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

PAUL W. GOLDBERG, University of Liverpool
CHRISTOS H. PAPADIMITRIOU, University of California at Berkeley
RAHUL SAVANI, University of Liverpool

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 Savani and von Stengel [2006]. These results show that our techniques can be widely applied and suggest that the PSPACE-completeness of implementing homotopy methods is a general principle.

Categories and Subject Descriptors: F.2.2 [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms and Problems

General Terms: Algorithms, Economics, Theory

Additional Key Words and Phrases: Fixpoint problems, game theory, equilibria, homotopy methods, Lemke-Howson algorithm

ACM Reference Format:
Goldberg, P. W., Papadimitriou, C. H., and Savani, R. 2013. The complexity of the homotopy method, equilibrium selection, and Lemke-Howson solutions. *ACM Trans. Econ. Comp.* 1, 2, Article 9 (May 2013), 25 pages.
DOI:http://dx.doi.org/10.1145/2465769.2465774

1. INTRODUCTION

According to Roger Myerson [1999], 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 [1999] argues convincingly that the concept of Nash equilibrium lies at the foundations of modern economic thought.

Seen from an algorithmic perspective, however, the Nash equilibrium suffers 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

This research was supported by EPSRC grant EP/G069239/1 "Efficient Decentralised Approaches in Algorithmic Game Theory", by an NSF grant and by a Google grant.
Authors' addresses: P. W. Goldberg, Department of Computer Science, University of Liverpool, Ashton Street, Liverpool L69 3BX, UK; C. H. Papadimitriou, Computer Science Division, University of California at Berkeley, Soda Hall, Berkeley, CA 94720; R. Savani (corresponding author), Department of Computer Science, University of Liverpool, Ashton Street, Liverpool L69 3BX, UK; email: rahul.savani@liverpool.ac.uk.
Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org.
© 2013 ACM 2167-8375/2013/05-ART9 $15.00
DOI:http://dx.doi.org/10.1145/2465769.2465774

been identified by economists since the 1950s, and much effort has been devoted to algorithms for finding Nash equilibria; see Scarf [1973], Lemke and Howson [1964], and Herings and Peeters [2001] for examples from an extensive literature. None of these algorithms came with polynomial-time guarantees, however, and the recent result [Chen et al. 2009; Daskalakis et al. 2009] 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 [Hirsch et al. 1989; Savani and von Stengel 2006]. One exception is an important algorithmic genre known as *homotopy methods* [Eaves 1972]; see Herings and Peeters [2010] 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 [1960] 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 nonmonotonic, going back and forth in time. As a result, equilibria vanish at its folds, leaving the market in turmoil [Balasko 1978]. In games, a proposed remedy for multiplicity is the so-called focal point theory; see, for example, Kreps [1990, page 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., Fudenberg and Levine [1998]) can also be considered a remedy for multiplicity. Harsanyi [1975] proposed the *tracing procedure* for battling equilibrium multiplicity, a theory further explicated in his joint book *A General Theory of Equilibrium Selection in Games* [Harsanyi and Selten 1988] (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 article 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 will not only undergo an exponential number of pivots (this was expected since Hirsch et al. [1989]), 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 Savani and von Stengel [2006]) of algorithmic hope in this field. This reinforces the "exponentially long paths" result of Savani and von Stengel [2006] 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 [Herings and Peeters 2010], 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 behavior) 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 in 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 Papadimitriou [1994], but this fact has so far remained unexploited for proving lower bounds for other problems.

Outline of the article. 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 Chen et al. [2009] and Daskalakis et al. [2009] are central to both Section 3 and Section 4 and are recalled and extended along the way. 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 article, 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 [Browder 1960] (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 [Eaves 1972; Herings and Peeters 2010] 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 Herings and Peeters [2010], we do not expect the path to be monotonic in t; indeed, we show in Appendix A.1 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 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 [Kearns et al. 2001] (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 [2002] achieves this (see Section 4.2). 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.

[1]A very interesting alternative consideration [Etessami and Yannakakis 2010] 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 Etessami and Yannakakis [2010] that this harder problem belongs to PSPACE.

Definition 2.1. An (S,P)-graph with parameter n is a graph on $\{0,1\}^n$ specified by circuits S and P, as described earlier, 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 [Papadimitriou 1994] 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 Papadimitriou [1994]), which is the starting point of our reductions.

THEOREM 2.2 [PAPADIMITRIOU 1994]. OEOTL *is* PSPACE-*complete.*

2.3. Our Approach

In Daskalakis et al. [2009], each instance I of END OF THE LINE is reduced to a game $\mathcal{G}_I$ in such a way that any Nash equilibrium of $\mathcal{G}_I$ efficiently encodes a solution to I. Here we reduce I to a homotopy problem defined by two games, $(\mathcal{G}_0, \mathcal{G}'_I)$, where $\mathcal{G}_0$ depends only on n, while $\mathcal{G}'_I$ encodes the circuits in I, and is an extension of $\mathcal{G}_I$ of Daskalakis et al. [2009]. We establish that the associated linear homotopy corresponds to the naive "follow the line" approach to solving OEOTL; technically, a suffix of the homotopy path corresponds to following the line, and the existence of the relevant suffix is established in a nonconstructive way, using the intermediate value theorem. In extending the result to the Lemke-Howson algorithm, the main technical obstacle is the initial choice of which "label to drop", leading to multiple paths in the mixed-strategy profile space. We have to ensure that all of the $2n$ solutions (one for each pure strategy) efficiently encode the solution to I, where I is treated as an instance of OEOTL. This is done by embedding two copies of the game $\mathcal{G}'_I$ inside a larger one in such a way that at least one copy does not contain the initially dropped label, and arguing that any Lemke-Howson equilibrium restricted to this copy ends up encoding the unique solution to I.

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. [2009], here applied to a three-dimensional domain. In Section 3.2, we review the techniques of Daskalakis et al. [2009] and Chen et al. [2009] for implementing Brouwer-mapping functions as arithmetic circuits. In Section 3.3, we prove Theorem 3.7, 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 Eq. (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 $\mathcal{F}_1$. In Theorem 3.7, we consider the problem of finding fixpoints within accuracy that is inverse exponential in n. In Appendix A.2, we show how to use

the snake embeddings of Chen et al. [2009] to strengthen the result so that it applies to finding fixpoints within accuracy that is inverse polynomial in n.

3.1. Definitions and Notation

Notation 1. Let K be the unit 3D cube $[0,1]^3$. For $n \in \mathbb{N}$ let $\mathcal{K}^{(n)}$ denote a partition of K into 2^{3n} "cubelets", $\mathcal{K}^{(n)} = \{K_{ijk} \ : \ 0 \le i,j,k \le 2^n - 1\}$; 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 Chen et al. [2009], here specialized to the case of three dimensions. We also introduce some variations of the definition, as follows.

