal{G}_1$ where the payoffs have been changed to give each player payoff 1 for his first strategy and 0 for the others.

We reduce from the graphical game problem of Proposition 3. Let $\mathcal{GG}^+$ be a linear graphical game that includes a player v_{switch} as per Proposition 3. First, modify $\mathcal{GG}^+$ to give v_{switch} a small payment (say, 0.01) to play 1, and zero to play 0.

We define a homotopy between two-player strategic-form games $\mathcal{G}_0$ and $\mathcal{G}_1$ such that equilibria of $\mathcal{G}_1$ efficiently encode equilibria of $\mathcal{GG}^+$, and equilibria of $\mathcal{G}_t$ encode equilibria of versions of $\mathcal{GG}^+$ where v_{switch} has a bias towards playing 0. We use the reduction of [6] (Section 6.1) from graphical games to 2-player games.

Given a mixed-strategy profile, let $\Pr[s]$ denote the probability allocated to pure strategy s by its player.

Definition 6: A *circuit-encoding* 2-player game $\mathcal{G}$ has a corresponding graphical game $\mathcal{GG}$ where the graph of $\mathcal{GG}$ is bipartite; denote it $G = (V_1 \cup V_2, E)$; each player (vertex) in $\mathcal{GG}$ has 2 actions (denote them 0 and 1) and payoffs that depend on the behavior of 2 other players in the opposite side of G's bipartition. Each vertex/action pair (v, a) of $\mathcal{GG}$ has a corresponding strategy in $\mathcal{G}$; for $v \in V_1$, (v, a) belongs to the row player and for $v \in V_2$, (v, a) belongs to the column player. The payoffs in $\mathcal{G}$ are designed to ensure that in a Nash equilibrium of $\mathcal{G}$

- $\Pr[(v, 0)] + \Pr[(v, 1)] \geq 1/2n$ where n is the number of players in $\mathcal{GG}$
- if in $\mathcal{GG}$, v plays 1 with probability $\Pr[(v, 1)]/(\Pr[(v, 0)] + \Pr[(v, 1)])$ then we have a Nash equilibrium of $\mathcal{GG}$.

Let $\mathcal{G}$ be a circuit-encoding game derived from $\mathcal{GG}^+$ according to Definition 6. Associate v_{switch} with 2 strategies of of the column player of $\mathcal{G}$, and let s^c_k and s^c_{k+1} be these strategies. Hence a Nash equilibrium of $\mathcal{G}$ corresponds to one of $\mathcal{GG}^+$ where the value $\mathbf{p}[v_{switch}]$ is given by the value $\Pr[s^c_{k+1}]/(\Pr[s^c_k] + \Pr[s^c_{k+1}])$.

Observation 4: If we take a circuit-encoding 2-player game, and award one of the players a small bonus to play (v, a), then this corresponds to incentivizing the player v in $\mathcal{GG}$ to select strategy a. The corresponding incentive for v will be larger, but only polynomially larger.

Let $\mathcal{G}_0$ be a $(n+1) \times (n+1)$ game with strategies $\{s^r_0, \dots s^r_n\}$ for the row player, and $\{s^c_0, \dots s^c_n\}$ for the column player. Payoffs are as follows: each player receives 1 for playing s^r_0 or s^c_0, and 0 for s^r_j or s^c_j for $j > 0$.

Rescale the payoffs of $\mathcal{G}$ to all lie in the range $[0\cdot 9, 1\cdot 1]$. Let $\mathcal{G}_1$ be a $(n+1)\times(n+1)$ game with strategies $\{s^r_0, \dots, s^r_n\}$ for the row player, and $\{s^c_0, \dots, s^c_n\}$ for the column player. Payoffs are as follows:

- (s^r_0, s^c_0) results in payoffs $(0, -1)$ for the players.[2]
- (s^r_0, s^c_j) for $j > 0$ results in payoffs $(0, \frac{3}{4})$.
- (s^r_j, s^c_0) for $j > 0$ results in payoffs $(-1, \frac{3}{4})$ for $j \neq k$, and $(-1, \frac{3}{4} + \delta)$ (for δ inverse polynomial in n) for $j = k$
- The rest of $\mathcal{G}_1$ is a copy of $\mathcal{G}$ above.

