The Complexity of Nash Equilibria

Constantinos Daskalakis

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2008-107
http://www.eecs.berkeley.edu/Pubs/TechRpts/2008/EECS-2008-107.html

August 28, 2008

The Complexity of Nash Equilibria

by

Constantinos Daskalakis

Diploma (National Technical University of Athens) 2004

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Computer Science

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Christos H. Papadimitriou, Chair
Professor Alistair J. Sinclair
Professor Satish Rao
Professor Ilan Adler

Fall 2008

The dissertation of Constantinos Daskalakis is approved:

Chair ________________________________ Date __________

________________________________ Date __________

________________________________ Date __________

________________________________ Date __________

University of California, Berkeley

Fall 2008

The Complexity of Nash Equilibria

Abstract

The Complexity of Nash Equilibria

by

Constantinos Daskalakis

Doctor of Philosophy in Computer Science

University of California, Berkeley

Professor Christos H. Papadimitriou, Chair

The Internet owes much of its complexity to the large number of entities that run it and use it. These entities have different and potentially conflicting interests, so their interactions are strategic in nature. Therefore, to understand these interactions, concepts from Economics and, most importantly, Game Theory are necessary. An important such concept is the notion of *Nash equilibrium*, which provides us with a rigorous way of predicting the behavior of strategic agents in situations of conflict. But the credibility of the Nash equilibrium as a framework for behavior-prediction depends on whether such equilibria are efficiently computable. After all, why should we expect a group of rational agents to behave in a fashion that requires exponential time to be computed? Motivated by this question, we study the *computational complexity of the Nash equilibrium*.

We show that computing a Nash equilibrium is an intractable problem. Since by Nash's theorem a Nash equilibrium always exists, the problem belongs to the family of total search problems in NP, and previous work establishes that it is unlikely that such problems are NP-complete. We show instead that the problem is as hard as solving any Brouwer fixed point computation problem, in a precise complexity-theoretic sense. The corresponding complexity class is called PPAD, for Polynomial Parity Argument in Directed graphs, and our precise result is that computing a Nash

equilibrium is a PPAD-complete problem.

In view of this hardness result, we are motivated to study the complexity of computing approximate Nash equilibria, with arbitrarily close approximation. In this regard, we consider a very natural and important class of games, called *anonymous games*. These are games in which every player is oblivious to the identities of the other players; examples arise in auction settings, congestion games, and social interactions. We give a polynomial time approximation scheme for anonymous games with a bounded number of strategies.

Professor Christos H. Papadimitriou
Dissertation Committee Chair

Acknowledgments

Christos once told me that I should think of my Ph.D. research as a walk through a field of exotic flowers. “You should not focus on the finish line, but enjoy the journey. And, in the end, you’ll have pollen from all sorts of different flowers on your clothes.” I want to thank Christos for guiding me through this journey and everyone else who contributed in making these past four years a wonderful experience.

I first met Christos in Crete, the island in Greece where my family comes from. Christos was giving a set of talks on Algorithmic Game Theory as part of the Onassis Foundation science lecture series on the Internet and the Web. I walked into the amphitheater a bit late, and the first thing I saw was a slide depicting the Internet as a cloud connecting a dozen of computers. This cloud started growing, and, as it grew, it devoured the computers and broke out of the boundaries of the screen. Then, a large question-mark appeared. In the next couple of slides Christos explained Game Theory and the concept of the Nash equilibrium as a framework for studying the Internet. I had no idea at that moment that this would be the motivation for my Ph.D. research. . .

I am indebted to Christos for mentoring me through this journey, and for teaching me how to look at the essence of things. His belief in me, his enthusiasm, and his support were essential for me to realize my potential. But I thank him even more for being there as a friend, for all our discussions over wine at Cesar, and for our cups of coffee at Nefeli Café. I only wish that I will be to my students as great of an advisor and friend as Christos has been to me.

I also want to thank all my other teachers at Berkeley, above all, the members of the theory group for the unique, friendly, and stimulating atmosphere they create. I thank Dick Karp for being a valuable collaborator and advisor and for guiding me through my first experience as a teacher. I will always admire and be inspired by his remarkable profoundness and simplicity. I thank Satish Rao for his support through the difficult time of choosing a research identity, for stimulating political discussions, and for our amusing basketball and soccer games. I thank Alistair Sinclair for his randomness and computation class, and his helpful guidance and advice. I thank Luca Trevisan for the complexity theory that he taught me and Umesh Vazirani for his unique character and spirit. Thanks to Ilan Adler for serving in my dissertation committee and to Tandy Warnow for involving me in CIPRES. Finally, many thanks are due to Elchanan Mossel, who has been a valuable collaborator and a great influence in turning my research interests towards the applications of Probability Theory in Computer Science and Biology. I thank him for the hours we spent thinking together, his inexhaustible enthusiasm, and his support.

I feel indebted to all my collaborators: Christos, Dick, Elchanan, Satish, Christian Borgs, Jennifer Chayes, Kamalika Chaudhuri, Alexandros Dimakis, Alex Fabrikant, Paul Goldberg, Cameron Hill, Alex Jaffe, Robert Kleinberg, Henry Lin, Aranyak Mehta, Radu Mihaescu, Samantha Riesenfeld, Sebastien Roch, Grant Schoenebeck, Gregory and Paul Valiant, Elad Verbin, Martin J. Wainwright, Dimitris Achlioptas, Sanjeev Arora, Arash Asadpour, Albert Atserias, Omid Etesami, Jason Hartline, Nicole Immorlica, Elias Koutsoupias, David Steurer, Shanghua Teng, Adrian Vetta, and Riccardo Zecchina. Thanks for the hours we spent thinking together, and the unique things I learned from each of you!

Thanks also to all the students in Berkeley's Computer Science, Electrical Engineering and Statistics departments for four wonderful years. Special thanks to Alex Dimakis, Omid Etesami, Slav Petrov, Sam Riesenfeld, Grant Schoenebeck, Kunal Talwar, Andrej Bogdanov, Alex Fabrikant, Simone Gambini, Brighten Godfrey, Alexandra Kolla, James Lee, Henry Lin, Mani Narayanan, Lorenzo Orrechia, Boriska Toth and Madhur Tulsiani. Thanks to Alex, Omid, Sam and Slav for being valuable friends. Thanks to Grant for a great China trip. Thanks to Alex, Omid and Slav for a wonderful road-trip in Oregon. And thanks to Alexandra and Slav for an amazing Costa Rica trip. Also, thanks to all students in the theory group for our Lake Tahoe retreat.

Outside of Berkley I feel very grateful towards Elias Koutsoupias, Timos Sellis and Stathis Zachos. I am grateful to Stathis for introducing me to theoretical computer science during my undergraduate studies and for turning my research interests towards the theory of computation. I thank Timos for being a wonderful advisor and friend back in my undergraduate years and Elias for being a good friend and great influence throughout my Ph.D. years.

I want to thank UC Berkeley, NSF, CIPRES, Microsoft and Yahoo! for supporting my research. I especially thank UC Regents for the fellowship I received my first year and Microsoft Research for the Graduate Research Fellowship I received my last year. I also thank Microsoft Research for two productive summer internships. Special thanks are due to my mentors Jennifer Chayes and Christian Borgs for being extremely supportive during my time as an intern and for their continuing appreciation and support.

I also feel deeply honored by the Game Theory and Computer Science prize that the Game Theory Society awarded to my collaborators and me for our research on the Complexity of Nash equilibria. Their appreciation of our research is extremely gratifying.

I want to thank the Greeks of the Bay Area for their friendship and support. Special thanks to Maria-Daphne, Constantinos, Manolis, Lefteris, Alexandros, Alexandra, George, Dimitris, Stergios, Christos, Tasos, Apostolis, Katerina, Eleni, Antonis and Antonis. Maria-Daphne, Lefteri and Alexandre, thanks for a wonderful Mexico trip. Costa, Manoli and Antoni, I will never forget our trip to Hawaii.

Thanks also to Ilias, Natassa, Maria-Daphne, Aliki, Nikos, Nicole, Jason, Alisar and Alexandra C. I am grateful to each of you for being in my life, for all your love and support.

Thanks to Berkeley itself for its magical ambience. I am grateful for all the moments I spent at Café Strada, Nefeli Café, Brewed Awakening, Cesar, the i-House, and the French Hotel Café. I also cannot forget the beautiful sunsets at the Theology School between Le Conte and Arch Streets.

Last but not least, I am grateful to my family. I have no way to thank them for their invaluable presence in my life, my memories, my whole essence, but to dedicate this dissertation to them.

To my parents Pericles and Sophia for inspiring my love for knowledge,

and to my brother Nikos for sharing this journey with me.

Contents

List of Figures

Chapter 1

Introduction

Do we understand the Internet? One possible response to this question is "Of course we do, since it is an engineered system". Indeed, at the very least, we do understand the design of its basic components and the very basic processes running on them. On the other hand, we are often *surprised* by singular events that occur on the Internet: in February 2008, for example, a mere Border Gateway Protocol (BGP) table update in a network in Pakistan resulted in a two-hour outage of YouTube accesses throughout the globe...

What we certainly understand is that the Internet is a remarkably complex system. And it owes much of its complexity to the large number of entities that run it and use it, through such familiar applications as routing, file sharing, online advertising, and social networking. These interactions occurring in the Internet, much like those happening in social and biological systems, are often strategic in nature, since the participating entities have different and potentially conflicting interests. Hence, to understand the Internet, it makes sense to use concepts and ideas from Economics and, most importantly, Game Theory.

One of Game Theory's most basic and influential concepts, which provides us with a rigorous way of describing the behaviors that may arise in a system of interacting agents, is the concept of the *Nash equilibrium*. And this dissertation is devoted to the study of the *computational complexity of the Nash equilibrium*. But why consider computational complexity? First, it *is* a very natural and useful question to answer. Second, because of the computational nature of the motivating application, it is natural to study the computational aspects of the concepts we introduce for its study. But the main justification for this question is philosophical: Equilibria are models of behavior of rational agents, and, as such, they should be efficiently computable. After all, it is doubtful that groups of rational agents are computationally more powerful than computers; and, if they were, it would be really remarkable. Hence, whether equilibria are efficiently computable is a question of fundamental significance for Game Theory, the field for which equilibrium is perhaps the most central concept.

1.1 Games and the Theory of Games

Game Theory is one of the most important and vibrant mathematical fields established in the 20th century. It studies the behavior of strategic agents in situations of conflict, called *games*; these, e.g., include markets, transportation networks, and the Internet.

- *But how is a game modeled mathematically?*

A game can be described by naming its players and specifying the strategies available to them. Then, for every selection of strategies by the players, each of them gets some (potentially negative) utility, called *payoff*. The payoffs can be given implicitly as functions of the players' strategies; or, if the number of strategies is finite, they can be given explicitly by tables.

For example, Figure 1.1 depicts a variant of the *Chicken Game* [OR94], called the *Railroad Crossing Game*: A car and a train approach an unprotected railroad crossing at collision speed. If both the car driver and the train operator choose to stop, or "chicken", then both of them lose time and fuel; if one of them stops, and the other goes, or "dares", the latter is happier than if he had "chickened"; but, if both of them decide to go, the car gets destroyed, and the train has severe damages. The table representation of the game given in Figure 1.1 assigns numerical payoffs to

		train's strategies	
		chicken	dare
car's strategies	chicken	-1, -10	-1, 10
	dare	1,-10	-10000,-100

Figure 1.1: The Railroad Crossing Game

the different outcomes of the game; in every box of the table the first payoff value corresponds to the car driver and the second to the train operator. The following question arises.

- *What should we expect the behavior of the players of a game to be?*

In the Railroad Crossing Game, it is reasonable to expect that not both the car driver and the train operator will "dare": in a world in which train operators always "dare", it is in the best interest of car drivers to always "chicken"; if the car drivers always "dare", then the train operators should always "chicken". Similarly, it is not reasonable to expect that they will both "chicken"; because it would then be in the best interest of either party to switch to the "dare" strategy. The following outcomes are, however, plausible: the car drivers "dare" and the train operators "chicken", *or* the train operators "dare" and the car drivers "chicken"; in any of these outcomes, neither player can improve her payoff by changing her strategy. In actual unprotected railroad crossings, the second outcome is what normally happens. Incidentally, this outcome also maximizes the social welfare, that is, the sum of players' payoffs.

The plausible outcomes of the Railroad Crossing Game discussed above are instances of an important equilibrium concept, called *pure Nash equilibrium*. This is defined as any collection of strategies, with one strategy per player of the game, such that, given the strategies of the other players, none of them can improve their payoff by switching to a different strategy. Hence, it is reasonable for every player to stick to the strategy prescribed to her.

To understand the pure Nash equilibrium as a concept of behavior prediction, let us adopt the following interpretation of a game, called the *steady state interpretation*: [1] We view a game as a model designed to explain some regularity observed in a *family* of similar situations. A player of the game forms her expectation about the other players' behavior on the basis of the information about how the game or a similar game was played in the past. That is, every player "knows" the equilibrium of the game that she is about to play and only tests the optimality of her behavior given this knowledge; and the pure Nash equilibrium specifies exactly the conditions that need to hold so that she does not need to adopt a different behavior. Observe that the steady state interpretation is what we used to argue about the equilibria of the Railroad Crossing Game: we viewed the game as a model of the interaction between two populations, the train operators and the car drivers, and each instance of the game took place when two members of these populations met at a railroad crossing. It is important to note that the pure Nash equilibrium is a convincing method of behavior-prediction only in the absence of strategic links between the different plays of the game. If there are inter-temporal strategic links between occurrences of the game, different equilibrium concepts are necessary.

The pure Nash equilibrium is a simple and convincing equilibrium concept. Alas, it does not exist in every game. Let us consider, for example, the *Penalty Shot Game* described in Figure 1.2. The numerical values in the table specify the following rules: if the goalkeeper and the penalty kicker choose the same strategy, then the goalkeeper wins a point, and the penalty kicker loses a point; if they choose different strategies, then the goalie loses, and the penalty kicker wins. Observe that there is no pure Nash

[1] See, e.g., Osborne and Rubinstein [OR94] for a more detailed discussion of the subject.

		penalty kicker's strategies	
		left	right
goalkeeper's strategies	left	1,-1	-1, 1
	right	-1, 1	1,-1

Figure 1.2: The Penalty Shot Game

equilibrium in this game.

- *In the absence of a pure Nash equilibrium, what behavior should we expect from the players of a game?*

Here is a suggestion: let us assume that the players of the game may choose to randomize by selecting a probability distribution over their strategies, called a *mixed strategy*. We will discuss shortly the meaning of randomization for a decision maker. Before that, let us revisit the penalty shot game given in Figure 1.2. Suppose that the goalkeeper chooses to randomize uniformly over 'left' and 'right', and so does the penalty kicker. Suppose also that the two players have information about each other's mixed strategies. If this is the case, then none of them would be able to increase their *expected payoff* by switching to a different mixed strategy, so they might as well keep their strategy.

The pair of uniform strategies for the Penalty Shot Game is an instance of an important equilibrium concept, called *mixed Nash equilibrium*, or simply *Nash equilibrium*. Formally, this is defined as a collection of mixed strategies, one for every player of the game, such that none of the players can improve their expected payoff by switching to a different mixed strategy; hence, it is reasonable for every player to stick to the mixed strategy prescribed to her. The plausibility of the concept of the Nash equilibrium depends, of course, on the answer to the following question.

- *What does it mean for decision makers to randomize?*

This question could be the beginning of a long and interesting discussion — see, e.g., Osborne and Rubinstein [OR94] for a detailed analysis. So, we only attempt an explanation here. To do this we revisit the steady state interpretation of a game, according to which a game models an environment in which players act repeatedly and ignore strategic links between different plays. By the same token, we can interpret the Nash equilibrium as a stochastic steady state as follows: Each player of the game collects statistical data about the frequencies with which different actions were taken in the previous plays of the game. And she chooses an action according to the beliefs she forms about the other players' strategies from these statistics. The Nash equilibrium then describes the frequencies with which different actions are played by the players of the game in the long run. Coming back to the Penalty Shot Game, it is reasonable to expect that in half of the penalty shots played in this year's EuroCup the penalty kicker shot right and in half of them the goalkeeper dived left.

The pure Nash equilibrium is of course more attractive than the mixed Nash equilibrium, since it does not require the players to randomize. However, as noted above, it does not exist in every game, and this makes its value as a framework for behavior prediction rather questionable. For the same reason, the usefulness and plausibility of the mixed Nash equilibrium is contingent upon a positive answer to the following question.

- *Is there a mixed Nash equilibrium in every game?*

1.2 The History of the Nash Equilibrium

In 1928, John von Neumann, extending work by Emile Borel, showed that any two-player zero-sum game — that is, a game in which every outcome has zero payoff-sum, such as the Penalty Shot Game of Figure 1.2 — has a mixed equilibrium [Neu28]. Two decades after von Neumann's result it was understood that the existence of an equilibrium in zero-sum games is equivalent to Linear Programming duality [AR86, Dan63], and, as was established another three decades later [Kha79], finding such an equilibrium is computationally tractable. In other words, computationally speaking, the state of affairs of equilibria in zero-sum games is quite satisfactory.

However, as it became clear with the seminal book by von Neumann and Morgenstern [NM44], zero-sum games are too specialized and fail to capture most interesting situations of conflict between rational strategic players. Hence, the following question became important.

- *Is there a Nash equilibrium in non-zero-sum multi-player games?*

The answer to this question came in 1951 with John Nash's important and deeply influential result: every game, independent of the number of players and strategies available to them (provided only that these numbers are finite) and of the properties of the players' payoffs, has an equilibrium in randomized strategies, henceforth called a Nash equilibrium [Nas51]. Nash's proof, based on Brouwer's fixed point theorem [KKM29] is mathematically beautiful, but non-constructive. Even the more recent combinatorial proofs of Brouwer's fixed point theorem based on Sperner's lemma (see, e.g., Papadimitriou [Pap94b]) result in exponential time algorithms. Due to the importance of the Nash equilibrium concept, soon after Nash's result the following question emerged.

- *Are there efficient algorithms for computing a Nash equilibrium?*

We will consider this question in the centralized model of computation. Of course, the computations performed by strategic agents during game-play are modeled more faithfully by distributed protocols; and these protocols should be of a very special kind, since they correspond to rational behavior of strategic agents. [2] Hence, it is not clear a priori whether an efficient centralized algorithm for computing a Nash equilibrium would imply a natural and efficient distributed protocol for the same

[2] The reader is referred to Fudenberg and Levine [FL99] for an extensive discussion of natural protocols for game-play.

task. However, it *is* true — and will be of central importance for the philosophical implications of our results discussed in Section 1.3 — that an intractability result for centralized algorithms implies a similar result for distributed algorithms. After all, the computational parallelism, inherent in the interaction of players during game-play, can only result in polynomial-time speedups.

Whether Nash equilibria can be computed efficiently has been studied extensively in the Economics and Optimization literature. At least for the two-player case, the hope for a positive answer was supported by a remarkable similarity of the problem to linear programming: there always exist rational solutions, and the Lemke-Howson algorithm [LH64], a simplex-like technique for solving two-player games, appears to be very efficient in practice. There *are* generalizations of the Lemke-Howson algorithm applying to the multi-player case [Ros71, Wil71]; however, as noted by Nash in his original paper [Nas51], there are three-player games with only irrational equilibria. This gives rise to the following question.

- *What does it mean to compute a Nash equilibrium in the presence of irrational equilibria?*

There are two obvious ways to define the problem: One is to ask for a collection of mixed strategies within a specified distance from a Nash equilibrium. And the other is to ask for mixed strategies such that no player has more than some (small) specified incentive to change her strategy; that is, a collection of mixed strategies such that every player is playing an approximate best response to the other players' strategies. The latter notion of approximation is arguably more natural for applications (since the players' goal in a game is to optimize their payoffs rather than the distance of their strategies from an equilibrium strategy), and we are going to adopt this notion in this dissertation. This is also the standard notion used in the literature of algorithms for equilibria, e.g., those based on the computation of fixed points [Sca67, Eav72, GLL73, LT79]. For the former notion of approximation, the reader is referred to the recent work of Etessami and Yannakakis [EY07].

Despite extensive research on the subject, none of the existing algorithms for computing Nash equilibria are known to be efficient. There are instead negative results [HPV89], most notably for the Lemke-Howson algorithm [SS04]. This brings about the following question.

- *Is computing a Nash equilibrium an inherently hard computational problem?*

Besides Game Theory, the 20th century saw the development of another great mathematical field of tremendous growth and impact, whose concepts enable us to answer questions of this sort: Computational Complexity. However, the mainstream concepts and techniques developed by complexity theorists — chief among them NP-completeness — are not directly applicable for fathoming the complexity of the Nash equilibrium. There *are* versions of the problem which are NP-complete, for example counting the number of equilibria, or deciding whether there are equilibria with certain properties [GZ89, CS03]. But answering these questions appears computationally harder than finding a (single) Nash equilibrium. So, it seems quite plausible that the Nash equilibrium problem could be easier than an NP-complete problem.

The heart of the complication in characterizing the complexity of the Nash equilibrium is ironically Nash's Theorem: NP-complete problems seem to draw much of their difficulty from the possibility that a solution may not exist; and, since a Nash equilibrium is always guaranteed to exist, NP-completeness does not seem useful in characterizing the complexity of finding one. What would a reduction from SATISFIABILITY to NASH (the problem of finding a Nash equilibrium) look like? Any obvious attempt to define such a reduction quickly leads to NP = coNP [MP91]. Hence, the following question arises.

- *If not NP-hard, exactly how hard is it to compute a Nash equilibrium?*

Motivated mainly by this question for the Nash equilibrium, Meggido and Papadimitriou [MP91] defined in the 1980s the complexity class TFNP (for "NP total functions"), consisting exactly of all search problems in NP for which every instance is guaranteed to have a solution. NASH of course belongs there, and so do many other important and natural problems, finitary versions of Brouwer's problem included. But here there is a difficulty of a different sort: TFNP is a "semantic class" [Pap94a], meaning that there is no easy way of recognizing nondeterministic Turing machines which define problems in TFNP — in fact the problem is undecidable; such classes are known to be devoid of complete problems.

To capture the complexity of NASH and other important problems in TFNP, another step is needed: One has to group together into subclasses of TFNP total functions whose proofs of totality are similar. Most of these proofs work by essentially constructing an exponentially large graph on the solution space (with edges that are computed by some algorithm), and then applying a simple graph-theoretic lemma establishing the existence of a particular kind of node. The node whose existence is guaranteed by the lemma is the desired solution of the given instance. Interestingly, essentially all known problems in TFNP can be shown total by one of the following arguments:

- *In any dag there must be a sink.* The corresponding class, PLS for "Polynomial Local Search", had already been defined in [JPY88] and contains many important complete problems.

- *In any directed graph with outdegree one and with one node with indegree zero, there must be a node with indegree at least two.* The corresponding class is PPP (for "Polynomial Pigeonhole Principle").

- *In any undirected graph with one odd-degree node, there must be another odd-degree node.* This defines a class called PPA for "Polynomial Parity Argument" [Pap94b], containing many important combinatorial problems (unfortunately none of them known to be complete).

- *In any directed graph with one unbalanced node (node with outdegree different from its indegree), there must be another unbalanced node.* The corresponding class is called PPAD for "Polynomial Parity Argument for Directed graphs," and it contains NASH, BROUWER, and BORSUK-ULAM (finding approximate fixed points of the kind guaranteed by Brouwer's Theorem and the Borsuk-Ulam Theorem, respectively, see [Pap94b]). The latter two were among the problems proven PPAD-complete in [Pap94b]. Unfortunately, NASH — the one problem which had motivated this line of research — was not shown PPAD-complete; it was conjectured that it is.

The central question arising from this line of research, and the starting point of this dissertation, is the following.

- *Is computing a Nash equilibrium PPAD-complete?*

1.3 Overview of Results

Our main result is that NASH*, the problem of computing a Nash equilibrium, is PPAD-complete.* Hence, we settle the questions about the computational complexity of the Nash equilibrium problem discussed in Section 1.2.

The proof of our main result is presented in Chapter 4. Our original argument (Section 4.1) works for games with three players or more, leaving open the question for two-player games. This case was thought to be computationally easier, since, as discussed in Section 1.2, linear programming-like techniques come into play, and solutions consisting of rational numbers are guaranteed to exist [LH64]; on the contrary, as exhibited in Nash's original paper [Nas51], there are three-player games with only irrational equilibria. Surprisingly, a few months after our result was circulated, Chen and Deng extended our hardness result to the two-player case [CD05, CD06]. In Section 4.2, we present a simple modification of our argument which also establishes the hardness of two-player games.

- *So, what is the implication of our PPAD-hardness result for Nash equilibria?*

First of all, a polynomial-time algorithm for computing Nash equilibria would imply a polynomial-time algorithm for computing Brouwer fixed points of (succinctly described) continuous and piece-wise linear functions, a problem for which quite strong lower bounds for large classes of algorithms are known [HPV89]. Moreover, there are oracles — that is, computational universes [Pap94a] — relative to which PPAD is different from P [BCE+98]. Hence, a polynomial-time algorithm for computing Nash equilibria would have to fail to relativize with respect to these oracles, which seems unlikely.

Our result gives an affirmative answer to another important question arising from Nash's Theorem, namely, whether the reliance of its proof on Brouwer's fixed point theorem is inherent. Our proof is essentially a reduction in the opposite direction to Nash's: an appropriately discretized and stylized PPAD-complete version of Brouwer's fixed point problem in 3 dimensions is reduced to Nash.

In fact, it is possible to eliminate the computational ingredient in this reduction to obtain a purely mathematical statement, establishing the equivalence between the existence of a Nash equilibrium in 2- and 3-player games and the existence of fixed points in continuous piecewise-linear and polynomial maps respectively. This important point is discussed briefly in Section 4.3 and explored in detail by Etessami and Yannakakis [EY07]. Mainly due to this realization, we have been able to show that a large class of equilibrium-computation problems belongs to the class PPAD; in particular, we can show this for all games for which, loosely speaking, the expected utility of a player can be computed by an *arithmetic circuit*[3] given the other players' mixed strategies [DFP06]. In the same spirit, Etessami and Yannakakis [EY07] relate the computation of Nash equilibria to computational problems, such as the square-root-sum problem (see, e.g., [GGJ76, Pap77]) and the value of simple stochastic games [Con92], the complexity of which is largely unknown.

But perhaps the most important implication of our result is a critique of the Nash equilibrium as a framework of behavior prediction — contingent, of course, upon the hardness of the class PPAD: Should we expect that the players of a game behave in a fashion which is too expensive computationally? Or, relative also to the steady state interpretation of a game, is it interesting to study a notion of player behavior which could only arise after a prohibitively large number of game-plays? In view of these objections, the following question becomes important.

- *In the absence of efficient algorithms for computing a Nash equilibrium, are there efficient algorithms for computing an approximate Nash equilibrium?*

As discussed in the previous section, we are interested in collections of mixed strategies such that no player has more than some small, say ϵ, incentive to change her strategy. Let us call such a collection an *ϵ-approximate Nash equilibrium*. From our result on the hardness of computing a Nash equilibrium, it follows that, if ϵ is inverse exponential in the size of the game, computing an ϵ-approximate Nash equilibrium is PPAD-complete. In fact, this hardness result was extended to the case where ϵ is inverse polynomial in the size of the game by Chen, Deng and Teng [CDT06a]. Hence, a fully polynomial-time approximation scheme seems unlikely. [4] The following question then emerges at the boundary of intractability.

[3]Arithmetic Circuits are analogous to Boolean Circuits, but instead of the Boolean operators $\wedge, \vee, \neg$, they use the arithmetic operators $+, -, \times$.

[4]A *polynomial-time approximation scheme*, or PTAS, is a family of approximation algorithms, running in time polynomial in the problem size, for every fixed value of the approximation ϵ. If the running time is also polynomial in $1/\epsilon$, the family is called a *fully polynomial-time approximation scheme*, or FPTAS.

- *Is there a polynomial-time approximation scheme for the Nash equilibrium problem? And, in any case, what would the existence of such a PTAS imply for the predictive power of the Nash equilibrium?*

In view of our hardness result for the Nash equilibrium problem, a PTAS would be rather important, since it would support the following interpretation of the Nash equilibrium as a framework for behavior prediction: Although it might take a long time to approach an exact Nash equilibrium, the game-play could converge — after a polynomial number of iterations — to a state where all players' regret is no more than ϵ, for any desired ϵ. If that ϵ is smaller than the numerical error (e.g., the quantization of the currency used by the players), then ϵ-regret might not even be visible to the players.

There has been a significant body of research devoted to the computation of approximate Nash equilibria [LMM03, KPS06, DMP06, FNS07, DMP07, BBM07, TS07], however no PTAS is known to date. We discuss the known results in detail in Chapter 5. Let us only note here that, even for the case of 2-player games, we only know how to efficiently compute ϵ-approximate Nash equilibria for finite values of ϵ.

Motivated by this challenge we consider an important class of multi-player games, called *anonymous games*. These are games in which each player's payoff depends on the strategy that she chooses and only the *number of other players* choosing each of the available strategies. That is, the payoff of a player does not differentiate among the identities of the other players. As an example, let us consider the decision faced by a driver when choosing a route between two towns: the travel time to her destination depends on her own choice of a route and the routes chosen by the other drivers, but not on the identities of these drivers. Anonymous games capture important aspects of congestion, as well as auctions and markets, and comprise a broad and well-studied class of games (see, e.g., [Mil96, Blo99, Blo05, Kal05] for recent work on the subject by economists).

In Chapter 5, we present a polynomial-time approximation scheme for anonymous games with many players and a constant number of strategies per player. Our algorithm, reported in [DP07, DP08a], extends to several generalizations of anonymous games, for example the case in which there are a few *types* of players, and the utilities depend on how many players of each type play each strategy; and to the case in which we have *extended families* (disjoint subsets of up to logarithmically many players, each with a utility depending in arbitrary — possibly non-anonymous — ways on the other members of the family, in addition to their anonymous — possibly typed — interest on everybody else).

Our PTAS for anonymous games shows that this broad and important class of games is free of the complications posed by our PPAD-completeness result. Moreover, it could be the precursor of practical algorithms for the problem; after all, it is not known if the Nash equilibrium problem for multi-player anonymous games with a constant number of strategies is PPAD-complete. [5] Towards this goal, in recent work we have developed an efficient PTAS[6] for the case of two-strategy anonymous games [Das08]; we discuss this algorithm in Section 5.9.

1.4 Discussion of Techniques

The starting point for our PPAD-completeness result is the definition of a PPAD-complete version of the computational problem related to the Brouwer fixed point theorem. The resulting problem, described in Section 2.4, asks for the computation of fixed points of continuous and piecewise-linear maps of a very special kind from the unit 3-dimensional cube to itself. These maps are described by circuits, which compute their values at the points of the discrete 3-dimensional grid of a certain size;

[5] On the other hand, the Nash equilibrium problem in anonymous games with a constant number of players is PPAD-complete (see Chapter 5).

[6] An efficient PTAS is a PTAS with running time whose dependence on the problem size is a polynomial of degree independent of the approximation ϵ.

the values at the other points of the cube are then specified by interpolation. An obvious algorithm for finding fixed points of such maps would be to enumerate over all cubelets defined by the discrete grid and check if there is a fixed point within each cubelet. However, the number of cubelets may be exponential in the size of the circuit which could make the problem more challenging. We call this problem BROUWER and show that it is PPAD-complete. (See Section 2.4 for details.)

The next step in our reduction would be to reduce BROUWER to NASH. Indeed, this is what we do, but, instead of reducing it to NASH for games with a constant number of players, we reduce it to NASH for a class of multi-player games with sparse player interactions, called *graphical games* [KLS01]: these are specified by giving a graph of player interactions, so that the payoff of a player depends only on her strategy and the strategies of her neighbors in the graph. We define graphical games formally in Chapter 2 and present our reduction from BROUWER to NASH for graphical games in Chapter 4.

The reduction goes roughly as follows. We represent a point in the three-dimensional unit cube by three players each of which has two strategies. Thus, every combination of mixed strategies for these players corresponds naturally to a point in the cube. Now, suppose we are given a function from the cube to itself represented by circuit. We construct a graphical game in which the best responses of the three players representing a point in the cube implement the given function, so that the Nash equilibria of the game must correspond to Brouwer fixed points. This is done by decoding the coordinates of the point in order to find the binary representation of the grid-points that surround it. Then the value of the function at these grid-points is computed by simulating the circuit computations with a graphical game. This part of the construction relies on certain "gadgets," small graphical games acting as arithmetical gates and comparators. The graphical game thus "computes" (in the sense of a mixed strategy over two strategies representing a real number) the values of the function at the grid points surrounding the point represented by the mixed strategies of the original three players; these values are used to compute (by interpolation) the value of the function at the original point, and the three players are then induced to add appropriate increments to their mixed strategy to shift to that value. This establishes a one-to-one correspondence between the fixed points of the given function and the Nash equilibria of the constructed graphical game, and it shows that Nash for graphical games is PPAD-complete.

One difficulty in this part of the reduction is related to *brittle comparators*. Our comparator gadget sets its output to 0 if the input players play mixed strategies x, y that satisfy $x < y$, to 1 if $x > y$, *and to anything if* $x = y$; moreover, it is not hard to see that no "robust" comparator gadget is possible, one that outputs a specific fixed value if the input is $x = y$. This in turn implies that no robust decoder from real to binary can be constructed; decoding will always be flaky for a non-empty subset of the unit cube, and, on that set, arbitrary values can be output by the decoder. On the other hand, real to binary decoding would be really useful since the circuit representing the given Brouwer function should be simulated in binary arithmetic. We take care of this difficulty by computing the Brouwer function on a "microlattice" around the point of interest and averaging the results, thus smoothing out any effects from boundaries of measure zero.

To continue to our main result for three-player games, we establish certain reductions between equilibrium problems. In particular, we show by reductions that the following three problems are polynomial-time equivalent:

- Nash for r-player (normal-form) games, for any constant $r > 3$.
- Nash for three-player games.
- Nash for graphical games with two strategies per player and maximum degree three (that is, of the exact type used in the simulation of Brouwer functions

given above).

These reductions are presented in Chapter 3. It follows that all the above problems and their generalizations are PPAD-complete (since the third one was already shown to be PPAD-complete).

Our techniques for anonymous games are entirely different. The PTAS for approximate Nash equilibria follows from a deep understanding of the probabilistic nature of the decision faced by a player, given that her payoff does not differentiate among the identities of the other players: we show that, for any ϵ, there is an ϵ-approximate Nash equilibrium in which the players' mixed strategies only assign to the strategies in their support probability mass which is an integer multiple of $1/z$, for some z which depends polynomially in $1/\epsilon$ and exponentially in the number of strategies, but does not depend on the number of players. To appreciate this, note that for general (non-anonymous) games a linear dependency on the number of players would be necessary.

At the heart of our argument, e.g., for the case of two strategies per player, we need to approximate the sum of a set of independent Bernoulli random variables with the sum of another set of independent Bernoulli random variables, restricted to have means which are integer multiples of $1/z$, so that the two sums are within $\epsilon = O(1/z)$ in total variation distance. To achieve this we appeal to results from the literature on Poisson and Normal approximations [BC05], providing us with finitary versions of the Law of Rare Events and the Central Limit Theorem respectively. Using these results, we can approximate the sums of Bernoulli random variables with Poisson or Normal distributions, as appropriate, and compare those instead. See Chapter 5 for more details.