Definition 3.1. (*Brouwer-Mapping Circuit/Function; Basic Brouwer-Mapping Function; DGP-Style Basic Brouwer-Mapping Function; Partial Basic Brouwer-Mapping Function*).

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 Daskalakis et al. [2009], and so is computable with a bmc of size polynomial in the size of circuits S and P. (Proposition 3.2 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 nonelements of $\mathcal{S}$.

PROPOSITION 3.2. *The following problem is* PSPACE*-complete. Given a Brouwer-mapping circuit B, find a cubelet vertex in K whose set of (typically 8) neighboring cubelets contains 4 that are 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 $\{1,2,3\}$. We note in passing that if we did not make the "connected to the origin" requirement, the problem would be PPAD-complete.

PROOF. We reduce from OEOTL (Theorem 2.2). Let G be an (S,P)-graph. Let f_B be a DGP-style bmf derived from G, whose circuit B efficiently encodes G.

Given a bmf f_B, define a $\{1,2,3\}$-*chromatic vertex* to be one that is shared by 3 cubelets with colors 1, 2, and 3. By construction, the only exterior $\{1,2,3\}$-chromatic vertex for any bmf is $2^{-n}(1,1,0)$. Form a digraph G_B on $\{1,2,3\}$-chromatic vertices by adding an arc between any pair that are connected by an edge of a cubelet, directed such that if it is pointing away from a viewer, its adjacent colors 1, 2, 3 will appear in clockwise order around it. The reduction of Daskalakis et al. [2009] ensures G_B has indegree/outdegree at most 1.

Define a *panchromatic vertex* to be one that belongs to 4 cubelets of all 4 different colors. By construction, for all bmf's there is a path of $\{1,2,3\}$-chromatic vertices

starting at $2^{-n}(1,1,0)$ and ending at a unique panchromatic vertex v_{end}. v_{end} is a solution; it can be found in polynomial space by following this path.

Let G_B be the graph on f_B's $\{1,2,3\}$-chromatic vertices as described before. The reduction of Daskalakis et al. [2009] has the following properties, from which the result follows.

(1) Each vertex v of G has an associated $\{1,2,3\}$-chromatic vertex $b(v)$ of G_B; v and $b(v)$ may be computed in polynomial time from each other. For $v = 0^n$, $b(v) = 2^{-n}(1,1,0)$.
(2) v is a solution to END OF THE LINE if and only if $b(v)$ is panchromatic.
(3) Each arc (v,w) of G corresponds to a sequence of edges of G_B that connect $b(v)$ to $b(w)$.
(4) Each connected component of cubelets colored with $\{1,2,3\}$, corresponds to a connected component of G. □

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 three-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 3.3. 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 3.4. 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 (i.e., a weighted sum where weights are nonnegative and sum to 1) 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 3.5. *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 Chen et al. [2009] and Daskalakis et al. [2009] 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; it can be extended to higher dimension using the "snake embeddings" of Chen et al. [2009]. (See Appendix A.2.) The general idea of the technique is to take a simplicial decomposition of the domain K, give rules for obtaining the values of $\mathcal{F}$ at the vertices of the decomposition, and linearly interpolate within each simplex.

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 3.6. *$\mathcal{F}$ as defined earlier, 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 bmf's f_0 and f_1, where f_0 is the *basic bmf* of Definition 3.1, and f_1 shall be a DGP-style bmf that encodes an instance of END OF THE LINE as constructed in Daskalakis et al. [2009]. 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 3.5. 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 Eq. (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 3.3 (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 3.7. *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 3.2 as follows. Let B be a Brouwer-mapping circuit derived from OEOTL-instance (S,P) using Proposition 3.2 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 3.5). $\mathcal{F}_t$ is given by Eq. (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 3.2 (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 nonzero 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 nonzero cubelet of f_B. This follows provided we assume that connected components of nonzero 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 nonzero cubelet of f_B, it is also at least 2^{-n} distant from any nonzero cubelet of f_0, since for any cubelet K_{ijk}, $f_B(K_{ijk}) = 0 \Longrightarrow f_0(K_{ijk}) = 0$. The implementation of any bmf f as a function $\mathcal{F}$ computed by a linear arithmetic circuit, as referred to in Theorem 3.5, 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.

$$\mathcal{G}_0 = \begin{array}{c|cccc} & s_0^c & s_1^c & \dots & s_{n-1}^c \\ \hline s_0^r & (1,1) & (1,0) & \dots & (1,0) \\ s_1^r & (0,1) & (0,0) & \dots & (0,0) \\ \vdots & \vdots & \vdots & & \vdots \\ s_{n-1}^r & (0,1) & (0,0) & \dots & (0,0) \end{array} \tag{2}$$

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 two players, we need a positive ϵ to ensure that solutions can be written down as rational numbers.

THEOREM 4.1. 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 Chen et al. [2009] to obtain a version where ϵ is inverse polynomial.

Our reduction uses the result of the previous section, along with earlier reductions between strategic form games and graphical games. $\mathcal{G}_0$ has a similar role to the basic Brouwer function $\mathcal{F}_0$, but the correspondence is indirect; generally $\mathcal{F}_0$ is associated with one of the "intermediate games" $\mathcal{G}_t$ for $t > 0$.

4.1. Brief Overview of the Proof Ideas

The following is a brief overview of the rest of Section 4. Membership of PSPACE can be deduced from Herings and van den Elzen [2002]. The reduction from the PSPACE-complete discrete Brouwer fixpoint problem of the previous section applies the idea from Daskalakis et al. [2009] 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 two-player game such that the linear-tracing procedure corresponds to this continuous path of equilibria in the graphical game.

4.2. Membership of PSPACE

Herings and van den Elzen [2002] show how to find approximate equilibria on multi-player games, implicitly constructing a degree-2 graph that has a vertex corresponding to $\mathcal{N}_0$, the Nash equilibrium of $\mathcal{G}_0$. Given a simplicial decomposition of $D \times [0,1]$ (where D is the space of mixed strategies of $\mathcal{G}$) vertices of the graph correspond to simplices

and subsimplices, and edges are implicitly defined by a lexicographical pivoting rule that governs a choice of movement from simplex to adjacent simplex, at each step of the algorithm. It can be checked that this algorithm establishes membership of PSPACE for multiplayer LINEAR TRACING.

4.3. Graphical Games

In a graphical game [Kearns et al. 2001], 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 as follows.