Let $\mathcal{G}_t = (1-t)\mathcal{G}_0 + t\mathcal{G}_1$. The above payoffs have been chosen so that Nash equilibria satisfy: in $\mathcal{G}_1$, players do not use s^r_0 or s^c_0; in $\mathcal{G}_{0\cdot 6}$, players both have a proper mixture of s^r_0 and s^c_0 with their other strategies. Since $\mathcal{G}$'s payoffs were rescaled to lie in $[0\cdot 9, 1\cdot 1]$, $\Pr[s^r_0]$ and $\Pr[s^c_0]$ can be shown to lie in $[0\cdot 1, 0\cdot 9]$, which can be checked from the following payoff ranges for $\mathcal{G}_{0\cdot 6}$:

	s^c_0	$s^c_1 \dots s^c_n$
s^r_0	$(0\cdot 4, -0\cdot 2)$	$(0\cdot 4, 0\cdot 45 + \delta)$
$s^r_1 \dots s^r_n$	$(-0\cdot 6, 0\cdot 85)$	$([0\cdot 54, 0\cdot 66], [0\cdot 54, 0\cdot 66])$

Thus a continuous path of equilibria should at some stage allocate gradually less and less probability to s^r_0 and s^c_0 as t increases.

Observation 5: In any Nash equilibrium $\mathcal{N}$ of $\mathcal{G}_1$, the players assign probability 0 to s^r_0 and s^c_0, and consequently $\mathcal{N}$ consists of a Nash equilibrium of $\mathcal{G}$, restricting to strategies s^r_j, $s^c_{j'}$ for $j, j' > 0$.

Since $\mathcal{G}_t = (1-t)\mathcal{G}_0 + t\mathcal{G}_1$, we can write $\mathcal{G}_t$ as

	s^c_0	$s^c_1 \cdots s^c_{k-1}$	s^c_k	$s^c_{k+1} \cdots s^c_n$
s^r_0	$(1-t, 1-2t)$	$(1-t, \frac{3}{4}t)$	$(1-t, (\frac{3}{4}+\delta)t)$	$(1-t, \frac{3}{4}t)$
s^r_1	$(-t, 1-\frac{1}{4}t)$			
$\vdots$	$\vdots$		$t\mathcal{G}$	
s^r_n	$(-t, 1-\frac{1}{4}t)$			

The general idea is as follows. Consider the Browder path of equilibria that begins from the unique equilibrium of $\mathcal{G}_0$ (where initially both players play s^r_0, s^c_0). As t increases, the players will start to use the other strategies. At that stage, consider the distribution of their mixed strategies restricted to $s^r_1, \dots, s^r_n$ and $s^c_1, \dots, s^c_n$. These distributions will constitute a Nash equilibrium of a version of $\mathcal{G}$ in which the column player receives a small bonus for playing s^c_k. As t increases to 1, the bonus decreases continuously to 0, and we recover

[2] The two-component payoff vectors assign the first component to the row player and the second component to the column player.

Observation 5. Now, recall from Definition 6 that the way [6, 4] reduce graphical games to two-player games, is to associate each player v in the graphical game with two strategies in the two-player game, both belonging to the same player. The division of probability between those two strategies represents the probability that v plays 1. Consider v_{switch} now, corresponding to s^c_k and s^c_{k+1}. v_{switch} is, in the graphical game, mildly incentivized to play 1, but for $t < 1$ the δ in the two-player game $\mathcal{G}_t$ pushes it the other way, towards 0. As a result, a Nash equilibrium of $\mathcal{G}_t$ may simulate a Nash equilibrium of $\mathcal{GG}^+_t$ where $\mathbf{p}[v_{switch}] \in (0, 1)$. As t increases and the contribution from δ decreases, this process corresponds to raising $\mathbf{p}[v_{switch}]$ continuously (but not monotonically) from 0 to 1.