1.5 Organization of the Dissertation

In Chapter 2, we provide the required background of Game Theory, including the formal definition of games and of the concept of the Nash equilibrium, as well as of a few notions of approximate Nash equilibrium. We then review the complexity theory of total search problems, define the complexity class PPAD, and show that computing a Nash equilibrium is in that class. We also define the problem BROUWER, a canonical version of the Brouwer fixed point computation problem, which is PPAD-complete and will be the starting point for our main reduction in Chapter 4. We conclude with a survey of related work on computing Nash equilibria and Brouwer fixed points in general.

In Chapter 3, we present the game-gadget machinery needed for our main reduction and establish the computational equivalence of various Nash equilibrium computation problems. In particular, we describe a polynomial-time reduction from the problem of computing a Nash equilibrium in games of any constant number of players or in graphical games of any constant degree to that of computing a Nash equilibrium of a 3-player game.

In Chapter 4, we show our main result that computing a Nash equilibrium of a 3-player game is PPAD-hard. We also present a simple modification of our argument, extending the result to 2-player games. We conclude with extensions of our techniques to other classes of games, as well as to other fixed point computation problems.

In Chapter 5, we turn to the computation of approximate Nash equilibria and review recent work on the subject. This leads us to the introduction of the class of anonymous games, for which we develop a polynomial-time approximation scheme for the case of a bounded number of strategies. We conclude with extensions of our algorithm to broader classes of games and a discussion of more efficient polynomial-time approximation schemes.

The results of Chapters 3 and 4 are joint work with Paul Goldberg and Christos Papadimitriou [GP06, DGP06, DP05], and those of Chapter 5 are joint work with Christos Papadimitriou [DP07, DP08a, Das08].

Chapter 2

Background

We start with the required background of Game Theory, in Section 2.1. We define games, mixed strategies, and the concept of the Nash equilibrium, and we discuss various notions of approximate Nash equilibria. We also define graphical games, which play a central role in our results.

In Section 2.2, we turn to Computational Complexity. We review the complexity theory of total search problems and define the complexity class PPAD. We also define the problems NASH and GRAPHICAL NASH, corresponding to the problem of computing a Nash equilibrium in normal-form and graphical games respectively.

In Section 2.3, we show that NASH is in PPAD. Our reduction is motivated by previous work on simplicial approximation algorithms for Brouwer fixed points.

In Section 2.4, we define the problem BROUWER, a canonical version of the computational problem related to Brouwer's fixed point theorem for continuous and piecewise linear functions, which will be the starting point for showing that NASH is PPAD-hard in Chapter 4. Here, we show that BROUWER is PPAD-hard.

We conclude the chapter with a review of related work on computing Nash equilibria and other fixed points in Section 2.5.

2.1 Basic Definitions from Game Theory

A *game in normal form*, or *normal-form game*, has $r \geq 2$ players, $1, \ldots, r$, and for each player $p \leq r$ a finite set S_p of pure strategies. The set S of *pure strategy profiles* is the Cartesian product of the S_p's. We denote the set of pure strategy profiles of all players other than p by S_{-p}. Also, for a subset T of the players we denote by S_T the set of pure strategy profiles of the players in T. Finally, for each p and $s \in S$ we have a *payoff* or *utility* $u_s^p \geq 0$ — also occasionally denoted u_{js}^p for $j \in S_p$ and $s \in S_{-p}$. We refer to the set $\{u_s^p\}_{s \in S}$ as the *payoff table* of player p. If all payoffs lie in $[0,1]$ the game is called *normalized*. Also, for notational convenience and unless otherwise specified, we will denote by $[t]$ the set $\{1, \ldots, t\}$, for all $t \in \mathbb{N}$.

A *mixed strategy* for player p is a distribution on S_p, that is, real numbers $x_j^p \geq 0$ for each strategy $j \in S_p$ such that $\sum_{j \in S_p} x_j^p = 1$. A set of r mixed strategies $\{x_j^p\}_{j \in S_p}, p \in [r]$, is called a *(mixed) Nash equilibrium* if, for each p, $\sum_{s \in S} u_s^p x_s$ is maximized over all mixed strategies of p —where for a strategy profile $s = (s_1, \ldots, s_r) \in S$, we denote by x_s the product $x_{s_1}^1 \cdot x_{s_2}^2 \cdots x_{s_r}^r$. That is, a Nash equilibrium is a set of mixed strategies from which no player has a unilateral incentive to deviate. It is well-known (see, e.g., [OR94]) that the following is an equivalent condition for a set of mixed strategies to be a Nash equilibrium:

$$\forall p \in [r], j, j' \in S_p : \sum_{s \in S_{-p}} u_{js}^p x_s > \sum_{s \in S_{-p}} u_{j's}^p x_s \Longrightarrow x_{j'}^p = 0. \tag{2.1}$$

The summation $\sum_{s \in S_{-p}} u_{js}^p x_s$ in the above equation is the expected utility of player p if p plays pure strategy $j \in S_p$ and the other players use the mixed strategies $\{x_j^q\}_{j \in S_q}, q \neq p$. Nash's theorem [Nas51] asserts that *every normal-form game has a Nash equilibrium.*

We next turn to approximate notions of equilibrium. We say that a set of mixed strategies x is an *ϵ-approximately well supported Nash equilibrium*, or *ϵ-Nash equilibrium* for short, if the following holds:

$$\forall p \in [r], j, j' \in S_p : \sum_{s \in S_{-p}} u^p_{js} x_s > \sum_{s \in S_{-p}} u^p_{j's} x_s + \epsilon \Longrightarrow x^p_{j'} = 0. \tag{2.2}$$

Condition (2.2) relaxes (2.1) in that it allows a strategy to have positive probability in the presence of another strategy whose expected payoff is better by at most ϵ.

This is the notion of approximate Nash equilibrium that we use. There is an alternative, and arguably more natural, notion, called *ϵ-approximate Nash equilibrium* [LMM03], in which the expected utility of each player is required to be within ϵ of the optimum response to the other players' strategies. This notion is less restrictive than that of an approximately well supported one. More precisely, for any ϵ, an ϵ-Nash equilibrium is also an ϵ-approximate Nash equilibrium, whereas the opposite need not be true. Nevertheless, the following lemma, proved in Section 3.7, establishes that the two concepts are computationally related (a weaker version of this fact was pointed out in [CDT06a]).

Lemma 2.1. *Given an ϵ-approximate Nash equilibrium $\{x^p_j\}_{j,p}$ of a game $\mathcal{G}$ we can compute in polynomial time a $\sqrt{\epsilon} \cdot (\sqrt{\epsilon} + 1 + 4(r-1)u_{\max})$-approximately well supported Nash equilibrium $\{\hat{x}^p_j\}_{j,p}$, where r is the number of players and $u_{\max}$ is the maximum entry in the payoff tables of $\mathcal{G}$.*

In the sequel we shall focus on the notion of approximately well-supported Nash equilibrium, but all our results will also hold for the notion of approximate Nash equilibrium. Notice that Nash's theorem ensures the existence of an ϵ-Nash equilibrium —and hence of an ϵ-approximate Nash equilibrium— for every $\epsilon \geq 0$; in particular, for every ϵ there exists an ϵ-Nash equilibrium whose probabilities are integer multiples of $\epsilon/(2r \times u_{\text{maxsum}})$, where u_{maxsum} is the maximum, over all players p, of the sum of all entries in the payoff table of p. This can be established by rounding a Nash equilibrium $\{x_j^p\}_{j,p}$ to a nearby (in total variation distance) set of mixed strategies $\{\hat{x}_j^p\}_{j,p}$ all the entries of which are integer multiples of $\epsilon/(2r \times u_{\text{maxsum}})$. Note, however, that a ϵ-Nash equilibrium may not be close to an exact Nash equilibrium; see [EY07] for much more on this important distinction.

A game in normal form requires $r|S|$ numbers for its description, an amount of information that is exponential in the number of players. A *graphical game* [KLS01] is defined in terms of an undirected graph $G = (V, E)$ together with a set of strategies S_v for each $v \in V$. We denote by $\mathcal{N}(v)$ the set consisting of v and v's neighbors in G, and by $S_{\mathcal{N}(v)}$ the set of all $|\mathcal{N}(v)|$-tuples of strategies, one from each vertex in $\mathcal{N}(v)$. In a graphical game, the utility of a vertex $v \in V$ only depends on the strategies of the vertices in $\mathcal{N}(v)$ so it can be represented by just $|S_{\mathcal{N}(v)}|$ numbers. In other words, a graphical game is a succinct representation of a multiplayer game, advantageous when it so happens that the utility of each player only depends on a few other players. A generalization of graphical games are the *directed graphical games*, where G is directed and $\mathcal{N}(v)$ consists of v and the predecessors of v. The two notions are almost identical; of course, the directed graphical games are more general than the undirected ones, but any directed graphical game can be represented, albeit less concisely, as an undirected game whose graph is the same except with no direction on the edges. We will not be very careful in distinguishing the two notions; our results will apply to both. The following is a useful definition.

Definition 2.2. *Suppose that $\mathcal{GG}$ is a graphical game with underlying graph $G = (V, E)$. The* affects-graph *$G' = (V, E')$ of $\mathcal{GG}$ is a directed graph with edge $(v_1, v_2) \in E'$ if the payoff to v_2 depends on the action of v_1, that is, the payoff to v_2 is a non-constant function of the action of v_1.*

In the above definition, an edge (v_1, v_2) in G' represents the relationship "v_1 affects v_2". Notice that if $(v_1, v_2) \in E'$ then $\{v_1, v_2\} \in E$, but the opposite need not be true

—it could very well be that some vertex v_2 is affected by another vertex v_1, but vertex v_1 is not affected by v_2.

Since graphical games are representations of multi-player games, it follows by Nash's theorem that every graphical game has a mixed Nash equilibrium. It can be checked that a set of mixed strategies $\{x_j^v\}_{j \in S_v}, v \in V$, is a mixed Nash equilibrium if and only if

$$\forall v \in V, j, j' \in S_v : \sum_{s \in S_{\mathcal{N}(v) \setminus \{v\}}} u_{js}^v x_s > \sum_{s \in S_{\mathcal{N}(v) \setminus \{v\}}} u_{j's}^v x_s \Longrightarrow x_{j'}^v = 0.$$

Similarly, the condition for an approximately well supported Nash equilibrium can be derived.

2.2 The Complexity Theory of Total Search Problems and the Class PPAD

A *search problem* $\mathcal{S}$ is a set of inputs $I_{\mathcal{S}} \subseteq \Sigma^*$ on some alphabet Σ such that for each $x \in I_{\mathcal{S}}$ there is an associated set of solutions $\mathcal{S}_x \subseteq \Sigma^{|x|^k}$ for some integer k, such that for each $x \in I_{\mathcal{S}}$ and $y \in \Sigma^{|x|^k}$ whether $y \in \mathcal{S}_x$ is decidable in polynomial time. Notice that this is precisely NP with an added emphasis on finding a witness.

For example, let us define r-Nash to be the search problem $\mathcal{S}$ in which each $x \in I_{\mathcal{S}}$ is an r-player game in normal form together with a binary integer A (the *accuracy specification*), and $\mathcal{S}_x$ is the set of $\frac{1}{A}$-Nash equilibria of the game (where the probabilities are rational numbers of bounded size as discussed). Similarly, d-graphical Nash is the search problem with inputs the set of all graphical games with degree at most d, plus an accuracy specification A, and solutions the set of all $\frac{1}{A}$-Nash equilibria. (For $r > 2$ it is important to specify the problem in terms of a search for *approximate* Nash equilibrium — exact solutions may need to be high-degree algebraic numbers, raising the question of how to represent them as bit strings.)

A search problem is *total* if $\mathcal{S}_x \neq \emptyset$ for all $x \in I_{\mathcal{S}}$. For example, Nash's 1951 theorem [Nas51] implies that r-NASH is total. Obviously, the same is true for d-GRAPHICAL NASH. The set of all total search problems is denoted TFNP. A *polynomial-time reduction* from total search problem $\mathcal{S}$ to total search problem $\mathcal{T}$ is a pair f, g of polynomial-time computable functions such that, for every input x of $\mathcal{S}$, $f(x)$ is an input of $\mathcal{T}$, and furthermore for every $y \in \mathcal{T}_{f(x)}$, $g(y) \in \mathcal{S}_x$.

TFNP is what in Complexity is sometimes called a "semantic" class [Pap94a], i.e., it has no generic complete problem. Therefore, the complexity of total functions is typically explored via "syntactic" subclasses of TFNP, such as PLS [JPY88], PPP, PPA and PPAD [Pap94b]. We focus on PPAD.

PPAD can be defined in many ways. As mentioned in the introduction, it is, informally, the set of all total functions whose totality is established by invoking the following simple lemma on a graph whose vertex set is the solution space of the instance:

> *In any directed graph with one unbalanced node (node with outdegree different from its indegree), there is another unbalanced node.*

This general principle can be specialized, without loss of generality or computational power, to the case in which every node has both indegree and outdegree at most one. In this case the lemma becomes:

> *In any directed graph in which all vertices have indegree and outdegree at most one, if there is a* source *(a node with indegree zero), then there must be a sink (a node with outdegree zero).*

Formally, we shall define PPAD as the class of all total search problems polynomial-time reducible to the following problem:

END OF THE LINE: *Given two circuits S and P, each with n input bits and n output bits, such that $P(0^n) = 0^n \neq S(0^n)$, find an input $x \in \{0,1\}^n$ such that $P(S(x)) \neq x$ or $S(P(x)) \neq x \neq 0^n$.*

Intuitively, END OF THE LINE creates a directed graph $G_{S,P}$ with vertex set $\{0,1\}^n$ and an edge from x to y whenever both $y = S(x)$ and $x = P(y)$; S and P stand for "successor candidate" and "predecessor candidate". All vertices in $G_{S,P}$ have indegree and outdegree at most one, and there is at least one source, namely 0^n, so there must be a sink. We seek either a sink, or a source other than 0^n. Notice that in this problem a sink *or a source other than* 0^n is sought; if we insist on a sink, another complexity class called PPADS, apparently larger than PPAD, results.

The other important classes PLS, PPP and PPA, and others, are defined in a similar fashion based on other elementary properties of finite graphs. These classes are of no relevance to our analysis so their definition will be skipped; the interested reader is referred to [Pap94b].

A search problem $\mathcal{S}$ in PPAD is called PPAD-*complete* if all problems in PPAD reduce to it. Obviously, END OF THE LINE is PPAD-complete; furthermore, it was shown in [Pap94b] that several problems related to topological fixed points and their combinatorial underpinnings are PPAD-complete: BROUWER, SPERNER, BORSUK-ULAM, TUCKER. Our main result (Theorem 4.1) states that so are the problems 3-NASH and 3-GRAPHICAL NASH.

2.3 Computing a Nash Equilibrium is in PPAD

We establish that computing an approximate Nash equilibrium in an r-player game is in PPAD. The $r = 2$ case was shown in [Pap94b].

Theorem 2.3. *r-NASH is in PPAD, for $r \geq 2$.*

Proof. We reduce r-NASH to END OF THE LINE. Note that Nash's original proof [Nas51] utilizes Brouwer's fixed point theorem — it is essentially a reduction from the problem of finding a Nash equilibrium to that of finding a Brouwer fixed point of a continuous function; the latter problem can be reduced, under certain continuity conditions, to END OF THE LINE, and is therefore in PPAD. The, rather elaborate, proof below makes this simple intuition precise.

Let $\mathcal{G}$ be a normal-form game with r players, $1, \ldots, r$, and strategy sets $S_p = [n]$, for all $p \in [r]$, and let $\{u_s^p : p \in [r], s \in S\}$ be the utilities of the players. Also let $\epsilon < 1$. In time polynomial in $|\mathcal{G}| + \log(1/\epsilon)$, we will specify two circuits S and P each with $N = poly(|\mathcal{G}|, \log(1/\epsilon))$ input and output bits and $P(0^N) = 0^N \neq S(0^N)$, so that, given any solution to END OF THE LINE on input S, P, one can construct in polynomial time an ϵ-approximate Nash equilibrium of $\mathcal{G}$. This is enough for reducing r-NASH to END OF THE LINE by virtue of Lemma 2.1. Our construction of S, P builds heavily upon the simplicial approximation algorithm of Laan and Talman [LT82] for computing fixed points of continuous functions from the product space of unit simplices to itself.

Let $\Delta_n = \{x \in \mathbb{R}^n_+ | \sum_{k=1}^n x_k = 1\}$ be the $(n-1)$-dimensional unit simplex. Then the space of mixed strategy profiles of the game is $\Delta_n^r := \times_{p=1}^r \Delta_n$. For notational convenience we embed Δ_n^r in $\mathbb{R}^{n \cdot r}$, and represent the elements of Δ_n^r as vectors in $\mathbb{R}^{n \cdot r}$. That is, if $(x^1, x^2, \ldots, x^r) \in \Delta_n^r$ is a mixed strategy profile of the game, we identify this strategy profile with a vector $x = (x^1; x^2; \ldots; x^r) \in \mathbb{R}^{n \cdot r}$ resulting from the concatenation of the mixed strategies. For $p \in [r]$ and $j \in [n]$ we denote by $x(p, j)$ the $((p-1)n + j)$-th coordinate of x, that is $x(p, j) := x_{(p-1)n+j}$.

We are about to describe our reduction from finding an ϵ-approximate Nash equilibrium to END OF THE LINE. The nodes of the END OF THE LINE graph will correspond to the simplices of a triangulation of Δ_n^r which we describe next.

Triangulation of the Product Space of Unit Simplices. For some d, to be specified later, we describe the triangulation of Δ_n^r induced by the regular grid of size d. For this purpose, let us denote by $\Delta_n(d)$ the set of points of Δ_n induced by the grid of size d, i.e.

$$\Delta_n(d) = \left\{ x \in \mathbb{R}_+^n \;\middle|\; x = \left(\frac{y_1}{d}, \frac{y_2}{d}, \ldots, \frac{y_n}{d}\right), y_j \in \mathbb{N}_0 \text{ and } \sum_j y_j = d \right\},$$

and, similarly, define $\Delta_n^r(d) = \times_{p=1}^r \Delta_n(d)$. Moreover, let us define the block diagonal matrix Q by

$$Q = \begin{pmatrix} Q^1 & 0 & \ldots & 0 & 0 \\ 0 & Q^2 & & 0 & 0 \\ 0 & & & & \\ \vdots & & \ddots & & \vdots \\ & & & Q^{r-1} & 0 \\ 0 & 0 & \ldots & 0 & Q^r \end{pmatrix},$$

where, for all $p \in [r]$, Q^p is the $n \times n$ matrix defined by

$$Q^p = \begin{pmatrix} -1 & 0 & \ldots & 0 & 1 \\ 1 & -1 & & 0 & 0 \\ 0 & 1 & & & \\ \vdots & & \ddots & & \vdots \\ & & & -1 & 0 \\ 0 & 0 & \ldots & 1 & -1 \end{pmatrix}.$$

Let us denote by $q(p, j)$ the $((p-1)n + j)$-th column of Q. It is clear that adding $q(p, j)^{\mathrm{T}}/d$ to a mixed strategy profile corresponds to shifting probability mass of $1/d$

from strategy j of player p to strategy $(j \mod n) + 1$ of player p.

For all $p \in [r]$ and $k \in [n]$, let us define the set of indices $I_{p,k} := \{(p,j)\}_{j \leq k}$. Also, let us define a collection $\mathcal{T}$ of sets of indices as follows

$$\mathcal{T} := \left\{ T \subseteq \bigcup_{p \in [r]} I_{p,n} \,\middle|\, \forall p \in [r], \exists k \in [n-1] : T \cap I_{p,n} = I_{p,k} \right\}.$$

Suppose, now, that q_0 is a mixed strategy profile in which every player plays strategy 1 with probability 1, that is $q_0(p,1) = 1$, for all $p \in [r]$, and for $T \in \mathcal{T}$ define the set

$$A(T) := \left\{ x \in \Delta_n^r \middle| x = q_0 + \sum_{(p,j) \in T} a(p,j) q(p,j)^{\mathrm{T}}/d \text{ for non-negative reals } a(p,j) \geq 0 \right\}.$$

Defining $T^* := \cup_{p \in [r]} I_{p,n-1}$, it is not hard to verify that

$$A\left(T^*\right) = \Delta_n^r.$$

Moreover, if, for $T \in \mathcal{T}$, we define $B(T) := A(T) \setminus \cup_{T' \in \mathcal{T}, T' \subset T} A(T')$, the collection $\{B(T)\}_{T \in \mathcal{T}}$ partitions the set Δ_n^r.

To define the triangulation of Δ_n^r let us fix some set $T \in \mathcal{T}$, some permutation $\pi : [|T|] \to T$ of the elements of T, and some $x_0 \in A(T) \cap \Delta_n^r(d)$. Let us then denote by $\sigma(x_0, \pi)$ the $|T|$-simplex which is the convex hull of the points $x_0, \ldots, x_{|T|}$ defined as follows

$$x_t = x_{t-1} + q(\pi(t))^{\mathrm{T}}/d, \quad \text{for all } t = 1, \ldots, |T|.$$

The following lemmas, whose proof can be found in [LT82], describe the triangulation of Δ_n^r. We define $A(T,d) := A(T) \cap \Delta_n^r(d)$, we denote by $\mathcal{P}_T$ the set of all permutations $\pi : [|T|] \to T$, and set

$$\Sigma_T := \{\sigma(x_0, \pi) \,|\, x_0 \in A(T, d),\ \pi \in \mathcal{P}_T,\ \sigma(x_0, \pi) \subseteq A(T)\}\,.$$

Lemma 2.4 ([LT82])**.** *For all $T \in \mathcal{T}$, the collection of $|T|$-simplices Σ_T triangulates $A(T)$.*

Corollary 2.5 ([LT82])**.** *Δ_n^r is triangulated by the collection of simplices Σ_{T^*}.*

The Vertices of the END OF THE LINE Graph. The vertices of the graph in our construction will correspond to the elements of the set

$$\Sigma := \bigcup_{T \in \mathcal{T}} \Sigma_T.$$

Let us encode the elements of Σ with strings $\{0,1\}^N$; choosing N polynomial in $|\mathcal{G}|$, the description size of $\mathcal{G}$, and $\log d$ is sufficient.

We proceed to define the edges of the END OF THE LINE graph in terms of a labeling of the points of the set $\Delta_n^r(d)$, which we describe next.

Labeling Rule. Recall the function $f : \Delta_n^r \to \Delta_n^r$ defined by Nash to establish the existence of an equilibrium [Nas51]. To describe f, let $U_j^p(x) := \sum_{s \in S_{-p}} u_{js}^p x_s$ be the expected utility of player p, if p plays pure strategy $j \in [n]$ and the other players use the mixed strategies $\{x_j^q\}_{j \in [n]}, q \neq p$; let also $U^p(x) := \sum_{s \in S} u_s^p x_s$ be the expected utility of player p if every player $q \in [r]$ uses mixed strategy $\{x_j^q\}_{j \in [n]}$. Then, the function f is described as follows:

$$f(x^1, x^2, \dots, x^r) = (y^1, y^2, \dots, y^r),$$

where, for each $p \in [r]$, $j \in [n]$,

$$y_j^p = \frac{x_j^p + \max\left(0, U_j^p(x) - U^p(x)\right)}{1 + \sum_{k \in [n]} \max\left(0, U_k^p(x) - U^p(x)\right)}.$$

It is not hard to see that f is continuous, and that $f(x)$ can be computed in time polynomial in the binary encoding size of x and $\mathcal{G}$. Moreover, it can be verified that any point $x \in \Delta_n^r$ such that $f(x) = x$ is a Nash equilibrium [Nas51]. The following lemma establishes that f is λ-Lipschitz for $\lambda := [1 + 2U_{\max} rn(n+1)]$, where $U_{\max}$ is the maximum entry in the payoff tables of the game.

Lemma 2.6. *For all $x, x' \in \Delta_n^r \subseteq \mathbb{R}^{n \cdot r}$ such that $||x - x'||_\infty \leq \delta$,*

$$||f(x) - f(x')||_\infty \leq [1 + 2U_{\max} rn(n+1)]\delta.$$

Proof. We use the following bound shown in Section 3.6, Lemma 3.32.

Lemma 2.7. *For any game $\mathcal{G}$, for all $p \leq r$, $j \in S_p$,*

$$\left| \sum_{s \in S_{-p}} u_{js}^p x_s - \sum_{s \in S_{-p}} u_{js}^p x'_s \right| \leq \max_{s \in S_{-p}} \{u_{js}^p\} \sum_{q \neq p} \sum_{i \in S_q} |x_i^q - x_i'^q|.$$

It follows that for all $p \in [r]$, $j \in [n]$,

$$\begin{aligned} & |U_j^p(x) - U_j^p(x')| \leq U_{\max} rn\delta \\ \text{and} \quad & |U^p(x) - U^p(x')| \leq U_{\max} rn\delta. \end{aligned}$$

Denoting $B_j^p(x) := \max\left(0, U_j^p(x) - U^p(x)\right)$, for all $p \in [r]$, $j \in [n]$, the above bounds imply that

$$|B_j^p(x) - B_j^p(x')| \leq 2U_{\max} rn\delta,$$
$$\left|\sum_{k\in[n]} B_k^p(x) - \sum_{k\in[n]} B_k^p(x')\right| \leq 2U_{\max} rn\delta \cdot n.$$

Combining the above bounds we get that, for all $p \in [r]$, $j \in [n]$,

$$\begin{aligned} |y_j^p(x) - y_j^p(x')| &\leq |x_j^p - x_j'^p| + |B_j^p(x) - B_j^p(x')| + \left|\sum_{k\in[n]} B_k^p(x) - \sum_{k\in[n]} B_k^p(x')\right| \\ &\leq \delta + 2U_{\max} rn\delta + 2U_{\max} rn\delta \cdot n \\ &\leq [1 + 2U_{\max} rn(n+1)]\delta, \end{aligned}$$

where we made use of the following lemma:

Lemma 2.8. *For any $x, x', y, y', z, z' \geq 0$ such that $\frac{x+y}{1+z} \leq 1$,*

$$\left|\frac{x+y}{1+z} - \frac{x'+y'}{1+z'}\right| \leq |x-x'| + |y-y'| + |z-z'|.$$

Proof.

$$\begin{aligned} \left|\frac{x+y}{1+z} - \frac{x'+y'}{1+z'}\right| &= \left|\frac{(x+y)(1+z') - (x'+y')(1+z)}{(1+z)(1+z')}\right| \\ &= \left|\frac{(x+y)(1+z') - (x+y)(1+z) - ((x'-x)+(y'-y))(1+z)}{(1+z)(1+z')}\right| \\ &\leq \left|\frac{(x+y)(1+z') - (x+y)(1+z)}{(1+z)(1+z')}\right| + \left|\frac{((x'-x)+(y'-y))(1+z)}{(1+z)(1+z')}\right| \\ &\leq \left|\frac{(x+y)(z'-z)}{(1+z)(1+z')}\right| + |x'-x| + |y'-y| \\ &\leq \frac{x+y}{1+z}|z'-z| + |x'-x| + |y'-y| \leq |z'-z| + |x'-x| + |y'-y|. \end{aligned}$$

□

□

We describe a labeling of the points of the set $\Delta_n^r(d)$ in terms of the function f. The labels that we are going to use are the elements of the set $\mathcal{L} := \cup_{p\in[r]} I_{p,n}$. In particular,

We assign to a point $x \in \Delta_n^r$ the label (p, j) iff (p, j) is the lexicographically least index such that $x_j^p > 0$ and $f(x)_j^p - x_j^p \leq f(x)_k^q - x_k^q$, for all $q \in [r], k \in [n]$.

This labeling rule satisfies the following properties:

- *Completeness*: Every point x is assigned a label; hence, we can define a labeling function $\ell : \Delta_n^r \to \mathcal{L}$.
- *Properness*: $x_j^p = 0$ implies $\ell(x) \neq (p, j)$.
- *Efficiency*: $\ell(x)$ is computable in time polynomial in the binary encoding size of x and $\mathcal{G}$.

A simplex $\sigma \in \Sigma$ is called *completely labeled* if all its vertices have different labels; a simplex $\sigma \in \Sigma$ is called *p-stopping* if it is completely labeled and, moreover, for all $j \in [n]$, there exists a vertex of σ with label (p, j). Our labeling satisfies the following important property.

Theorem 2.9 ([LT82])**.** *Suppose a simplex $\sigma \in \Sigma$ is p-stopping for some $p \in [r]$. Then all points $x \in \sigma \subseteq \mathbb{R}^{n\cdot r}$ satisfy*

$$||f(x) - x||_\infty \leq \frac{1}{d}(\lambda + 1)n(n - 1).$$

Proof. It is not hard to verify that, for any simplex $\sigma \in \Sigma$ and for all pairs of points $x, x' \in \sigma$,

$$||x - x'||_\infty \leq \frac{1}{d}.$$

Suppose now that a simplex $\sigma \in \Sigma$ is p-stopping, for some $p \in [r]$, and that, for all $j \in [n]$, $z(j)$ is the vertex of σ with label (p, j). Since, for any x, $\sum_{i\in[n]} x_i^p = 1 = \sum_{i\in[n]} f(x)_i^p$, it follows from the labeling rule that

$$f(z(j))_j^p - z(j)_j^p \leq 0, \forall j \in [n].$$

Hence, for all $x \in \sigma$, $j \in [n]$,

$$f(x)_j^p - x_j^p \leq f(z(j))_j^p - z(j)_j^p + (\lambda+1)\frac{1}{d} \leq (\lambda+1)\frac{1}{d},$$

where we used the fact that the diameter of σ is $\frac{1}{d}$ (in the infinity norm) and the function f is λ-Lipschitz. Hence, in the opposite direction, for all $x \in \sigma$, $j \in [n]$, we have

$$f(x)_j^p - x_j^p = -\sum_{i\in[n]\setminus\{j\}} (f(x)_i^p - x_i^p) \geq -(n-1)(\lambda+1)\frac{1}{d}.$$

Now, by the definition of the labeling rule, we have, for all $x \in \sigma$, $q \in [r]$, $j \in [n]$,

$$\begin{aligned} f(x)_j^q - x_j^q &\geq f(z(1))_j^q - z(1)_j^q - (\lambda+1)\frac{1}{d} \\ &\geq f(z(1))_1^p - z(1)_1^p - (\lambda+1)\frac{1}{d} \\ &\geq -(n-1)(\lambda+1)\frac{1}{d} - (\lambda+1)\frac{1}{d} = -n(\lambda+1)\frac{1}{d}, \end{aligned}$$

whereas

$$\begin{aligned} f(x)_j^q - x_j^q &= -\sum_{i\in[n]\setminus\{j\}} (f(x)_i^q - x_i^q) \\ &\leq (n-1)n(\lambda+1)\frac{1}{d}. \end{aligned}$$

Combining the above, it follows that, for all $x \in \sigma$,

$$||f(x) - x||_\infty \leq \frac{1}{d}(\lambda + 1)n(n-1).$$

□

The Approximation Guarantee. By virtue of Theorem 2.9, if we choose

$$d := \frac{1}{\epsilon'}[2 + 2U_{\max} rn(n+1)]n(n-1),$$

then a p-stopping simplex $\sigma \in \Sigma$, for any $p \in [r]$, satisfies that, for all $x \in \sigma$,

$$||f(x) - x||_\infty \leq \epsilon',$$

which, by Lemma 2.10 below, implies that x is an approximate Nash equilibrium achieving approximation

$$n\sqrt{\epsilon'(1 + nU_{\max})}\left(1 + \sqrt{\epsilon'(1 + nU_{\max})}\right)\max\{U_{\max}, 1\}.$$

Choosing

$$\epsilon' := \frac{1}{1 + nU_{\max}}\left(\frac{\epsilon}{2n\max\{U_{\max}, 1\}}\right)^2,$$

we have that x is an ϵ-approximate Nash equilibrium.

Lemma 2.10. *If a vector $x = (x^1; x^2; \ldots; x^r) \in \mathbb{R}^{n \cdot r}$ satisfies*

$$||f(x) - x||_\infty \leq \epsilon',$$

then x is a $n\sqrt{\epsilon'(1 + nU_{\max})}\left(1 + \sqrt{\epsilon'(1 + nU_{\max})}\right)\max\{U_{\max}, 1\}$*-approximate Nash* *equilibrium.*

Proof. Let us fix some player $p \in [r]$, and assume, without loss of generality, that

$$U_1^p(x) \geq U_2^p(x) \geq \ldots \geq U_k^p(x) \geq U^p(x) \geq U_{k+1}^p(x) \geq \ldots \geq U_n^p(x).$$

For all $j \in [n]$, observe that $|f(x)_j^p - x_j^p| \leq \epsilon'$ implies

$$x_j^p \sum_{i \in [n]} B_i^p(x) \leq B_j^p(x) + \epsilon' \left(1 + \sum_{i \in [n]} B_i^p(x) \right).$$

Setting $\epsilon'' := \epsilon'(1 + nU_{\max})$, the above inequality implies

$$x_j^p \sum_{i \in [n]} B_i^p(x) \leq B_j^p(x) + \epsilon''. \tag{2.3}$$

Let us define $t := x_{k+1}^p + x_{k+2}^p + \ldots + x_n^p$, and let us distinguish the following cases

- If $t \geq \frac{\sqrt{\epsilon''}}{U_{\max}}$, then summing Equation (2.3) for $j = k+1, \ldots, n$ implies

$$t \sum_{i \in [n]} B_i^p(x) \leq (n-k)\epsilon'',$$

  which gives

$$B_1^p \leq \sum_{i \in [n]} B_i^p(x) \leq n\sqrt{\epsilon''}U_{\max}. \tag{2.4}$$

- If $t \leq \frac{\sqrt{\epsilon''}}{U_{\max}}$, then multiplying Equation (2.3) by x_j^p and summing over $j = 1, \ldots, n$ gives

$$\sum_{j \in [n]} (x_j^p)^2 \sum_{i \in [n]} B_i^p(x) \leq \sum_{j \in [n]} x_j^p B_j^p(x) + \epsilon''. \tag{2.5}$$

Now observe that for any setting of the probabilities x_j^p, $j \in [n]$, it holds that

$$\sum_{j\in[n]} (x_j^p)^2 \geq \frac{1}{n}. \tag{2.6}$$

Moreover, observe that, since $U^p(x) = \sum_{j\in[n]} x_j^p U_j^p(x)$, it follows that

$$\sum_{j\in[n]} x_j^p (U_j^p(x) - U^p(x)) = 0,$$

which implies that

$$\sum_{j\in[n]} x_j^p B_j^p(x) + \sum_{j\geq k+1} x_j^p (U_j^p(x) - U^p(x)) = 0.$$

Plugging this into (2.5) implies

$$\sum_{j\in[n]} (x_j^p)^2 \sum_{i\in[n]} B_i^p(x) \leq \sum_{j\geq k+1} x_j^p (U^p(x) - U_j^p(x)) + \epsilon''.$$

Further, using (2.6) gives

$$\frac{1}{n} \sum_{i\in[n]} B_i^p(x) \leq \sum_{j\geq k+1} x_j^p (U^p(x) - U_j^p(x)) + \epsilon'',$$

which implies

$$\sum_{i\in[n]} B_i^p(x) \leq n(tU_{\max} + \epsilon'').$$

The last inequality then implies

$$B_1^p(x) \leq n(\sqrt{\epsilon''} + \epsilon''). \tag{2.7}$$

Combining (2.4) and (2.7), we have the following uniform bound

$$B_1^p(x) \leq n(\sqrt{\epsilon''} + \epsilon'') \max\{U_{\max}, 1\} =: \epsilon'''. \tag{2.8}$$

Since $B_1^p(x) = U_1^p(x) - U(x)$, it follows that player p cannot improve her payoff by more that ϵ''' by changing her strategy. This is true for every player, hence x is a ϵ'''-approximate Nash equilibrium. □

The Edges of the END OF THE LINE **Graph.** Laan and Talman [LT82] describe a pivoting algorithm which operates on the set Σ, by specifying the following:

- a simplex $\sigma_0 \in \Sigma$, which is the *starting simplex*; σ_0 contains the point q_0 and is uniquely determined by the labeling rule;
- a partial one-to-one function $h : \Sigma \to \Sigma$, mapping a simplex to a neighboring simplex, which defines a *pivoting rule*; h has the following properties: [1]
  - σ_0 has no pre-image;
  - any simplex $\sigma \in \Sigma$ that has no image is a p-stopping simplex for some p; and, any simplex $\sigma \in \Sigma \setminus \{\sigma_0\}$ that has no pre-image is a p-stopping simplex for some p;
  - both $h(\sigma)$ and $h^{-1}(\sigma)$ are computable in time polynomial in the binary encoding size of σ, that is N, and $\mathcal{G}$ —given that the labeling function ℓ is efficiently computable;

The algorithm of Laan and Talman starts off with the simplex σ_0 and employs the pivoting rule h until a simplex σ with no image is encountered. By the properties

[1] More precisely, the pivoting rule h of Laan and Talman is defined on a subset Σ' of Σ. For our purposes, let us extend their pivoting rule h to the set Σ by setting $h(\sigma) = \sigma$ for all $\sigma \in \Sigma \setminus \Sigma'$.

of h, σ must be p-stopping for some $p \in [r]$ and, by the discussion above, any point $x \in \sigma$ is an ϵ-approximate Nash equilibrium.