PROPOSITION 4.2. *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 3.1, 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 4.3 (*Linear Graphical Game; Simulation of bmf's and Partial bmf's*).
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 Daskalakis et al. [2009] 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 poly(ϵ)-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 preceding 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 4.4. *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 3.1, 3.4, and 4.3. $\mathcal{GG}_i$ 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

$$\begin{aligned} \mathbf{p}[\bar{v}_x^+] &= (\mathbf{p}[v_x^0] - \mathbf{p}[v_{switch}]) + (\mathbf{p}[v_x^1] - (1 - \mathbf{p}[v_{switch}])) \\ \mathbf{p}[v_x^+] &= \max(\mathbf{p}[\bar{v}_x^+], \min(\mathbf{p}[v_x^0], \mathbf{p}[v_x^1])) \end{aligned} \tag{3}$$

(and similar expressions for v_y^+ and v_z^+) where the parentheses in the previous 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 4.2: We reduce from the circuit homotopy of Theorem 3.7. 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 4.3. Construct $\mathcal{GG}^+$ as in Lemma 4.4, 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 Eq. (3) is essentially the same as Eq. (1)).

That concludes the proof of Proposition 4.2. □

The following version of Lemma 4.4 is useful in the construction for Lemke-Howson solutions, later on.

COROLLARY 4.5. *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. Furthermore, it is* PSPACE*-complete to compute a Nash equilibrium of $\mathcal{GG}^+$ where v_{switch} and v'_{switch} play 1, that is connected to one where either or both play 0 (via equilibria of games where v_{switch}, v'_{switch} play intermediate strategies).*

PROOF. The proof of the previous lemma is modified as follows. We reuse Eq. (3) for players (v_x^+, v_y^+, v_z^+). We have an additional 3 players $(v_x^{++}, v_y^{++}, v_z^{++})$ whose mixed strategies represent a point in K, and have behavior governed by

$$\begin{aligned} \mathbf{p}[\bar{v}_x^{++}] &= (\mathbf{p}[v_x^0] - \mathbf{p}[v'_{switch}]) + (\mathbf{p}[v_x^+] - (1 - \mathbf{p}[v'_{switch}])) \\ \mathbf{p}[v_x^{++}] &= \max(\mathbf{p}[\bar{v}_x^{++}], \min(\mathbf{p}[v_x^0], \mathbf{p}[v_x^+])) \end{aligned} \tag{4}$$

again with similar expressions for v_y^{++} and v_z^{++}.

The PSPACE-completeness follows in a similar way to the proof of Proposition 4.2; Observation 3 applies to $\mathcal{GG}^+$. □

4.4. From Graphical to Two-Player Strategic Form Games

In this subsection we prove the following theorem, from which Theorem 4.1 follows since we have previously noted membership of PSPACE.

THEOREM 4.6. *It is* PSPACE*-hard to compute the Nash equilibrium of a given two-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 4.2. Let $\mathcal{GG}^+$ be a linear graphical game that includes a player v_{switch} as per Proposition 4.2. First, modify $\mathcal{GG}^+$ to give v_{switch} a small payment (say, 0.01) to play 1, and zero to play 0. Thus in equilibria of $\mathcal{GG}^+$, v_{switch} plays 1.

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 Daskalakis et al. [2009, Section 6.1] from graphical games to two-player games (a similar reduction is used in Chen et al. [2009, Section 7] to express *generalized circuits* (similar to our linear arithmetic circuits) as two-player games).

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

Definition 4.7. A *circuit-encoding* two-player game $\mathcal{G}$ has a corresponding graphical game $\mathcal{GG}$ ($\mathcal{G}$ is assumed to be derived from $\mathcal{GG}$ by the reduction of Daskalakis et al. [2009]) 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 4.7. v_{switch} has 2 corresponding strategies $(v_{switch}, 0)$ and $(v_{switch}, 1)$ which we may assume belong to the column player; denote them s^c_k and s^c_{k+1}. 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 two-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 $\{s^r_0, \ldots s^r_n\}$ for the row player, and $\{s^c_0, \ldots s^c_n\}$ 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 $j > 0$. (Thus $\mathcal{G}_0$ is a $(n+1) \times (n+1)$ version of Eq. (2).)

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 $\{s_0^r, \ldots, s_n^r\}$ for the row player, and $\{s_0^c, \ldots, s_n^c\}$ for the column player. Payoffs are as follows:

— (s_0^r, s_0^c) results in payoffs $(0, -1)$ for the players.[2]
— (s_0^r, s_j^c) for $j > 0$ results in payoffs $(0, \frac{3}{4})$.
— (s_j^r, s_0^c) 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 $\mathcal{G}$ given before.

$$\mathcal{G}_1 = \begin{array}{c|cccc} & s_0^c & s_1^c \cdots s_{k-1}^c & s_k^c & s_{k+1}^c \cdots s_n^c \\ \hline s_0^r & (0,-1) & (0,\frac{3}{4}) \cdots (0,\frac{3}{4}) & (0,\frac{3}{4}+\delta) & (0,\frac{3}{4}) \cdots (0,\frac{3}{4}) \\ s_1^r & (-1,\frac{3}{4}) & & & \\ \vdots & \vdots & & \mathcal{G} & \\ s_n^r & (-1,\frac{3}{4}) & & & \end{array}$$

Let $\mathcal{G}_t = (1-t)\mathcal{G}_0 + t\mathcal{G}_1$. The preceding payoffs have been chosen so that Nash equilibria satisfy: in $\mathcal{G}_1$, players do not use s_0^r or s_0^c; in $\mathcal{G}_{0{\cdot}6}$, players both have a proper mixture of s_0^r and s_0^c with their other strategies. Since $\mathcal{G}$'s payoffs were rescaled to lie in $[0{\cdot}9, 1{\cdot}1]$, $\Pr[s_0^r]$ and $\Pr[s_0^c]$ 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_0^c	$s_1^c \ldots s_n^c$
s_0^r	$(0{\cdot}4, -0{\cdot}2)$	$(0{\cdot}4, 0{\cdot}45+\delta)$
$s_1^r \ldots s_n^r$	$(-0{\cdot}6, 0{\cdot}85)$	$([0{\cdot}54, 0{\cdot}66], [0{\cdot}54, 0{\cdot}66])$

Thus a continuous path of equilibria should at some stage allocate gradually less and less probability to s_0^r and s_0^c as t increases.

Observation 5. In any Nash equilibrium $\mathcal{N}$ of $\mathcal{G}_1$, the players assign probability 0 to s_0^r and s_0^c, and consequently $\mathcal{N}$ consists of a Nash equilibrium of $\mathcal{G}$, restricting to strategies s_j^r, $s_{j'}^c$ for $j, j' > 0$.