Lemma 2: Let $\mathcal{N}$ be a Nash equilibrium of $\mathcal{G}_t$ in which $\Pr[s^r_0] < 1$ and $\Pr[s^c_0] < 1$. Let $\mathcal{P}$ be the probability distributions over $\{s^r_1, \ldots, s^r_n\}$ and $\{s^c_1, \ldots, s^c_n\}$ obtained by taking each value $\Pr[s^i_j]$ (for $i \in \{r, c\}$, $1 \le j \le n$) and dividing it by $1 - \Pr[s^i_0]$.

Then $\mathcal{P}$ is a Nash equilibrium of a version of $\mathcal{G}$ where the column player receives a bonus of $\delta \Pr[s^r_0]/(1 - \Pr[s^r_0])$ for s^c_k.

Proof: In Nash equilibrium $\mathcal{N}$, c's strategy s^c_0 contributes the same quantity to each one of r's strategies $s^r_1, \ldots, s^r_n$. So the values $\Pr[s^r_1], \ldots, \Pr[s^r_n]$ must form a best response to c's mixed strategy from $\mathcal{P}$.

The column player receives a bonus $\delta t \Pr[s^r_0]$ specific to s^c_k, arising from the possibility that row player plays 0. He also receives an additional $\frac{3}{4}t$ for all strategies s^c_j for $j > 0$, but that uniform bonus has no further effect on his preference amongst $s^c_1, \ldots, s^c_n$.

So in $\mathcal{N}$, $\Pr[s^c_1], \ldots, \Pr[s^c_n]$ is a best response to a mixture of $\mathcal{G}$ weighted by $1 - \Pr[s^r_0]$ and the probability $\Pr[s^r_0]$ of a bonus $\delta \Pr[s^r_0]$ for playing s^c_k. This is equivalent to a best response to a version of $\mathcal{G}$ with a bonus of $\delta \Pr[s^r_0]/(1 - \Pr[s^r_0])$ for playing s^c_k. ■

Consider the path of equilibria connecting equilibrium $\mathcal{N}_0$ of $\mathcal{G}_0$ to equilibrium $\mathcal{N}_1$ of $\mathcal{G}_1$. By Lemma 2 we can choose δ such that in any equilibrium of $\mathcal{G}_{0.5}$ we have $\Pr[s^c_{k+1}] = 0$. We also have that in any equilibrium of $\mathcal{G}_1$, $\Pr[s^c_k] = 0$. Consider the longest suffix of the path for which $t \ge 0.5$ for all games $\mathcal{G}_t$ that appear in that suffix. The corresponding equilibria assign weight strictly less than 1 to s^r_0 and s^c_0, so Lemma 2 may be used to recover corresponding equilibria of versions of $\mathcal{G}$ which in turn correspond to versions of $\mathcal{GG}^+$ in which initially, v_{switch} is incentivized to play 0, and finally, v_{switch} is incentivized to play 1.

5. From Linear Tracing to the homotopies of van den Elzen-Talman, Herings-van den Elzen, and Herings-Peeters

In the previous section, we showed the PSPACE-completeness of finding the Nash equilibrium of a two-player game that is associated with a homotopy that uses a specific simple starting-game that is not derived from the game of interest. In the literature on homotopy methods, starting with Harsanyi [10], the starting-game is usually derived from the game of interest by positing a prior distribution over the players' pure strategies, and using a starting-game whose payoffs are the result of playing against this prior distribution. In this section, we extend the result of Section 4 to handle these starting-games and thus obtain results for the Herings-van den Elzen [13] and Herings-Peeters [14] algorithms, which use the same underlying homotopy, and the van den Elzen-Talman [15] algorithm, which uses a different homotopy. All three algorithms have been shown under certain conditions to mimic the Harsayni-Selten linear tracing procedure. For each algorithm, we use the uniform distribution as the prior distribution, which is a natural choice.

The van den Elzen-Talman algorithm uses a homotopy based on a starting mixed-strategy profile v. Letting Σ be the set of mixed-strategy profiles, let $\Sigma(t)$ be the set of convex combinations $(1 - t)\{v\} + t\Sigma$. In the notation of [15], the van den Elzen-Talman algorithm —restricted to the two-player case— uses the homotopy

$$H(t, \sigma) = \beta^1_{\sigma^1(t)}(\sigma) \times \beta^2_{\sigma^2(t)}(\sigma)$$

where for $i = 1, 2$, $\beta^i_{\sigma^i(t)}(\sigma)$ denotes the best responses of player i to mixed strategy σ, restricted to $\Sigma(t)$.