In our construction, the edges of the END OF THE LINE graph are defined in terms of the function h: if $h(\sigma) = \sigma'$, then there is a directed edge from σ to σ'. Moreover, the string 0^N is identified with the simplex σ_0. Any solution to the END OF THE LINE problem thus defined corresponds by the above discussion to a simplex σ such that any point $x \in \sigma$ is an ϵ-approximate Nash equilibrium of $\mathcal{G}$. This concludes the construction. □

2.4 Brouwer: a PPAD-Complete Fixed Point Computation Problem

To show that Nash is PPAD-hard, we use a problem we call Brouwer, which is a discrete and simplified version of the search problem associated with Brouwer's fixed point theorem. We are given a continuous function ϕ from the 3-dimensional unit cube to itself, defined in terms of its values at the centers of 2^{3n} cubelets with side 2^{-n}, for some $n \geq 0$. [2] At the center c_{ijk} of the cubelet K_{ijk} defined as

$$K_{ijk} = \{(x, y, z) : \; i \cdot 2^{-n} \leq x \leq (i+1) \cdot 2^{-n},$$
$$j \cdot 2^{-n} \leq y \leq (j+1) \cdot 2^{-n},$$
$$k \cdot 2^{-n} \leq z \leq (k+1) \cdot 2^{-n}\},$$

where i, j, k are integers in $\{0, 1, \ldots, 2^n - 1\}$, the value of ϕ is $\phi(c_{ijk}) = c_{ijk} + \delta_{ijk}$, where δ_{ijk} is one of the following four vectors (also referred to as colors):

[2] The value of the function near the boundaries of the cubelets could be determined by interpolation —there are many simple ways to do this, and the precise method is of no importance to our discussion.

- $\delta_1 = (\alpha, 0, 0)$
- $\delta_2 = (0, \alpha, 0)$
- $\delta_3 = (0, 0, \alpha)$
- $\delta_0 = (-\alpha, -\alpha, -\alpha)$

Here $\alpha > 0$ is much smaller than the cubelet side, say 2^{-2n}.

Thus, to compute ϕ at the center of the cubelet K_{ijk} we only need to know which of the four displacements to add. This is computed by a circuit C (which is the only input to the problem) with $3n$ input bits and 2 output bits; $C(i,j,k)$ is the index r such that, if c is the center of cubelet K_{ijk}, $\phi(c) = c + \delta_r$. C is such that $C(0,j,k) = 1$, $C(i,0,k) = 2$, $C(i,j,0) = 3$, and $C(2^n - 1, j, k) = C(i, 2^n - 1, k) = C(i, j, 2^n - 1) = 0$ (with conflicts resolved arbitrarily), so that the function ϕ maps the boundary to the interior of the cube. A vertex of a cubelet is called *panchromatic* if among the cubelets adjacent to it there are four that have all four displacements $\delta_0, \delta_1, \delta_2, \delta_3$. Sperner's Lemma guarantees that, for any circuit C satisfying the above properties, a panchromatic vertex exists, see, e.g., [Pap94b]. An alternative proof of this fact follows as a consequence of Theorem 2.11 below.

Brouwer is thus the following total problem: Given a circuit C as described above, find a panchromatic vertex. The relationship with Brouwer fixed points is that *fixed points of ϕ only ever occur in the vicinity of a panchromatic vertex.* We next show:

Theorem 2.11. Brouwer *is* PPAD-*complete.*

Proof. That Brouwer is in PPAD follows from Theorem 4.1, which is a reduction from Brouwer to r-Nash, which has been shown to be in PPAD in Theorem 2.3.

To show hardness, we shall reduce END OF THE LINE to Brouwer. Given circuits S and P with n inputs and outputs, as prescribed in that problem, we shall construct an "equivalent" instance of BROUWER, that is, another circuit C with $3m = 3(n+4)$ inputs and two outputs that computes the color of each cubelet of side 2^{-m}, that is to say, the index i such that δ_i is the correct displacement of the Brouwer function at the center of the cubelet encoded into the $3m$ bits of the input. We shall first describe the Brouwer function ϕ explicitly, and then argue that it can be computed by a circuit.

Our description of ϕ proceeds as follows: We shall first describe a 1-dimensional subset L of the 3-dimensional unit cube, intuitively an embedding of the path-like directed graph $G_{S,P}$ implicitly given by S and P. Then we shall describe the 4-coloring of the 2^{3m} cubelets based on the description of L. Finally, we shall argue that colors are easy to compute locally, and that panchromatic vertices correspond to endpoints other than the standard source 0^n of $G_{S,P}$.

We assume that the graph $G_{S,P}$ is such that for each edge (u, v), one of the vertices is even (ends in 0) and the other is odd; this is easy to guarantee by duplicating the vertices of $G_{S,P}$.

L will be orthonormal, that is, each of its segments will be parallel to one of the axes; all coordinates of endpoints of segments are integer multiples of 2^{-m}, a factor that we omit in the discussion below. Let $u \in \{0,1\}^n$ be a vertex of $G_{S,P}$. By $\langle u \rangle$ we denote the integer between 0 and $2^n - 1$ whose binary representation is u. Associated with u there are two line segments of length 4 of L. The first, called the *principal* segment of u, has endpoints $u_1 = (8\langle u \rangle + 2, 3, 3)$ and $u'_1 = (8\langle u \rangle + 6, 3, 3)$. The other *auxiliary* segment has endpoints $u_2 = (3, 8\langle u \rangle + 6, 2^m - 3)$ and $u'_2 = (3, 8\langle u \rangle + 10, 2^m - 3)$. Informally, these segments form two dashed lines (each segment being a dash) that run along two edges of the cube and slightly in its interior (see Figure 2.1).

Now, for every vertex u of $G_{S,P}$, we connect u'_1 to u_2 by a line with three straight segments, with joints $u_3 = (8\langle u \rangle + 6, 8\langle u \rangle + 6, 3)$ and $u_4 = (8\langle u \rangle + 6, 8\langle u \rangle + 6, 2^m - 3)$.

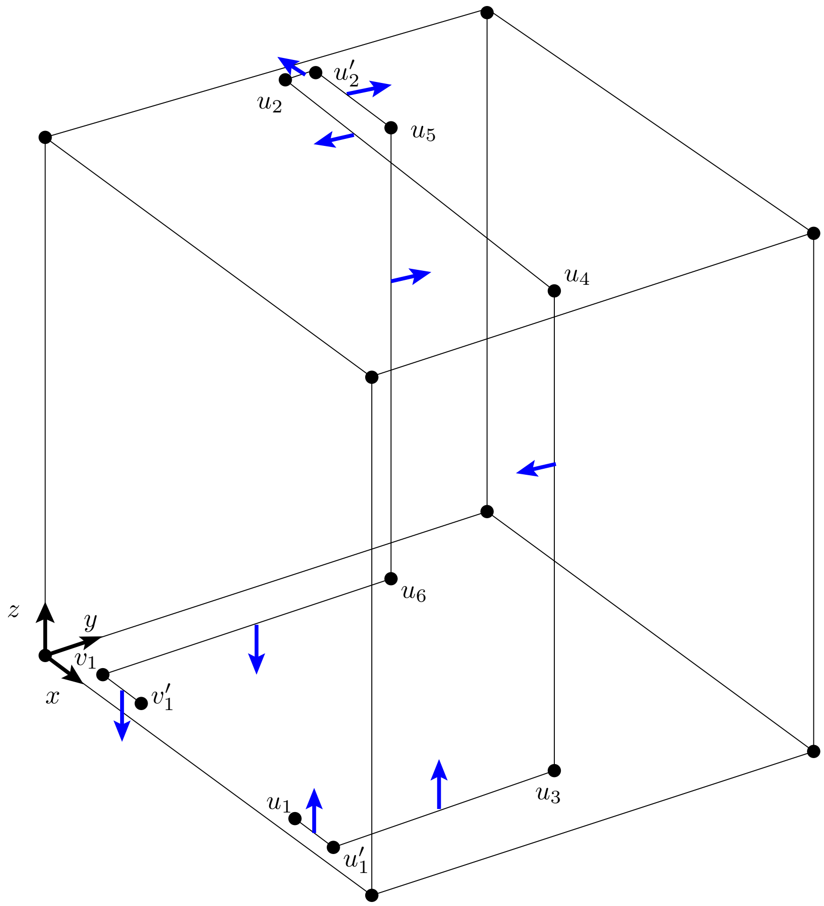

Figure 2.1: The orthonormal path connecting vertices (u,v); the arrows indicate the orientation of colors surrounding the path.

Finally, if there is an edge (u, v) in $G_{S,P}$, we connect u_2' to v_1 by a jointed line with breakpoints $u_5 = (8\langle v\rangle + 2, 8\langle u\rangle + 10, 2^m - 3)$ and $u_6 = (8\langle v\rangle + 2, 8\langle u\rangle + 10, 3)$. This completes the description of the line L if we do the following perturbation: exceptionally, the principal segment of $u = 0^n$ has endpoints $0_1 = (2, 2, 2)$ and $0_1' = (6, 2, 2)$ and the corresponding joint is $0_3 = (6, 6, 2)$.

It is easy to see that L traverses the interior of the cube without ever "nearly crossing itself"; that is, two points p, p' of L are closer than $3 \cdot 2^{-m}$ in Euclidean distance only if they are connected by a part of L that has length $8 \cdot 2^{-m}$ or less. (This is important in order for the coloring described below of the cubelets surrounding L to be well-defined.) To check this, just notice that segments of different types (e.g.,

$[u_3, u_4]$ and $[u'_2, u_5]$) come closer than $3 \cdot 2^{-m}$ only if they share an endpoint; segments of the same type on the $z = 3$ or the $z = 2^m - 3$ plane are parallel and at least 4 apart; and segments parallel to the z axis differ by at least 4 in either their x or y coordinates.

We now describe the coloring of the 2^{3m} cubelets by four colors corresponding to the four displacements. Consistent with the requirements for a Brouwer circuit, we color any cubelet K_{ijk} where any one of i, j, k is $2^m - 1$, with 0. Given that, any other cubelet with $i = 0$ gets color 1; with this fixed, any other cubelet with $j = 0$ gets color 2, while the remaining cubelets with $k = 0$ get color 3. Having colored the boundaries, we now have to color the interior cubelets. An interior cubelet is always colored 0 *unless one of its vertices is a point of the interior of line* L, in which case it is colored by one of the three other colors in a manner to be explained shortly. Intuitively, at each point of the line L, starting from $(2, 2, 2)$ (the beginning of the principle segment of the string $u = 0^n$) the line L is "protected" from color 0 from all 4 sides. As a result, the only place where the four colors can meet is vertex u'_2 or $u_1, u \neq 0^n$, where u is an end of the line...

In particular, near the beginning of L at $(2, 2, 2)$ the 27 cubelets K_{ijk} with $i, j, k \leq 2$ are colored as shown in Figure 2.2. From then on, for any length-1 segment of L of the form $[(x, y, z), (x', y', z')]$ consider the four cubelets containing this segment. Two of these cubelets are colored 3, and the other two are colored 1 and 2, in this order clockwise (from the point of view of an observer at (x, y, z)). The remaining cubelets touching L are the ones at the joints where L turns. Each of these cubelets, a total of two per turn, takes the color of the two other cubelets adjacent to L with which it shares a face.

Now it only remains to describe, for each line segment $[a, b]$ of L, the direction d in which the two cubelets that are colored 3 lie. The rules are these (in Figure 2.1 the directions d are shown as arrows):

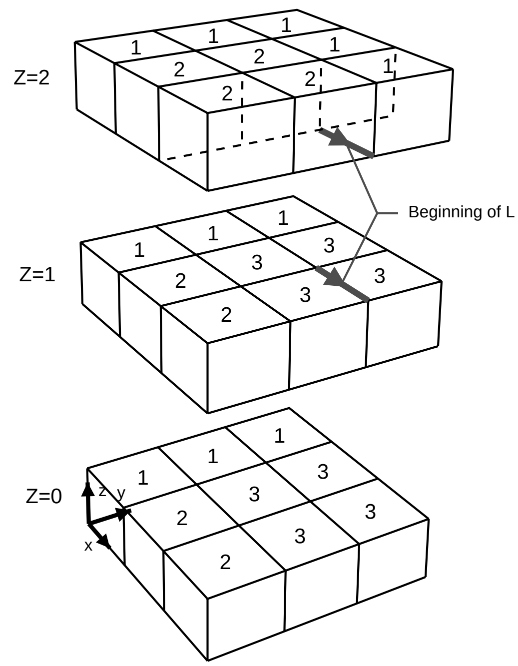

Figure 2.2: The 27 cubelets around the beginning of line L.

- If $[a, b] = [u_1, u'_1]$ then $d = (0, 0, -1)$ if u is even and $d = (0, 0, 1)$ if u is odd.
- If $[a, b] = [u'_1, u_3]$ then $d = (0, 0, -1)$ if u is even and $d = (0, 0, 1)$ if u is odd.
- If $[a, b] = [u_3, u_4]$ then $d = (0, 1, 0)$ if u is even and $d = (0, -1, 0)$ if u is odd.
- If $[a, b] = [u_4, u_2]$ then $d = (0, 1, 0)$ if u is even and $d = (0, -1, 0)$ if u is odd.
- If $[a, b] = [u_2, u'_2]$ then $d = (1, 0, 0)$ if u is even and $d = (-1, 0, 0)$ if u is odd.
- If $[a, b] = [u'_2, u_5]$ then $d = (0, -1, 0)$ if u is even and $d = (0, 1, 0)$ if u is odd.
- If $[a, b] = [u_5, u_6]$ then $d = (0, -1, 0)$ if u is even and $d = (0, 1, 0)$ if u is odd.
- If $[a, b] = [u_6, v_1]$ then $d = (0, 0, 1)$ if u is even and $d = (0, 0, -1)$ if u is odd.

This completes the description of the construction. Notice that, for this to work, we need our assumption that edges in $G_{S,P}$ go between odd and even vertices. Regarding the alternating orientation of colored cubelets around L, note that we could not simply introduce "twists" to make them always point in (say) direction $d = (0, 0, -1)$ for all $[u_1, u'_1]$. That would create a panchromatic vertex at the location of a twist.

The result now follows from the following two claims:

1. A point in the cube is panchromatic in the described coloring if and only if it is

   (a) an endpoint u'_2 of a sink vertex u of $G_{S,P}$, or

   (b) an endpoint u_1 of a source vertex $u \neq 0^n$ of $G_{S,P}$

2. A circuit C can be constructed in time polynomial in $|S|+|P|$, which computes, for each triple of binary integers $i, j, k < 2^m$, the color of cubelet K_{ijk}.

Regarding the first claim, the endpoint u'_2 of a sink vertex u, or the endpoint u_1 of a source vertex u other than 0^n, will be a point where L meets color 0, hence a panchromatic vertex. There is no alternative way that L can meet color 0 and no other way a panchromatic vertex can occur.

Regarding the second claim, circuit C is doing the following. $C(0, j, k) = 1$, for $j, k < 2^m - 1$, $C(i, 0, k) = 2$ for $i > 0$, $i, k < 2^m - 1$, $C(i, j, 0) = 3$ for $i, j > 0$, $i, j < 2^m - 1$. Then by default, $C(i, j, k) = 0$. However the following tests yield alternative values for $C(i, j, k)$, for cubelets adjacent to L. $LSB(x)$ denotes the least significant bit of x, equal to 1 if x is odd, 0 if x is even, and undefined if x is not an integer. For example, a $[u'_1, u_3], u \neq 0^n$ segment is given by (letting $x = \langle u \rangle$):

1. If $k = 2$ and $i = 8x + 5$ and $LSB(x) = 1$ and $j \in \{3, \ldots, 8x + 6\}$ then $C(i, j, k) = 2$.

2. If $k = 2$ and $i = 8x + 6$ and $LSB(x) = 1$ and $j \in \{2, \ldots, 8x + 6\}$ then $C(i, j, k) = 1$.

3. If $k = 3$ and ($i = 8x+5$ or $i = 8x+6$) and $LSB(x) = 1$ and $j \in \{2, \ldots, 8x+5\}$ then $C(i, j, k) = 3$.

4. If $k = 2$ and ($i = 8x+5$ or $i = 8x+6$) and $LSB(x) = 0$ and $j \in \{2, \ldots, 8x+6\}$ then $C(i, j, k) = 3$.

5. If $k = 3$ and $i = 8x + 5$ and $LSB(x) = 0$ and $j \in \{3, \ldots, 8x + 5\}$ then $C(i, j, k) = 1$.

6. If $k = 3$ and $i = 8x + 6$ and $LSB(x) = 0$ and $j \in \{2, \ldots, 8x + 5\}$ then $C(i, j, k) = 2$.

A $[u_2', u_5]$ segment uses the circuits P and S, and, in the case $LSB(x) = 1$, $x = \langle u \rangle$, is given by:

1. If ($k = 2^m - 3$ or $k = 2^m - 4$) and $j = 8x + 10$ and $S(x) = x'$ and $P(x') = x$ and $i \in \{2, \ldots, 8x' + 2\}$ then $C(i, j, k) = 3$.

2. If $k = 2^m - 3$ and and $j = 8x + 9$ and $S(x) = x'$ and $P(x') = x$ and $i \in \{3, \ldots, 8x' + 2\}$ then $C(i, j, k) = 1$.

3. If $k = 2^m - 4$ and $j = 8x+9$ and $S(x) = x'$ and $P(x') = x$ and $i \in \{3, \ldots, 8x'+1\}$ then $C(i, j, k) = 2$.

The other segments are done in a similar way, and so the second claim follows. This completes the proof of hardness.

□

2.5 Related Work on Computing Nash Equilibria and Other Fixed Points

Over the past fifty years, a wealth of studies in the Economics, Optimization, and Computer Science literature address the problem of computing Nash equilibria. A celebrated algorithm for the case of two-player games is the Lemke-Howson algorithm [LH64], which is remarkably similar to the simplex method, and appears to be very efficient in practice. Rosenmüller [Ros71] and Wilson [Wil71] generalize the Lemke-Howson algorithm to the multi-player case; however, this generalization results in significant loss in efficiency. More practical algorithms for the multi-player case are based on general purpose methods for approximating Brouwer fixed points, most notably on algorithms that walk on simplicial subdivisions of the space where the equilibria lie, so-called *simplicial algorithms* [Sca67, GLL73, LT79, LT82, Eav72]. Despite much research on the subject, none of the existing methods for computing Nash equilibria are known to run in polynomial time, and there are negative results [HPV89], even for the Lemke-Howson algorithm [SS04].

Lipton and Markakis [LM04] study the algebraic properties of Nash equilibria and point out that standard quantifier elimination algorithms can be used to solve them, but these do not run in polynomial time in general. Papadimitriou and Roughgarden [PR05] show that, in the case of *symmetric* games, quantifier elimination results in polynomial-time algorithms for a broad range of parameters. Lipton, Markakis and Mehta [LMM03] show that, if we only require an ϵ-approximate Nash equilibrium, then a subexponential algorithm is possible. If the Nash equilibria sought are required to have any special properties, for example optimize total utility, the problem typically becomes NP-complete [GZ89, CS03].

In addition to our results in Chapter 3, other researchers have explored reductions between alternative types of games (see, e.g., [Bub79, AKV05, CSVY06, SV06]). In particular, the reductions by Bubelis [Bub79] in the 1970s comprise a remarkable early precursor of our work; it is astonishing that these important results had not been pursued for three decades. Bubelis established that the Nash equilibrium problem for 3 players captures the computational complexity of the same problem with any number of players. In Chapter 3, we show the same result in an indirect way, via the Nash equilibrium problem for graphical games — a connection that is crucial for our PPAD-completeness reduction. Bubelis also demonstrated in [Bub79] that any algebraic number can be the basis of a Nash equilibrium, something that follows easily from our results (see Theorem 4.14).

Etessami and Yannakakis study in [EY07] the problem of computing a Nash equilibrium *exactly* (a problem that is well-motivated in the context of stochastic games) and provide an interesting characterization of its complexity (considerably higher than PPAD), along with that of several other problems. In Section 4.3, we mention certain interesting results at the interface of their approach with ours.

Finally, Adler and Verma show that several important sub-classes of the Linear Complementarity Problem (LCP) belong to the class PPAD [AV07]. Using our results from Chapter 4, they also establish that some of these classes, e.g., the LCP for strictly co-positive and for strictly semi-monotone matrices, are PPAD-complete.

Chapter 3

Reductions Among Equilibrium Problems

In Chapter 4, we show that r-Nash is PPAD-hard by reducing Brouwer to it. Rather than r-Nash, it will be more convenient to first reduce Brouwer to d-graphical Nash, the problem of computing a Nash equilibrium in graphical games of degree d, defined in Section 2.2. Therefore, we need to show that the latter reduces to r-Nash. This will be the purpose of the present chapter; in fact, we will establish something stronger, namely that

Theorem 3.1. *For every fixed $d, r \geq 3$,*

- *Every r-player normal-form game and every graphical game of degree d can be mapped in polynomial time to (a) a 3-player normal-form game and (b) a graphical game with degree 3 and 2 strategies per player, such that there is a polynomial-time computable surjective mapping from the set of Nash equilibria of the latter to the set of Nash equilibria of the former.*

- *There are polynomial-time reductions from r-Nash and d-graphical Nash to both 3-Nash and 3-graphical Nash.*

Note that the first part of the theorem establishes mappings of exact equilibrium points among different games, whereas the second asserts that computing approximate equilibrium points in all these games is polynomial-time equivalent. The proof, which is quite involved, is presented in the following sections. In Section 3.1, we present some useful ideas that enable the reductions described in Theorem 3.1, as well as prepare the necessary machinery for the reduction from BROUWER to d-GRAPHICAL NASH in Section 4.1. Sections 3.2 through 3.6 provide the proof of the theorem. We note that a mapping from r-player games to 3-player games was already known by Bubelis [Bub79].

In Section 3.7, we establish the computational equivalence of the two notions of approximation discussed in Section 2.1. In particular, we give a polynomial-time reduction from the problem of computing an approximately well supported Nash equilibrium to the problem of computing an approximate Nash equilibrium. The opposite reduction is trivial, as discussed in Section 2.1.

3.1 Preliminaries: Game Gadgets

We describe the building blocks of our constructions. As we have observed earlier, if a player v has two pure strategies, say 0 and 1, then every mixed strategy of that player corresponds to a real number $\mathbf{p}[v] \in [0, 1]$ which is precisely the probability that the player plays strategy 1. Identifying players with these numbers, we are interested in constructing games that perform simple arithmetical operations on mixed strategies; for example, we are interested in constructing a game with two "input" players v_1 and v_2 and another "output" player v_3 so that in any Nash equilibrium the latter plays the sum of the former, i.e., $\mathbf{p}[v_3] = \min\{\mathbf{p}[v_1] + \mathbf{p}[v_2], 1\}$. Such constructions are considered below.

Notation: We use $x = y \pm \epsilon$ to denote $y - \epsilon \leq x \leq y + \epsilon$.

Proposition 3.2. *Let α be a non-negative real number. Let v_1, v_2, w be players in a graphical game $\mathcal{GG}$ with two strategies per player, and suppose that the payoffs to v_2 and w are as follows.*

Payoffs to v_2 :

	w *plays* 0	w *plays* 1
v_2 *plays* 0	0	1
v_2 *plays* 1	1	0

Payoffs to w:

w plays 0

	v_2 *plays* 0	v_2 *plays* 1
v_1 *plays* 0	0	0
v_1 *plays* 1	α	α

w plays 1

	v_2 *plays* 0	v_2 *plays* 1
v_1 *plays* 0	0	1
v_1 *plays* 1	0	1

Then, for $\epsilon < 1$, in every ϵ-Nash equilibrium of game $\mathcal{GG}$, $\mathbf{p}[v_2] = \min(\alpha \mathbf{p}[v_1], 1) \pm \epsilon$. In particular, in every Nash equilibrium of game $\mathcal{GG}$, $\mathbf{p}[v_2] = \min(\alpha \mathbf{p}[v_1], 1)$.

Proof. If w plays 1, then the expected payoff to w is $\mathbf{p}[v_2]$, and, if w plays 0, the expected payoff to w is $\alpha \mathbf{p}[v_1]$. Therefore, in an ϵ-Nash equilibrium of $\mathcal{GG}$, if $\mathbf{p}[v_2] > \alpha \mathbf{p}[v_1] + \epsilon$ then $\mathbf{p}[w] = 1$. However, note also that if $\mathbf{p}[w] = 1$ then $\mathbf{p}[v_2] = 0$. (Payoffs to v_2 make it prefer to disagree with w.) Consequently, $\mathbf{p}[v_2]$ cannot be larger than $\alpha \mathbf{p}[v_1] + \epsilon$, so it cannot be larger than $\min(\alpha \mathbf{p}[v_1], 1) + \epsilon$. Similarly, if $\mathbf{p}[v_2] < \min(\alpha \mathbf{p}[v_1], 1) - \epsilon$, then $\mathbf{p}[v_2] < \alpha \mathbf{p}[v_1] - \epsilon$, so $\mathbf{p}[w] = 0$, which implies —again since v_2 has the biggest payoff by disagreeing with w— that $\mathbf{p}[v_2] = 1 \geq$

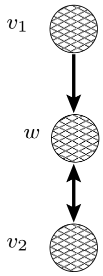

Figure 3.1: $\mathcal{G}_{\times\alpha}$, $\mathcal{G}_=$

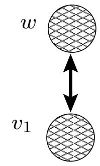

Figure 3.2: $\mathcal{G}_\alpha$

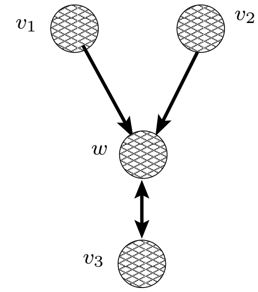

Figure 3.3: $\mathcal{G}_+$,$\mathcal{G}_*$,$\mathcal{G}_-$

$1-\epsilon$, a contradiction to $\mathbf{p}[v_2] < \min(\alpha\mathbf{p}[v_1], 1) - \epsilon$. Hence $\mathbf{p}[v_2]$ cannot be less than $\min(\alpha\mathbf{p}[v_1], 1) - \epsilon$. □

We will denote by $\mathcal{G}_{\times\alpha}$ the (directed) graphical game shown in Figure 3.1, where the payoffs to players v_2 and w are specified as in Proposition 3.2 and the payoff of player v_1 is completely unconstrained: v_1 could have any dependence on other players of a larger graphical game $\mathcal{GG}$ that contains $\mathcal{G}_{\times\alpha}$ or even depend on the strategies of v_2 and w; as long as the payoffs of v_2 and w are specified as above the conclusion of the proposition will be true. Note in particular that using the above construction with $\alpha = 1$, v_2 becomes a "copy" of v_1; we denote the corresponding graphical game by $\mathcal{G}_=$. These graphical games will be used as building blocks in our constructions; the way to incorporate them into some larger graphical game is to make player v_1 depend (incoming edges) on other players of the game and make v_2 affect (outgoing edges) other players of the game. For example, we can make a sequence of copies of any vertex, which form a path in the graph. The copies will then alternate with distinct w vertices.

Proposition 3.3. *Let α, β, γ be non-negative real numbers. Let v_1, v_2, v_3, w be players in a graphical game $\mathcal{GG}$ with two strategies per player, and suppose that the* *payoffs to v_3 and w are as follows.*

Payoffs to v_3 :

	w plays 0	*w plays 1*
v_3 plays 0	0	1
v_3 plays 1	1	0

Payoffs to w:

w plays 0

	v_2 plays 0	*v_2 plays 1*
v_1 plays 0	0	β
v_1 plays 1	α	$\alpha + \beta + \gamma$

w plays 1

v_3 plays 0	0
v_3 plays 1	1

Then, for $\epsilon < 1$, in any ϵ-Nash equilibrium of $\mathcal{GG}$, $\mathbf{p}[v_3] = \min(\alpha\mathbf{p}[v_1] + \beta\mathbf{p}[v_2] + \gamma\mathbf{p}[v_1]\mathbf{p}[v_2], 1) \pm \epsilon$. In particular, in every Nash equilibrium of $\mathcal{GG}$, $\mathbf{p}[v_3] = \min(\alpha\mathbf{p}[v_1] + \beta\mathbf{p}[v_2] + \gamma\mathbf{p}[v_1]\mathbf{p}[v_2], 1)$.

Proof. If w plays 1, then the expected payoff to w is $\mathbf{p}[v_3]$, and if w plays 0 then the expected payoff to w is $\alpha\mathbf{p}[v_1] + \beta\mathbf{p}[v_2] + \gamma\mathbf{p}[v_1]\mathbf{p}[v_2]$. Therefore, in an ϵ-Nash equilibrium of $\mathcal{GG}$, if $\mathbf{p}[v_3] > \alpha\mathbf{p}[v_1]+\beta\mathbf{p}[v_2]+\gamma\mathbf{p}[v_1]\mathbf{p}[v_2]+\epsilon$ then $\mathbf{p}[w] = 1$. However, note from the payoffs to v_3 that if $\mathbf{p}[w] = 1$ then $\mathbf{p}[v_3] = 0$. Consequently, $\mathbf{p}[v_3]$ cannot be strictly larger than $\alpha\mathbf{p}[v_1] + \beta\mathbf{p}[v_2] + \gamma\mathbf{p}[v_1]\mathbf{p}[v_2] + \epsilon$. Similarly, if $\mathbf{p}[v_3] < \min(\alpha\mathbf{p}[v_1]+\beta\mathbf{p}[v_2]+\gamma\mathbf{p}[v_1]\mathbf{p}[v_2], 1)-\epsilon$, then $\mathbf{p}[v_3] < \alpha\mathbf{p}[v_1]+\beta\mathbf{p}[v_2]+\gamma\mathbf{p}[v_1]\mathbf{p}[v_2]-\epsilon$ and, due to the payoffs to w, $\mathbf{p}[w] = 0$. This in turn implies —since v_3 has the biggest payoff by disagreeing with w— that $\mathbf{p}[v_3] = 1 \geq 1 - \epsilon$, a contradiction to $\mathbf{p}[v_3] < \min(\alpha\mathbf{p}[v_1] + \beta\mathbf{p}[v_2] + \gamma\mathbf{p}[v_1]\mathbf{p}[v_2], 1) - \epsilon$. Hence $\mathbf{p}[v_3]$ cannot be less than $\min(\alpha\mathbf{p}[v_1] + \beta\mathbf{p}[v_2] + \gamma\mathbf{p}[v_1]\mathbf{p}[v_2], 1) - \epsilon$. $\square$

Remark 3.4. *It is not hard to verify that, if v_1, v_2, v_3, w are players of a graphical game $\mathcal{GG}$ and the payoffs to v_3, w are specified as in Proposition 3.3 with $\alpha = 1$, $\beta = -1$ and $\gamma = 0$, then, in every ϵ-Nash equilibrium of the game $\mathcal{GG}$, $\mathbf{p}[v_3] = \max(0, \mathbf{p}[v_1] - \mathbf{p}[v_2]) \pm \epsilon$; in particular, in every Nash equilibrium, $\mathbf{p}[v_3] = \max(0, \mathbf{p}[v_1] - \mathbf{p}[v_2])$.*

Let us denote by $\mathcal{G}_+$ and $\mathcal{G}_*$ the (directed) graphical game shown in Figure 3.3, where the payoffs to players v_3 and w are specified as in Proposition 3.3 taking (α, β, γ) equal to $(1, 1, 0)$ (addition) and $(0, 0, 1)$ (multiplication) respectively. Also, let $\mathcal{G}_-$ be the game when the payoffs of v_3 and w are specified as in Remark 3.4.

Proposition 3.5. *Let v_1, v_2, v_3, v_4, v_5, v_6, w_1, w_2, w_3, w_4 be vertices in a graphical game $\mathcal{GG}$ with two strategies per player, and suppose that the payoffs to vertices other than v_1 and v_2 are as follows.*

Payoffs to w_1:

w_1 *plays* 0

	v_2 *plays* 0	v_2 *plays* 1
v_1 *plays* 0	0	0
v_1 *plays* 1	1	1

w_1 *plays* 1

	v_2 *plays* 0	v_2 *plays* 1
v_1 *plays* 0	0	1
v_1 *plays* 1	0	1

Payoffs to v_5 :

	w_1 *plays* 0	w_1 *plays* 1
v_5 *plays* 0	1	0
v_5 *plays* 1	0	1

Payoffs to w_2 and v_3 are chosen using Proposition 3.3 to ensure $\mathbf{p}[v_3] = \mathbf{p}[v_1](1-$

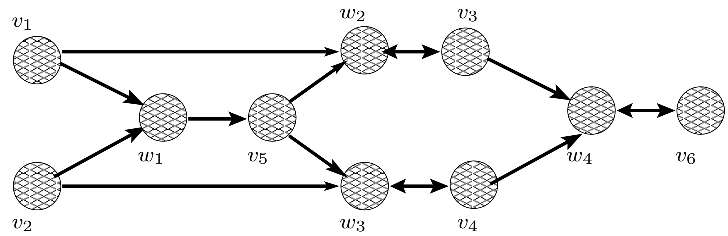

Figure 3.4: $\mathcal{G}_{\max}$

$\mathbf{p}[v_5]) \pm \epsilon$, [1] *in every ϵ-Nash equilibrium of game $\mathcal{GG}$.*

Payoffs to w_3 and v_4 are chosen using Proposition 3.3 to ensure $\mathbf{p}[v_4] = \mathbf{p}[v_2]\mathbf{p}[v_5] \pm \epsilon$, in every ϵ-Nash equilibrium of game $\mathcal{GG}$.