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

$$\begin{array}{c|cccc} & s_0^c & s_1^c \cdots s_{k-1}^c & s_k^c & s_{k+1}^c \cdots s_n^c \\ \hline s_0^r & (1-t, 1-2t) & (1-t, \frac{3}{4}t) \cdots (1-t, \frac{3}{4}t) & (1-t, (\frac{3}{4}+\delta)t) & (1-t, \frac{3}{4}t) \cdots (1-t, \frac{3}{4}t) \\ s_1^r & (-t, 1-\frac{1}{4}t) & & & \\ \vdots & \vdots & & t\mathcal{G} & \\ s_n^r & (-t, 1-\frac{1}{4}t) & & & \end{array}$$

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_0^r, s_0^c). As t increases, the players will start to use the other strategies. At that stage, consider the

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

distribution of their mixed strategies restricted to $s_1^r, \ldots, s_n^r$ and $s_1^c, \ldots, s_n^c$. 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_k^c. As t increases to 1, the bonus decreases continuously to 0, and we recover Observation 5. Now, recall from Definition 4.7 that the way Daskalakis et al. [2009] and Chen et al. [2009] 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 these two strategies represents the probability that v plays 1. Consider v_{switch} now, corresponding to s_k^c and s_{k+1}^c. 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 4.8. *Let $\mathcal{N}$ be a Nash equilibrium of $\mathcal{G}_t$ in which* $\Pr[s_0^r] < 1$ *and* $\Pr[s_0^c] < 1$. *Let $\mathcal{P}$ be the probability distributions over $\{s_1^r, \ldots, s_n^r\}$ and $\{s_1^c, \ldots, s_n^c\}$ obtained by taking each value* $\Pr[s_j^i]$ *(for $i \in \{r, c\}$, $1 \le j \le n$) and dividing it by* $1 - \Pr[s_0^i]$.

Then $\mathcal{P}$ is a Nash equilibrium of a version of $\mathcal{G}$ in which the column player receives an additional bonus of $\delta \Pr[s_0^r] / (1 - \Pr[s_0^r])$ *for playing s_k^c.*

PROOF. In Nash equilibrium $\mathcal{N}$, c's strategy s_0^c contributes the same quantity to each one of r's strategies $s_1^r, \ldots, s_n^r$. So the values $\Pr[s_1^r], \ldots, \Pr[s_n^r]$ must form a best response to c's mixed strategy from $\mathcal{P}$.

The column player receives a bonus $\delta t \Pr[s_0^r]$ specific to s_k^c, arising from the possibility that row player plays 0. He also receives an additional $\frac{3}{4}t$ for all strategies s_j^c for $j > 0$, but that uniform bonus has no further effect on his preference amongst $s_1^c, \ldots, s_n^c$.

So in $\mathcal{N}$, $\Pr[s_1^c], \ldots, \Pr[s_n^c]$ is a best response to a mixture of $\mathcal{G}$ weighted by $1 - \Pr[s_0^r]$ and the probability $\Pr[s_0^r]$ of a bonus $\delta \Pr[s_0^r]$ for playing s_k^c. This is equivalent to a best response to a version of $\mathcal{G}$ with a bonus of $\delta \Pr[s_0^r] / (1 - \Pr[s_0^r])$ for playing s_k^c. □

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 4.8 we can choose δ such that in any equilibrium of $\mathcal{G}_{0.5}$ we have $\Pr[s_{k+1}^c] = 0$. We also have that in any equilibrium of $\mathcal{G}_1$, $\Pr[s_k^c] = 0$. Consider the longest suffix of this path of equilibria 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_0^r and s_0^c, so Lemma 4.8 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 [1975], 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 and van den Elzen [2002] and Herings and Peeters [2001] algorithms, which use the same underlying homotopy, and

the van den Elzen-Talman [Herings and Peeters 2010] 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 Herings and Peeters [2010], 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 5.1. *It is* PSPACE*-complete to compute equilibria that result from the aforesaid van den Elzen-Talman homotopy.*

PROOF. (sketch) It can be checked that the algorithm uses polynomial space. For the hardness, we reduce from LINEAR TRACING; consider a game $\mathcal{G}$ for which we seek an equilibrium that results from starting with $\mathcal{G}_0$ of the form of (2). Suppose we take a game $\mathcal{G}$ from Section 4 and give each player an additional strategy as follows. Let s^r_n and s^c_n be the new strategies, for the row and column player respectively. s^r_n has a payoff of -10 for the row player, regardless of how the column player plays (thus, s^r_n is dominated by all the other strategies). The payoffs to the column player arising from s^r_n are chosen in such a way that, if in fact the row player uses the uniform distribution over $s^r_0, \ldots, s^r_n$, then the column player's payoffs will be 1 for s^c_0 and 0 for s^c_j, for $j > 0$. The new strategy s^c_n has a similar definition. Note that the new payoffs are at most n in absolute value. These new strategies ensure that we have the desired $\mathcal{G}_0$ of Section 4, when we restrict each player to his first n strategies. Strategies s^r_n and s^c_n are not used in any Nash equilibrium of $\mathcal{G}_t$, since they are strictly dominated for all t.

We let v be the uniform distribution. In $\mathcal{G}_t$, the row plays a mixture $(1-t)v + t\sigma^r_t$ while the column player plays $(1-t)v + t\sigma^c_t$, where σ^r_t and σ^c_t are mixed strategies whose support do not include s^r_n and s^c_n, so they constitute a Nash equilibrium of a version of $\mathcal{G}$ in which there is a bonus to play s^r_0 and s^c_0. This bonus drops continuously to zero, so it is equivalent to the linear-tracing homotopy. □

The algorithms of Herings and van den Elzen [2002] and Herings and Peeters [2001] are based on an identical homotopy and differ only in the numerical technique used to follow the homotopy path. We can show using essentially the same construction as before that it is PSPACE-hard to compute the equilibria found by these homotopies. To do so we can again construct a starting game by giving the row/column players new strategies s^r_n and s^c_n chosen to have low payoffs to the row (respectively, column) players, but whose payoffs to the opponent are chosen such that if either player played the uniform distribution, the opponent would receive a higher payoff for his first strategy (either s^r_0 or s^c_0) than the others, which would all receive the same (lower) payoffs.

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 [Herings and Peeters 2010], where an arbitrary strategy s (the one whose label is dropped initially) is initially given a large "bonus" B to be played; that is, the payoffs are changed so that

the player who has s receives an additional B to play s. We refer to this bonus as the "L-H bonus". For large enough B, strategy s strictly dominates all other strategies, and s together with a best response from the other player constitutes an equilibrium; the homotopy arises from reducing the bonus continuously to zero.

THEOREM 6.1. *It is* PSPACE-*complete to find* any *of the solutions of a two-player game that are constructed by the Lemke-Howson algorithm.*

The remainder of this section proves Theorem 6.1, the hardness being established by a reduction from the graphical game problem of Proposition 4.2, extending the ideas of the reduction for LINEAR TRACING (Theorems 4.1 and 4.6). 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) $2n$ solutions encodes the single solution to some instance of LINEAR TRACING.