Theorem 6: It is PSPACE-complete to compute equilibria that result from the above van den Elzen-Talman homotopy.

The proof works by giving each player a dummy strategy which does not get used in any equilibrium, but whose payoffs ensure that the uniform distribution (at $t = 0$) over all players' strategies results in payoffs that looks like $\mathcal{G}_0$. A similar trick works for the homotopy of [13, 14].

6. From Linear Tracing to Lemke-Howson

The Lemke-Howson (L-H) algorithm is an important and rich research subject in and by itself within Game Theory; for the purposes of this reduction, it is helpful to take a point of view that considers the L-H algorithm as a homotopy [15], where an arbitrary strategy (the one whose label is dropped initially) is given a large "bonus" to be played, so that the unique equilibrium consists of that strategy together with its best response from the other player; the homotopy arises from reducing that bonus continuously to zero.

Theorem 7: It is PSPACE-complete to find *any* of the solutions of a 2-player game that are constructed by the Lemke-Howson algorithm.

The remainder of this section proves Theorem 7, the hardness being established by a reduction from the graphical game problem of Proposition 3, extending the ideas of the reduction for Linear tracing (Theorems 4, 5). A new technical challenge here is that the choice of initially dropped label results in $2n$ alternative homotopy paths, and we must ensure that any of the (up to) $2n$ solution can encode the single solution to some instance of Linear tracing.

Suppose that some strategy has been given this "L-H bonus", and a Browder path of Nash equilibria is obtained from reducing that bonus to zero. As before let $t \in [0, 1]$ be a parameter that denotes the distance from the starting game of the homotopy to the game of interest, so that $1 - t$ is a multiplicative weight for the bonus in intermediate games. Consider the Browder path. It is piecewise linear, a topologically well-behaved line. Let $T \in [0, 1]$ parameterize points along the Browder path — an equilibrium $\mathcal{N}_T$ is the one that is a fraction T of the distance along the path (starting at the version of the game with the L-H bonus). So, multiple values of T can correspond to the same value of t. Here we mostly focus on T rather than t.

The following construction addresses the issue that an arbitrary strategy may receive the L-H bonus. We embed two copies of a circuit-encoding game $\mathcal{G}$ (Definition 6) into a game instance for the Lemke-Howson algorithm. At least one of those copies of $\mathcal{G}$ will not contain the strategy that receives the L-H bonus. The L-H homotopy, restricted to that copy of $\mathcal{G}$, will simulate the homotopy of Section 4.

In Figure 1, $\mathcal{G}$ denotes a circuit-encoding $n \times n$ game (note the two copies) whose payoffs have been rescaled to lie in the interval $[0{\cdot}4, 0{\cdot}6]$. $\mathcal{G}$ is assumed to have an associated graphical game with two "switch" players v^r_{switch}, v^c_{switch} that affect the equilibria of $\mathcal{G}$ according to Corollary 1. They will correspond to the first pair of each of $\mathcal{G}$'s players' strategies (s^r_0, s^r_1) and (s^c_0, s^c_1) such that,

- if both $\mathbf{p}[v^r_{switch}] = 1$ and $\mathbf{p}[v^c_{switch}] = 1$, $\mathcal{G}$'s equilibrium encodes a solution to an END OF THE LINE instance that is efficiently encoded by $\mathcal{G}$;
- if either $\mathbf{p}[v^r_{switch}] = 0$ or $\mathbf{p}[v^c_{switch}] = 0$, $\mathcal{G}$ encodes the "basic" Brouwer-mapping function;
- if we add a bonus to the row player for his first strategy s^r_0 that is less than some threshold τ, it will result in $\Pr[s^r_0] = 0$ and hence $\mathbf{p}[v^r_{switch}] = 1$, and similarly for the column player with respect to s^c_0 and v^c_{switch}. (We will see that such bonuses occur, and they decrease at $T \longrightarrow 1$.)