Payoffs to w_4 and v_6 are chosen using Proposition 3.3 to ensure that, in every ϵ-Nash equilibrium of game $\mathcal{GG}$, $\mathbf{p}[v_6] = \min(1, \mathbf{p}[v_3] + \mathbf{p}[v_4]) \pm \epsilon$.

Then, for $\epsilon < 1$, in every ϵ-Nash equilibrium of game $\mathcal{GG}$, $\mathbf{p}[v_6] = \max(\mathbf{p}[v_1], \mathbf{p}[v_2]) \pm 4\epsilon$. In particular, in every Nash equilibrium, $\mathbf{p}[v_6] = \max(\mathbf{p}[v_1], \mathbf{p}[v_2])$.

The graph of the game looks as in Figure 3.4. It is actually possible to "merge" w_1 and v_5, but we prefer to keep the game as is in order to maintain the bipartite structure of the graph in which one side of the partition contains all the vertices corresponding to arithmetic expressions (the v_i vertices) and the other side all the intermediate w_i vertices.

Proof. If, in an ϵ-Nash equilibrium, we have $\mathbf{p}[v_1] < \mathbf{p}[v_2] - \epsilon$, then it follows from w_1's payoffs that $\mathbf{p}[w_1] = 1$. It then follows that $\mathbf{p}[v_5] = 1$ since v_5's payoffs induce it to imitate w_1. Hence, $\mathbf{p}[v_3] = \pm\epsilon$ and $\mathbf{p}[v_4] = \mathbf{p}[v_2] \pm \epsilon$, and, consequently, $\mathbf{p}[v_3] + \mathbf{p}[v_4] = \mathbf{p}[v_2] \pm 2\epsilon$. This implies $\mathbf{p}[v_6] = \mathbf{p}[v_2] \pm 3\epsilon$, as required. A similar argument shows that, if $\mathbf{p}[v_1] > \mathbf{p}[v_2] + \epsilon$, then $\mathbf{p}[v_6] = \mathbf{p}[v_1] \pm 3\epsilon$.

[1] We can use Proposition 3.3 to multiply by $(1 - \mathbf{p}[v_5])$ in a similar way to multiplication by $\mathbf{p}[v_5]$; the payoffs to w_2 have v_5's strategies reversed.

If $|\mathbf{p}[v_1] - \mathbf{p}[v_2]| \leq \epsilon$, then $\mathbf{p}[w_1]$ and, consequently, $\mathbf{p}[v_5]$ may take any value. Assuming, without loss of generality that $\mathbf{p}[v_1] \geq \mathbf{p}[v_2]$, we have

$$\begin{aligned}\mathbf{p}[v_3] &= \mathbf{p}[v_1](1 - \mathbf{p}[v_5]) \pm \epsilon \\ \mathbf{p}[v_4] &= \mathbf{p}[v_2]\mathbf{p}[v_5] \pm \epsilon = \mathbf{p}[v_1]\mathbf{p}[v_5] \pm 2\epsilon,\end{aligned}$$

which implies

$$\mathbf{p}[v_3] + \mathbf{p}[v_4] = \mathbf{p}[v_1] \pm 3\epsilon,$$

and, therefore,

$$\mathbf{p}[v_6] = \mathbf{p}[v_1] \pm 4\epsilon, \text{ as required.}$$

□

We conclude the section with the simple construction of a graphical game $\mathcal{G}_\alpha$, depicted in Figure 3.2, which performs the assignment of some fixed value $\alpha \geq 0$ to a player. The proof is similar in spirit to our proof of Propositions 3.2 and 3.3 and will be skipped.

Proposition 3.6. *Let α be a non-negative real number. Let w, v_1 be players in a graphical game $\mathcal{GG}$ with two strategies per player and let the payoffs to w, v_1 be specified as follows.*

Payoffs to v_1 :

	w plays 0	*w plays 1*
v_1 plays 0	0	1
v_1 plays 1	1	0

Payoffs to w :

	v_1 plays 0	*v_1 plays 1*
w plays 0	α	α
w plays 1	0	1

Then, for $\epsilon < 1$, in every ϵ-Nash equilibrium of game $\mathcal{GG}$, $\mathbf{p}[v_1] = \min(\alpha, 1) \pm \epsilon$. In particular, in every Nash equilibrium of $\mathcal{GG}$, $\mathbf{p}[v_1] = \min(\alpha, 1)$.

Before concluding the section we give a useful definition.

Definition 3.7. *Let $v_1, v_2, \ldots, v_k, v$ be players of a graphical game $\mathcal{G}_f$ such that, in every Nash equilibrium, it holds that $\mathbf{p}[v] = f(\mathbf{p}[v_1], \ldots, \mathbf{p}[v_k])$, where f is some function with k arguments and range $[0, 1]$. We say that the game $\mathcal{G}_f$ has* error amplification at most c *if, in every ϵ-Nash equilibrium, it holds that $\mathbf{p}[v] = f(\mathbf{p}[v_1], \ldots, \mathbf{p}[v_k]) \pm c\epsilon$.*

In particular, the games $\mathcal{G}_=$, $\mathcal{G}_+$, $\mathcal{G}_-$, $\mathcal{G}_*$, $\mathcal{G}_\alpha$ described above have error amplifications at most 1, whereas the game $\mathcal{G}_{\max}$ has error amplification at most 4.

3.2 Reducing Graphical Games to Normal-Form Games

We establish a mapping from graphical games to normal-form games as specified by the following theorem.

Theorem 3.8. *For every $d > 1$, a graphical game (directed or undirected) $\mathcal{GG}$ of maximum degree d can be mapped in polynomial time to a $(d^2 + 1)$-player normal-form game $\mathcal{G}$ so that there is a polynomial-time computable surjective mapping g from the Nash equilibria of the latter to the Nash equilibria of the former.*

Proof.

Overview:

Figure 3.5 shows the construction of $\mathcal{G} = f(\mathcal{GG})$. We will explain the construction in detail as well as show that it can be computed in polynomial time. We will also establish that there is a surjective mapping from the Nash equilibria of $\mathcal{G}$ to the Nash equilibria of $\mathcal{GG}$. In the following discussion we will refer to the players of the graphical game as "vertices" to distinguish them from the players of the normal-form game.

Input: Degree d graphical game $\mathcal{GG}$: vertices V, $|V| = n'$, $|S_v| = t$ for all $v \in V$.
Output: Normal-form game $\mathcal{G}$.

1. If needed, rescale the entries in the payoff tables of $\mathcal{GG}$ so that they lie in the range $[0, 1]$. One way to do so is to divide all payoff entries by $\max\{u\}$, where $\max\{u\}$ is the largest entry in the payoff tables of $\mathcal{GG}$.
2. Let $r = d^2$ or $r = d^2 + 1$; r chosen to be even.
3. Let $c : V \longrightarrow \{1, \dots, r\}$ be a r-coloring of $\mathcal{GG}$ such that no two adjacent vertices have the same color, and, furthermore, no two vertices having a common successor —in the affects graph of the game— have the same color. Assume that each color is assigned to the same number of vertices, adding to V extra isolated vertices to make up any shortfall; extend mapping c to these vertices. Let $\{v_1^{(i)}, \dots, v_{n/r}^{(i)}\}$ denote $\{v \ : \ c(v) = i\}$, where $n \geq n'$.
4. For each $p \in [r]$, game $\mathcal{G}$ will have a player, labeled p, with strategy set S_p; S_p will be the union (assumed disjoint) of all S_v with $c(v) = p$, i.e.,
$$S_p = \{(v, a) \ : \ c(v) = p, a \in S_v\}, \ \ |S_p| = t \ \tfrac{n}{r}.$$
5. Taking S to be the cartesian product of the S_p's, let $s \in S$ be a strategy profile of game $\mathcal{G}$. For $p \in [r]$, u_s^p is defined as follows:
   (a) Initially, all utilities are 0.
   (b) For $v_0 \in V$ having predecessors $v_1, \dots, v_{d'}$ in the affects graph of $\mathcal{GG}$, if $c(v_0) = p$ (that is, $v_0 = v_j^{(p)}$ for some j) and, for $i = 0, \dots, d'$, s contains (v_i, a_i), then $u_s^p = u_{s'}^{v_0}$ for s' a strategy profile of $\mathcal{GG}$ in which v_i plays a_i for $i = 0, \dots, d'$.
   (c) Let $M > 2 \ \frac{n}{r}$.
   (d) For odd number $p < r$, if player p plays $(v_i^{(p)}, a)$ and $p + 1$ plays $(v_i^{(p+1)}, a')$, for any i, a, a', then add M to u_s^p and subtract M from u_s^{p+1}.

Figure 3.5: Reduction from the graphical game $\mathcal{GG}$ to the normal-form game $\mathcal{G}$

We first rescale all payoffs so that they are nonnegative and at most 1 (Step 1); it is easy to see that the set of Nash equilibria is preserved under this transformation. Also, without loss of generality, we assume that all vertices $v \in V$ have the same number of strategies, $|S_v| = t$. We color the vertices of G, where $G = (V, E)$ is the affects graph of $\mathcal{GG}$, so that any two adjacent vertices have different colors, but also any two vertices with a common successor have different colors (Step 3). Since this type of coloring will be important for our discussion we will define it formally.

Definition 3.9. *Let $\mathcal{GG}$ be a graphical game with affects graph $G = (V, E)$. We say that $\mathcal{GG}$ can be* legally colored with k colors *if there exists a mapping $c : V \to \{1, 2, \dots, k\}$ such that, for all $e = (v, u) \in E$, $c(v) \neq c(u)$ and, moreover, for all $e_1 = (v, w), e_2 = (u, w) \in E$ with $v \neq u$, $c(v) \neq c(u)$. We call such coloring a* legal k-coloring of $\mathcal{GG}$.

To get such coloring, it is sufficient to color the union of the underlying undirected graph G' with its square (with self-loops removed) so that no adjacent vertices have the same color; this can be done with at most d^2 colors —see, e.g., [CKK$^+$00]— since G' has degree d by assumption; we are going to use $r = d^2$ or $r = d^2 + 1$ colors, whichever is even, for reasons to become clear shortly. We assume for simplicity that each color class has the same number of vertices, adding dummy vertices if needed to satisfy this property. Henceforth, we assume that n is an integer multiple of r so that every color class has $\frac{n}{r}$ vertices.

We construct a normal-form game $\mathcal{G}$ with $r \leq d^2 + 1$ players. Each of them corresponds to a color and has $t\frac{n}{r}$ strategies, the t strategies of each of the $\frac{n}{r}$ vertices in its color class (Step 4). Since r is even, we can divide the r players into pairs and make each pair play a generalized Matching Pennies game (see Definition 3.10 below) at very high stakes, so as to ensure that all players will randomize uniformly over the vertices assigned to them. [2] Within the set of strategies associated with each vertex, the Matching Pennies game expresses no preference, and payoffs are augmented to correspond to the payoffs that would arise in the original graphical game $\mathcal{GG}$ (see

[2] A similar trick is used in Theorem 7.3 of [SV06], a hardness result for a class of circuit games.

Step 5 for the exact specification of the payoffs).

Definition 3.10. *The (2-player) game* Generalized Matching Pennies *is defined as follows. Call the 2 players the* pursuer *and the* evader*, and let* $[n]$ *denote their strategies. If for any* $i \in [n]$ *both players play* i*, then the pursuer receives a positive payoff* $u > 0$ *and the evader receives a payoff of* $-u$*. Otherwise both players receive* 0*. It is not hard to check that the game has a unique Nash equilibrium in which both players use the uniform distribution.*

Polynomial size of $\mathcal{G} = f(\mathcal{GG})$:

The input size is $|\mathcal{GG}| = \Theta(n' \cdot t^{d+1} \cdot q)$, where n' is the number of vertices in $\mathcal{GG}$ and q the size of the values in the payoff matrices in the logarithmic cost model. The normal-form game $\mathcal{G}$ has $r \in \{d^2, d^2 + 1\}$ players, each having tn/r strategies, where $n \leq rn'$ is the number of vertices in $\mathcal{GG}$ after the possible addition of dummy vertices to make sure that all color classes have the same number of vertices. Hence, there are $r \cdot \left(tn/r\right)^r \leq \left((d^2+1)\left(tn'\right)^{d^2+1}\right)$ payoff entries in $\mathcal{G}$. This is polynomial in $|\mathcal{G}G|$ so long as d is constant. Moreover, each payoff entry will be of polynomial size since M is of polynomial size and each payoff entry of the game $\mathcal{G}$ is the sum of 0 or M and a payoff entry of $\mathcal{GG}$.

Construction of the mapping g:

Given a Nash equilibrium $N_{\mathcal{G}} = \{x^p_{(v,a)}\}_{p,v,a}$ of $\mathcal{G} = f(\mathcal{GG})$, we claim that we can recover a Nash equilibrium $\{x^v_a\}_{v,a}$ of $\mathcal{GG}$, $N_{\mathcal{GG}} = g(N_{\mathcal{G}})$, as follows:

$$x^v_a := x^{c(v)}_{(v,a)} \Big/ \sum_{j \in S_v} x^{c(v)}_{(v,j)}, \quad \forall a \in S_v, v \in V. \tag{3.1}$$

Clearly g is computable in polynomial time.

Proof that g maps Nash equilibria of $\mathcal{G}$ to Nash equilibria of $\mathcal{GG}$:

Call $\mathcal{GG}'$ the graphical game resulting from $\mathcal{GG}$ by rescaling the utilities so that they lie in the range $[0, 1]$. It is easy to see that any Nash equilibrium of game $\mathcal{GG}$ is, also, a Nash equilibrium of game $\mathcal{GG}'$ and vice versa. Therefore, it is enough to establish that the mapping g maps every Nash equilibrium of game $\mathcal{G}$ to a Nash equilibrium of game $\mathcal{GG}'$.

For $v \in V$, $c(v) = p$, let "p plays v" denote the event that p plays (v, a) for some $a \in S_v$. We show that in a Nash equilibrium $N_{\mathcal{G}}$ of game $\mathcal{G}$, for every player p and every $v \in V$ with $c(v) = p$, $\Pr[p \text{ plays } v] \in [\lambda - \frac{1}{M}, \lambda + \frac{1}{M}]$, where $\lambda = \binom{n}{r}^{-1}$. Note that the "fair share" for v is λ.

Lemma 3.11. *For all $v \in V$, in a Nash equilibrium of $\mathcal{G}$,* $\Pr[c(v) \text{ plays } v] \in [\lambda - \frac{1}{M}, \lambda + \frac{1}{M}]$.

Proof. Suppose, for a contradiction, that in a Nash equilibrium of $\mathcal{G}$, $\Pr\left[p \text{ plays } v_i^{(p)}\right] < \lambda - \frac{1}{M}$ for some i, p. Then there exists some j such that $\Pr\left[p \text{ plays } v_j^{(p)}\right] > \lambda + \frac{1}{M}\lambda$.

If p is odd (a pursuer) then $p+1$ (the evader) will have utility of at least $-\lambda M+1$ for playing any strategy $\left(v_i^{(p+1)}, a\right)$, $a \in S_{v_i^{(p+1)}}$, whereas utility of at most $-\lambda M-\lambda+1$ for playing any strategy $(v_j^{(p+1)}, a)$, $a \in S_{v_j^{(p+1)}}$. Since $-\lambda M+1 > -\lambda M-\lambda+1$, in a Nash equilibrium, $\Pr\left[p+1 \text{ plays } v_j^{(p+1)}\right] = 0$. Therefore, there exists some k such that $\Pr\left[p+1 \text{ plays } v_k^{(p+1)}\right] > \lambda$. Now the payoff of p for playing any strategy $\left(v_j^{(p)}, a\right)$, $a \in S_{v_j^{(p)}}$, is at most 1, whereas the payoff for playing any strategy $\left(v_k^{(p)}, a\right)$, $a \in S_{v_k^{(p)}}$ is at least λM. Thus, in a Nash equilibrium, player p should not include any strategy $\left(v_j^{(p)}, a\right)$, $a \in S_{v_j^{(p)}}$, in her support; hence $\Pr\left[p \text{ plays } v_j^{(p)}\right] = 0$, a contradiction.

If p is even, then $p-1$ will have utility of at most $(\lambda - \frac{1}{M})M+1$ for playing any strategy $\left(v_i^{(p-1)}, a\right)$, $a \in S_{v_i^{(p-1)}}$, whereas utility of at least $(\lambda + \frac{1}{M}\lambda)M$ for playing any strategy $(v_j^{(p-1)}, a)$, $a \in S_{v_j^{(p-1)}}$. Hence, in a Nash equilibrium $\Pr\left[p-1 \text{ plays } v_i^{(p-1)}\right] = 0$, which implies that there exists some k such that $\Pr\left[p-1 \text{ plays } v_k^{(p-1)}\right] > \lambda$. But, p will then have utility of at least 0 for playing any strategy $\left(v_i^{(p)}, a\right)$, $a \in S_{v_i^{(p)}}$, whereas utility of at most $-\lambda M+1$ for playing any strategy $(v_k^{(p)}, a)$, $a \in S_{v_k^{(p)}}$. Since

$0 > -\lambda M + 1$, in a Nash equilibrium, $\Pr\left[p \text{ plays } v_k^{(p)}\right] = 0$. Therefore, there exists some k' such that $\Pr\left[p \text{ plays } v_{k'}^{(p)}\right] > \lambda$. Now the payoff of $p-1$ for playing any strategy $\left(v_k^{(p-1)}, a\right)$, $a \in S_{v_k^{(p-1)}}$, is at most 1, whereas the payoff for playing any strategy $\left(v_{k'}^{(p-1)}, a\right)$, $a \in S_{v_{k'}^{(p-1)}}$ is at least λM. Thus, in a Nash equilibrium, player $p-1$ should not include any strategy $\left(v_k^{(p-1)}, a\right)$, $a \in S_{v_k^{(p-1)}}$, in her support; hence $\Pr\left[p-1 \text{ plays } v_k^{(p-1)}\right] = 0$, a contradiction.

From the above discussion, it follows that every vertex is chosen with probability at least $\lambda - \frac{1}{M}$ by the player that represents its color class. A similar argument shows that no vertex is chosen with probability greater than $\lambda + \frac{1}{M}$. Indeed, suppose, for a contradiction, that in a Nash equilibrium of $\mathcal{G}$, $\Pr\left[p \text{ plays } v_j^{(p)}\right] > \lambda + \frac{1}{M}$ for some j, p; then there exists some i such that $\Pr\left[p \text{ plays } v_i^{(p)}\right] < \lambda - \frac{1}{M}\lambda$; now, distinguish two cases depending on whether p is even or odd and proceed in the same fashion as in the argument used above to show that no vertex is chosen with probability smaller than $\lambda - 1/M$. □

To see that $\{x_a^v\}_{v,a}$, defined by (3.1), corresponds to a Nash equilibrium of $\mathcal{GG}'$ note that, for any player p and vertex $v \in V$ such that $c(v) = p$, the division of $\Pr[p \text{ plays } v]$ into $\Pr[p \text{ plays } (v,a)]$, for various values of $a \in S_v$, is driven entirely by the same payoffs as in $\mathcal{GG}'$; moreover, note that there is some positive probability $p(v) \geq (\lambda - \frac{1}{M})^d > 0$ that the predecessors of v are chosen by the other players of $\mathcal{G}$ and the additional expected payoff to p resulting from choosing (v,a), for some $a \in S_v$, is $p(v)$ times the expected payoff of v in $\mathcal{GG}'$ if v chooses action a and all other vertices play as specified by (3.1). More formally, suppose that $p = c(v)$ for some vertex v of the graphical game $\mathcal{GG}'$ and, without loss of generality, assume that p is odd (pursuer) and that v is the vertex $v_i^{(p)}$ in the notation of Figure 3.5. Then, in a Nash equilibrium of the game $\mathcal{G}$, we have, by the definition of a Nash equilibrium, that for all strategies $a, a' \in S_v$ of vertex v:

$$\mathbb{E}\left[\text{payoff to } p \text{ for playing } (v,a)\right] > \mathbb{E}\left[\text{payoff to } p \text{ for playing } (v,a')\right] \Rightarrow x^p_{(v,a')} = 0. \tag{3.2}$$

But

$$\mathbb{E}\left[\text{payoff to } p \text{ for playing } (v,a)\right] = \\ M \cdot \Pr\left[p+1 \text{ plays } v_i^{(p+1)}\right] + \sum_{s \in S_{\mathcal{N}(v)\setminus\{v\}}} u^v_{as} \prod_{u \in \mathcal{N}(v)\setminus\{v\}} x^{c(u)}_{(u,s_u)}$$

and, similarly, for a'. Therefore, (3.2) implies

$$\sum_{s \in S_{\mathcal{N}(v)\setminus\{v\}}} u^v_{as} \prod_{u \in \mathcal{N}(v)\setminus\{v\}} x^{c(u)}_{(u,s_u)} > \sum_{s \in S_{\mathcal{N}(v)\setminus\{v\}}} u^v_{a's} \prod_{u \in \mathcal{N}(v)\setminus\{v\}} x^{c(u)}_{(u,s_u)} \Rightarrow x^p_{(v,a')} = 0.$$

Dividing by $\prod_{u \in \mathcal{N}(v)\setminus\{v\}} \sum_{j \in S_u} x^{c(u)}_{(u,j)} = \prod_{u \in \mathcal{N}(v)\setminus\{v\}} \Pr\left[c(u) \text{ plays } u\right] = p(v)$ and invoking (3.1) gives

$$\sum_{s \in S_{\mathcal{N}(v)\setminus\{v\}}} u^v_{as} \prod_{u \in \mathcal{N}(v)\setminus\{v\}} x^u_{s_u} > \sum_{s \in S_{\mathcal{N}(v)\setminus\{v\}}} u^v_{a's} \prod_{u \in \mathcal{N}(v)\setminus\{v\}} x^u_{s_u} \Rightarrow x^v_{a'} = 0,$$

where we used that $p(v) \geq (\lambda - \frac{1}{M})^d > 0$, which follows by Lemma 3.11.

Mapping g is surjective on the Nash equilibria of $\mathcal{GG}'$ and, therefore, $\mathcal{GG}$: We will show that, for every Nash equilibrium $N_{\mathcal{GG}'} = \{x^v_a\}_{v,a}$ of $\mathcal{GG}'$, there exists a Nash equilibrium $N_{\mathcal{G}} = \{x^p_{(v,a)}\}_{p,v,a}$ of $\mathcal{G}$ such that (3.1) holds. The existence can be easily established via the existence of a Nash equilibrium in a game $\mathcal{G}'$ defined as follows. Suppose that, in $N_{\mathcal{GG}'}$, every vertex $v \in V$ receives an expected payoff of u_v from every strategy in the support of $\{x^v_a\}_a$. Define the following game $\mathcal{G}'$ whose structure results from $\mathcal{G}$ by merging the strategies $\{(v,a)\}_a$ of player $p = c(v)$ into one strategy s_v^p, for every v such that $c(v) = p$. So the strategy set of player p in $\mathcal{G}'$ will be $\{s_v^p \mid c(v) = p\}$ also denoted as $\{s_1^{(p)}, \ldots, s_{n/r}^{(p)}\}$ for ease of notation. Define now the payoffs to the players as follows. Initialize the payoff matrices with all entries equal to 0. For every strategy profile s,

- for $v_0 \in V$ having predecessors $v_1, \ldots, v_{d'}$ in the affects graph of $\mathcal{G}\mathcal{G}'$, if, for $i = 0, \ldots, d'$, s contains $s_{v_i}^{c(v_i)}$, then add u_{v_0} to $u_s^{c(v_0)}$.

- for odd number $p < r$ if player p plays strategy $s_i^{(p)}$ and player $p{+}1$ plays strategy $s_i^{(p+1)}$ then add M to u_s^p and subtract M from u_s^{p+1} (Generalized Matching Pennies).

Note the similarity between the definitions of the payoff matrices of $\mathcal{G}$ and $\mathcal{G}'$. From Nash's theorem, game $\mathcal{G}'$ has a Nash equilibrium $\{y_{s_v^p}^p\}_{p,v}$ and it is not hard to verify that $\{x_{(v,a)}^p\}_{p,v,a}$ is a Nash equilibrium of game $\mathcal{G}$, where $x_{(v,a)}^p := y_{s_v^p}^p \cdot x_a^v$, for all p, $v \in V$ such that $c(v) = p$, and $a \in S_v$.

□

3.3 Reducing Normal-Form Games to Graphical Games

We establish the following mapping from normal-form games to graphical games.

Theorem 3.12. *For every $r > 1$, a normal-form game with r players can be mapped in polynomial time to an undirected graphical game of maximum degree 3 and two strategies per player so that there is a polynomial-time computable surjective mapping g from the Nash equilibria of the latter to the Nash equilibria of the former.*

Given a normal-form game $\mathcal{G}$ having r players, $1, \ldots, r$, and n strategies per player, say $S_p = [n]$ for all $p \in [r]$, we will construct a graphical game $\mathcal{GG}$, with a bipartite graph of maximum degree 3, and 2 strategies per player, say $\{0, 1\}$, with description length polynomial in the description length of $\mathcal{G}$, so that from every Nash equilibrium of $\mathcal{GG}$ we can recover a Nash equilibrium of $\mathcal{G}$. In the following discussion we will refer to the players of the graphical game as "vertices" to distinguish them from the players of the normal-form game. It will be easy to check that the graph of $\mathcal{GG}$ is bipartite and has degree 3; this graph will be denoted $G = (V \cup W, E)$, where W and V are disjoint, and each edge in E goes between V and W. For every vertex v of the graphical game, we will denote by $\mathbf{p}[v]$ the probability that v plays pure strategy 1.

Recall that $\mathcal{G}$ is specified by the quantities $\{u_s^p \,:\, p \in [r], s \in S\}$. A mixed strategy profile of $\mathcal{G}$ is given by probabilities $\{x_j^p \,:\, p \in [r], j \in S_p\}$. $\mathcal{GG}$ will contain a vertex $v(x_j^p) \in V$ for each player p and strategy $j \in S_p$, and the construction of $\mathcal{GG}$ will ensure that in any Nash equilibrium of $\mathcal{GG}$, the quantities $\{\mathbf{p}[v(x_j^p)] : p \in [r], j \in S_p\}$, if interpreted as values $\{x_j^p\}_{p,j}$, will constitute a Nash equilibrium of $\mathcal{G}$. Extending this notation, for various arithmetic expressions A involving any x_j^p and u_s^p, vertex $v(A) \in V$ will be used, and be constructed such that in any Nash equilibrium of $\mathcal{GG}$, $\mathbf{p}[v(A)]$ is equal to A evaluated at the given values of u_s^p and with x_j^p equal to $\mathbf{p}[v(x_j^p)]$. Elements of W are used to mediate between elements of V, so that the latter ones obey the intended arithmetic relationships.

We use Propositions 3.2-3.6 as building blocks of $\mathcal{GG}$, starting with r subgraphs that represent mixed strategies for the players of $\mathcal{G}$. In the following, we construct a graphical game containing vertices $\{v(x_j^p)\}_{j \in [n]}$, whose probabilities sum to 1, and internal vertices v_j^p, which control the distribution of the one unit of probability mass among the vertices $v(x_j^p)$. See Figure 3.6 for an illustration.

Proposition 3.13. *Consider a graphical game that contains*

- *for* $j \in [n]$ *a vertex* $v(x_j^p)$
- *for* $j \in [n-1]$ *a vertex* v_j^p
- *for* $j \in [n]$ *a vertex* $v(\sum_{i=1}^{j} x_i^p)$
- *for* $j \in [n-1]$ *a vertex* $w_j(p)$ *used to ensure* $\mathbf{p}[v(\sum_{i=1}^{j} x_i^p)] = \mathbf{p}[v(\sum_{i=1}^{j+1} x_i^p)](1 - \mathbf{p}[v_j^p])$
- *for* $j \in [n-1]$ *a vertex* $w'_j(p)$ *used to ensure* $\mathbf{p}[v(x_{j+1}^p)] = \mathbf{p}[v(\sum_{i=1}^{j+1} x_i^p)]\mathbf{p}[v_j^p]$
- *a vertex* $w'_0(p)$ *used to ensure* $\mathbf{p}[v(x_1^p)] = \mathbf{p}[v(\sum_{i=1}^{1} x_i^p)]$

Also, let $v(\sum_{i=1}^{n} x_i^p)$ *have payoff of* 1 *when it plays* 1 *and* 0 *otherwise. Then, in any Nash equilibrium of the graphical game,* $\sum_{i=1}^{n} \mathbf{p}[v(x_i^p)] = 1$ *and moreover* $\mathbf{p}[v(\sum_{i=1}^{j} x_i^p)] = \sum_{i=1}^{j} \mathbf{p}[v(x_i^p)]$*, and the graph is bipartite and of degree 3.*

Proof. It is not hard to verify that the graph has degree 3. Most of the degree 3 vertices are the w vertices used in Propositions 3.2 and 3.3 to connect the pairs or triples of graph players whose probabilities are supposed to obey an arithmetic relationship. In a Nash equilibrium, $v(\sum_{i=1}^{n} x_i^p)$ plays 1. The vertices v_j^p split the probability $\mathbf{p}[v(\sum_{i=1}^{j+1} x_i^p)]$ between $\mathbf{p}[v(\sum_{i=1}^{j} x_i^p)]$ and $\mathbf{p}[v(x_{j+1}^p)]$. □

Comment. The values $\mathbf{p}[v_j^p]$ control the distribution of probability (summing to 1) amongst the n vertices $v(x_j^p)$. These vertices can set to zero any proper subset of the probabilities $\mathbf{p}[v(x_j^p)]$.

Notation. For $s \in S_{-p}$ let $x_s = x_{s_1}^1 \cdot x_{s_2}^2 \cdots x_{s_{p-1}}^{p-1} \cdot x_{s_{p+1}}^{p+1} \cdots x_{s_r}^r$. Also, let $U_j^p = \sum_{s \in S_{-p}} u_{js}^p x_s$ be the utility to p for playing j in the context of a given mixed profile $\{x_s\}_{s \in S_{-p}}$.

Lemma 3.14. *Suppose all utilities* u_s^p *(of* $\mathcal{G}$*) lie in the range* $[0, 1]$ *for some* $p \in [r]$*. We can construct a degree 3 bipartite graph having a total of* $O(rn^r)$ *vertices, including*

The vertices whose labels include U do not form part of Proposition 3.13; they have been included to show how the gadget fits into the rest of the construction, as described in Figure 3.7. Unshaded vertices belong to V, shaded vertices belong to W (V and W being the two parts of the bipartite graph). A directed edge from u to v indicates that u's choice can affect v's payoff.

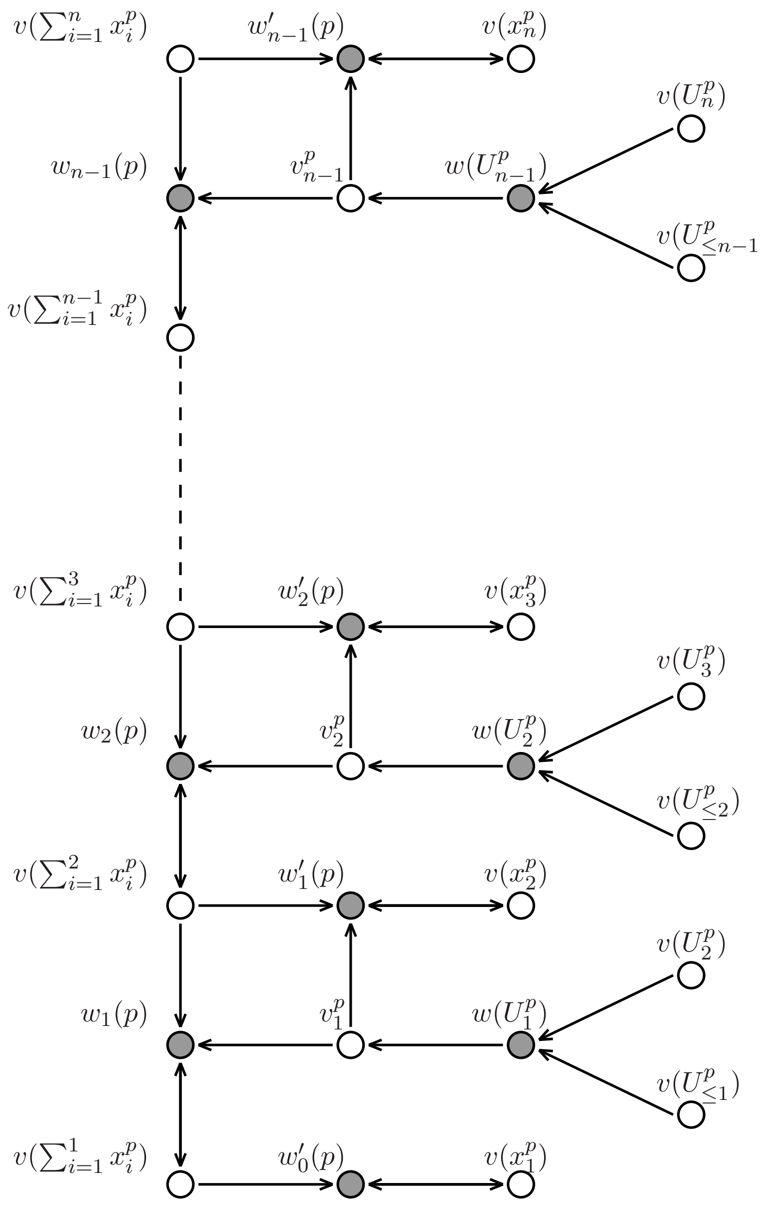

Figure 3.6: Diagram of Proposition 3.13

vertices $v(x_j^p)$, $v(U_j^p)$, $v(U_{\le j}^p)$, for all $j \in [n]$, such that in any Nash equilibrium,

$$\mathbf{p}[v(U_j^p)] = \sum_{s \in S_{-p}} u_{js}^p \prod_{q \neq p} \mathbf{p}[v(x_{s_q}^q)], \tag{3.3}$$

$$\mathbf{p}[v(U_{\le j}^p)] = \max_{i \le j} \sum_{s \in S_{-p}} u_{is}^p \prod_{q \neq p} \mathbf{p}[v(x_{s_q}^q)]. \tag{3.4}$$

The general idea is to note that the expressions for $\mathbf{p}[v(U_j^p)]$ and $\mathbf{p}[v(U_{\le j}^p)]$ are constructed from arithmetic subexpressions using the operations of addition, multiplication and maximization. If each subexpression A has a vertex $v(A)$, then using Propositions 3.2 through 3.6 we can assemble them into a graphical game such that in any Nash equilibrium, $\mathbf{p}[v(A)]$ is equal to the value of A with input $\mathbf{p}[v(x_j^p)]$, $p \in [r]$, $j \in [n]$. We just need to limit our usage to $O(rn^r)$ subexpressions and ensure that their values all lie in $[0, 1]$.