Suppose that some strategy is given the 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.

Overview. The construction described shortly (and in Figure 1) 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 4.7) into a game instance for the L-H 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. Observation 6 identifies a point on the homotopy path where the players' mixed strategies, restricted to the relevant copy of $\mathcal{G}$, simulate play with bonuses for each player to use their first strategies. The subsequent discussion and Lemma 6.2 establish that these bonuses are large but get smaller as we continue along the homotopy path. Lemma 6.3 shows that at $t = 1$ these bonuses are significantly smaller. Lemma 6.4 shows a positive lower bound on the probability allocated to the strategies of both copies of $\mathcal{G}$, so that we may validly consider distributions that are restricted to those strategies.

The reduction. 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 4.5. 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 as $T \longrightarrow 1$.)

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

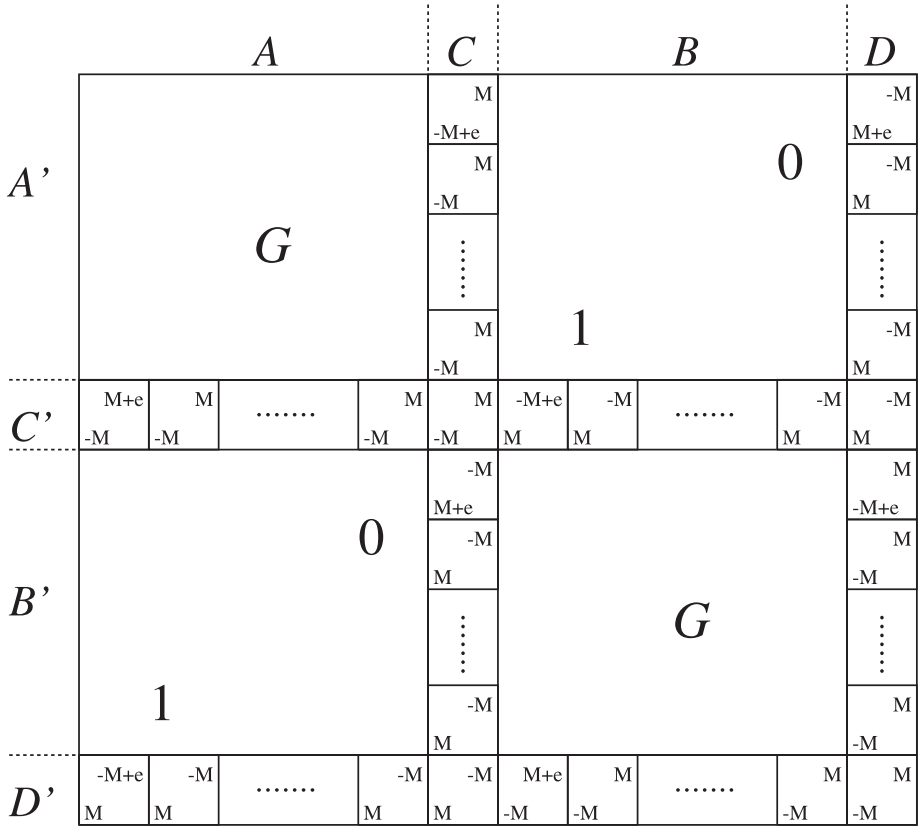

Fig. 1. The game has 2 copies of $n \times n$ game $\mathcal{G}$ embedded in the top-left and bottom-right regions, with payoff rescaled to $[0{\cdot}4, 0{\cdot}6]$. In the top-right and bottom-left regions are copies of a $n \times n$ game that give the column player a payoff of 0 and the row player 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 $M = 1000$, $e = 1$.

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 6.3)

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

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 $X/2$ to play s_0^p 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. As T increases from T' to 1, $X(T)$ goes down from $\frac{1}{4}$ to at most $\frac{1}{25}$ (in more detail, $\frac{4}{M}$ is shown

in Lemma 6.3, and we choose $M \geq 100$). 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_0^r or s_0^c) has an additional bonus (enough to ensure $\Pr[s_1^r] = 0$ and $\mathbf{p}[v_{switch}^r] = 0$, 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 $\Pr[s_0^p] = 1$ in $\bar{\mathcal{G}}$ (for $p \in \{r, c\}$). Between T' and $T = 1$ we pass through a continuum of equilibria where $\Pr[s_0^p]$ changes from 1 to 0; equivalently $\mathbf{p}[v_{switch}^p]$ changes from 0 to 1, and the resulting equilibrium at $T = 1$ corresponds to a solution to OEOTL.

LEMMA 6.2. *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 normalized 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_0^r, and the column player has an additional bonus of $e(\Pr[C'] + \Pr[D'])/\Pr[B]$ to play his first strategy s_0^c.*

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 6.3. *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 $X(1) \leq \frac{1}{25}$.*

PROOF. of Lemma 6.3. We give the proof that $\Pr[C] \leq \frac{1}{M}$; by symmetry the other claims are similar.

Assume for contradiction that $\Pr[C] > \frac{1}{M}$. We know that at least one strategy from $A \cup B$ gives positive payoff, since the places where they can obtain a negative payoff are "equal and opposite". Indeed, it can be checked that the payoff to at least one member of A and B is at least $\frac{1}{2}\min\{0{\cdot}4, e\}$. The (column player's) payoffs from C and D must sum to zero, so if $\Pr[C] > 0$ then $\Pr[D] = 0$ (D gets negative payoff and is a strictly worse response A or B.) A similar argument for the row player's payoffs establishes that one or both of C' and D' gets zero probability.

Given that $\Pr[C] > \frac{1}{M}$ we can deduce that $\Pr[A'] = 0$ due to being a worse response than B': C contributes at least $\frac{2M}{M} = 2$ to *payoff*(B') − *payoff*(A'); A contributes a positive amount; D has zero probability so contributes nothing; B contributes $\geq -0{\cdot}6$. We noted before that $\Pr[C'] = 0$ or $\Pr[D'] = 0$ (or both). Consider the following two cases.

Case 1. $\Pr[D'] = 0$. Deleting strategies with probability zero, we are left with the following structure.

	A	C	B
B'	$(1, 0)$	$([M, M+e], -M)$	$([0{\cdot}4, 0{\cdot}6], [0{\cdot}4, 0{\cdot}6])$
C'	$(-M, [M, M+e])$	$(-M, M)$	$(M, [-M, -M+e])$

Comparing C with A, we need $\Pr[C'] = 1$ to avoid A being a better response than C (which is supposed to have positive probability $> \frac{1}{M}$). If $\Pr[C'] = 1$, B is a worse

response than the others, but when the column player uses only A and C, C' has much lower payoff than B', contradicting assumption that $\Pr[C']$ is positive. This leaves us with Case 2.