Notation. A, B, C, D and A', B', C', D' denote sets of the players' strategies as shown in Figure 1. In the context of a mixed-strategy profile, $\Pr[C]$ denotes the probability that the column player uses C; $\Pr[A]$ that he chooses an element of A, and so on. Let $X(T) = \Pr[C] + \Pr[D] + \Pr[C'] + \Pr[D']$, a function of distance along the Browder path. We note the following facts

- $X(0) \geq 1$ (if, say, a column player strategy receives the L-H bonus, then the row player will play some pure best response, either C' or D'; so $\Pr[C'] = 1$ or $\Pr[D'] = 1$.)
- $X(1) \leq \frac{1}{25}$ (shown in Lemma 4)

together with the key observation that $X(T)$ is a continuous function of T, implying:

Observation 6: For some $T' \in [0, 1]$, $X(T') = \frac{1}{4}$, and for $T > T'$, $X(T) < \frac{1}{4}$.

Let $\bar{\mathcal{G}}$ be the copy of $\mathcal{G}$ that does not contain the strategy that receives the L-H bonus. (If one of C, D, C' or D' receive the L-H bonus, then $\bar{\mathcal{G}}$ may be either copy of $\mathcal{G}$.)

For any X, at least one player p has an additional bonus at least $X/4$ to play s^p_0 in $\bar{\mathcal{G}}$ (suppose for example $\Pr[C] + \Pr[D] \geq X/2$ and p is the row player; Figure 1 awards additional $e = 1$ to p when C or D is played). But neither player's bonus exceeds $X/2$. As T increases from T' to 1, $X(T)$ goes down from $\frac{1}{4}$ to at most $\frac{4}{M}$. We will establish that when $X(T) = \frac{1}{4}$, $\mathcal{N}_T$ contains a solution to a "biased" version of $\bar{\mathcal{G}}$ where one of the players' first strategies (i.e. s^r_0 or s^c_0) has an additional bonus (enough to ensure $\Pr[s^r_1] = 0$ and $\mathbf{p}[v^r_{switch}] = 0$, in the case of the row player). Furthermore, when $T = 1$, we have that $\mathcal{N}_T$ contains a solution to $\bar{\mathcal{G}}$, only with smaller biases. These biases are associated with "switch" strategies in the graphical game associated with $\mathcal{G}$.

Let T' be the largest value of T where $X(T)$ is large enough that one of the bonuses sets $\Pr[s^p_0] = 1$ in $\bar{\mathcal{G}}$ (for $p \in \{r, c\}$). Between T' and $T = 1$ we pass through a continuum of equilibria where $\Pr[s^p_0]$ changes from 1 to 0; equivalently $\mathbf{p}[v^p_{switch}]$ changes from 0 to 1, and the resulting equilibrium at $T = 1$ corresponds to a solution to OEOTL.

Lemma 3: Let $\mathcal{N}_T$ be a solution of $\mathcal{G}_T$. If $\bar{\mathcal{G}}$ is the bottom right-hand copy of $\mathcal{G}$ in Figure 1, then if the distributions over B and B' are normalised to 1, we have a Nash equilibrium of a game $\hat{\mathcal{G}}$ where the row player has an additional bonus of $e(\Pr[C] + \Pr[D]) / \Pr[B']$ to play his first strategy s^r_0, and the column player has an additional bonus of $e(\Pr[C'] + \Pr[D']) / \Pr[B]$ to play his first strategy s^c_0.

By symmetry, a similar result also holds in the case that $\bar{\mathcal{G}}$ is the top right-hand copy of the $\mathcal{G}$.

Proof: Payoffs to the row player are unaffected by the column player's distribution over A. Meanwhile, C and D lead to an additional bonus of e (weighted by the probability that C and D are used by the column player) for the row player to use the top row of B'. ■

Lemma 4: At $t = 1$ (equivalently, $T = 1$) we have in any Nash equilibrium, that $\Pr[C] \leq \frac{1}{M}$, $\Pr[D] \leq \frac{1}{M}$, $\Pr[C'] \leq \frac{1}{M}$ and $\Pr[D'] \leq \frac{1}{M}$. Since $M \geq 100$ we have $X(1) \leq \frac{1}{25}$.