Proof. Note that

$$U_{\le j}^p = \max\{U_j^p, U_{\le j-1}^p\}, \qquad U_j^p = \sum_{s \in S_{-p}} u_{js}^p x_s = \sum_{s \in S_{-p}} u_{js}^p x_{s_1}^1 \cdots x_{s_{p-1}}^{p-1} x_{s_{p+1}}^{p+1} \cdots x_{s_r}^r.$$

Let $S_{-p} = \{S_{-p}(1), \ldots, S_{-p}(n^{r-1})\}$, so that

$$\sum_{s \in S_{-p}} u_{js}^p x_s = \sum_{\ell=1}^{n^{r-1}} u_{jS_{-p}(\ell)}^p x_{S_{-p}(\ell)}.$$

Include vertex $v(\sum_{\ell=1}^{z} u_{jS_{-p}(\ell)}^p x_{S_{-p}(\ell)})$, for each partial sum $\sum_{\ell=1}^{z} u_{jS_{-p}(\ell)}^p x_{S_{-p}(\ell)}$, $1 \le z \le n^{r-1}$. Similarly, include vertex $v(u_{js}^p \prod_{p \neq q \le z} x_{s_q}^q)$, for each partial product of the summands $u_{js}^p \prod_{p \neq q \le z} x_{s_q}^q$, $0 \le z \le r$. So, for each strategy $j \in S_p$, there are n^{r-1} partial sums and $r+1$ partial products for each summand. Then, there are n partial sequences over which we have to maximize. Note that, since all utilities are assumed to lie in the set $[0,1]$, all partial sums and products must also lie in $[0,1]$, so the truncation at 1 in the computations of Propositions 3.2, 3.3, 3.5 and 3.6 is not a problem. So using a vertex for each of the $2n + (r+1)n^r$ arithmetic subexpressions, a Nash equilibrium will compute the desired quantities. □

We repeat the construction specified by Lemma 3.14 for all $p \in [r]$. Note that, to avoid large degrees in the resulting graphical game, each time we need to make use of a value $x^q_{s_q}$ we create a new copy of the vertex $v(x^q_{s_q})$ using the gadget $\mathcal{G}_=$ and, then, use the new copy for the computation of the desired partial product; an easy calculation shows that we have to make $(r-1)n^{r-1}$ copies of $v(x^q_{s_q})$, for all $q \leq r$, $s_q \in S_q$. To limit the degree of each vertex to 3 we create a binary tree of copies of $v(x^q_{s_q})$ with $(r-1)n^{r-1}$ leaves and use each leaf once.

Proof of Theorem 3.12: Let $\mathcal{G}$ be a r-player normal-form game with n strategies per player and construct $\mathcal{GG} = f(\mathcal{G})$ as shown in Figure 3.7. The graph of $\mathcal{GG}$ has degree 3, by the graph structure of our gadgets from Propositions 3.2 through 3.6 and the fact that we use separate copies of the $v(x^p_j)$ vertices to influence different $v(U^p_j)$ vertices (see Step 4 and discussion after Lemma 3.14).

Polynomial size of $\mathcal{GG} = f(\mathcal{G})$:

The size of $\mathcal{GG}$ is polynomial in the description length $r \cdot n^r q$ of $\mathcal{G}$, where q is the size of the values in the payoff tables in the logarithmic cost model.

Construction of $g(N_{\mathcal{GG}})$ (where $N_{\mathcal{GG}}$ denotes a Nash equilibrium of $\mathcal{GG}$):

Given a Nash equilibrium $g(N_{\mathcal{GG}})$ of $\mathcal{GG}$, we claim that we can recover a Nash equilibrium $\{x^p_j\}_{p,j}$ of $\mathcal{G}$ by taking $x^p_j = \mathbf{p}[v(x^p_j)]$. This is clearly computable in polynomial time.

Proof that the reduction preserves Nash equilibria:

Call $\mathcal{G}'$ the game resulting from $\mathcal{G}$ by rescaling the utilities so that they lie in the

Input: Normal-form game $\mathcal{G}$ with r players, n strategies per player, utilities $\{u_s^p : p \in [r], s \in S\}$.
Output: Graphical game $\mathcal{GG}$ with bipartite graph $(V \cup W, E)$.

1. If needed, rescale the utilities u_s^p so that they lie in the range $[0,1]$. One way to do so is to divide all utilities by $\max\{u_s^p\}$.

2. For each player/strategy pair (p,j) let $v(x_j^p) \in V$ be a vertex in $\mathcal{GG}$.

3. For each $p \in [r]$ construct a subgraph as described in Proposition 3.13 so that in a Nash equilibrium of $\mathcal{GG}$, we have $\sum_j \mathbf{p}[v(x_j^p)] = 1$.

4. Use the construction of Proposition 3.2 with $\alpha = 1$ to make $(r-1)n^{r-1}$ copies of the $v(x_j^p)$ vertices (which are added to V). More precisely, create a binary tree with copies of $v(x_j^p)$ which has $(r-1)n^{r-1}$ leaves.

5. Use the construction of Lemma 3.14 to introduce (add to V) vertices $v(U_j^p)$, $v(U_{\leq j}^p)$, for all $p \in [r]$, $j \in [n]$. Each $v(U_j^p)$ uses its own set of copies of the vertices $v(x_j^p)$. For $p \in [r]$, $j \in [n]$ introduce (add to W) $w(U_j^p)$ with
   (a) If $w(U_j^p)$ plays 0 then $w(U_j^p)$ gets payoff 1 whenever $v(U_{\leq j}^p)$ plays 1, else 0.
   (b) If $w(U_j^p)$ plays 1 then $w(U_j^p)$ gets payoff 1 whenever $v(U_{j+1}^p)$ plays 1, else 0.

6. Give the following payoffs to the vertices v_j^p (the additional vertices used in Proposition 3.13 whose payoffs were not specified).
   (a) If v_j^p plays 0 then v_j^p has a payoff of 1 whenever $w(U_j^p)$ plays 0, otherwise 0.
   (b) If v_j^p plays 1 then v_j^p has a payoff of 1 whenever $w(U_j^p)$ plays 1, otherwise 0.

7. Return the underlying undirected graphical game $\mathcal{GG}$.

Figure 3.7: Reduction from normal-form game $\mathcal{G}$ to graphical game $\mathcal{GG}$

range $[0, 1]$. It is easy to see that any Nash equilibrium of game $\mathcal{G}$ is, also, a Nash equilibrium of game $\mathcal{G}'$ and vice versa. Therefore, it is enough to establish that the mapping $g(\cdot)$ maps every Nash equilibrium of game $\mathcal{GG}$ to a Nash equilibrium of game $\mathcal{G}'$. By Proposition 3.13, we have that $\sum_j x_j^p = 1$, for all $p \in [r]$. It remains to show that, for all p, j, j',

$$\sum_{s \in S_{-p}} u^p_{js} x_s > \sum_{s \in S_{-p}} u^p_{j's} x_s \Longrightarrow x^p_{j'} = 0.$$

We distinguish the cases:

- If there exists some $j'' < j'$ such that $\sum_{s \in S_{-p}} u^p_{j''s} x_s > \sum_{s \in S_{-p}} u^p_{j's} x_s$, then, by Lemma 3.14, $\mathbf{p}[v(U^p_{\leq j'-1})] > \mathbf{p}[v(U^p_{j'})]$. Thus, $\mathbf{p}[v^p_{j'-1}] = 0$ and, consequently, $v(x^p_{j'})$ plays 0 as required, since

$$\mathbf{p}[v(x^p_{j'})] = \mathbf{p}[v^p_{j'-1}]\mathbf{p}\left[v\left(\sum_{i=1}^{j'} x^p_i\right)\right].$$

- The case $j < j'$ reduces trivially to the previous case.

- It remains to deal with the case $j > j'$, under the assumption that, for all $j'' < j'$, $\sum_{s \in S_{-p}} u^p_{j''s} x_s \leq \sum_{s \in S_{-p}} u^p_{j's} x_s$, or, equivalently,

$$\mathbf{p}[v(U^p_{j''})] \leq \mathbf{p}[v(U^p_{j'})],$$

which in turn implies that

$$\mathbf{p}[v(U^p_{\leq j'})] \leq \mathbf{p}[v(U^p_{j'})].$$

It follows that there exists some k, $j' + 1 \leq k \leq j$, such that $\mathbf{p}[v(U^p_k)] > \mathbf{p}[v(U^p_{\leq k-1})]$. Otherwise, $\mathbf{p}[v(U^p_{\leq j'})] \geq \mathbf{p}[v(U^p_{\leq j'+1})] \geq \ldots \geq \mathbf{p}[v(U^p_{\leq j})] \geq \mathbf{p}[v(U^p_j)] >$

$\mathbf{p}[v(U^p_{j'})]$, which is a contradiction to $\mathbf{p}[v(U^p_{\leq j'})] \leq \mathbf{p}[v(U^p_{j'})]$. Since $\mathbf{p}[v(U^p_k)] > \mathbf{p}[v(U^p_{\leq k-1})]$, it follows that $\mathbf{p}[w(U^p_{k-1})] = 1 \Rightarrow \mathbf{p}[v^p_{k-1}] = 1$ and, therefore,

$$\mathbf{p}\left[v\left(\sum_{i=1}^{k-1} x^p_i\right)\right] = \mathbf{p}\left[v\left(\sum_{i=1}^{k} x^p_i\right)\right](1 - \mathbf{p}[v^p_{k-1}]) = 0$$

$$\Rightarrow \mathbf{p}\left[v\left(\sum_{i=1}^{j'} x^p_i\right)\right] = 0 \Rightarrow \mathbf{p}\left[v(x^p_{j'})\right] = 0.$$

Mapping g is surjective on the Nash equilibria of $\mathcal{G}'$ and, therefore, $\mathcal{G}$: We will show that given a Nash equilibrium $N_{\mathcal{G}'}$ of $\mathcal{G}'$ there is a Nash equilibrium $N_{\mathcal{GG}}$ of $\mathcal{GG}$ such that $g(N_{\mathcal{GG}}) = N_{\mathcal{G}'}$. Let $N_{\mathcal{G}'} = \{x^p_j : p \leq r, j \in S_p\}$. In $N_{\mathcal{GG}}$, let $\mathbf{p}[v(x^p_j)] = x^p_j$. Lemma 3.14 shows that the values $\mathbf{p}[v(U^p_j)]$ are the expected utilities to player p for playing strategy j, given that all other players use the mixed strategy $\{x^p_j : p \leq r, j \in S_p\}$. We identify values for $\mathbf{p}[v^p_j]$ that complete a Nash equilibrium for $\mathcal{GG}$.

Based on the payoffs to v^p_j described in Figure 3.7 we have

- If $\mathbf{p}[v(U^p_{\leq j})] > \mathbf{p}[v(U^p_{j+1})]$ then $\mathbf{p}[w(U^p_j)] = 0$; $\mathbf{p}[v^p_j] = 0$;
- If $\mathbf{p}[v(U^p_{\leq j})] < \mathbf{p}[v(U^p_{j+1})]$ then $\mathbf{p}[w(U^p_j)] = 1$; $\mathbf{p}[v^p_j] = 1$;
- If $\mathbf{p}[v(U^p_{\leq j})] = \mathbf{p}[v(U^p_{j+1})]$ then choose $\mathbf{p}[w(U^p_j)] = \frac{1}{2}$; $\mathbf{p}[v^p_j]$ is arbitrary (we may assign it any value)

Given the above constraints on the values $\mathbf{p}[v^p_j]$ we must check that we can choose them (and there is a unique choice) so as to make them consistent with the probabilities $\mathbf{p}[v(x^p_j)]$. We use the fact the values x^p_j form a Nash equilibrium of $\mathcal{G}$. In particular, we know that $\mathbf{p}[v(x^p_j)] = 0$ if there exists j' with $U^p_{j'} > U^p_j$. We claim that for j satisfying $\mathbf{p}[v(U^p_{\leq j})] = \mathbf{p}[v(U^p_{j+1})]$, if we choose

$$\mathbf{p}[v^p_j] = \mathbf{p}[v(x^p_{j+1})] / \sum_{i=1}^{j+1} \mathbf{p}[v(x^p_i)],$$

then the values $\mathbf{p}[v(x^p_j)]$ are consistent. □

3.4 Combining the Reductions

Suppose that we take either a graphical or a normal-form game, and apply to it both of the reductions described in the previous sections. Then we obtain a game of the same type and a surjective mapping from the Nash equilibria of the latter to the Nash equilibria of the former.

Corollary 3.15. *For any fixed d, a (directed or undirected) graphical game of maximum degree d can be mapped in polynomial time to an undirected graphical game of maximum degree 3 so that there is a polynomial-time computable surjective mapping g from the Nash equilibria of the latter to the Nash equilibria of the former.*

The following also follows directly from Theorems 3.12 and 3.8, but is not as strong as Theorem 3.17 below.

Corollary 3.16. *For any fixed $r > 1$, a r-player normal-form game can be mapped in polynomial time to a 10-player normal-form game so that there is a polynomial-time computable surjective mapping g from the Nash equilibria of the latter to the Nash equilibria of the former.*

Proof. Theorem 3.12 converts a r-player game $\mathcal{G}$ into a graphical game $\mathcal{GG}$ based on a graph of degree 3. Theorem 3.8 converts $\mathcal{GG}$ to a 10-player game $\mathcal{G}'$, whose Nash equilibria encode the Nash equilibria of $\mathcal{GG}$ and hence of $\mathcal{G}$. (Note that for d an odd number, the proof of Theorem 3.8 implies a reduction to a (d^2+1)-player normal-form game.) □

We next prove a stronger result, by exploiting in more detail the structure of the graphical games $\mathcal{GG}$ constructed in the proof of Theorem 3.12. The technique used here will be used in Section 3.5 to strengthen the result even further.

Theorem 3.17. *For any fixed $r > 1$, a r-player normal-form game can be mapped in polynomial time to a 4-player normal-form game so that there is a polynomial-time computable surjective mapping g from the Nash equilibria of the latter to the Nash equilibria of the former.*

Proof. Construct $\mathcal{G}'$ from $\mathcal{G}$ as shown in Figure 3.8.

Polynomial size of $\mathcal{G}' = f(\mathcal{G})$.

By Theorem 3.12, $\mathcal{GG}$ (as constructed in Figure 3.8) is of polynomial size. The size of $\mathcal{GG}'$ is at most 3 times the size of $\mathcal{GG}$ since we do not need to apply Step 3 to any edges that are themselves constructed by an earlier iteration of Step 3. Finally, the size of $\mathcal{G}'$ is polynomial in the size of $\mathcal{GG}'$ from Theorem 3.8.

Construction of $g(N_{\mathcal{G}'})$ (for $N_{\mathcal{G}'}$ a Nash equilibrium of $\mathcal{G}'$).

Let g_1 be a surjective mapping from the Nash equilibria of $\mathcal{GG}$ to the Nash equilibria of $\mathcal{G}$, which is guaranteed to exist by Theorem 3.12. It is trivial to construct a surjective mapping g_2 from the Nash equilibria of $\mathcal{GG}'$ to the Nash equilibria of $\mathcal{GG}$. By Theorem 3.8, there exists a surjective mapping g_3 from the Nash equilibria of $\mathcal{G}'$ to the Nash equilibria of $\mathcal{GG}'$. Therefore, $g_3 \circ g_2 \circ g_1$ is a surjective mapping from the Nash equilibria of $\mathcal{G}'$ to the Nash equilibria of $\mathcal{G}$. □

Input: Normal-form game $\mathcal{G}$ with r players, n strategies per player, utilities $\{u_s^p : p \le r, s \in S\}$.
Output: 4-player Normal-form game $\mathcal{G}'$.

1. Let $\mathcal{GG}$ be the graphical game constructed from $\mathcal{G}$ according to Figure 3.7. Recall that the affects graph $G = (V \cup W, E)$ of $\mathcal{GG}$ has the following properties:

   - Every edge $e \in E$ is from a vertex of set V to a vertex of set W or vice versa.
   - Every vertex of set W has indegree at most 3 and outdegree at most 1 and every vertex of set V has indegree at most 1 and outdegree at most 2.

2. Color the graph $(V \cup W, E)$ of $\mathcal{GG}$ as follows: let $c(w) = 1$ for all W-vertices w and $c(v) = 2$ for all V-vertices v.

3. Construct a new graphical game $\mathcal{GG}'$ from $\mathcal{GG}$ as follows. While there exist v_1, $v_2 \in V$, $w \in W$, $(v_1, w), (v_2, w) \in E$ with $c(v_1) = c(v_2)$:

   (a) Every W-vertex has at most 1 outgoing edge, so assume $(w, v_1) \notin E$.

   (b) Add $v(v_1)$ to V, add $w(v_1)$ to W.

   (c) Replace (v_1, w) with $(v_1, w(v_1))$, $(w(v_1), v(v_1))$, $(v(v_1), w(v_1))$, $(v(v_1), w)$. Let $c(w(v_1)) = 1$, choose $c(v(v_1)) \in \{2, 3, 4\} \neq c(v')$ for any v' with $(v', w) \in E$. Payoffs for $w(v_1)$ and $v(v_1)$ are chosen using Proposition 3.2 with $\alpha = 1$ such that in any Nash equilibrium, $\mathbf{p}[v(v_1)] = \mathbf{p}[v_1]$.

4. The coloring $c : V \cup W \to \{1, 2, 3, 4\}$ has the property that, for every vertex v of $\mathcal{GG}'$, its neighborhood $\mathcal{N}(v)$ in the affects graph of the game —recall it consists of v and all its predecessors— is colored with $|\mathcal{N}(v)|$ distinct colors. Rescale all utilities of $\mathcal{GG}'$ to [0,1] and map game $\mathcal{GG}'$ to a 4-player normal-form game $\mathcal{G}'$ following the steps 3 through 5 of figure 3.5.

Figure 3.8: Reduction from normal-form game $\mathcal{G}$ to 4-player game $\mathcal{G}'$

3.5 Reducing to Three Players

We will strengthen Theorem 3.17 to reduce a r-player normal-form game to a 3-player normal-form game. The following theorem together with Theorems 3.8 and 3.12 imply the first part of Theorem 3.1.

Theorem 3.18. *For any fixed $r > 1$, a r-player normal-form game can be mapped in polynomial time to a 3-player normal-form game so that there is a polynomial-time computable surjective mapping g from the Nash equilibria of the latter to the Nash equilibria of the former.*

Proof. The bottleneck of the construction of Figure 3.8 in terms of the number k of players of the resulting normal-form game $\mathcal{G}'$ lies entirely on the ability or lack thereof to color the vertices of the affects graphs of $\mathcal{GG}$ with k colors so that, for every vertex v, its neighborhood $\mathcal{N}(v)$ in the affects graph is colored with $|\mathcal{N}(v)|$ distinct colors, i.e. on whether there exists a legal k coloring. In Figure 3.8, we show how to design a graphical game $\mathcal{GG}'$ which is equivalent to $\mathcal{GG}$ —in the sense that there exists a surjective mapping from the Nash equilibria of the former to the Nash equilibria of the latter— and can be legally colored using 4 colors. However, this cannot be improved to 3 colors since the addition game $\mathcal{G}_+$ and the multiplication game $\mathcal{G}_*$, which are essential building blocks of $\mathcal{GG}$, have vertices with indegree 3 (see Figure 3.3) and, therefore, need at least 4 colors to be legally colored. Therefore, to improve our result we need to redesign addition and multiplication games which can be legally colored using 3 colors.

Notation: In the following,

- $x = y \pm \epsilon$ denotes $y - \epsilon \leq x \leq y + \epsilon$
- $v : s$ denotes "player v plays strategy s"

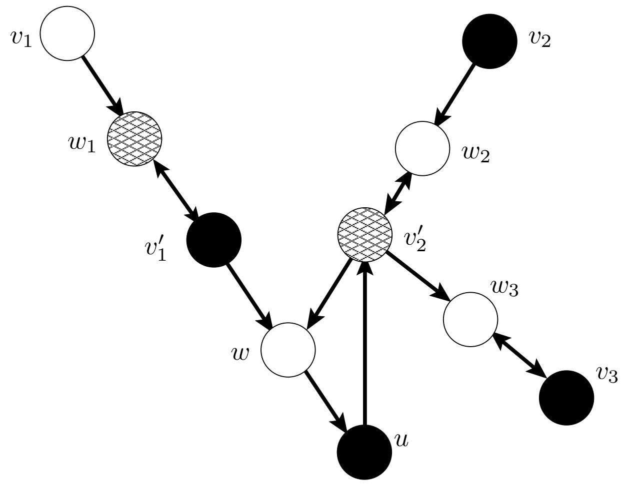

Figure 3.9: The new addition/multiplication game and its legal 3-coloring.

Proposition 3.19. *Let α, β, γ be non-negative integers such that $\alpha+\beta+\gamma \leq 3$. There is a graphical game $\mathcal{G}_{+,*}$ with two "input players" v_1 and v_2, one "output player" v_3 and several intermediate players, with the following properties:*

- *the graph of the game can be legally colored using 3 colors*
- *for any $\epsilon \in [0, 0.01]$, at any ϵ-Nash equilibrium of game $\mathcal{G}_{+,*}$ it holds that $\mathbf{p}[v_3] = \min\{1, \alpha\mathbf{p}[v_1]+\beta\mathbf{p}[v_2]+\gamma\mathbf{p}[v_1]\mathbf{p}[v_2]\} \pm 81\epsilon$; in particular at any Nash equilibrium $\mathbf{p}[v_3] = \min\{1, \alpha\mathbf{p}[v_1] + \beta\mathbf{p}[v_2] + \gamma\mathbf{p}[v_1]\mathbf{p}[v_2]\}$.*

Proof. The graph of the game and the labeling of the vertices is shown in Figure 3.9. All players of $\mathcal{G}_{+,*}$ have strategy set $\{0, 1\}$ except for player v_2' who has three strategies $\{0, 1, *\}$. Below we give the payoff tables of all the players of the game. For ease of understanding we partition the game $\mathcal{G}_{+,*}$ into four subgames:

1. Game played by players v_1, w_1, v_1':

Payoffs to v_1' :

	w_1 : 0	w_1 : 1
v_1' : 0	0	1
v_1' : 1	1	0

Payoffs to w_1:

$w_1 : 0$:

	$v_1{}' : 0$	$v_1{}' : 1$
$v_1 : 0$	0	0
$v_1 : 1$	1/8	1/8

$w_1 : 1$:

	$v_1{}' : 0$	$v_1{}' : 1$
$v_1 : 0$	0	1
$v_1 : 1$	0	1

2. Game played by players $v_2{}', w_3, v_3$:

Payoffs to v_3 :

	$w_3 : 0$	$w_3 : 1$
$v_3 : 0$	0	1
$v_3 : 1$	1	0

Payoffs to w_3:

$w_3 : 0$:

	$v_3 : 0$	$v_3 : 1$
$v_2' : 0$	0	0
$v_2' : 1$	0	0
$v_2' : *$	8	8

$w_3 : 1$:

	$v_3 : 0$	$v_3 : 1$
$v_2' : 0$	0	1
$v_2' : 1$	0	1
$v_2' : *$	0	1

3. Game played by players v_2, w_2, v_2':

Payoffs to w_2:

$w_2 : 0$:

	$v_2 : 0$	$v_2 : 1$
$v_2' : 0$	0	1/8
$v_2' : 1$	0	1/8
$v_2' : *$	0	1/8

$w_2 : 1$:

	$v_2 : 0$	$v_2 : 1$
$v_2' : 0$	0	0
$v_2' : 1$	1	1
$v_2' : *$	0	0

Payoffs to v_2':

$v_2' : 0$:

	$w_2 : 0$	$w_2 : 1$
$u : 0$	0	1
$u : 1$	0	0

$v_2' : 1$:

	$w_2 : 0$	$w_2 : 1$
$u : 0$	1	0
$u : 1$	1	0

$v_2' : *$:

	$w_2 : 0$	$w_2 : 1$
$u : 0$	0	0
$u : 1$	0	1

4. Game played by players v_1', v_2', w, u:

Payoffs to w:

$w : 0$:

	$v_1' : 0$	$v_1' : 1$
$v_2' : 0$	0	α
$v_2' : 1$	$1+\beta$	$1+\alpha+\beta+8\gamma$
$v_2' : *$	0	α

$w : 1$:

	$v_1' : 0$	$v_1' : 1$
$v_2' : 0$	0	0
$v_2' : 1$	1	1
$v_2' : *$	1	1

Payoffs to u:

	$w : 0$	$w : 1$
$u : 0$	0	1
$u : 1$	1	0

Claim 3.20. *At any ϵ-Nash equilibrium of $\mathcal{G}_{+,*}$:* $\mathbf{p}[v_1'] = \frac{1}{8}\mathbf{p}[v_1] \pm \epsilon$.

Proof. If w_1 plays 0, then the expected payoff to w_1 is $\frac{1}{8}\mathbf{p}[v_1]$, whereas if w_1 plays 1, the expected payoff to w_1 is $\mathbf{p}[v_1']$. Therefore, in an ϵ-Nash equilibrium, if $\frac{1}{8}\mathbf{p}[v_1] > \mathbf{p}[v_1'] + \epsilon$, then $\mathbf{p}[w_1] = 0$. However, note also that if $\mathbf{p}[w_1] = 0$ then $\mathbf{p}[v_1'] = 1$, which is a contradiction to $\frac{1}{8}\mathbf{p}[v_1] > \mathbf{p}[v_1'] + \epsilon$. Consequently, $\frac{1}{8}\mathbf{p}[v_1]$ cannot be strictly larger than $\mathbf{p}[v_1'] + \epsilon$. On the other hand, if $\mathbf{p}[v_1'] > \frac{1}{8}\mathbf{p}[v_1] + \epsilon$, then $\mathbf{p}[w_1] = 1$ and consequently $\mathbf{p}[v_1'] = 0$, a contradiction. The claim follows from the above observations. □

Claim 3.21. *At any ϵ-Nash equilibrium of $\mathcal{G}_{+,*}$:* $\mathbf{p}[v_2' : 1] = \frac{1}{8}\mathbf{p}[v_2] \pm \epsilon$.

Proof. If w_2 plays 0, then the expected payoff to w_2 is $\frac{1}{8}\mathbf{p}[v_2]$, whereas, if w_2 plays 1, the expected payoff to w_2 is $\mathbf{p}[v_2' : 1]$.

If, in an ϵ-Nash equilibrium, $\frac{1}{8}\mathbf{p}[v_2] > \mathbf{p}[v_2' : 1] + \epsilon$, then $\mathbf{p}[w_2] = 0$. In this regime, the payoff to player v_2' is 0 if v_2' plays 0, 1 if v_2' plays 1 and 0 if v_2' plays $*$. Therefore, $\mathbf{p}[v_2' : 1] = 1$ and this contradicts the hypothesis that $\frac{1}{8}\mathbf{p}[v_2] > \mathbf{p}[v_2' : 1] + \epsilon$.

On the other hand, if, in an ϵ-Nash equilibrium, $\mathbf{p}[v_2' : 1] > \frac{1}{8}\mathbf{p}[v_2] + \epsilon$, then $\mathbf{p}[w_2] = 1$. In this regime, the payoff to player v_2' is $\mathbf{p}[u : 0]$ if v_2' plays 0, 0 if v_2' plays 1 and $\mathbf{p}[u : 1]$ if v_2' plays $*$. Since $\mathbf{p}[u : 0] + \mathbf{p}[u : 1] = 1$, it follows that $\mathbf{p}[v_2' : 1] = 0$ because at least one of $\mathbf{p}[u : 0]$, $\mathbf{p}[u : 1]$ will be greater than ϵ. This contradicts the hypothesis that $\mathbf{p}[v_2' : 1] > \frac{1}{8}\mathbf{p}[v_2] + \epsilon$ and the claim follows from the above observations. □

Claim 3.22. *At any ϵ-Nash equilibrium of $\mathcal{G}_{+,*}$:* $\mathbf{p}[v_2' : *] = \frac{\alpha}{8}\mathbf{p}[v_1] + \frac{\beta}{8}\mathbf{p}[v_2] + \frac{\gamma}{8}\mathbf{p}[v_1]\mathbf{p}[v_2] \pm 10\epsilon$.

Proof. If w plays 0, then the expected payoff to w is $\alpha\mathbf{p}[v_1'] + (1+\beta)\mathbf{p}[v_2' : 1] + 8\gamma\mathbf{p}[v_1']\mathbf{p}[v_2' : 1]$, whereas, if w plays 1, the expected payoff to w is $\mathbf{p}[v_2' : 1] + \mathbf{p}[v_2' : *]$.

If, in a ϵ-Nash equilibrium, $\alpha\mathbf{p}[v_1'] + (1+\beta)\mathbf{p}[v_2' : 1] + 8\gamma\mathbf{p}[v_1']\mathbf{p}[v_2' : 1] > \mathbf{p}[v_2' : 1] + \mathbf{p}[v_2' : *] + \epsilon$, then $\mathbf{p}[w] = 0$ and, consequently, $\mathbf{p}[u] = 1$. In this regime, the payoff to player v_2' is 0 if v_2' plays 0, $\mathbf{p}[w_2 : 0]$ if v_2' plays 1 and $\mathbf{p}[w_2 : 1]$ if v_2' plays $*$. Since $\mathbf{p}[w_2 : 0] + \mathbf{p}[w_2 : 1] = 1$, it follows that at least one of $\mathbf{p}[w_2 : 0]$, $\mathbf{p}[w_2 : 1]$ will be larger than ϵ so that $\mathbf{p}[v_2' : 0] = 0$ or, equivalently, that $\mathbf{p}[v_2' : 1] + \mathbf{p}[v_2' : *] = 1$. So the hypothesis can be rewritten as $\alpha\mathbf{p}[v_1'] + (1+\beta)\mathbf{p}[v_2' : 1] + 8\gamma\mathbf{p}[v_1']\mathbf{p}[v_2' :$

$1] > 1 + \epsilon$. Using Claims 3.20 and 3.21 and the fact that $\epsilon \leq 0.01$ this inequality implies $\frac{\alpha}{8}\mathbf{p}[v_1] + \frac{1+\beta}{8}\mathbf{p}[v_2] + \frac{\gamma}{8}\mathbf{p}[v_1]\mathbf{p}[v_2] + (\alpha + 1 + \beta + 3\gamma)\epsilon > 1 + \epsilon$ and further that $\frac{\alpha+1+\beta+\gamma}{8} + (\alpha+1+\beta+3\gamma)\epsilon > 1+\epsilon$. We supposed $\alpha+\beta+\gamma \leq 3$ therefore the previous inequality implies $\frac{1}{2} + 10\epsilon > 1 + \epsilon$, a contradiction since we assumed $\epsilon \leq 0.01$.

On the other hand, if, in a ϵ-Nash equilibrium, $\mathbf{p}[v_2' : 1] + \mathbf{p}[v_2' : *] > \alpha\mathbf{p}[v_1'] + (1 + \beta)\mathbf{p}[v_2' : 1] + 8\gamma\mathbf{p}[v_1']\mathbf{p}[v_2' : 1] + \epsilon$, then $\mathbf{p}[w] = 1$ and consequently $\mathbf{p}[u] = 0$. In this regime, the payoff to player v_2' is $\mathbf{p}[w_2 : 1]$ if v_2' plays 0, $\mathbf{p}[w_2 : 0]$ if v_2' plays 1 and 0 if v_2' plays $*$. Since $\mathbf{p}[w_2 : 0] + \mathbf{p}[w_2 : 1] = 1$, it follows that $\mathbf{p}[v_2' : *] = 0$. So the hypothesis can be rewritten as $0 > \alpha\mathbf{p}[v_1'] + \beta\mathbf{p}[v_2' : 1] + 8\gamma\mathbf{p}[v_1']\mathbf{p}[v_2' : 1] + \epsilon$ which is a contradiction.

Therefore, in any ϵ-Nash equilibrium, $\mathbf{p}[v_2' : 1] + \mathbf{p}[v_2' : *] = \alpha\mathbf{p}[v_1'] + (1 + \beta)\mathbf{p}[v_2' : 1] + 8\gamma\mathbf{p}[v_1']\mathbf{p}[v_2' : 1] \pm \epsilon$, or, equivalently, $\mathbf{p}[v_2' : *] = \alpha\mathbf{p}[v_1'] + \beta\mathbf{p}[v_2' : 1] + 8\gamma\mathbf{p}[v_1']\mathbf{p}[v_2' : 1] \pm \epsilon$. Using Claims 3.20 and 3.21 this can be restated as $\mathbf{p}[v_2' : *] = \frac{\alpha}{8}\mathbf{p}[v_1] + \frac{\beta}{8}\mathbf{p}[v_2] + \frac{\gamma}{8}\mathbf{p}[v_1]\mathbf{p}[v_2] \pm 10\epsilon$ □

Claim 3.23. *At any ϵ-Nash equilibrium of $\mathcal{G}_{+,*}$:* $\mathbf{p}[v_3] = \min\{1, \alpha\mathbf{p}[v_1] + \beta\mathbf{p}[v_2] + \gamma\mathbf{p}[v_1]\mathbf{p}[v_2]\} \pm 81\epsilon$.

Proof. If w_3 plays 0, the expected payoff to w_3 is $8\mathbf{p}[v_2' : *]$, whereas, if w_3 plays 1, the expected payoff to w_3 is $\mathbf{p}[v_3]$. Therefore, in a ϵ-Nash equilibrium, if $\mathbf{p}[v_3] > 8\mathbf{p}[v_2' : *] + \epsilon$, then $\mathbf{p}[w_3] = 1$ and, consequently, $\mathbf{p}[v_3] = 0$, which is a contradiction to $\mathbf{p}[v_3] > 8\mathbf{p}[v_2' : *] + \epsilon$.

On the other hand, if $8\mathbf{p}[v_2' : *] > \mathbf{p}[v_3] + \epsilon$, then $\mathbf{p}[w_3] = 0$ and consequently $\mathbf{p}[v_3] = 1$. Hence, $\mathbf{p}[v_3]$ cannot be less than $\min\{1, 8\mathbf{p}[v_2' : *] - \epsilon\}$.

From the above observations it follows that $\mathbf{p}[v_3] = \min\{1, 8\mathbf{p}[v_2' : *]\} \pm \epsilon$ and, using Claim 3.22, $\mathbf{p}[v_3] = \min\{1, \alpha\mathbf{p}[v_1] + \beta\mathbf{p}[v_2] + \gamma\mathbf{p}[v_1]\mathbf{p}[v_2]\} \pm 81\epsilon$. □

It remains to show that the graph of the game can be legally colored using 3 colors. The coloring is shown in Figure 3.9. □

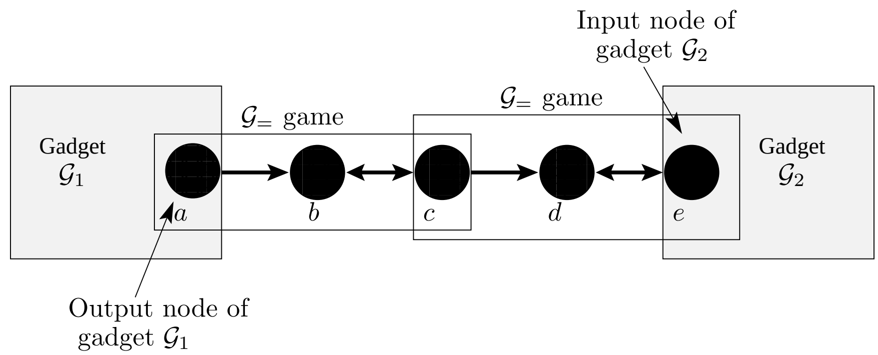

Figure 3.10: The interposition of two $\mathcal{G}_=$ games between gadgets $\mathcal{G}_1$ and $\mathcal{G}_2$ does not change the game.