Case 2. $\Pr[C'] = 0$. We get, after deleting zero-probability strategies, the following.

	A	C	B
B'	$(1,0)$	$([M, M+e], -M)$	$([0{\cdot}4, 0{\cdot}6], [0{\cdot}4, 0{\cdot}6])$
D'	$(M, [-M, -M+e])$	$(M, -M)$	$(-M, [M, M+e])$

Here the contradiction is immediate since B is a strictly better response than C, preventing $\Pr[C] > 0$. □

LEMMA 6.4. *Assume that $e \leq 1$ in Figure 1 and that $M \geq 100$. Suppose that $X(T) \leq \frac{1}{4}$. Then* $\Pr[A] \geq \frac{1}{10}$, $\Pr[B] \geq \frac{1}{10}$, $\Pr[A'] \geq \frac{1}{10}$, $\Pr[B'] \geq \frac{1}{10}$.

PROOF. We need to consider two cases in detail: case 1 assumes that an element of A' received the L-H bonus and case 2 assumes that C received the bonus. All other possibilities are essentially the same as these, by symmetry.

Case 1. Suppose first that a strategy from A' has been given the L-H bonus, and we are at a Nash equilibrium where $X = \frac{1}{4}$.

First we prove that the row player strategies satisfy $\Pr[A'] \geq \frac{1}{10}$, $\Pr[B'] \geq \frac{1}{10}$. Since $X \leq \frac{1}{4}$, we have $\Pr[A'] + \Pr[B'] \geq \frac{3}{4}$. Suppose for a contradiction that $\Pr[B'] < \frac{1}{10}$, so that $\Pr[A'] \geq 0{\cdot}65$. Then (for $e \leq 1$), C is the unique best response for the column player and hence $\Pr[C] = 1$. This implies $X \geq 1$, contradicting the assumption that $X \leq \frac{1}{4}$. Similarly, if $\Pr[A'] < \frac{1}{10}$ then $\Pr[B'] \geq 0{\cdot}65$, then D is the column player's unique best response, hence $\Pr[D] = 1$, again contradicting $X \leq \frac{1}{4}$.

Next we prove that the column player strategies satisfy $\Pr[A] \geq \frac{1}{10}$, $\Pr[B] \geq \frac{1}{10}$. Suppose for a contradiction that $\Pr[B] < \frac{1}{10}$, so that $\Pr[A] \geq 0{\cdot}65$. Then for the row player, D' is a better response than B' and C' (and regarding A', some strategy from A' received the L-H bonus, so we do not claim A' is suboptimal). If B' is not a best response, so $\Pr[B'] = 0$, since $X \leq \frac{1}{4}$ we have $\Pr[A'] \geq \frac{3}{4}$. Then C is strictly better than A, contradicting $\Pr[A] \geq 0{\cdot}65$. Alternatively suppose that $\Pr[A] < \frac{1}{10}$, so that $\Pr[B] \geq 0{\cdot}65$. Then C' is a better response than B' and D', since *payoff*$(C') \geq (0{\cdot}65 - 0{\cdot}35)M = 0{\cdot}3M$; *payoff*$(B') \leq \frac{1}{4}M$ since $\Pr[C] \leq \frac{1}{4}$ by assumption that $X \leq \frac{1}{4}$; *payoff*(D') is negative. With D' and B' eliminated, C is a strictly better response than B, contradicting $\Pr[B] > 0$.

Case 2. Suppose alternatively that it was strategy C' that received the L-H bonus.

We show first that the row player's strategies satisfy $\Pr[A'] \geq \frac{1}{10}$, $\Pr[B'] \geq \frac{1}{10}$. Suppose $\Pr[A'] < \frac{1}{10}$, so that $\Pr[B'] \geq 0{\cdot}65$. D is a better response than A and B. But from this it follows that $\Pr[D] + \Pr[C] = 1$, contradicting $X = \frac{1}{4}$. Suppose $\Pr[B'] < \frac{1}{10}$, so that $\Pr[A'] \geq 0{\cdot}65$. C is a better response than A and B, so $\Pr[C] + \Pr[D] = 1$ contradicting $X = \frac{1}{4}$.

Next we show that the column player's strategies satisfy $\Pr[A] \geq \frac{1}{10}$, $\Pr[B] \geq \frac{1}{10}$. Suppose $\Pr[B] < \frac{1}{10}$, so $\Pr[A] > 0{\cdot}65$. D''s payoff is greater than $0{\cdot}3M$ while A' and B' have payoff at most $\frac{1}{4}M + 1$ so D' is a better response than A' and B', contradicting

$X \leq \frac{1}{4}$. Suppose $\Pr[A] < \frac{1}{10}$, so $\Pr[B] > 0{\cdot}65$. C' is a better response than A' and B' even ignoring the L-H bonus. □

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, 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 6.4, 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.

A somewhat tangential question: Is the two-dimensional analog of OEOTL PSPACE-complete? (i.e., consider the PPAD-complete problem 2D-SPERNER [Chen and Deng 2009]; suppose we ask for the trichromatic triangle identified in the proof of Sperner's lemma.) In two dimensions, the gadget that is used to allow "edges" to cross each other, rearranges the structure of those edges, such that the corresponding solutions to END OF THE LINE are the same, but not the unique solution to OEOTL. Generally, there are many ways to modify the edges of a given (S, P)-graph so that the degree-1 vertices are unchanged, but the structure of the graph is in other respects completely different.

The tracing procedure is used by von Stengel et al. [2002] to solve extensive two-person games, and a *normal-form perfect* equilibrium is obtained by starting from a completely mixed starting vector. What is the complexity of computing a normal-form perfect solution using this (or other) methods? (They note [von Stengel et al. 2002, page 707] that on strategic form games this procedure mimics the linear tracing procedure of Harsanyi and Selten [1988].)

APPENDIXES

A.1. Exponentially Many Changes of Direction

We give an outline of how to modify our first construction, so as to show that in following a sequence of approximate fixpoints of Brouwer functions, or equilibria of games, t may have to change direction exponentially many times, and furthermore, oscillate between values whose difference is bounded away from zero.