A proof of Lemma 4 may be found in the full version.

Lemma 5: Assume that $e \leq 1$ in Figure 1 and that $M \geq 100$. Suppose that $X(T) \leq \frac{1}{4}$. Then $\Pr[A] \geq \frac{1}{10}$, $\Pr[B] \geq \frac{1}{10}$, $\Pr[A'] \geq \frac{1}{10}$, $\Pr[B'] \geq \frac{1}{10}$.

A proof of Lemma 5 may be found in the full version.

At $X(T) = \frac{1}{4}$ we have that at least one of $\Pr[C]$, $\Pr[D]$, $\Pr[C']$, $\Pr[D']$ is at least $\frac{1}{16}$, while at the end of the Browder path, we know that all these quantities are at most $\frac{1}{100}$. We set the switch threshold probability to be somewhere between these, but we have to use lower bounds on $\Pr[A]$, $\Pr[B]$, $\Pr[A']$, $\Pr[B']$ at $t = 1$ and upper bounds on these at $X = \frac{1}{4}$ (as well as lower bounds on these at $X = \frac{1}{4}$ to ensure that a Nash equilibrium of the "biased game" is being encoded).

Finally, we need to show that there exists τ such that the bonus from at least one switch strategy in $\bar{\mathcal{G}}$ changes continuously above τ to below it, while the bonus for the other switch strategy ends up below τ, thus initially,

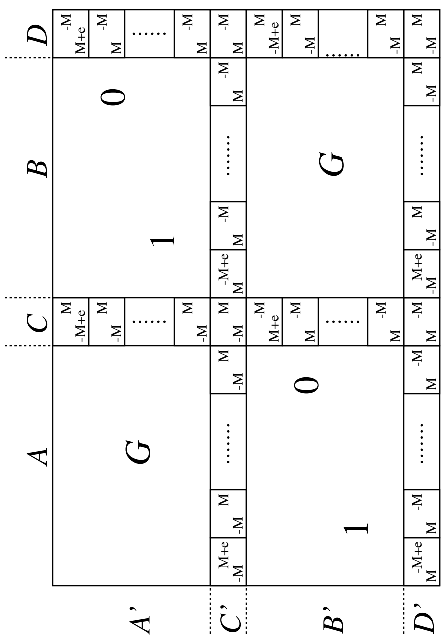

Figure 1. The game has 2 copies of $n \times n$ game $\mathcal{G}$ embedded in the top-left and bottom-right regions, with payoff rescaled to $[0{\cdot}4, 0{\cdot}6]$. In the top-right and bottom-left regions are copies of a $n \times n$ game that give the column player a payoff of 0 and the row player a payoff of 1.
Each of A, B, A', B' denotes a set of n strategies. C, D, C' and D' are individual strategies.
In the proofs we put $M = 1000$, $e = 1$.

at least one value of $\mathbf{p}[v^r_{switch}]$ and $\mathbf{p}[v^c_{switch}]$ is zero, but at the end both evaluate to 1. This needs to take into account the variable amount of probability allocated to the strategies in $\bar{G}$, since that affects the impact of the bonuses on s^p_0.

For any $T \in [T', 1]$ the weight assigned by each player to $\bar{\mathcal{G}}$'s strategies is at least $\frac{1}{10}$ by Lemma 5, so that the bonus for player p to play s^p_0, falls by a larger factor than the probability that $\bar{G}$ is played. That means that τ can indeed be chosen as required.

7. Discussion and Open Problems

Should a more general result be obtainable? For example, perhaps it should be possible to identify general classes of "path-following algorithms" that include the ones we analyzed here, for which it is PSPACE-complete to compute their output.

Acknowledgements: Goldberg was supported by EPSRC Grant EP/G069239/1 "Efficient Decentralised Approaches in Algorithmic Game Theory". Papadimitriou was supported by an NSF grant and by a Google grant.