Now that we have our hands on the game $\mathcal{G}_{+,*}$ of Proposition 3.19, we can reduce r-player games to 3-player games, for any fixed r, using the algorithm of Figure 3.8 with the following tweak: in the construction of game $\mathcal{GG}$ at Step 1 of the algorithm, instead of using the addition and multiplication gadgets $\mathcal{G}_+$, $\mathcal{G}_*$ of Section 3.1, we use our more elaborate $\mathcal{G}_{+,*}$ gadget. Let us call the resulting game $\mathcal{GG}$. We will show that we can construct a graphical game $\mathcal{GG}'$ which is equivalent to $\mathcal{GG}$ in the sense that there is a surjective mapping from the Nash equilibria of $\mathcal{GG}'$ to the Nash equilibria of $\mathcal{GG}$ and which, moreover, can be legally colored using three colors. Then we can proceed as in Step 4 of Figure 3.8 to get the desired 3-player normal-form game G'.

The construction of $\mathcal{GG}'$ and its coloring can be done as follows: Recall that all our gadgets have some distinguished vertices which are the *inputs* and one distinguished vertex which is the *output*. The gadgets are put together to construct $\mathcal{GG}$ by identifying the output vertices of some gadgets as the input vertices of other gadgets. It is easy to see that we get a graphical game with the same functionality if, instead of identifying the output vertex of some gadget with the input of another gadget, we interpose a sequence of two $\mathcal{G}_=$ games between the two gadgets to be connected, as shown in Figure 3.10. If we "glue" our gadgets in this way then the resulting graphical game $\mathcal{GG}'$ can be legally colored using 3 colors:

i. (stage 1) legally color the vertices inside the "initial gadgets" using 3 colors

ii. (stage 2) extend the coloring to the vertices that serve as "connections" between gadgets; any 3-coloring of the initial gadgets can be extended to a 3-coloring of $\mathcal{GG}'$ because, for any pair of gadgets $\mathcal{G}_1$, $\mathcal{G}_2$ which are connected (Figure 3.10) and for any colors assigned to the output vertex a of gadget $\mathcal{G}_1$ and the input vertex e of gadget $\mathcal{G}_2$, the intermediate vertices b, c and d can be also colored legally. For example, if vertex a gets color 1 and vertex e color 2 at stage 1, then, at stage 2, b can be colored 2, c can be colored 3 and d can be colored 1.

This completes the proof of the theorem. □

3.6 Preservation of Approximate equilibria

Our reductions so far map exact equilibrium points. In this section we generalize to approximate equilibria and prove the second part of Theorem 3.1. We claim that the reductions of the previous sections translate the problem of finding an ϵ-Nash equilibrium of a game to the problem of finding an ϵ'-Nash equilibrium of its image, for ϵ' polynomial in ϵ and inverse polynomial in the size of the game. As a consequence, we obtain polynomial-time equivalence results for the problems r-Nash and d-graphical-Nash. To prove the second part of Theorem 3.1, we extend Theorems 3.8, 3.12 and 3.18 of the previous sections.

Theorem 3.24. *For every fixed $d > 1$, there is a polynomial-time reduction from d-graphical-Nash to $(d^2 + 1)$-Nash.*

Proof. Let $\widetilde{\mathcal{GG}}$ be a graphical game of maximum degree d and $\mathcal{GG}$ the resulting graphical game after rescaling all utilities by $1/\max\{\tilde{u}\}$, where $\max\{\tilde{u}\}$ is the largest entry in the utility tables of game $\widetilde{\mathcal{GG}}$, so that they lie in the set $[0,1]$, as in the first step of Figure 3.5. Assume that $\epsilon < 1$. In time polynomial in $|\mathcal{GG}| + \log(1/\epsilon)$, we will specify a normal-form game $\mathcal{G}$ and an accuracy ϵ' with the property that, given an ϵ'-Nash equilibrium of $\mathcal{G}$, one can recover in polynomial time an ϵ-Nash equilibrium of $\mathcal{GG}$. This will be enough, since an ϵ-Nash equilibrium of $\mathcal{GG}$ is trivially an $\epsilon \cdot \max\{\tilde{u}\}$-Nash equilibrium of game $\widetilde{\mathcal{GG}}$ and, moreover, $|\mathcal{GG}|$ is polynomial in $|\widetilde{\mathcal{GG}}|$.

We construct $\mathcal{G}$ using the algorithm of Figure 3.5; recall that $M \geq 2\frac{n}{r}$, where r is the number of color classes specified in Figure 3.5 and n is the number of vertices in $\mathcal{GG}$ after the possible addition of dummy vertices to make sure that all color classes have the same number of vertices (as in Step 3 of Figure 3.5). Let us choose $\epsilon' \leq \epsilon(\frac{r}{n} - \frac{1}{M})^d$; we will argue that from any ϵ'-Nash equilibrium of game $\mathcal{G}$ one can construct in polynomial time an ϵ-Nash equilibrium of game $\mathcal{GG}$.

Suppose that $p = c(v)$ for some vertex v of the graphical game $\mathcal{GG}$. As in the proof of Theorem 3.8, Lemma 3.11, it can be shown that in any ϵ'-Nash equilibrium of the game $\mathcal{G}$,

$$\Pr[p \text{ plays } v] \in \left[\frac{r}{n} - \frac{1}{M}, \frac{r}{n} + \frac{1}{M}\right].$$

Now, without loss of generality, assume that p is odd (pursuer) and suppose that v is vertex $v_i^{(p)}$ in the notation of Figure 3.5. Then, in an ϵ'-Nash equilibrium of the game $\mathcal{G}$, we have, by the definition of a Nash equilibrium, that for all strategies $a, a' \in S_v$ of vertex v:

$$\mathbb{E}\left[\text{payoff to } p \text{ for playing } (v,a)\right] > \mathbb{E}\left[\text{payoff to } p \text{ for playing } (v,a')\right] + \epsilon' \Rightarrow x^p_{(v,a')} = 0.$$

But

$$\begin{aligned}&\mathbb{E}\left[\text{payoff to } p \text{ for playing } (v,a)\right]\\ &\qquad = M \cdot \Pr\left[p+1 \text{ plays } v_i^{(p+1)}\right] + \sum_{s \in S_{\mathcal{N}(v)\setminus\{v\}}} u^v_{as} \prod_{u \in \mathcal{N}(v)\setminus\{v\}} x^{c(u)}_{(u,s_u)}\end{aligned}$$

and, similarly, for a'. Therefore, the previous inequality implies

$$\sum_{s \in S_{\mathcal{N}(v)\setminus\{v\}}} u^v_{as} \prod_{u \in \mathcal{N}(v)\setminus\{v\}} x^{c(u)}_{(u,s_u)} > \sum_{s \in S_{\mathcal{N}(v)\setminus\{v\}}} u^v_{a's} \prod_{u \in \mathcal{N}(v)\setminus\{v\}} x^{c(u)}_{(u,s_u)} + \epsilon' \Rightarrow x^p_{(v,a')} = 0$$

So letting

$$x^v_a = x^{c(v)}_{(v,a)} \Big/ \sum_{j \in S_v} x^{c(v)}_{(v,j)}, \quad \forall v \in V, a \in S_v,$$

as we did in the proof of Theorem 3.8, we get that, for all $v \in V$, $a, a' \in S_v$,

$$\sum_{s \in S_{\mathcal{N}(v)\setminus\{v\}}} u^v_{as} \prod_{u \in \mathcal{N}(v)\setminus\{v\}} x^u_{s_u} > \sum_{s \in S_{\mathcal{N}(v)\setminus\{v\}}} u^v_{a's} \prod_{u \in \mathcal{N}(v)\setminus\{v\}} x^u_{s_u} + \epsilon'/\mathcal{T} \Rightarrow x^v_{a'} = 0, \tag{3.5}$$

where $\mathcal{T} = \prod_{u \in \mathcal{N}(v)\setminus\{v\}} \sum_{j \in S_u} x^{c(u)}_{(u,j)} = \prod_{u \in \mathcal{N}(v)\setminus\{v\}} \Pr[c(u) \text{ plays } u] \geq (\frac{r}{n} - \frac{1}{M})^d$. By the definition of ϵ' it follows that $\epsilon'/\mathcal{T} \leq \epsilon$. Hence, from (3.5) it follows that $\{x^v_a\}_{v,a}$ is an ϵ-Nash equilibrium of the game $\mathcal{GG}$. □

We have the following extension of Theorem 3.12.

Theorem 3.25. *For every fixed $r > 1$, there is a polynomial-time reduction from r-*Nash *to* 3-graphical Nash *with two strategies per vertex.*

Proof. Let $\tilde{\mathcal{G}}$ be a normal-form game with r players, $1, 2, \ldots, r$, and strategy sets $S_p = [n]$, for all $p \in [r]$, and let $\{\tilde{u}^p_s : p \in [r], s \in S\}$ be the utilities of the players. Denote by $\mathcal{G}$ the game constructed at the first step of Figure 3.7 which results from $\tilde{\mathcal{G}}$ after rescaling all utilities by $1/\max\{\tilde{u}^p_s\}$ so that they lie in $[0, 1]$; let $\{u^p_s : p \in [r], s \in S\}$ be the utilities of the players in game $\mathcal{G}$. Also, let $\epsilon < 1$. In time polynomial in $|\mathcal{G}| + \log(1/\epsilon)$, we will specify a graphical game $\mathcal{GG}$ and an accuracy ϵ' with the property that, given an ϵ'-Nash equilibrium of $\mathcal{GG}$, one can recover in polynomial time an ϵ-Nash equilibrium of $\mathcal{G}$. This will be enough, since an ϵ-Nash equilibrium of $\mathcal{G}$ is trivially an $\epsilon \cdot \max\{\tilde{u}_s^p\}$-Nash equilibrium of game $\tilde{\mathcal{G}}$ and, moreover, $|\mathcal{G}|$ is polynomial in $|\tilde{\mathcal{G}}|$. In our reduction, the graphical game $\mathcal{GG}$ will be the same as the one described in the proof of Theorem 3.12 (Figure 3.7), while the accuracy specification will be of the form $\epsilon' = \epsilon/p(|\mathcal{G}|)$, where $p(\cdot)$ is a polynomial that will be be specified later. We will use the same labels for the vertices of the game $\mathcal{GG}$ that we used in the proof Theorem 3.12.

Suppose $N_{\mathcal{GG}}$ is some ϵ'-Nash equilibrium of the game $\mathcal{GG}$ and let $\{\mathbf{p}[v(x_j^p)]\}_{j,p}$ denote the probabilities with which the vertices $v(x_j^p)$ of $\mathcal{GG}$ play strategy 1. In the proof of Theorem 3.12 we considered the following mapping from the Nash equilibria of game $\mathcal{GG}$ to the Nash equilibria of game $\mathcal{G}$:

$$x_j^p := \mathbf{p}[v(x_j^p)], \text{ for all } p \text{ and } j. \tag{3.6}$$

Although (3.6) succeeds in mapping exact equilibrium points, it fails for approximate equilibria, as specified by the following remark —its justification follows from the proof of Lemma 3.27.

Remark 3.26. *For any $\epsilon' > 0$, there exists an ϵ'-Nash equilibrium of game $\mathcal{GG}$ such that $\sum_j \mathbf{p}[v(x_j^p)] \neq 1$, for some player $p \leq r$, and, moreover, $\mathbf{p}[v(U_j^p)] > \mathbf{p}[v(U_{j'}^p)] + \epsilon'$, for some $p \leq r$, j and j', and, yet, $\mathbf{p}[v(x_{j'}^p)] > 0$.*

Recall from Section 3.3, that, for all p, j, the probability $\mathbf{p}[v(U_j^p)]$ represents the utility of player p for playing pure strategy j, when the other players play according to $\{x_j^q := \mathbf{p}[v(x_j^q)]\}_{j,q\neq p}$. [3] Therefore, not only the $\{x_j^p := \mathbf{p}[v(x_j^p)]\}_j$ do not necessarily constitute a distribution —this could be easily fixed by rescaling— but, also, the defining property of an approximate equilibrium (2.2) is in question. The following lemma bounds the deviation from the approximate equilibrium conditions.

[3] Note, however, that, since we are considering an ϵ'-Nash equilibrium of game $\mathcal{GG}$, Equation (3.3) of Section 3.3 will be only satisfied approximately as specified by Lemma 3.29.

Lemma 3.27. *In any ϵ'-Nash equilibrium of the game $\mathcal{GG}$,*

(i) *for all $p \in [r]$, $|\sum_j \mathbf{p}[v(x_j^p)] - 1| \leq 2cn\epsilon'$, and,*

(ii) *for all $p \in [r]$, $j, j' \in [n]$, $\mathbf{p}[v(U_j^p)] > \mathbf{p}[v(U_{j'}^p)] + 5cn\epsilon' \Rightarrow \mathbf{p}[v(x_{j'}^p)] \in [0, cn\epsilon']$,*

where $c \geq 1$ is the maximum error amplification of the gadgets used in the construction of $\mathcal{GG}$.

Proof. Note that at an ϵ'-Nash equilibrium of game $\mathcal{GG}$ the following properties are satisfied for all $p \in [r]$ by the vertices of game $\mathcal{GG}$, since the error amplification of the gadgets is at most c:

$$\mathbf{p}\left[v\left(\sum_{i=1}^{n} x_i^p\right)\right] = 1 \tag{3.7}$$

$$\mathbf{p}\left[v\left(\sum_{i=1}^{j} x_i^p\right)\right] = \mathbf{p}\left[v\left(\sum_{i=1}^{j+1} x_i^p\right)\right] \cdot (1 - \mathbf{p}[v_j^p]) \pm c\epsilon', \forall j < n \tag{3.8}$$

$$\mathbf{p}\left[v(x_{j+1}^p)\right] = \mathbf{p}\left[v\left(\sum_{i=1}^{j+1} x_i^p\right)\right] \cdot \mathbf{p}[v_j^p] \pm c\epsilon', \forall j < n \tag{3.9}$$

$$\mathbf{p}\left[v(x_1^p)\right] = \mathbf{p}\left[v\left(\sum_{i=1}^{1} x_i^p\right)\right] \pm c\epsilon' \tag{3.10}$$

Proof of (i): By successive applications of (3.8) and (3.9), we deduce

$$\begin{aligned}
\sum_{j=1}^{n} \mathbf{p}[v(x_j^p)] &= \sum_{j=2}^{n} \left\{ \mathbf{p}\left[v\left(\sum_{i=1}^{j} x_i^p \right) \right] \cdot \mathbf{p}[v_{j-1}^p] \right\} + \mathbf{p}\left[v\left(\sum_{i=1}^{1} x_i^p \right) \right] \pm cn\epsilon' \\
&= \sum_{j=2}^{n} \left\{ \mathbf{p}\left[v\left(\sum_{i=1}^{j} x_i^p \right) \right] \cdot \mathbf{p}[v_{j-1}^p] \right\} + \left(\mathbf{p}\left[v\left(\sum_{i=1}^{2} x_i^p \right) \right] \cdot (1 - \mathbf{p}[v_1^p]) \pm c\epsilon' \right) \pm cn\epsilon' \\
&= \sum_{j=3}^{n} \left\{ \mathbf{p}\left[v\left(\sum_{i=1}^{j} x_i^p \right) \right] \cdot \mathbf{p}[v_{j-1}^p] \right\} + \mathbf{p}\left[v\left(\sum_{i=1}^{2} x_i^p \right) \right] \pm c(n+1)\epsilon' \\
&= \dots \\
&= \mathbf{p}\left[v\left(\sum_{i=1}^{n} x_i^p \right) \right] \pm c(2n-1)\epsilon' \\
&= 1 \pm c(2n-1)\epsilon'
\end{aligned}$$

Proof of (ii): Let us first observe the behavior of vertices $w(U_j^p)$ and v_j^p in an ϵ'-Nash equilibrium.

- **Behavior of $w(U_j^p)$ vertices:** The utility of vertex $w(U_j^p)$ for playing strategy 0 is $\mathbf{p}[v(U_{\leq j}^p)]$, whereas for playing 1 it is $\mathbf{p}[v(U_{j+1}^p)]$. Therefore,

$$\mathbf{p}[v(U_{\leq j}^p)] > \mathbf{p}[v(U_{j+1}^p)] + \epsilon' \Rightarrow \mathbf{p}[w(U_j^p)] = 0$$

$$\mathbf{p}[v(U_{j+1}^p)] > \mathbf{p}[v(U_{\leq j}^p)] + \epsilon' \Rightarrow \mathbf{p}[w(U_j^p)] = 1$$

$$|\mathbf{p}[v(U_{j+1}^p)] - \mathbf{p}[v(U_{\leq j}^p)]| \leq \epsilon' \Rightarrow \mathbf{p}[w(U_j^p)] \text{ can be anything}$$

- **Behavior of v_j^p vertices:** The utility of vertex v_j^p for playing strategy 0 is $1 - \mathbf{p}[w(U_j^p)]$, whereas for playing 1 it is $\mathbf{p}[w(U_j^p)]$. Therefore,

$$\mathbf{p}[w(U_j^p)] < \tfrac{1-\epsilon'}{2} \Rightarrow \mathbf{p}[v_j^p] = 0$$

$$\mathbf{p}[w(U_j^p)] > \tfrac{1+\epsilon'}{2} \Rightarrow \mathbf{p}[v_j^p] = 1$$

$$|\mathbf{p}[w(U_j^p)] - \tfrac{1}{2}| \leq \tfrac{\epsilon'}{2} \Rightarrow \mathbf{p}[v_j^p] \text{ can be anything}$$

Note that, since the error amplification of the gadget $\mathcal{G}_{\max}$ is at most c and computing $\mathbf{p}[v(U^p_{\leq j})]$, for all j, requires j applications of $\mathcal{G}_{\max}$,

$$\mathbf{p}[v(U^p_{\leq j})] = \max_{i \leq j} \mathbf{p}[v(U^p_i)] \pm c\epsilon' j. \tag{3.11}$$

To establish the second part of the claim, we need to show that, for all p, j, j',

$$\mathbf{p}[v(U^p_j)] > \mathbf{p}[v(U^p_{j'})] + 5cn\epsilon' \Rightarrow \mathbf{p}[v(x^p_{j'})] \in [0, nc\epsilon'].$$

1. Note that, if there exists some $j'' < j'$ such that $\mathbf{p}[v(U^p_{j''})] > \mathbf{p}[v(U^p_{j'})] + c\epsilon' n$, then

$$\begin{aligned}\mathbf{p}[v(U^p_{\leq j'-1})] &= \max_{i \leq j'-1} \mathbf{p}[v(U^p_i)] \pm c\epsilon'(j'-1) \\ &\geq \mathbf{p}[v(U^p_{j''})] - c\epsilon'(j'-1) \\ &> \mathbf{p}[v(U^p_{j'})] + cn\epsilon' - c\epsilon'(j'-1) \geq \mathbf{p}[v(U^p_{j'})] + \epsilon'.\end{aligned}$$

   Then, because $\mathbf{p}[v(U^p_{\leq j'-1})] > \mathbf{p}[v(U^p_{j'})] + \epsilon'$, it follows that $\mathbf{p}[w(U^p_{j'-1})] = 0$ and $\mathbf{p}[v^p_{j'-1}] = 0$. Therefore,

$$\mathbf{p}[v(x^p_{j'})] = \mathbf{p}\left[v\left(\sum_{i=1}^{j'} x^p_i\right)\right] \cdot \mathbf{p}[v^p_{j'-1}] \pm c\epsilon' = \pm c\epsilon'.$$

2. The case $j < j'$ reduces to the previous for $j'' = j$.

3. It remains to deal with the case $j > j'$, under the assumption that, for all $j'' < j'$,

$$\mathbf{p}[v(U^p_{j''})] \leq \mathbf{p}[v(U^p_{j'})] + c\epsilon' n.$$

which, in turn, implies

$$\mathbf{p}[v(U^p_{\leq j'})] < \mathbf{p}[v(U^p_{j'})] + 2c\epsilon' n. \qquad \text{(by (3.11))}$$

Let us further distinguish the following subcases

(a) If there exists some k, $j'+1 \leq k \leq j$, such that $\mathbf{p}[v(U^p_k)] > \mathbf{p}[v(U^p_{\leq k-1})]+\epsilon'$, then

$$\begin{aligned}
&\mathbf{p}[w(U^p_{k-1})] = 1 \Rightarrow \mathbf{p}[v^p_{k-1}] = 1 \\
&\quad \Rightarrow \mathbf{p}\left[v\left(\sum_{i=1}^{k-1} x^p_i\right)\right] = \mathbf{p}\left[v\left(\sum_{i=1}^{k} x^p_i\right)\right](1-\mathbf{p}[v^p_{k-1}]) \pm c\epsilon' = \pm c\epsilon' \\
&\quad \Rightarrow \mathbf{p}\left[v\left(\sum_{i=1}^{j'} x^p_i\right)\right] = \pm(k-j')c\epsilon' \qquad \left(\begin{array}{l}\text{by successive applications} \\ \text{of equation (3.8)}\end{array}\right) \\
&\quad \Rightarrow \mathbf{p}\left[v(x^p_{j'})\right] = \pm nc\epsilon'. \qquad \text{(by (3.9), (3.10))}
\end{aligned}$$

(b) If, for all k, $j'+1 \leq k \leq j$, it holds that $\mathbf{p}[v(U^p_k)] \leq \mathbf{p}[v(U^p_{\leq k-1})] + \epsilon'$, we will show a contradiction; hence, only the previous case can hold. Towards a contradiction,we argue first that

$$\mathbf{p}[v(U^p_{\leq j'+1})] \geq \mathbf{p}[v(U^p_j)] - 2cn\epsilon'.$$

To show this, we distinguish the cases $j = j'+1$, $j > j'+1$.

- In the case $j = j'+1$, we have

$$\begin{aligned}
\mathbf{p}[v(U^p_{\leq j'+1})] &\geq \max\left\{\mathbf{p}[v(U^p_{j'+1})], \mathbf{p}[v(U^p_{\leq j'})]\right\} - c\epsilon' \\
&\geq \mathbf{p}[v(U^p_{j'+1})] - c\epsilon' = \mathbf{p}[v(U^p_j)] - c\epsilon'.
\end{aligned}$$

- In the case $j > j' + 1$, we have for all k, $j' + 2 \leq k \leq j$,

  $$\mathbf{p}[v(U^p_{\leq k-1})] \geq \max\{\mathbf{p}[v(U^p_{\leq k-1})], \mathbf{p}[v(U^p_k)]\} - \epsilon' \geq \mathbf{p}[v(U^p_{\leq k})] - c\epsilon' - \epsilon',$$

  where the last inequality holds since the game $\mathcal{G}_{\max}$ has error amplification at most c. Summing these inequalities for $j' + 2 \leq k \leq j$, we deduce that

  $$\begin{aligned}\mathbf{p}[v(U^p_{\leq j'+1})] &\geq \mathbf{p}[v(U^p_{\leq j})] - (c\epsilon' + \epsilon')(n-2) \\ &\geq \max\{\mathbf{p}[v(U^p_j)], \mathbf{p}[v(U^p_{\leq j-1})]\} - c\epsilon' - (c\epsilon' + \epsilon')(n-2) \\ &\geq \mathbf{p}[v(U^p_j)] - 2c\epsilon' n.\end{aligned}$$

It follows that

$$\mathbf{p}[v(U^p_{\leq j'+1})] > \mathbf{p}[v(U^p_{j'})] + 3cn\epsilon'.$$

But,

$$\mathbf{p}[v(U^p_{\leq j'+1})] \leq \max\{\mathbf{p}[v(U^p_{j'+1})], \mathbf{p}[v(U^p_{\leq j'})]\} + c\epsilon'$$

and recall that

$$\mathbf{p}[v(U^p_{\leq j'})] < \mathbf{p}[v(U^p_{j'})] + 2c\epsilon' n.$$

We can deduce that

$$\max\{\mathbf{p}[v(U^p_{j'+1})], \mathbf{p}[v(U^p_{\leq j'})]\} = \mathbf{p}[v(U^p_{j'+1})],$$

which combined with the above implies

$$\mathbf{p}[v(U^p_{j'+1})] \geq \mathbf{p}[v(U^p_{j'})] + 3cn\epsilon' - c\epsilon' > \mathbf{p}[v(U^p_{\leq j'})] + \epsilon'.$$

$\square$

From Lemma 3.27, it follows that the extraction of an ϵ-Nash equilibrium of game $\mathcal{G}$ from an ϵ'-Nash equilibrium of game $\mathcal{GG}$ cannot be done by just interpreting the values $\{x_j^p := \mathbf{p}[v(x_j^p)]\}_j$ as the mixed strategy of player p. What we show next is that, for the right choice of ϵ', a *trim and renormalize* transformation succeeds in deriving an ϵ-Nash equilibrium of game $\mathcal{G}$ from an ϵ'-Nash equilibrium of game $\mathcal{GG}$. Indeed, for all $p \leq r$, suppose that $\{\hat{x}_j^p\}_j$ are the values derived from $\{x_j^p\}_j$ by setting

$$\hat{x}_j^p = \begin{cases} 0, & \text{if } x_j^p \leq cn\epsilon' \\ x_j^p, & \text{otherwise} \end{cases}$$

and then renormalizing the resulting values $\{\hat{x}_j^p\}_j$ so that $\sum_j \hat{x}_j^p = 1$.

Lemma 3.28. *There exists a polynomial $p(\cdot)$ such that, if $\{\{x_j^p\}_j\}_p$ is an $\epsilon/p(|\mathcal{G}|)$-Nash equilibrium of game $\mathcal{GG}$, then the trimmed and renormalized values $\{\{\hat{x}_j^p\}_j\}_p$ constitute an ϵ-Nash equilibrium of game $\mathcal{G}$.*

Proof. We first establish the following useful lemma

Lemma 3.29. *At an ϵ'-Nash equilibrium of game $\mathcal{GG}$, for all p, j, it holds that*

$$\mathbf{p}[v(U_j^p)] = \sum_{s \in S_{-p}} u_{js}^p x_{s_1}^1 \cdots x_{s_{p-1}}^{p-1} x_{s_{p+1}}^{p+1} \cdots x_{s_r}^r \pm 2n^{r-1}\zeta_r,$$

where c is the maximum error amplification of the gadgets used in the construction of $\mathcal{GG}$, $\zeta_r = c\epsilon' + ((1+\zeta)^r - 1)(c\epsilon' + 1)$, $\zeta = 2r \log n \; c\epsilon'$.

Proof. Using the same notation as in Section 3.3, let $S_{-p} = \{S_{-p}(1), \ldots, S_{-p}(n^{r-1})\}$, so that

$$\sum_{s \in S_{-p}} u_{js}^p x_s = \sum_{\ell=1}^{n^{r-1}} u_{jS_{-p}(\ell)}^p x_{S_{-p}(\ell)}.$$

Recall that in $\mathcal{GG}$, for each partial sum $\sum_{\ell=1}^{z} u^p_{jS_{-p}(\ell)} x_{S_{-p}(\ell)}$, $1 \leq z \leq n^{r-1}$, we have included vertex $v(\sum_{\ell=1}^{z} u^p_{jS_{-p}(\ell)} x_{S_{-p}(\ell)})$. Similarly, for each partial product of the summands $u^p_{js} \prod_{p \neq q \leq z} x^q_{s_q}$, $0 \leq z \leq r$, we have included vertex $v(u^p_{js} \prod_{p \neq q \leq z} x^q_{s_q})$. Note that, since we have rescaled the utilities to the set $[0,1]$, all partial sums and products must also lie in $[0,1]$. Note, moreover, that, to avoid large degrees in the resulting graphical game, each time we need to make use of a value $x^q_{s_q}$ we create a new copy of the vertex $v(x^q_{s_q})$ using the gadget $\mathcal{G}_=$ and, then, use the new copy for the computation of the desired partial product; an easy calculation shows that we have to make $(r-1)n^{r-1}$ copies of $v(x^q_{s_q})$, for all $q \leq r$, $s_q \in S_q$. To limit the degree of each vertex to 3 we create a binary tree of copies of $v(x^q_{s_q})$ with $(r-1)n^{r-1}$ leaves and use each leaf once. Then, because of the error amplification of $\mathcal{G}_=$, this already induces an error of $\pm \lceil \log{(r-1)n^{r-1}} \rceil c\epsilon'$ to each of the factors of the partial products. The following lemma characterizes the error that results from the error amplification of our gadgets in the computation of the partial products and can be proved easily by induction.

Lemma 3.30. *For all $p \leq r$, $j \in S_p$, $s \in S_{-p}$ and $z \leq r$,*

$$\mathbf{p}\left[v\left(u^p_{js} \prod_{p \neq \ell \leq z} x^\ell_{s_\ell}\right)\right] = u^p_{js} \prod_{p \neq \ell \leq z} x^\ell_{s_\ell} \pm \zeta_z, \tag{3.12}$$

where $\zeta_z = c\epsilon' + ((1+\zeta)^z - 1)(c\epsilon' + 1)$, $\zeta = 2r \log n \; c\epsilon'$.

The following lemma characterizes the error in the computation of the partial sums and can be proved by induction using the previous lemma for the base case.

Lemma 3.31. *For all $p \leq r$, $j \in S_p$ and $z \leq n^{r-1}$,*

$$\mathbf{p}\left[v\left(\sum_{\ell=1}^{z} u^p_{jS_{-p}(\ell)} x_{S_{-p}(\ell)}\right)\right] = \sum_{\ell=1}^{z} u^p_{jS_{-p}(\ell)} x_{S_{-p}(\ell)} \pm (z\zeta_r + (z-1)c\epsilon'), \tag{3.13}$$

where ζ_r is defined as in Lemma 3.30.

From Lemma 3.31 we can deduce, in particular, that for all $p \leq r$, $j \in S_p$,

$$\mathbf{p}[v(U_j^p)] = \sum_{s \in S_{-p}} u_{js}^p x_s \pm 2n^{r-1}\zeta_r.$$

□

Lemma 3.32. *For all $p \leq r$, $j \in S_p$,*

$$\left| \sum_{s \in S_{-p}} u_{js}^p x_s - \sum_{s \in S_{-p}} u_{js}^p y_s \right| \leq \max_{s \in S_{-p}} \{u_{js}^p\} \sum_{q \neq p} \sum_{i \in S_q} |x_i^q - y_i^q|.$$

Proof. We have

$$\left| \sum_{s \in S_{-p}} u_{js}^p x_s - \sum_{s \in S_{-p}} u_{js}^p y_s \right| \leq \sum_{s \in S_{-p}} u_{js}^p \left| x_s - y_s \right| \leq \max_{s \in S_{-p}} \{u_{js}^p\} \sum_{s \in S_{-p}} \left| x_s - y_s \right|. \tag{3.14}$$

Let us denote by $\mathcal{X}^q$ the random variable, ranging over the set S_q, which represents the mixed strategy $\{x_i^q\}_{i \in S_q}$, $q \leq r$. Similarly, define the random variable $\mathcal{Y}^q$ from the mixed strategy $\{y_i^q\}_{i \in S_q}$, $q \leq r$. Note, then, that $\frac{1}{2}\sum_{s \in S_{-p}} |x_s - y_s|$ is precisely the total variation distance between the vector random variable $(\mathcal{X}^q)_{q \neq p}$ and the vector random variable $(\mathcal{Y}^q)_{q \neq p}$. That is,

$$\frac{1}{2} \sum_{s \in S_{-p}} |x_s - y_s| = ||(\mathcal{X}^q)_{q \neq p} - (\mathcal{Y}^q)_{q \neq p}||_{TV}. \tag{3.15}$$

By the coupling lemma, we have that

$$||(\mathcal{X}^q)_{q \neq p} - (\mathcal{Y}^q)_{q \neq p}||_{TV} \leq \Pr\left[(\mathcal{X}^q)_{q \neq p} \neq (\mathcal{Y}^q)_{q \neq p}\right],$$

for any coupling of $(\mathcal{X}^q)_{q \neq p}$ and $(\mathcal{Y}^q)_{q \neq p}$. Applying a union bound to the right hand side of the above implies

$$||(\mathcal{X}^q)_{q\neq p} - (\mathcal{Y}^q)_{q\neq p}||_{TV} \leq \sum_{q\neq p} \Pr\left[\mathcal{X}^q \neq \mathcal{Y}^q\right]. \tag{3.16}$$

Now let us fix a coupling between $(\mathcal{X}^q)_{q\neq p}$ and $(\mathcal{Y}^q)_{q\neq p}$ so that, for all $q \neq p$,

$$\Pr\left[\mathcal{X}^q \neq \mathcal{Y}^q\right] = ||\mathcal{X}^q - \mathcal{Y}^q||_{TV}.$$

Such a coupling exists by the coupling lemma for each $q \neq p$ individually, and for the whole vectors $(\mathcal{X}^q)_{q\neq p}$ and $(\mathcal{Y}^q)_{q\neq p}$ it exists because also the $\mathcal{X}^q$'s are independent and so are the $\mathcal{Y}^q$'s. Then (3.16) implies that

$$||(\mathcal{X}^q)_{q\neq p} - (\mathcal{Y}^q)_{q\neq p}||_{TV} \leq \sum_{q\neq p} ||\mathcal{X}^q - \mathcal{Y}^q||_{TV},$$

so that from (3.14), (3.15) we get

$$\left|\sum_{s\in S_{-p}} u^p_{js} x_s - \sum_{s\in S_{-p}} u^p_{js} y_s\right| \leq \max_{s\in S_{-p}} \{u^p_{js}\} 2 \sum_{q\neq p} ||\mathcal{X}^q - \mathcal{Y}^q||_{TV}. \tag{3.17}$$

Now, note that, for all q,

$$||\mathcal{X}^q - \mathcal{Y}^q||_{TV} = \frac{1}{2}\sum_{i\in S_q} |x^q_i - y^q_i|.$$

Hence, (3.17) implies

$$\left|\sum_{s\in S_{-p}} u^p_{js} x_s - \sum_{s\in S_{-p}} u^p_{js} y_s\right| \leq \max_{s\in S_{-p}} \{u^p_{js}\} \sum_{q\neq p} \sum_{i\in S_q} |x^q_i - y^q_i|.$$

□

We can conclude the proof of Lemma 3.28, by invoking Lemmas 3.29 and 3.32. Indeed, by the definition of the $\{\hat{x}_j^p\}$, it follows that for all p, $j \in S_p$,

$$\hat{x}_j^p = \begin{cases} \frac{x_j^p}{\Lambda^p}, & x_j^p > cn\epsilon' \\ 0, & x_j^p \leq cn\epsilon' \end{cases},$$

where

$$1 \geq \Lambda^p = \sum_{j \in S_p} x_j^p \mathcal{X}_{\{x_j^p > cn\epsilon'\}} = 1 - \sum_{j \in S_p} x_j^p \mathcal{X}_{\{x_j^p \leq cn\epsilon'\}} \geq 1 - n \cdot cn\epsilon',$$

where $\mathcal{X}_{\{\cdot\}}$ is the indicator function. Therefore,

$$|\hat{x}_j^p - x_j^p| = \begin{cases} \frac{x_j^p}{\Lambda^p} - x_j^p, & x_j^p > cn\epsilon' \\ x_j^p, & x_j^p \leq cn\epsilon' \end{cases},$$

which implies

$$|\hat{x}_j^p - x_j^p| \leq \begin{cases} \frac{1}{\Lambda^p} - 1, & x_j^p > cn\epsilon' \\ cn\epsilon', & x_j^p \leq cn\epsilon' \end{cases}.$$

So,

$$|\hat{x}_j^p - x_j^p| \leq \max\left\{cn\epsilon', \frac{n^2 c\epsilon'}{1 - n^2 c\epsilon'}\right\} =: \delta_1,$$

which by Lemma 3.32 implies that

$$\left|\sum_{s \in S_{-p}} u_{js}^p x_s - \sum_{s \in S_{-p}} u_{js}^p \hat{x}_s\right| \leq \max_{s \in S_{-p}} \{u_{js}^p\}(r-1)n\delta_1 \leq (r-1)n\delta_1 =: \delta_2, \qquad (3.18)$$

where the second inequality follows from the fact that we have rescaled the utilities so that they lie in $[0, 1]$.