$\mathcal{F}_0$ shall be the same as in Theorem 3.7. We construct a modified form of $\mathcal{F}_1$, which we will call $\mathcal{F}^m_1$, as follows. We identify two subsets of the cubelets K_{ijk}, R_1 and R_2,

defined as $R_1 = \{K_{ijk} \ : \ 10 \le k \le 12\}$, and $R_2 = \{K_{ijk} \ : \ 20 \le k \le 22\}$. These subsets are "thin layers" of cubelets that are perpendicular to the z-axis. Now we define how $\mathcal{F}_1^m$ behaves on points at the centers of cubelets. For x at the center of K_{ijk},

$$\begin{aligned} &\text{if } K_{ijk} \in R_1,\ \mathcal{F}_1^m(x) - x = 10(\mathcal{F}_1(x) - x),\\ &\text{if } K_{ijk} \in R_2,\ \mathcal{F}_1^m(x) - x = \tfrac{1}{10}(\mathcal{F}_1(x) - x),\\ &\qquad\text{otherwise, } \mathcal{F}_1^m(x) = \mathcal{F}_1(x). \end{aligned}$$

For points not at the centers of cubelets, $\mathcal{F}_1^m$ shall interpolate between the values at the nearest cubelet centers, using the same general approach as $\mathcal{F}_1$. Let $\mathcal{F}_t^m = (1-t)\mathcal{F}_0 + t\mathcal{F}_1^m$.

Let D be the set of cubelets K_{ijk} for which $f_0(K_{ijk}) \ne f_1(K_{ijk})$, so D is the region where fixpoints of $\mathcal{F}_t^m$ may exist. Let D^- be the connected component of D which contains the cubelet $K_{1,1,1}$, so D^- is the region within which fixpoints of $\mathcal{F}_t^m$ will occur on the homotopy path.

By construction, D^- has the property that $D^- \setminus R_1$ (and similarly $D^- \setminus R_2$) has exponentially many connected components. This follows from the construction of Daskalakis et al. [2009] from which D is derived; D simulates a (S,P)-graph by, for each edge of the (S,P)-graph, including a long sequence of cubelets that passes through both R_1 and R_2.

Now consider points in $D^- \cap R_1$. The claim is that fixpoints x of $\mathcal{F}_t^m$ for which $x \in D^- \cap R_1$, satisfy $t \le \frac{1}{4}$, and that for fixpoints x of $\mathcal{F}_t^m$ with $x \in D^- \cap R_2$ we have $t \ge \frac{3}{4}$. The general idea (in the first case; the second case is similar) is that $\mathcal{F}_1^m$ will map points z in R_1 to points z' for which at least one component of $z' - z$ is greater than 10α (recall $\alpha = 2^{-2n}$). Hence for $t \ge \frac{1}{4}$, $\mathcal{F}_t^m$ raises the value of this component (the positive contribution from $\mathcal{F}_1^m$ exceeds the negative contribution from $\mathcal{F}_0$) and prevents it from being a fixpoint. We can see this applies to z at the center of any cubelet in $D^- \cap R_1$; away from cubelet centers, consider the hyperplane spanned by cubelet centers of R_1. For points z in this hyperplane, we have by construction that some component of $\mathcal{F}_1^m(z) - z$ is at least 5α, so we need $t < \frac{1}{4}$ to have one of these points a fixpoint of $\mathcal{F}_t^m$.

The homotopy path must pass through this long sequence of regions that require $t > \frac{3}{4}$ or else $t < \frac{3}{4}$. Moreover, the two types of regions alternate, so we establish the following result.

THEOREM A.1. *For continuous functions defined using arithmetic circuits, the sequence of fixpoints along the path given by the linear homotopy* $(1-t)\mathcal{F}_0 + t\mathcal{F}_1$ *has exponentially many alternations of the value of* t.

We obtain the following corollary.

COROLLARY A.2. *For graphical or two-player games, suppose* $\mathcal{G}_0$ *is a game that assigns each player a dominating strategy, and* $\mathcal{G}$ *is an arbitrary game. The linear tracing procedure for the homotopy* $(1-t)\mathcal{G}_0 + t\mathcal{G}$ *will, in the worst case, have exponentially many reversals of* t.

The corollary follows since the way we represent Brouwer functions parameterized by t in terms of games parameterized by t does not change the value of t. Since we did not change $\mathcal{F}_0$, the associated game $\mathcal{G}_0$ is the same "dominating strategy" game of Section 4.

A.2. Polynomially Small Error

We can use the machinery of Chen et al. [2009] so that when we talk about the hardness of finding an ϵ-fixpoint, ϵ is allowed to be inverse polynomial rather than inverse exponential. This is achieved by using the snake embeddings of Chen et al. [2009].

Snake embeddings. A snake embedding reduces a low-dimensional Brouwer-mapping function (bmf) having 2^n cubelets in each dimension, to a $\Theta(n)$-dimensional bmf having $O(1)$ cubelets in each dimension, in such a way that panchromatic vertices of the high-dimensional bmf efficiently encode panchromatic vertices of the low-dimensional bmf. The reduction can be decomposed into a sequence of $\Theta(n)$ iterations, in which at each iteration, the number of cubelets along some axis is reduced by a constant factor, and we acquire an additional axis having $O(1)$ cubelets (in Chen et al. [2009] it is in fact 8 cubelets). (Intuitively, the space is folded a constant number of times and gains thickness along the new dimension.)

A specific snake embedding, and some notation. We consider a snake embedding of a three-dimensional bmf f_B of the type of Proposition 3.2. Initially the colors are $\{0, 1, 2, 3\}$; let c_i denote the new color at the i-th iteration and let s be the number of iterations required to reduce to 8 the number of cubelets along each axis.

Let $n = 4 + c_s$ be the dimension of the new snake embedding.

Definition A.3. Let K^n be the unit n-dimensional cube. Partition K^n into cubelets $K^n_{\mathbf{v}}$ where $\mathbf{v} \in \{0, 1, ..., 7\}^n$ represents a cubelet of edge length 1/8. A *Brouwer-mapping circuit* maps each such cubelet to one of the colors $\{0, 1, 2..., n\}$. Again, a bmf should be polynomial-time computable, and map exterior cubelets to color i for cubelets whose i-th coordinate contains the first 0 (when $\mathbf{v}$ contains a 0), otherwise color 0. Other types of bmf that correspond to Definition 3.1 are defined analogously.

The high-dimensional bmf can be computed by a Brouwer-mapping circuit B' that is polynomial in the size of B. Let $f_{B'}$ be the function computed. The challenge is to implement $f_{B'}$ using an arithmetic circuit that is polynomial in the size of the circuit that computes the bmf, and computes a Lipschitz continuous function. The simplicial decomposition technique of Theorem 3.5 no longer works when we move to nonconstant dimension, since the number of simplices per cubelet is exponential in the dimension.

Observation 7. After iteration i, we have a $(i + 3)$-dimensional bmf in which c_i becomes the "background color" corresponding to color 0 in the original three-dimensional instance.