References

[1] Y. Balasko. Economic Equilibrium and Catastrophe Theory: An Introduction. *Econometrica*, **46**(3) 557-569. (1978)
[2] F.E. Browder. On continuity of fixed points under deformations of continuous mappings. *Summa Brasiliensis Math* **4**, 183-191. (1960)
[3] X. Chen and X. Deng. On the Complexity of 2D Discrete Fixed Point Problem. *Proceedings of 33rd ICALP Conference*, LNCS 4051, 489-500. (2006)
[4] X. Chen, X. Deng and S-H Teng. Settling the Complexity of Computing Two-player Nash Equilibria. *Journal of the ACM*, **56**(3), 1-57. (2009)
[5] P. Crescenzi and C. H. Papadimitriou. Reversible Simulation of Space-Bounded Computations. *Theoretical Computer Science* **143**(1), 159-165. (1995)
[6] C. Daskalakis, P.W. Goldberg and C.H. Papadimitriou. The Complexity of Computing a Nash Equilibrium. *SIAM Journal on Computing*, **39**(1), 195-259. (2009)
[7] B.C. Eaves. Homotopies for computation of fixed points. *Mathematical Programming*, **3**(1), 1-22. (1972)
[8] K. Etessami and M. Yannakakis. On the Complexity of Nash Equilibria and Other Fixed Points. *SIAM Journal on Computing*, **39**(6), 2531-2597. (2010)
[9] D. Fudenberg and D.K. Levine. *The Theory of Learning in Games*. MIT Press. (1998)
[10] J.C. Harsanyi. The tracing procedure: a Bayesian approach to defining a solution for n-person noncooperative games. *International Journal of Game Theory*, **4**, 61-95. (1975)
[11] J.C. Harsanyi and R. Selten. *A general theory of equilibrium selection in games.* Cambridge: MIT Press. (1988)
[12] P.J-J. Herings. Two simple proofs of the feasibility of the linear tracing procedure. *Economic Theory*, **15**, 485-490. (2000)
[13] P.J-J. Herings and A. van den Elzen. Computation of the Nash Equilibrium Selected by the Tracing Procedure in N-Person Games. *Games and Economic Behavior*, **38**, 89-117. (2002)
[14] P.J-J Herings and R.J.A.P. Peeters. A differentiable homotopy to compute Nash equilibria of n-person games. *Economic Theory*, **18**(1), 159-185. (2001)
[15] P.J-J. Herings and R. Peeters. Homotopy methods to compute equilibria in game theory. *Economic Theory*, **42**(1), 119-156. (2010)
[16] M. D. Hirsch, C. H. Papadimitriou and S. A. Vavasis. Exponential lower bounds for finding Brouwer fixed points. *Journal of Complexity*, **5**(4), 379-416. (1989)
[17] M. Kearns, M. Littman and S. Singh. Graphical Models for Game Theory. *Proceedings of UAI*, 253-260. (2001)
[18] D. M. Kreps. *A Course in Microeconomic Theory.* Princeton Univ. Press. (1990)
[19] C. E Lemke and J. T. Howson, Jr. Equilibrium points of bimatrix games. *SIAM J. Appl. Math*, **12**(2), 413-423. (1964)
[20] R. B. Myerson. Nash equilibrium and the history of economic theory. *Journal of Economic Literature*, **37**(3), 1067-1082. (1999)
[21] J. Nash. Noncooperative Games. *Annals of Mathematics*, **54**(2), 289-295. (1951)
[22] M.J. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press. (1994)
[23] C.H. Papadimitriou. On the Complexity of the Parity Argument and Other Inefficient Proofs of Existence. *J. Comput. Syst. Sci.* **48**(3), 498-532. (1994)
[24] R. Savani and B. von Stengel. Hard-to-Solve Bimatrix Games. *Econometrica* **74**(2), 397-429. (2006)
[25] H. E Scarf and T. Hansen. *The computation of economic equilibria*. Yale Univ. Press. (1973)
[26] B. von Stengel, A. van den Elzen and D. Talman. Computing normal form perfect equilibria for extensive two-person games, *Econometrica* **70**(2), 693-715. (2002)