Choosing $\epsilon' = \frac{\epsilon}{40cr^2n^{r+1}}$, we will argue that the conditions of an ϵ-Nash equilibrium are satisfied by the mixed strategies $\{\hat{x}^p_j\}_{p,j}$. First, note that:

$$(1 + 2r \log n\ c\epsilon')^r - 1 \le \left(1 + \frac{\epsilon}{20rn^r}\right)^r - 1 \le \exp\left\{\frac{\epsilon}{20n^r}\right\} - 1 \le \frac{\epsilon}{10n^r},$$

which implies that

$$2n^{r-1}\zeta_r \le 2n^{r-1}\left(c\epsilon' + \frac{\epsilon}{10n^r}(c\epsilon' + 1)\right) \le 2n^{r-1}\frac{1.5\epsilon}{10n^r} = \frac{3\epsilon}{10n} \le \frac{0.3\epsilon}{n}.$$

Also, note that

$$\delta_1 = \max\left\{cn\epsilon', \frac{n^2c\epsilon'}{1 - n^2c\epsilon'}\right\} \le 2n^2c\epsilon',$$

which gives

$$\delta_2 = (r-1)n\delta_1 \le rn2n^2c\frac{\epsilon}{40cr^2n^{r+1}} \le \frac{\epsilon}{20r}.$$

Thus, for all $p \le r$, $j, j' \in S_p$, we have that

$$\begin{aligned}
&\sum_{s \in S_{-p}} u^p_{js}\hat{x}_s > \sum_{s \in S_{-p}} u^p_{j's}\hat{x}_s + \epsilon \\
&\quad\Rightarrow \sum_{s \in S_{-p}} u^p_{js}x_s + \delta_2 > \sum_{s \in S_{-p}} u^p_{j's}x_s - \delta_2 + \epsilon \quad \text{(using (3.18))} \\
&\quad\Rightarrow \sum_{s \in S_{-p}} u^p_{js}x_s > \sum_{s \in S_{-p}} u^p_{j's}x_s + \epsilon - 2\delta_2 \\
&\quad\Rightarrow \mathbf{p}[v(U^p_j)] + 2n^{r-1}\zeta_r > \mathbf{p}[v(U^p_{j'})] - 2n^{r-1}\zeta_r + \epsilon - 2\delta_2 \text{ (using Lemma 3.29)} \\
&\quad\Rightarrow \mathbf{p}[v(U^p_j)] > \mathbf{p}[v(U^p_{j'})] - 4n^{r-1}\zeta_r + \epsilon - 2\delta_2 \\
&\quad\Rightarrow \mathbf{p}[v(U^p_j)] > \mathbf{p}[v(U^p_{j'})] + 5cn\epsilon' \\
&\quad\Rightarrow x^p_{j'} \le cn\epsilon' \ \text{ (using Lemma 3.27)} \\
&\quad\Rightarrow \hat{x}^p_{j'} = 0.
\end{aligned}$$

Therefore, $\{\hat{x}_j^p\}$ is indeed an ϵ-Nash equilibrium of game $\mathcal{G}$, which concludes the proof of the lemma. □

□

We have the following extension of Theorem 3.18.

Theorem 3.33. *For every fixed $r > 1$, there is a polynomial-time reduction from r-*Nash *to* 3-Nash.

Proof. The proof follows immediately from the proofs of Theorems 3.24 and 3.25. Indeed, observe that the reduction of Theorem 3.25 still holds when we use the gadget $\mathcal{G}_{+,*}$ of Section 3.5 for the construction our graphical games, since the gadget $\mathcal{G}_{+,*}$ has constant error amplification. Therefore, the problem of computing an ϵ-Nash equilibrium of a r-player normal-form game $\mathcal{G}$ can be polynomially reduced to computing an ϵ'-Nash equilibrium of a graphical game $\mathcal{GG}'$ which can be legally colored with 3 colors (after performing the "glueing" step described in the end of the proof of Theorem 3.18 and appropriately adjusting the ϵ' specified in the proof of Theorem 3.25). Observe, further, that the reduction of Theorem 3.24 can be used to map the latter to computing an ϵ''-Nash equilibrium of a 3-player normal-form game $\mathcal{G}''$, since the number of players that are required for $\mathcal{G}''$ is equal to the minimum number of colors needed for a legal coloring of $\mathcal{GG}'$. The claim follows by combining the reductions.

□

3.7 Reductions Between Different Notions of Approximation

We establish a polynomial-time reduction from the problem of computing an approximately well supported Nash equilibrium to the problem of computing an approximate

Nash equilibrium. As pointed out in Section 2.1, the reduction in the opposite direction is trivial, since an ϵ-approximately well supported Nash equilibrium is also an ϵ-approximate Nash equilibrium.

Lemma 3.34. *Given an ϵ-approximate Nash equilibrium $\{x_j^p\}_{j,p}$ of a game $\mathcal{G}$ we can compute in polynomial time a $\sqrt{\epsilon} \cdot (\sqrt{\epsilon} + 1 + 4(r-1)\max\{u\})$-approximately well supported Nash equilibrium $\{\hat{x}_j^p\}_{j,p}$, where r is the number of players in $\mathcal{G}$ and $\max\{u\}$ is the maximum entry in the payoff tables of $\mathcal{G}$.*

Proof. Since $\{x_j^p\}_{j,p}$ is an ϵ-approximate Nash equilibrium, it follows that for every player $p \leq r$ and every mixed strategy $\{y_j^p\}_j$ for that player

$$\sum_{s \in S} u_s^p \cdot x_{s_{-p}} \cdot x_{s_p}^p \geq \sum_{s \in S} u_s^p \cdot x_{s_{-p}} \cdot y_{s_p}^p - \epsilon.$$

Equivalently,

$$\forall p \leq r,\ \forall\ \{y_j^p\}_{j \in S_p} : \ \sum_{j \in S_p} \left[\sum_{s_{-p} \in S_{-p}} u_{js_{-p}}^p \ x_{s_{-p}} \right] x_j^p \geq \sum_{j \in S_p} \left[\sum_{s_{-p} \in S_{-p}} u_{js_{-p}}^p \ x_{s_{-p}} \right] y_j^p - \epsilon. \tag{3.19}$$

For all $p \leq r$, denote $\mathcal{U}_j^p = \sum_{s_{-p} \in S_{-p}} u_{js_{-p}}^p \ x_{s_{-p}}$, for all $j \in S_p$, and $\mathcal{U}_{\max}^p = \max_j \mathcal{U}_j^p$. Then, if we choose $\{y_j^p\}_j$ to be some pure strategy from the set $\arg\max_j \mathcal{U}_j^p$, (3.19) implies

$$\forall p \leq r : \qquad \sum_{j \in S_p} \mathcal{U}_j^p x_j^p \geq \mathcal{U}_{\max}^p - \epsilon. \tag{3.20}$$

Now, let us fix some player $p \leq r$. We want to upper bound the probability mass that the distribution $\{x_j^p\}_j$ assigns to pure strategies $j \in S_p$ which give expected utility $\mathcal{U}_j^p$ more than an additive ϵk smaller than $\mathcal{U}_{\max}^p$, for some k to be specified later. The following bound is easy to derive using (3.20).

Claim 3.35. *For all p, set*

$$z^p = \sum_{j \in S_p} x_j^p \cdot \mathcal{X}_{\{\mathcal{U}_j^p < \mathcal{U}_{\max}^p - \epsilon k\}},$$

where $\mathcal{X}_A$ is the characteristic function of the event A. Then

$$z^p \leq \frac{1}{k}.$$

Let us then consider the strategy profile $\{\hat{x}_j^p\}_{j,p}$ defined as follows

$$\forall p, j \in S_p : \quad \hat{x}_j^p = \begin{cases} \frac{x_j^p}{1-z^p}, & \mathcal{U}_j^p \geq \mathcal{U}_{\max}^p - \epsilon k \\ \\ 0, & \text{otherwise} \end{cases}$$

We establish the following bound on the L_1 distance between the strategy profiles $\{x_j^p\}_j$ and $\{\hat{x}_j^p\}_j$.

Claim 3.36. *For all p, $\sum_{j \in S_p} |x_j^p - \hat{x}_j^p| \leq \frac{2}{k-1}$.*

Proof. Denote $S_{p,1} := \{j \mid j \in S_p, \mathcal{U}_j^p \geq \mathcal{U}_{\max}^p - \epsilon k\}$ and $S_{p,2} := S_p \setminus S_{p,1}$. Then

$$\begin{aligned} \sum_{j \in S_p} |x_j^p - \hat{x}_j^p| &= \sum_{j \in S_{p,1}} |x_j^p - \hat{x}_j^p| + \sum_{j \in S_{p,2}} |x_j^p - \hat{x}_j^p| \\ &= \sum_{j \in S_{p,1}} \left| x_j^p - \frac{x_j^p}{1 - z^p} \right| + \sum_{j \in S_{p,2}} |x_j^p| \\ &= \sum_{j \in S_{p,1}} \left| x_j^p - \frac{x_j^p}{1 - z^p} \right| + z^p \\ &\leq \frac{z^p}{1 - z^p} \sum_{j \in S_{p,1}} x_j^p + z^p \\ &\leq \frac{1}{k-1} + \frac{1}{k} \leq \frac{2}{k-1}. \end{aligned}$$

□

Now, for all players p, let $\hat{\mathcal{U}}_j^p$ and $\hat{\mathcal{U}}_{\max}^p$ be defined similarly to $\mathcal{U}_j^p$ and $\mathcal{U}_{\max}^p$. Recall Lemma 3.32 from Section 3.6.

Lemma 3.37. *For all p, $j \in S_p$,*

$$|\mathcal{U}_j^p - \hat{\mathcal{U}}_j^p| \le \max_{s \in S_{-p}} \{u_{js}^p\} \sum_{p' \neq p} \sum_{j \in S_{p'}} |x_j^{p'} - \hat{x}_j^{p'}|.$$

Let us then take $\Delta_2 = 2\frac{r-1}{k-1} \max_{p, j \in S_p, s \in S_{-p}} \{u_{js}^p\}$. Claim 3.36 and Lemma 3.37 imply that the strategy profile $\{\hat{x}_j^p\}_{j,p}$ satisfies

$$\forall p, \forall j \in S_p : \quad |\mathcal{U}_j^p - \hat{\mathcal{U}}_j^p| \le \Delta_2.$$

We will establish that $\{\hat{x}_j^p\}_{j,p}$ is a $(\epsilon k + 2\Delta_2)$-Nash equilibrium. Equivalently, we shall establish that

$$\forall p, \forall i, j \in S_p : \hat{\mathcal{U}}_j^p < \hat{\mathcal{U}}_i^p - (\epsilon k + 2\Delta_2) \Rightarrow \hat{x}_j^p = 0.$$

Indeed,

$$\begin{aligned}
\hat{\mathcal{U}}_j^p < \hat{\mathcal{U}}_i^p - (\epsilon k + 2\Delta_2) &\Rightarrow \mathcal{U}_j^p - \Delta_2 < \mathcal{U}_i^p + \Delta_2 - (\epsilon k + 2\Delta_2) \\
&\Rightarrow \mathcal{U}_j^p < \mathcal{U}_i^p - (\epsilon k + 2\Delta_2 - 2\Delta_2) \\
&\Rightarrow \mathcal{U}_j^p < \mathcal{U}_{\max}^p - \epsilon k \\
&\Rightarrow \hat{x}_j^p = 0.
\end{aligned}$$

Taking $k = 1 + \frac{1}{\sqrt{\epsilon}}$, it follows that $\{\hat{x}_j^p\}_{j,p}$ is a $\sqrt{\epsilon} \cdot (\sqrt{\epsilon} + 1 + 4(r-1)\max\{u_{js}^p\})$-Nash equilibrium. □

Chapter 4

The Complexity of Computing a Nash Equilibrium

In Section 4.1, we present our main result that 3-NASH, the problem of computing a Nash equilibrium of a 3-player normal-form game, is PPAD-complete. Since by Theorem 3.1 r-NASH and d-GRAPHICAL NASH are polynomial-time reducible to 3-NASH, for all $r, d \geq 3$, the same hardness holds for graphical games of degree 3 or larger and normal-form games with more than 3 players. This leaves open the complexity of 2-player games which were also shown to be PPAD-complete by Chen and Deng [CD06], by modifying our construction. In Section 4.2, we present an alternative small modification of our argument establishing the PPAD-completeness of 2-NASH. In Section 4.3, we discuss the application of our techniques for establishing that the Nash equilibrium problem of other important classes of games, such as congestion games [Ros73, FPT04] and extensive form games [OR94], is in PPAD. We also argue that solving simple stochastic games [Con92] is in PPAD. Finally, we establish that computing exact Nash equilibria and exact fixed points of polynomial functions are polynomial-time equivalent.

4.1 The Complexity of Games with Three or More Players

We show that computing a Nash equilibrium is a hard computational problem for normal-form games of at least 3 players and graphical games of degree at least 3, namely

Theorem 4.1. *Both* 3-Nash *and* 3-graphical Nash *are* PPAD-*complete.*

Proof. That 3-Nash is in PPAD follows from Theorem 2.3. That 3-graphical Nash is in PPAD follows by reducing it to 3-Nash, by Theorem 3.1, and then invoking Theorem 2.3. We hence focus on establishing the PPAD-hardness of the problems.

The reduction is from the problem Brouwer defined in Section 2.4. Given an instance of Brouwer, that is a circuit C with $3n$ input bits and 2 output bits describing a Brouwer function as specified in Section 2.4, we construct a graphical game $\mathcal{G}$, with maximum degree three, that simulates the circuit C, and specify an accuracy ϵ, so that, given an ϵ-Nash equilibrium of $\mathcal{G}$, one can find in polynomial time a panchromatic vertex of the Brouwer instance. Then, since, by Theorem 3.1, 3-graphical Nash reduces to 3-Nash, this completes the proof.

The graphical game $\mathcal{G}$ that we construct will be *binary*, in that each vertex v in it will have two strategies, and thus, at equilibrium, will represent a real number in $[0, 1]$, denoted $\mathbf{p}[v]$. (Letting 0 and 1 denote the strategies, $\mathbf{p}[v]$ is the probability that v plays 1.) There will be three distinguished vertices v_x, v_y, and v_z which will represent the coordinates of a point in the three dimensional cube and the construction will guarantee that in any Nash equilibrium of game $\mathcal{G}$ this point will be close to a panchromatic vertex of the given Brouwer instance.

The building blocks of $\mathcal{G}$ will be the game-gadgets $\mathcal{G}_\alpha, \mathcal{G}_{\times\alpha}, \mathcal{G}_=, \mathcal{G}_+, \mathcal{G}_-, \mathcal{G}_*$ that we constructed in Section 3.1 plus a few new gadgets. Recall from Propositions 3.2, 3.3 and 3.6, Figures 3.2, 3.1 and 3.3, that

Lemma 4.2. *There exist binary graphical games $\mathcal{G}_\alpha$, where α is any rational in $[0, 1]$, $\mathcal{G}_{\times\alpha}$, where α is any non-negative rational, $\mathcal{G}_=, \mathcal{G}_+, \mathcal{G}_-, \mathcal{G}_*$, with at most four players a, b, c, d each, such that, in all games, the payoffs of a and b do not depend on the choices of the other vertices c, d, and, for $\epsilon < 1$,*

1. *in every ϵ-Nash equilibrium of game $\mathcal{G}_\alpha$, we have $\mathbf{p}[d] = \alpha \pm \epsilon$;*
2. *in every ϵ-Nash equilibrium of game $\mathcal{G}_{\times\alpha}$, we have $\mathbf{p}[d] = \min(1, \alpha\mathbf{p}[a]) \pm \epsilon$;*
3. *in every ϵ-Nash equilibrium of game $\mathcal{G}_=$, we have $\mathbf{p}[d] = \mathbf{p}[a] \pm \epsilon$;*
4. *in every ϵ-Nash equilibrium of game $\mathcal{G}_+$, we have $\mathbf{p}[d] = \min\{1, \mathbf{p}[a] + \mathbf{p}[b]\} \pm \epsilon$;*
5. *in every ϵ-Nash equilibrium of game $\mathcal{G}_-$, we have $\mathbf{p}[d] = \max\{0, \mathbf{p}[a] - \mathbf{p}[b]\} \pm \epsilon$;*
6. *in every ϵ-Nash equilibrium of game $\mathcal{G}_*$, we have $\mathbf{p}[d] = \mathbf{p}[a] \cdot \mathbf{p}[b] \pm \epsilon$;*

where by $x = y \pm \epsilon$ we denote $y - \epsilon \le x \le y + \epsilon$.

Let us, further, define a comparator game $\mathcal{G}_<$.

Lemma 4.3. *There exists a binary graphical game $\mathcal{G}_<$ with three players a, b and d such that the payoffs of a and b do not depend on the choices of d and, in every ϵ-Nash equilibrium of the game, with $\epsilon < 1$, it holds that $\mathbf{p}[d] = 1$, if $\mathbf{p}[a] < \mathbf{p}[b] - \epsilon$, and $\mathbf{p}[d] = 0$, if $\mathbf{p}[a] > \mathbf{p}[b] + \epsilon$.*

Proof. Let us define the payoff table of player d as follows: d receives a payoff of 1 if d plays 0 and a plays 1, and d receives a payoff of 1 if d plays 1 and b plays 1, otherwise d receives a payoff of 0. Equivalently, d receives an expected payoff of $\mathbf{p}[a]$, if d plays 0, and an expected payoff of $\mathbf{p}[b]$, if d plays 1. It immediately follows that,

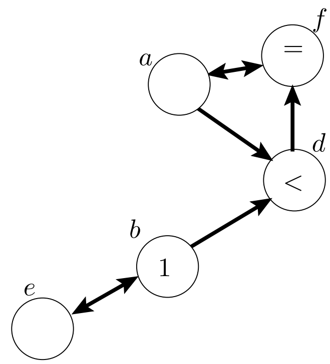

Figure 4.1: Brittleness of Comparator Games.

if in an ϵ-Nash equilibrium $\mathbf{p}[a] < \mathbf{p}[b] - \epsilon$, then $\mathbf{p}[d] = 1$, whereas, if $\mathbf{p}[a] > \mathbf{p}[b] + \epsilon$, $\mathbf{p}[d] = 0$. □

Notice that, in $\mathcal{G}_<$, $\mathbf{p}[d]$ is arbitrary if $|\mathbf{p}[a] - \mathbf{p}[b]| \leq \epsilon$; hence we call it the *brittle comparator*. As an aside, it is not hard to see that a robust comparator, one in which d is guaranteed, in an exact Nash equilibrium, to be, say, 0 if $\mathbf{p}[a] = \mathbf{p}[b]$, cannot exist, since it could be used to produce a simple graphical game with no Nash equilibrium, contradicting Nash's theorem. For completeness we present such a game in Figure 4.1, where vertices e and b constitute a $\mathcal{G}_1$ game so that, in any Nash equilibrium, $\mathbf{p}[b] = 1$, vertices d, f, a constitute a $\mathcal{G}_=$ game so that, in any Nash equilibrium, $\mathbf{p}[a] = \mathbf{p}[d]$ and vertices a, b, d constitute a comparator game with the hypothetical behavior that $\mathbf{p}[d] = 1$, if $\mathbf{p}[a] < \mathbf{p}[b]$ and $\mathbf{p}[d] = 0$, if $\mathbf{p}[a] \geq \mathbf{p}[b]$. Then it is not hard to argue that the game of Figure 4.1 does not have a Nash equilibrium contrary to Nash's theorem: indeed if, in a Nash equilibrium, $\mathbf{p}[a] = 1$, then $\mathbf{p}[d] = 0$, since $\mathbf{p}[a] = 1 = \mathbf{p}[b]$, and so $\mathbf{p}[a] = \mathbf{p}[d] = 0$, by $\mathcal{G}_=$, a contradiction; on the other hand, if, in a Nash equilibrium, $\mathbf{p}[a] < 1$, then $\mathbf{p}[d] = 1$, since $\mathbf{p}[a] < 1 = \mathbf{p}[b]$, and so $\mathbf{p}[a] = \mathbf{p}[d] = 1$, by $\mathcal{G}_=$, again a contradiction.

To continue with our reduction from Brouwer to 3-graphical nash, we include the following vertices to the graphical game $\mathcal{G}$.

- the three coordinate vertices v_x, v_y, v_z,

- for $i \in \{1, 2, \ldots, n\}$, vertices $v_{b_i(x)}$, $v_{b_i(y)}$ and $v_{b_i(z)}$, whose $\mathbf{p}$-values correspond to the i-th most significant bit of $\mathbf{p}[v_x]$, $\mathbf{p}[v_y]$, $\mathbf{p}[v_z]$,

- for $i \in \{1, 2, \ldots, n\}$, vertices v_{x_i}, v_{y_i} and v_{z_i}, whose $\mathbf{p}$-values correspond to the fractional number resulting from subtracting from $\mathbf{p}[v_x]$, $\mathbf{p}[v_y]$, $\mathbf{p}[v_z]$ the fractional numbers corresponding to the $i-1$ most significant bits of $\mathbf{p}[v_x]$, $\mathbf{p}[v_y]$, $\mathbf{p}[v_z]$ respectively.

We can extract these values by computing the binary representation of $\lfloor \mathbf{p}[v_x] 2^n \rfloor$ and, similarly, for v_y and v_z, that is, the binary representations of the integers i, j, k such that $(x, y, z) = (\mathbf{p}[v_x], \mathbf{p}[v_y], \mathbf{p}[v_z])$ lies in the cubelet K_{ijk}. This is done by a graphical game that simulates, using the arithmetical gadgets of Lemmas 4.2 and 4.3, the following algorithm ($<(a, b)$ is 1 if $a \leq b$ and 0 if $a > b$):

$x_1 = x$;

for $i = 1, \ldots, n$ *do:*

$\{b_i(x) :=< (2^{-i}, x_i);\ x_{i+1} := x_i - b_i(x) \cdot 2^{-i}\}$;

similarly, for y and z;

This is accomplished in $\mathcal{G}$ by connecting these vertices as prescribed by Lemmas 4.2 and 4.3, so that $\mathbf{p}[v_{x_i}], \mathbf{p}[v_{b_i(x)}]$, etc. approximate the value of x_i, $b_i(x)$ etc. as computed by the above algorithm. The following lemma (when applied with $m = n$) shows that this device properly decodes the first n bits of the binary expansion of $x = \mathbf{p}[v_x]$, as long as x is not too close to a multiple of 2^{-n} (suppose $\epsilon << 2^{-n}$ to be fixed later).

Lemma 4.4. *For* $m \leq n$, *if* $\sum_{i=1}^{m} b_i 2^{-i} + 3m\epsilon < \mathbf{p}[v_x] < \sum_{i=1}^{m} b_i 2^{-i} + 2^{-m} - 3m\epsilon$ *for some* $b_1, \ldots, b_m \in \{0, 1\}$, *then, in every* ϵ*-Nash equilibrium of* $\mathcal{G}$, $\mathbf{p}[v_{b_j(x)}] = b_j$, *and* $\mathbf{p}[v_{x_{j+1}}] = \mathbf{p}[v_x] - \sum_{i=1}^{j} b_i 2^{-i} \pm 3j\epsilon$, *for all* $j \leq m$.

Proof. The proof is by induction on j. For $j = 1$, the hypothesis $\sum_{i=1}^{m} b_i 2^{-i} + 3m\epsilon <$

$\mathbf{p}[v_x] < \sum_{i=1}^{m} b_i 2^{-i} + 2^{-m} - 3m\epsilon$ implies, in particular, that

$$\frac{b_1}{2} + 3\epsilon \leq \sum_{i=1}^{m} b_i 2^{-i} + 3m\epsilon < \mathbf{p}[v_x] < \sum_{i=1}^{m} b_i 2^{-i} + 2^{-m} - 3m\epsilon \leq \frac{b_1}{2} + \frac{1}{2} - 3\epsilon$$

and, since $\mathbf{p}[v_{x_1}] = \mathbf{p}[v_x] \pm \epsilon$, it follows that

$$\frac{b_1}{2} + 2\epsilon < \mathbf{p}[v_{x_1}] < \frac{b_1}{2} + \frac{1}{2} - 2\epsilon.$$

By Lemma 4.3, this implies that $\mathbf{p}[v_{b_1(x)}] = b_1$; note that the preparation of the constant $\frac{1}{2}$ —against which a comparator game compares the value $\mathbf{p}[v_{x_1}]$— is done via a $\mathcal{G}_{\frac{1}{2}}$ game which introduces an error of $\pm\epsilon$. For the computation of $\mathbf{p}[v_{x_2}]$, the multiplication of $\mathbf{p}[v_{b_1(x)}]$ by $\frac{1}{2}$ and the subtraction of the product from $\mathbf{p}[v_{x_1}]$ introduce an error of $\pm\epsilon$ each and, therefore, $\mathbf{p}[v_{x_2}] = \mathbf{p}[v_{x_1}] - b_1\frac{1}{2} \pm 2\epsilon$. And, since $\mathbf{p}[v_{x_1}] = \mathbf{p}[v_x] \pm \epsilon$, it follows that $\mathbf{p}[v_{x_2}] = \mathbf{p}[v_x] - b_1\frac{1}{2} \pm 3\epsilon$, as required.

Supposing that the claim holds up to $j-1 \leq m-1$, we will show that it holds for j. By the induction hypothesis, we have that $\mathbf{p}[v_{x_j}] = \mathbf{p}[v_x] - \sum_{i=1}^{j-1} b_i 2^{-i} \pm 3(j-1)\epsilon$. Combining this with $\sum_{i=1}^{m} b_i 2^{-i} + 3m\epsilon < \mathbf{p}[v_x] < \sum_{i=1}^{m} b_i 2^{-i} + 2^{-m} - 3m\epsilon$, it follows that

$$\sum_{i=j}^{m} b_i 2^{-i} + 3(m-(j-1))\epsilon < \mathbf{p}[v_{x_j}] < \sum_{i=j}^{m} b_i 2^{-i} + 2^{-m} - 3(m-(j-1))\epsilon$$

which implies

$$\frac{b_j}{2^j} + 2\epsilon < \mathbf{p}[v_{x_j}] < \frac{b_j}{2^j} + \frac{1}{2^j} - 2\epsilon.$$

Continue as in the base case. □

Assuming that $x = \mathbf{p}[v_x]$, $y = \mathbf{p}[v_y]$, $z = \mathbf{p}[v_z]$ are all at distance greater than $3n\epsilon$ from any multiple of 2^{-n}, the part of $\mathcal{G}$ that implements the above algorithm computes i, j, k such that the point (x, y, z) lies in the cubelet K_{ijk}; that is, there are $3n$ vertices of the game $\mathcal{G}$ whose $\mathbf{p}$ values are equal to the n bits of the binary representation of i, j, k. Once we have the binary representations of i, j, k, we can feed them into another part of $\mathcal{G}$ that simulates the circuit C. We could simulate the circuit by having vertices that represent gates, using addition (with ceiling 1) to simulate *or*, multiplication for *and*, and $1 - x$ for negation. However, there is a simpler way, one that avoids the complications related to accuracy, to simulate Boolean functions under the assumption that the inputs are 0 or 1:

Lemma 4.5. *There are binary graphical games $\mathcal{G}_\vee, \mathcal{G}_\wedge, \mathcal{G}_\neg$ with two input players a, b (one input player a for $\mathcal{G}_\neg$) and an output player c such that the payoffs of a and b do not depend on the choices of c, and, at any ϵ-Nash equilibrium with $\epsilon < 1/4$ in which $\mathbf{p}[a], \mathbf{p}[b] \in \{0, 1\}$, $\mathbf{p}[c]$ is also in $\{0, 1\}$, and is in fact the result of applying the corresponding Boolean function to the inputs.*

Proof. These games are in the same spirit as $G_<$. In $G_\vee$, for example, the payoff to c is $1/2$ if it plays 0; if c plays 1 its payoff is 1 if at least one of a, b plays 1, and it is 0 if they both play 0. Similarly, for $G_\wedge$ and $G_\neg$. □

It would seem that all we have to do now is to close the loop as follows: in addition to the part of $\mathcal{G}$ that computes the bits of i, j, k, we could have a part that simulates circuit C in the neighborhood of K_{ijk} and decides whether among the vertices of the cubelet K_{ijk} there is a panchromatic one; if not, the vertices v_x, v_y and v_z could be incentivized to change their $\mathbf{p}$ values, say in the direction $\delta_{C(i,j,k)}$, otherwise stay put. To simulate a circuit evaluation in $\mathcal{G}$ we could have one vertex for each gate of the circuit so that, in any ϵ-Nash equilibrium in which all the $\mathbf{p}[v_{b_i(x)}]$'s are $0 - 1$, the vertices corresponding to the outputs of the circuit also play pure strategies, and, furthermore, these strategies correspond correctly to the outputs of the circuit.

But, as we mentioned above, there is a problem: Because of the brittle comparators, at the boundaries of the cubelets the vertices that should represent the values of the bits of i, j, k hold in fact arbitrary reals and, therefore, so do the vertices that represent the outputs of the circuit, and this noise in the calculation can create spurious Nash equilibria. Suppose for example that (x, y, z) lies on the boundary between two cubelets that have color 1, i.e. their centers are assigned vector δ_1 by C, and none of these cubelets have a panchromatic vertex. Then there ought not to be a Nash equilibrium with $\mathbf{p}[v_x] = x$, $\mathbf{p}[v_y] = y$, $\mathbf{p}[v_z] = z$. We would want that, when $\mathbf{p}[v_x] = x$, $\mathbf{p}[v_y] = y$, $\mathbf{p}[v_z] = z$, the vertices v_x, v_y, v_z have the incentive to shift their $\mathbf{p}$ values in direction δ_1, so that v_x prefers to increase $\mathbf{p}[v_x]$. However, on a boundary between two cubelets, some of the "bit values" that get loaded into the vertices $v_{b_i(x)}$, could be other than 0 and 1, and then there is nothing we can say about the output of the circuit that processes these values.

To overcome this difficulty, we resort to the following *averaging maneuver*: We repeat the above computation not just for the point (x, y, z), but also for all $M = (2m+1)^3$ points of the form $(x+p\cdot\alpha, y+q\cdot\alpha, z+s\cdot\alpha)$, for $-m \leq p, q, s \leq m$, where m is a large enough constant to be fixed later (we show below that $m = 20$ is sufficient). The vertices v_x, v_y, v_z are then incentivized to update their values according to the consensus of the results of these computations, most of which are reliable, as we shall show next.

Let us first describe this averaging in more detail. It will be convenient to assume that the output of C is a little more explicit than 3 bits: let us say that C computes six bits Δx^+, Δx^-, Δy^+, Δy^-, Δz^+, Δz^-, such that at most one of Δx^+, Δx^- is 1, at most one of Δy^+, Δy^- is 1, and, similarly, for z, and the increment of the Brouwer function at the center of K_{ijk} is $\alpha \cdot (\Delta x^+ - \Delta x^-, \Delta y^+ - \Delta y^-, \Delta z^+ - \Delta z^-)$, equal to one of the vectors δ_0, δ_1, δ_2, δ_3 specified in the definition of Brouwer, where recall $\alpha = 2^{-2n}$.

The game $\mathcal{G}$ has the following structure: Starting from (x, y, z), some part of the game is devoted to calculating the points $(x+p\cdot\alpha,\ y+q\cdot\alpha,\ z+s\cdot\alpha)$, $-m \leq p,q,s \leq m$. Then, another part evaluates the circuit C on the binary representation of each of these points yielding $6M$ output bits, $\Delta x_1^+, \ldots, \Delta z_M^-$. A final part calculates the following averages

$$(\delta x^+, \delta y^+, \delta z^+) = \frac{\alpha}{M}\sum_{t=1}^{M}(\Delta x_t^+, \Delta y_t^+, \Delta z_t^+), \tag{4.1}$$

$$(\delta x^-, \delta y^-, \delta z^-) = \frac{\alpha}{M}\sum_{t=1}^{M}(\Delta x_t^-, \Delta y_t^-, \Delta z_t^-), \tag{4.2}$$

which correspond to the average positive, respectively negative, shift of all M points.

We have already described above how to implement the bit extraction and the evaluation of a circuit using the gadgets of Lemmas 4.2, 4.3 and 4.5. The computation of points $(x + p \cdot \alpha, y + q \cdot \alpha, z + s \cdot \alpha)$, for all $-m \leq p,q,s \leq m$, is also easy to implement by preparing the values $\alpha|p|$, $\alpha|q|$, $\alpha|s|$, using gadgets $\mathcal{G}_{\alpha|p|}$, $\mathcal{G}_{\alpha|q|}$, $\mathcal{G}_{\alpha|s|}$, and then adding or subtracting the results to x, y and z respectively, depending on whether p is positive or not and, similarly, for q and s. Of course, these computations are subject to truncations at 0 and 1 (see Lemma 4.2).

To implement the averaging of Equations (4.1) and (4.2) we must be careful on the order of operations. Specifically, we first have to multiply the 6 outputs, Δx_t^+, Δx_t^-, Δy_t^+, Δy_t^-, Δz_t^+, Δz_t^-, of each circuit evaluation by $\frac{\alpha}{M}$ using the $\mathcal{G}_{\times\frac{\alpha}{M}}$ gadget and, having done so, we then implement the additions (4.1) and (4.2). Since α will be a very small constant, by doing so we avoid undesired truncations at 0 and 1.

We can now close the loop by inserting equality, addition and subtraction gadgets, $\mathcal{G}_=$, $\mathcal{G}_+$, $\mathcal{G}_-$, that force, at equilibrium, x to be equal to $(x' + \delta x^+) - \delta x^-$, where x' is a copy of x created using $\mathcal{G}_=$, and, similarly, for y and z. Note that in $\mathcal{G}$ we respect the order of operations when implementing $(x'+\delta x^+)-\delta x^-$ to avoid undesired truncations at 0 or 1 as we shall see next. This concludes the reduction; it is clear that it can be carried out in polynomial time.

Our proof is concluded by the following claim. For the following lemma we choose $\epsilon = \alpha^2$. Recall from our definition of BROUWER that $\alpha = 2^{-2n}$.