The cubelets having colors $\{0, 1, 2, 3, c_1, \ldots, c_{i-1}\}$ are mapped to cubelets in the $(i + 3)$-dimensional instance in such a way as to have the same neighborhood structure, but with some duplication at the folds of the embedding.

The continuous implementation. Define $\mathcal{F}_{B'} : K^n \longrightarrow K^n$ as follows. If x lies at the center of a cubelet (of length 1/8), letting $j = f_{B'}(x)$, $\mathcal{F}_{B'}(x) = x + \delta_j$, where $\delta_j = (-\alpha, -\alpha, \ldots, -\alpha)$ if j is the background color c_s, otherwise $\delta_j = (0, 0, \ldots, 0, \alpha, 0, 0, \ldots, 0)$ where the position of the nonzero entry depends on the color j, and is chosen to satisfy the boundary conditions of a bmf.

If x does not lie at the center of a cubelet, we claim that for each color j, we can efficiently compute the L_∞ distance from x to the closest center of a cubelet having color j, using a linear arithmetic circuit. Let $d_j(x) \in [0, 1]$ be this distance. Let $\lambda_j(x) = \max(0, (\frac{1}{10} - d_j(x)))$. Then define $\mathcal{F}_{B'}(x) = x + \sum_j \delta_j.\lambda_j(x)$. In that expression for $\mathcal{F}_{B'}(x)$, δ_j is a constant vector, so we are not multiplying two computed quantities together (which is disallowed in a linear arithmetic circuit).

Why it works. If we are not within distance $\frac{1}{10}$ of a panchromatic vertex, then $\lambda_j = 0$ where j is one of the missing colors. However, there is some j' for which $\lambda_{j'} > \frac{1}{16}$. The choice of the vectors δ_j ensures that $|\mathcal{F}_{B'}(x) - x| \geq \frac{1}{16}$.

Consequently any approximate fixpoint of $\mathcal{F}_{B'}$ is close to a panchromatic vertex of $f_{B'}$. To show that it is close enough to permit that panchromatic vertex to be efficiently reconstructed from the coordinates of the fixpoint, it is easiest to assume that there are 24 rather than 8 cubelets along each axis, with the original cubelets having been divided into 27 smaller ones all having the same color. Then an approximate fixpoint can be assumed to lie within $\frac{1}{30}$ of a panchromatic vertex.

We also need to point out that the high-dimensional bmf $f_{B'}$ has a path of $\{0, 1, 2, 3, c_1, \ldots, c_{s-1}\}$-chromatic cubelets which simulates the path of $\{1, 2, 3\}$-chromatic cubelets in the bmf f_B. Thus, the one obtained by following the path in $f_{B'}$, encodes the one obtained by following the corresponding path in f_B.

By way of a final remark, it is necessary for us to make a snake embedding of our 3D graph into higher dimension, rather than (as in Chen et al. [2009]) reduce from the 2D version of the problem. This is because the PPAD-completeness of 2D SPERNER [Chen and Deng 2009] is a reduction that alters the structure of the END OF THE LINE graph being encoded, and so would not (in an obvious way) apply in a reduction from OEOTL.

ACKNOWLEDGMENTS

We thank Andy McLennan and two anonymous referees for providing useful comments on an earlier version.

REFERENCES

Balasko, Y. 1978. Economic equilibrium and catastrophe theory: An introduction. *Econometrica 46,* 3, 557–569.

Browder, F. E. 1960. On continuity of fixed points under deformations of continuous mappings. *Summa Brasil. Mat. 4*, 183–191.

Chen, X. and Deng, X. 2009. On the complexity of 2D discrete fixed point problem. *Theoret. Comput. Sci. 410,* 44, 4448–4456.

Chen, X., Deng, X., and Teng, S.-H. 2009. Settling the complexity of computing two-player nash equilibria. *J. ACM 56,* 3, 57.

Daskalakis, C., Goldberg, P. W., and Papadimitriou, C. H. 2009. The complexity of computing a nash equilibrium. *SIAM J. Comput. 39,* 1, 195–259.

Eaves, B. C. 1972. Homotopies for computation of fixed points. *Math. Program. 3*, 1–22.

Etessami, K. and Yannakakis, M. 2010. On the complexity of nash equilibria and other fixed points. *SIAM J. Comput. 39,* 6, 2531–2597.

Fudenberg, D. and Levine, D. K. 1998. *The Theory of Learning in Games*. Economic Learning and Social Evolution Series, vol. 2, MIT Press, Cambridge, MA.

Harsanyi, J. C. 1975. The tracing procedure: A bayesian approach to defining a solution for n-person noncooperative games. *Int. J. Game Theory 4,* 1–2, 61–94.

Harsanyi, J. C. and Selten, R. 1988. *A General Theory of Equilibrium Selection in Games*. MIT Press, Cambridge, MA.

Herings, P. J.-J. and Peeters, R. 2010. Homotopy methods to compute equilibria in game theory. *Econ. Theory 42,* 1, 119–156.

Herings, P. J.-J. and Peeters, R. J. A. P. 2001. A differentiable homotopy to compute nash equilibria of n-person games. *Econ. Theory 18,* 1, 159–185.

Herings, P. J.-J. and van den Elzen, A. 2002. Computation of the nash equilibrium selected by the tracing procedure in n-person games. *Games Econ. Behav. 38,* 1, 89–117.

Hirsch, M. D., Papadimitriou, C. H., and Vavasis, S. A. 1989. Exponential lower bounds for finding Brouwer fixed points. *J. Complex. 5,* 4, 379–416.

Kearns, M. J., Littman, M. L., and Singh, S. P. 2001. Graphical models for game theory. In *Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence (UAI'01)*. Morgan Kaufmann, San Francisco, CA, 253–260.

Kreps, D. M. 1990. *A Course in Microeconomic Theory*. Princeton University Press.

Lemke, C. E. and Howson Jr., J. T. 1964. Equilibrium points of bimatrix games. *J. Soc. Indust. Appl. Math. 12*, 413–423.

Myerson, R. B. 1999. Nash equilibrium and the history of economic theory. *J. Econ. Literat. 37,* 3, 1067–1082.

Papadimitriou, C. H. 1994. On the complexity of the parity argument and other inefficient proofs of existence. *J. Comput. Syst. Sci. 48,* 3, 498–532.

Savani, R. and von Stengel, B. 2006. Hard-to-solve bimatrix games. *Econometrica 74,* 2, 397–429.

Scarf, H. 1973. *The Computation of Economic Equilibria*. Yale University Press, New Haven, CT.

von Stengel, B., van den Elzen, A., and Talman, D. 2002. Computing normal form perfect equilibria for extensive two-person games. *Econometrica 70,* 2, 693–715.

Received August 2011; revised December 2011; accepted January 2012