Lemma 4.6. *In any ϵ-Nash equilibrium of the game $\mathcal{G}$, one of the vertices of the cubelet(s) that contain $(\mathbf{p}[v_x], \mathbf{p}[v_y], \mathbf{p}[v_z])$ is panchromatic.*

Proof. We start by pointing out a simple property of the increments $\delta_0, \ldots, \delta_3$:

Lemma 4.7. *Suppose that for nonnegative integers $k_0, \ldots, k_3$ all three coordinates of $\sum_{i=0}^{3} k_i \delta_i$ are smaller in absolute value than $\frac{\alpha K}{5}$ where $K = \sum_{i=0}^{3} k_i$. Then all four k_i are positive.*

Proof. For the sake of contradiction, suppose that $k_1 = 0$. It follows that $k_0 < K/5$ (otherwise the negative x coordinate of $\sum_{i=0}^{3} k_i \delta_i$ would be too large), and thus one of k_2, k_3 is larger than $2K/5$, which makes the corresponding coordinate of $\sum_{i=0}^{3} k_i \delta_i$ too large, a contradiction. Similarly, if $k_2 = 0$ or $k_3 = 0$. Finally, if $k_0 = 0$ then one of k_1, k_2, k_3 is at least $K/3$ and the associated coordinate of $\sum_{i=0}^{3} k_i \delta_i$ at least $\alpha K/3$, again a contradiction. □

Let us denote by $v_{\delta x^+}$, $\{v_{\Delta x_t^+}\}_{1 \le t \le M}$ the vertices of $\mathcal{G}$ that represent the values δx^+, $\{\Delta x_t^+\}_{1 \le t \le M}$. To implement the averaging

$$\delta x^+ = \frac{\alpha}{M} \sum_{t=1}^{M} \Delta x_t^+$$

inside $\mathcal{G}$, we first multiply each $\mathbf{p}[v_{\Delta x_t^+}]$ by $\frac{\alpha}{M}$ using a $\mathcal{G}_{\frac{\alpha}{M}}$ gadget, and then sum the results by a sequence of addition gadgets. Since each of these operations induces an error of $\pm\epsilon$ and there are $2M - 1$ operations it follows that

$$\mathbf{p}[v_{\delta x^+}] = \frac{\alpha}{M} \sum_{t=1}^{M} \mathbf{p}[v_{\Delta x_t^+}] \pm (2M-1)\epsilon. \tag{4.3}$$

Similarly, denoting by $v_{\delta x^-}$, $\{v_{\Delta x_t^-}\}_{1\le t\le M}$ the vertices of $\mathcal{G}$ that represent the values δx^-, $\{\Delta x_t^-\}_{1\le t\le M}$, it follows that

$$\mathbf{p}[v_{\delta x^-}] = \frac{\alpha}{M}\sum_{t=1}^{M}\mathbf{p}[v_{\Delta x_t^-}] \pm (2M-1)\epsilon, \tag{4.4}$$

and, similarly, for directions y and z.

We continue the proof by distinguishing two subcases for the location of $(x, y, z) = (\mathbf{p}[v_x], \mathbf{p}[v_y], \mathbf{p}[v_z])$

(a) the point $(\mathbf{p}[v_x], \mathbf{p}[v_y], \mathbf{p}[v_z])$ is further than $(m+1)\alpha$ from every face of the cube $[0,1]^3$,

(b) the point $(\mathbf{p}[v_x], \mathbf{p}[v_y], \mathbf{p}[v_z])$ is at distance at most $(m+1)\alpha$ from some face of the cube $[0,1]^3$.

Case (a): Denoting by $v_{x+p\cdot\alpha}$ the player of $\mathcal{G}$ that represents $x+p\cdot\alpha$, the small value of ϵ relative to α implies that at most one of the values $\mathbf{p}[v_{x+p\cdot\alpha}]$, $-m \le p \le m$, can be $3n\epsilon$-close to a multiple of 2^{-n}, and, similarly, for the directions y and z. Indeed, recall that $x+p\cdot\alpha$ is computed from x by first preparing the value $|p|\alpha$ via a $\mathcal{G}_{|p|\alpha}$ gadget and then adding or subtracting the result to x —depending on whether p is positive or not— using $\mathcal{G}_+$ or $\mathcal{G}_-$. It follows that

$$\mathbf{p}[v_{x+p\cdot\alpha}] = \mathbf{p}[v_x] + p\cdot\alpha \pm 2\epsilon, \tag{4.5}$$

since each gadget introduces an error of $\pm\epsilon$, where note that there are no truncations at 0 or 1, because, by assumption, $(m+1)\alpha < \mathbf{p}[v_x] < 1-(m+1)\alpha$. Consequently, for $p > p'$,

$$\mathbf{p}[v_{x+p\cdot\alpha}] - \mathbf{p}[v_{x+p'\cdot\alpha}] \ge (p-p')\cdot\alpha - 4\epsilon > 6n\epsilon,$$

and, moreover,

$$\mathbf{p}[v_{x+m\cdot\alpha}] - \mathbf{p}[v_{x-m\cdot\alpha}] \leq 2m\cdot\alpha + 4\epsilon << 2^{-n}, \tag{4.6}$$

since m is a constant, $\alpha = 2^{-2n}$, $\epsilon = \alpha^2$, and n is assumed to be large enough. Hence, from among the $M = (2m+1)^3$ circuit evaluations, all but at most $3(2m+1)^2$, or at least $K = (2m-2)(2m+1)^2$, compute legitimate, i.e. binary, Δx^+ etc. values.

Let us denote by $\mathcal{K} \subseteq \{-m,\ldots,m\}^3$, $|\mathcal{K}| \geq K$, the set of values (p,q,r) for which the bit extraction from $(\mathbf{p}[v_{x+p\cdot\alpha}], \mathbf{p}[v_{y+q\cdot\alpha}], \mathbf{p}[v_{z+r\cdot\alpha}])$ results in binary outputs and, consequently, so does the circuit evaluation. Let

$$S_{\mathcal{K}} = \frac{\alpha}{M}\sum_{t\in\mathcal{K}}(\mathbf{p}[v_{\Delta x_t^+}] - \mathbf{p}[v_{\Delta x_t^-}], \mathbf{p}[v_{\Delta y_t^+}] - \mathbf{p}[v_{\Delta y_t^-}], \mathbf{p}[v_{\Delta z_t^+}] - \mathbf{p}[v_{\Delta z_t^-}]), \tag{4.7}$$

$$S_{\mathcal{K}^c} = \frac{\alpha}{M}\sum_{t\notin\mathcal{K}}(\mathbf{p}[v_{\Delta x_t^+}] - \mathbf{p}[v_{\Delta x_t^-}], \mathbf{p}[v_{\Delta y_t^+}] - \mathbf{p}[v_{\Delta y_t^-}], \mathbf{p}[v_{\Delta z_t^+}] - \mathbf{p}[v_{\Delta z_t^-}]). \tag{4.8}$$

Recall that we have inserted gadgets $\mathcal{G}_+$, $\mathcal{G}_-$ and $\mathcal{G}_=$ in $\mathcal{G}$ to enforce that in a Nash equilibrium $x = x' + \delta x^+ - \delta x^-$, where x' is a copy of x. Because of the defection of the gadgets this will not be exactly tight in an ϵ-Nash equilibrium. More precisely, denoting by $v_{x'}$ the player of $\mathcal{G}$ corresponding to x', the following are true in an ϵ-Nash equilibrium

$$\mathbf{p}[v_{x'}] = \mathbf{p}[v_x] \pm \epsilon$$

$$\mathbf{p}[v_x] = \mathbf{p}[v_x'] + \mathbf{p}[v_{\delta x^+}] - \mathbf{p}[v_{\delta x^-}] \pm 2\epsilon,$$

where for the second observe that both $\mathbf{p}[v_{\delta x^+}]$ and $\mathbf{p}[v_{\delta x^-}]$ are bounded above by $\alpha + (2M-1)\epsilon$ so there will be no truncations at 0 or 1 when adding $\mathbf{p}[v_{\delta x^+}]$ to $\mathbf{p}[v_x']$

and then subtracting $\mathbf{p}[v_{\delta x^-}]$. By combining the above we get

$$\mathbf{p}[v_{\delta x^+}] - \mathbf{p}[v_{\delta x^-}] = \pm 3\epsilon$$

and, similarly, for y and z

$$\mathbf{p}[v_{\delta y^+}] - \mathbf{p}[v_{\delta y^-}] = \pm 3\epsilon$$

$$\mathbf{p}[v_{\delta z^+}] - \mathbf{p}[v_{\delta z^-}] = \pm 3\epsilon.$$

Now, if we use (4.3), (4.4), (4.7), (4.8) we derive

$$\left|S_{\mathcal{K}_\ell} + S_{\mathcal{K}^c{}_\ell}\right| \leq (4M+1)\epsilon, \text{ for } \ell = x, y, z,$$

where $S_{\mathcal{K}_\ell}$, $S_{\mathcal{K}^c{}_\ell}$ is the ℓ coordinate of $S_{\mathcal{K}}$, $S_{\mathcal{K}^c}$. Moreover, since $|\mathcal{K}| \geq K$, the summation $S_{\mathcal{K}^c{}_\ell}$ has at most $M - K$ summands and because each of them is at most $\frac{\alpha}{M}$ in absolute value it follows that $|S_{\mathcal{K}^c{}_\ell}| \leq \frac{\alpha}{M}(M-K)$, for all $\ell = x, y, z$. Therefore, we have that

$$\left|S_{\mathcal{K}_\ell}\right| \leq (4M+1)\epsilon + \frac{M-K}{M}\alpha, \text{ for } \ell = x, y, z.$$

Finally, note by the definition of the set $\mathcal{K}$ that, for all $(p, q, r) \in \mathcal{K}$, the bit extraction from $(\mathbf{p}[v_{x+p\cdot\alpha}], \mathbf{p}[v_{y+q\cdot\alpha}], \mathbf{p}[v_{z+r\cdot\alpha}])$ and the following circuit evaluation result in binary outputs. Therefore, $S_{\mathcal{K}} = \frac{1}{M}\sum_{i=0}^{3} k_i\delta_i$ for some nonnegative integers $k_0, \ldots, k_3$ adding up to $|\mathcal{K}|$. From the above we get that

$$\left|\sum_{i=0}^{3} k_i\delta_i\right|_\infty \leq (4M+1)M\epsilon + (M-K)\alpha \leq (4M+1)M\epsilon + 3(2m+1)^2\alpha.$$

By choosing $m = 20$, the bound becomes less than $\alpha K/5$, and so Lemma 4.7 applies. It follows that, among the results of the $|\mathcal{K}|$ circuit computations, all four $\delta_0, \ldots, \delta_3$

appeared. And, since every point on which the circuit C is evaluated is within ℓ_1 distance at most $3m\alpha + 6\epsilon << 2^{-n}$ from the point (x, y, z), as Equation (4.5) dictates, this implies that among the corners of the cubelet(s) containing (x, y, z) there must be one panchromatic corner, completing the proof of Lemma 4.6 for case (a).

Case (b): We will show that there is no ϵ-Nash equilibrium in which $(\mathbf{p}[v_x], \mathbf{p}[v_y], \mathbf{p}[v_z])$ is within distance $(m+1)\alpha$ from a face of $[0,1]^3$. We will argue so only for the case

$$\mathbf{p}[v_x] \leq (m+1)\alpha,$$

$$(m+1)\alpha < \mathbf{p}[v_y] < 1 - (m+1)\alpha,$$

$$(m+1)\alpha < \mathbf{p}[v_z] < 1 - (m+1)\alpha;$$

the other cases follow similarly.

First, we show that, for all $-m \leq p \leq m$, the bit extraction from $\mathbf{p}[v_{x+p\cdot\alpha}]$ results in binary outputs. From the proof of Lemma 4.4 it follows that, to show this, it is enough to establish that $\mathbf{p}[v_{x+p\alpha}] < 2^{-n} - 3n\epsilon$, for all p. Indeed, for $p \geq 0$, Equation (4.5) applies because there are no truncations at 1 at the addition gadget. So for $p \geq 0$ we get

$$\mathbf{p}[v_{x+p\cdot\alpha}] \leq \mathbf{p}[v_x] + p\cdot\alpha + 2\epsilon \leq (m+1)\alpha + m\alpha + 2\epsilon << 2^{-n} - 3n\epsilon$$

On the other hand, for $p < 0$, there might be a truncation at 0 when we subtract the value $|p|\alpha$ from $\mathbf{p}[v_x]$. Nevertheless, we have that

$$\begin{aligned}\mathbf{p}[v_{x+p\cdot\alpha}] = \max\{\ 0\ ,\ \mathbf{p}[v_x] - (|p|\alpha \pm \epsilon)\ \} \pm \epsilon \leq \mathbf{p}[v_x] + 2\epsilon \\ \leq (m+1)\alpha + 2\epsilon << 2^{-n} - 3n\epsilon.\end{aligned}$$

Therefore, for all $-m \leq p \leq m$, the bit extraction from $\mathbf{p}[v_{x+p\cdot\alpha}]$ is successful, i.e. results in binary outputs.

For the directions y and z the picture is exactly the same as in case (a) and, therefore, there exists at most one q, $-m \leq q \leq m$, and at most one r, $-m \leq r \leq m$, for which the bit extraction from $\mathbf{p}[v_{y+q\cdot\alpha}]$ and $\mathbf{p}[v_{z+r\cdot\alpha}]$ fails. Therefore, from among the $M = (2m+1)^3$ points of the form $(\mathbf{p}[v_{x+p\cdot\alpha}], \mathbf{p}[v_{y+q\cdot\alpha}], \mathbf{p}[v_{z+r\cdot\alpha}])$ the bit extraction succeeds in all but at most $2(2m+1)^2$ of them.

Therefore, at least $K' = (2m-1)(2m+1)^2$ circuit evaluations are successful, i.e. in binary arithmetic, and, moreover, they correspond to points inside cubelets of the form K_{ijk} with $i = 0$. In particular, from Equation (4.6) and the analogous equations for the y and z coordinates, it follows that the successful circuit evaluations correspond to points inside at most 4 neighboring cubelets of the form K_{0jk}. Since these cubelets are adjacent to the $x = 0$ face of the cube, from the properties of the circuit C in the definition of the problem BROUWER, it follows that, among the outputs of these evaluations, one of the vectors δ_0, δ_1, δ_2, δ_3 is missing. Without loss of generality, let us assume that δ_0 is missing. Then, since there are K' successful evaluations, one of δ_1, δ_2, δ_3 appears at least $K'/3$ times.

If this is vector δ_1 (a similar argument applies for the cases δ_2, δ_3), then denoting by $v_{x'+\delta x^+}$ the player corresponding to $x' + \delta x^+$, the following should be true in an ϵ-Nash equilibrium.

$$\mathbf{p}[v_x] + \epsilon \geq \mathbf{p}[v_{x'}] \geq \mathbf{p}[v_x] - \epsilon,$$

$$\alpha + (2M-1)\epsilon \geq \mathbf{p}[v_{\delta x^+}] \geq \frac{K'}{3M}\alpha - (2M-1)\epsilon,$$

$$\mathbf{p}[v_{x'+\delta x^+}] \geq \min(1, \mathbf{p}[v_{x'}] + \mathbf{p}[v_{\delta x^+}]) - \epsilon \geq \mathbf{p}[v_{x'}] + \mathbf{p}[v_{\delta x^+}] - \epsilon,$$

$$\frac{M-K'}{M}\alpha + (2M-1)\epsilon \geq \mathbf{p}[v_{\delta x^-}],$$

$$\mathbf{p}[v_x] \geq \max(0, \mathbf{p}[v_{x'+\delta x^+}] - \mathbf{p}[v_{\delta x^-}]) - \epsilon \geq \mathbf{p}[v_{x'+\delta x^+}] - \mathbf{p}[v_{\delta x^-}] - \epsilon;$$

in the second inequality of the third line above, we used that $\mathbf{p}[v_x] \leq (m+1)\alpha$. Combining the above we get

$$\begin{aligned}\mathbf{p}[v_x] &\geq \mathbf{p}[v_{x'}] + \mathbf{p}[v_{\delta x^+}] - \mathbf{p}[v_{\delta x^-}] - 2\epsilon \\ &\geq \mathbf{p}[v_x] + \mathbf{p}[v_{\delta x^+}] - \mathbf{p}[v_{\delta x^-}] - 3\epsilon\end{aligned}$$

or equivalently that

$$\mathbf{p}[v_{\delta x^-}] \geq \mathbf{p}[v_{\delta x^+}] - 3\epsilon,$$

which implies

$$\frac{M-K'}{M}\alpha + (4M+1)\epsilon \geq \frac{K'}{3M}\alpha,$$

which is not satisfied by our selection of parameters.

□

To conclude the proof of Theorem 4.1, if we find any ϵ-Nash equilibrium of $\mathcal{G}$, Lemma 4.6 has shown that by reading off the first n binary digits of $\mathbf{p}[v_x]$, $\mathbf{p}[v_y]$ and $\mathbf{p}[v_z]$ we obtain a solution to the corresponding instance of Brouwer. □

4.2 Two-Player Games

Soon after our proof became available, Chen and Deng [CD06] showed that our PPAD-completeness result can be extended to the important two-player case. Here we present a rather simple modification of our proof from the previous section establishing this result.

Theorem 4.8 ([CD06])**.** 2-Nash *is* PPAD*-complete.*

Proof. Let us define d-additive graphical Nash to be the problem d-graphical Nash restricted to *bipartite graphical games with additive utility functions* defined next.

Definition 4.9. *Let $\mathcal{GG}$ be a graphical game with underlying graph $G = (V, E)$. We call $\mathcal{GG}$ a* bipartite graphical game with additive utility functions *if G is a bipartite graph and, moreover, for each vertex $v \in V$ and for every pure strategy $s_v \in S_v$ of that player, the expected payoff of v for playing the pure strategy s_v is a linear function of the mixed strategies of the vertices in $\mathcal{N}_v \setminus \{v\}$ with rational coefficients; that is, there exist rational numbers $\{\alpha^{s_v}_{u,s_u}\}_{u \in \mathcal{N}_v \setminus \{v\}, s_u \in S_u}$, $\alpha^{s_v}_{u,s_u} \in [0,1]$ for all $u \in \mathcal{N}(v) \setminus \{v\}$, $s_u \in S_u$, such that the expected payoff to vertex v for playing pure strategy s_v is*

$$\sum_{u \in \mathcal{N}_v \setminus \{v\}, s_u \in S_u} \alpha^{s_v}_{u,s_u} \mathbf{p}[u : s_u],$$

where $\mathbf{p}[u : s_u]$ denotes the probability that vertex u plays pure strategy s_u.

The proof is based on the following lemmas.

Lemma 4.10. Brouwer *is poly-time reducible to* 3-additive graphical Nash.

Lemma 4.11. 3-additive graphical Nash *is poly-time reducible to* 2-Nash.

Proof of Lemma 4.10: The reduction is almost identical to the one in the proof of Theorem 4.1. Recall that given an instance of Brouwer a graphical game was constructed using the gadgets $\mathcal{G}_\alpha, \mathcal{G}_{\times\alpha}, \mathcal{G}_=, \mathcal{G}_+, \mathcal{G}_-, \mathcal{G}_*$, $\mathcal{G}_\vee, \mathcal{G}_\wedge, \mathcal{G}_\neg$, and $\mathcal{G}_>$. In fact, gadget $\mathcal{G}_*$ is not required, since only multiplication by a constant is needed which can be accomplished via the use of gadget $\mathcal{G}_{\times\alpha}$. Moreover, it is not hard to see by looking at the payoff tables of the gadgets defined in Section 3.1 and Lemma 4.3 that, in gadgets $\mathcal{G}_\alpha$, $\mathcal{G}_{\times\alpha}$, $\mathcal{G}_=$, $\mathcal{G}_+$, $\mathcal{G}_-$, and $\mathcal{G}_>$, the non-input vertices have the *additive utility functions* property of Definition 4.9. Let us further modify the games $\mathcal{G}_\vee, \mathcal{G}_\wedge, \mathcal{G}_\neg$ so that their output vertices have the *additive utility functions* property.

Lemma 4.12. *There are binary graphical games $\mathcal{G}_\vee, \mathcal{G}_\wedge, \mathcal{G}_\neg$ with two input players a, b (one input player a for $\mathcal{G}_\neg$) and an output player c such that the payoffs of a and b do not depend on the choices of c, c's payoff satisfies the additive utility functions* *property, and, in any ϵ-Nash equilibrium with $\epsilon < 1/4$ in which $\mathbf{p}[a], \mathbf{p}[b] \in \{0,1\}$, $\mathbf{p}[c]$ is also in $\{0,1\}$, and is in fact the result of applying the corresponding Boolean function to the inputs.*

Proof. For $\mathcal{G}_\vee$, the payoff of player c is $0.5\mathbf{p}[a]+0.5\mathbf{p}[b]$ for playing 1 and $\frac{1}{4}$ for playing 0. For $\mathcal{G}_\wedge$, the payoff of player c is $0.5\mathbf{p}[a]+0.5\mathbf{p}[b]$ for playing 1 and $\frac{3}{4}$ for playing 0. For $\mathcal{G}_\neg$, the payoff of player c is $\mathbf{p}[a]$ for playing 0 and $\mathbf{p}[a:0]$ for playing 1. □

If the modified gadgets $\mathcal{G}_\vee, \mathcal{G}_\wedge, \mathcal{G}_\neg$ specified by Lemma 4.12 are used in the construction of Theorem 4.1, all vertices of the resulting graphical game satisfy the *additive utility functions* property of Definition 4.9. To make sure that the graphical game is also bipartite we modify the gadgets $\mathcal{G}_\vee, \mathcal{G}_\wedge, \mathcal{G}_\neg$, and $\mathcal{G}_>$ with the insertion of an extra output vertex. The modification is the same for all 4 gadgets: let c be the output vertex of any of these gadgets; we introduce a new output vertex e, whose payoff only depends on the strategy of c, but c's payoff does not depend on the strategy of e, and such that the payoff of e is $\mathbf{p}[c]$ for playing 1 and $\mathbf{p}[c:0]$ for playing 0 (i.e. e "copies" c, if c's strategy is pure). It is not hard to see that, for every gadget, the new output vertex has the same behavior with regards to the strategies of the input vertices as the old output vertex, as specified by Lemmas 4.3 and 4.12. Moreover, it is not hard to verify that the graphical game resulting from the construction of Theorem 4.1 with the use of the modified gadgets $\mathcal{G}_\vee, \mathcal{G}_\wedge, \mathcal{G}_\neg$, and $\mathcal{G}_>$ is bipartite; indeed, it is sufficient to color blue the input and output vertices of all $\mathcal{G}_{\times\alpha}$, $\mathcal{G}_=$, $\mathcal{G}_+$, $\mathcal{G}_-$, $\mathcal{G}_\vee$, $\mathcal{G}_\wedge$, $\mathcal{G}_\neg$, and $\mathcal{G}_>$ gadgets used in the construction, blue the output vertices of all $\mathcal{G}_\alpha$ gadgets used, and red the remaining vertices. □

Proof of Lemma 4.11: Let $\widetilde{\mathcal{GG}}$ be a bipartite graphical game of maximum degree 3 with additive utility functions and $\mathcal{GG}$ the graphical game resulting after rescaling all utilities to the set $[0,1]$, e.g. by dividing all utilities by $\max\{\tilde{u}\}$, where $\max\{\tilde{u}\}$ is the largest entry in the payoff tables of game $\widetilde{\mathcal{GG}}$. Also, let $\epsilon < 1$. In time polynomial in

$|\mathcal{GG}|+\log(1/\epsilon)$, we will specify a 2-player normal-form game $\mathcal{G}$ and an accuracy ϵ' with the property that, given an ϵ'-Nash equilibrium of $\mathcal{G}$, one can recover in polynomial time an ϵ-Nash equilibrium of $\mathcal{GG}$. This will be enough, since an ϵ-Nash equilibrium of $\mathcal{GG}$ is trivially an $\epsilon \cdot \max\{\tilde{u}\}$-Nash equilibrium of game $\widetilde{\mathcal{GG}}$ and, moreover, $|\mathcal{GG}|$ is polynomial in $|\widetilde{\mathcal{GG}}|$.

The construction of $\mathcal{G}$ from $\mathcal{GG}$ is almost identical to the one described in Figure 3.5. Let $V = V_1 \sqcup V_2$ be the bipartition of the vertices of set V so that all edges are between a vertex in V_1 and a vertex in V_2. Let us define $c : V \to \{1, 2\}$ as $c(v) = 1$ iff $v \in V_1$ and let us assume, without loss of generality, that $|v : c(v) = 1| = |v : c(v) = 2|$; otherwise, we can add to $\mathcal{GG}$ isolated vertices to make up any shortfall. Suppose that n is the number of vertices in $\mathcal{GG}$ (after the possible addition of isolated vertices) and t the cardinality of the strategy sets of the vertices in V, and let $\epsilon' = \epsilon/n$. Let us then employ the Steps 4 and 5 of the algorithm in Figure 3.5 to construct the normal-form game $\mathcal{G}$ from the graphical game $\mathcal{GG}$; however, we choose $M = \frac{6tn}{\epsilon}$, and modify Step 5b to read as follows

(b)' for $v \in V$ and $s_v \in S_v$, if $c(v) = p$ and s contains (v, s_v) and (u, s_u) for some $u \in \mathcal{N}(v) \setminus \{v\}$, $s_u \in S_u$, then $u^p_s = \alpha^{s_v}_{u,s_u}$,

where we used the notation from Definition 4.9.

We argue next that, given an ϵ'-Nash equilibrium $\{x^p_{(v,a)}\}_{p,v,a}$ of $\mathcal{G}$, $\{x^v_a\}_{v,a}$ is an ϵ-Nash equilibrium of $\mathcal{GG}$, where

$$x^v_a = x^{c(v)}_{(v,a)} \Big/ \sum_{j \in S_v} x^{c(v)}_{(v,j)}, \quad \forall v \in V, a \in S_v.$$

Suppose that $p = c(v)$ for some vertex v of the graphical game $\mathcal{GG}$. As in the proof of Theorem 3.8, Lemma 3.11, it can be shown that in any ϵ'-Nash equilibrium of the game $\mathcal{G}$,

$$\Pr[p \text{ plays } v] \in \left[\frac{2}{n} - \frac{1}{M}, \frac{2}{n} + \frac{1}{M}\right].$$

Now, without loss of generality assume that $p = 1$ (the pursuer) and suppose v is vertex $v_i^{(p)}$, in the notation of Figure 3.5. Then, in an ϵ'-Nash equilibrium of the game $\mathcal{G}$, we have, by the definition of a Nash equilibrium, that for all strategies $s_v, s'_v \in S_v$ of vertex v:

$$\mathbb{E}\left[\text{payoff to } p \text{ for playing } (v, s_v)\right] > \mathbb{E}\left[\text{payoff to } p \text{ for playing } (v, s'_v)\right] + \epsilon' \Rightarrow x^p_{(v,s'_v)} = 0. \tag{4.9}$$

But

$$\mathbb{E}\left[\text{payoff to } p \text{ for playing } (v, s_v)\right] = M \cdot \Pr\left[p + 1 \text{ plays } v_i^{(p+1)}\right] + \sum_{u \in \mathcal{N}_v \setminus \{v\}, s_u \in S_u} \alpha^{s_v}_{u,s_u} x^{c(u)}_{(u,s_u)}$$

and, similarly, for s'_v. Therefore, (4.9) implies

$$\sum_{u \in \mathcal{N}_v \setminus \{v\}, s_u \in S_u} \alpha^{s_v}_{u,s_u} x^{c(u)}_{(u,s_u)} > \sum_{u \in \mathcal{N}_v \setminus \{v\}, s_u \in S_u} \alpha^{s'_v}_{u,s_u} x^{c(u)}_{(u,s_u)} + \epsilon' \Rightarrow x^p_{(v,s'_v)} = 0. \tag{4.10}$$

Lemma 4.13. *For all v, $a \in S_v$,*

$$\left| x^v_a - \frac{x^{c(v)}_{(v,a)}}{2/n} \right| \leq \frac{n}{2M}.$$

Proof. We have

$$\begin{aligned} \left| x^v_a - \frac{x^{c(v)}_{(v,a)}}{2/n} \right| &= \left| \frac{x^{c(v)}_{(v,a)}}{\Pr[c(v) \text{ plays } v]} - \frac{x^{c(v)}_{(v,a)}}{2/n} \right| \\ &= \frac{x^{c(v)}_{(v,a)}}{\Pr[c(v) \text{ plays } v]} \frac{|\Pr[c(v) \text{ plays } v] - 2/n|}{2/n} \leq \frac{n}{2M}, \end{aligned}$$

where we used that $\sum_{j \in S_v} x_{(v,j)}^{c(v)} = \Pr[c(v) \text{ plays } v]$ and $|\Pr[c(v) \text{ plays } v] - 2/n| \leq \frac{1}{M}$. □

By (4.10) and Lemma 4.13, we get that, for all $v \in V$, $s_v, s'_v \in S_v$,

$$\sum_{u \in \mathcal{N}_v \setminus \{v\}, s_u \in S_u} \alpha_{u,s_u}^{s_v} x_{s_u}^u > \sum_{u \in \mathcal{N}_v \setminus \{v\}, s_u \in S_u} \alpha_{u,s_u}^{s'_v} x_{s_u}^u + \frac{n}{2}\epsilon' + |\mathcal{N}_v \setminus \{v\}| t \frac{n}{M} \Rightarrow x_{s'_v}^v = 0.$$

Since $\frac{n}{2}\epsilon' + |\mathcal{N}_v \setminus \{v\}| t \frac{n}{M} \leq \epsilon$, it follows that $\{x_a^v\}_{v,a}$ is an ϵ-Nash equilibrium of the game $\mathcal{GG}$. □ □

4.3 Other Classes of Games and Fixed Points

There are several special cases of the Nash equilibrium problem for which PPAD-hardness persists. It has been shown, for example, that finding a Nash equilibrium of two-player normal-form games in which all utilities are restricted to take values 0 or 1 (the so-called *win-lose* case) remains PPAD-complete [AKV05, CTV07]. The Nash equilibrium problem in two-player *symmetric games* — that is, games in which the two players have the same strategy sets, and their utility is the same function of their own and the other player's strategy — is also PPAD-complete. [1] Moreover, rather surprisingly, it is essentially PPAD-complete to even play *repeated games* [BCI$^+$08] (the so-called "Folk Theorem for repeated games" [Rub79] notwithstanding).

And, what is known about the complexity of the Nash Equilibrium problem in other classes of succinctly representable games with many players (besides the graphical games which we have resolved)? For example, are these problems even in PPAD? [2] In [DFP06], we provide a general sufficient condition, satisfied by all known succinct

[1]This follows from a symmetrization argument of von Neumann [BN50] providing a reduction from the Nash equilibrium problem in general two-player games to that in symmetric games (see, also, the construction of Gale, Kuhn, Tucker [GKT50]).

[2]It is typically easy to see that they cannot be easier than the normal-form case.

representations of games, such as *congestion games* [Ros73, FPT04] and *extensive-form games* [OR94], for membership of the Nash equilibrium problem in the class PPAD. The basic idea is using the "arithmetical" gadgets in our present proof to simulate the calculation of utilities in these succinct games.

Our techniques can be used to treat two other open problems in complexity. One is that of the complexity of simple stochastic games defined in [Con92], heretofore known to be in TFNP, but not in any of the more specialized classes like PPAD or PLS. Now, it is known that this problem is equivalent to evaluating combinational circuits with MAX, MIN, and AVERAGE gates. Since all three kinds of gates can be implemented by the graphical games in our construction, it follows that solving simple stochastic games is in PPAD. [3]

Similarly, by an explicit construction we can show the following.

Theorem 4.14. *Let* $p : [0,1] \to \mathbb{R}$ *be any polynomial function such that* $p(0) < 0$ *and* $p(1) > 0$. *Then there exists a graphical game in which all vertices have two strategies,* 0 *and* 1*, and in which the mixed Nash equilibria correspond to a particular vertex* v *playing strategy* 1 *with probability equal to the roots of* $p(x)$ *between* 0 *and* 1.

Proof Sketch. Let p be described by its coefficients $\alpha_0, \alpha_1, \ldots, \alpha_n$, so that

$$p(x) := \alpha_n x^n + \alpha_{n-1} x^{n-1} + \ldots + \alpha_1 x + \alpha_0.$$

Taking $A := (\sum_{i=0}^{n} |\alpha_i|)^{-1}$, it is easy to see that the range of the polynomial $q(x) := \frac{1}{2}Ap(x) + \frac{1}{2}$ is $[0,1]$, that $q(0) < \frac{1}{2}$, $q(1) > 1/2$, and that every point $r \in [0,1]$ such that $q(r) = \frac{1}{2}$ is a root of p. We define next a graphical game $\mathcal{GG}$ in which all vertices have two strategies, 0 and 1, and a designated vertex v of $\mathcal{GG}$ satisfies the following

(i) in any mixed Nash equilibrium of $\mathcal{GG}$ the probability x_1^v by which v plays strategy 1 satisfies $q(x_1^v) = 1/2$;

[3] One has to pay attention to the approximation; see [EY07] for details.

(ii) for any root r of p in $[0, 1]$, there exists a mixed Nash equilibrium of $\mathcal{GG}$ in which $x_1^v = r$;

The graphical game has the following structure:

- there is a component graphical game $\mathcal{GG}_q$ with an "input vertex" v and an "output vertex" u such that, in any Nash equilibrium of $\mathcal{GG}$, the mixed strategies of u and v satisfy $x_1^u = q(x_1^v)$; a graphical game which progressively performs the computations required for the evaluation of $q(\cdot)$ on x_1^v can be easily constructed using our game-gadgets; note that the computations can be arranged in such an order that no truncations at 0 or 1 happen (recall the rescaling by $\frac{1}{2}A$ and the shifting around 1/2 done above);

- a comparator game $\mathcal{G}_>$ (see Lemma 4.3) compares the mixed strategy of u with the value $\frac{1}{2}$, prepared by a $\mathcal{G}_{1/2}$ gadget (see Section 3.1), so that the output vertex of the comparator game plays 0 if $x_1^u > \frac{1}{2}$, 1 if $x_1^u < \frac{1}{2}$, and anything if $x_1^u = \frac{1}{2}$;

- we identify the output player of $\mathcal{G}_>$ with player v;

It is not hard to see that $\mathcal{GG}$ satisfies Properties (i) and (ii).

□

As a corollary of Theorem 4.14, it follows that fixed points of polynomials can be computed by computing (exact) Nash equilibria of graphical games. Computing fixed points of polynomials via exact Nash equilibria in graphical games can be extended to the multi-variate case again via the use of game gadgets to evaluate the polynomial and the use of a series of $\mathcal{G}_=$ gadgets to set the output equal to the input.

Both this result and the result about simple stochastic games noted above were shown independently by [EY07], while Theorem 4.14 was already shown by Bubelis [Bub79].

Chapter 5

Computing Approximate Equilibria

In the previous chapters, we establish that r-Nash is PPAD-complete, for $r \geq 2$. This result implies that it is PPAD-complete to compute an ϵ-Nash equilibrium of a normal-form game with at least two players, for any approximation ϵ scaling as an inverse exponential function of the size of game. The same is true for graphical games of degree 3 or larger, since d-graphical Nash was also shown to be PPAD-complete, for all $d \geq 3$. This brings about the following question.

- *Is computing an ϵ-Nash equilibrium easier, if ϵ is larger?*

It turns out that, for any ϵ which is inverse polynomial in n, computing an ϵ-Nash equilibrium of a 2-player n-strategy game remains PPAD-complete. This result, established by Chen, Deng and Teng [CDT06a], follows from a modification of our reduction in which the starting Brouwer problem is defined not on the 3-dimensional cube, but on the n-dimensional hypercube. Intuitively, the difference is this: In order to create the exponentially many cells needed to embed the "line" of the end of the line problem, our construction had to resort to exponentially small cell size. On the other hand, the n-dimensional hypercube contains exponentially many cells, all of reasonably large size. This observation implies that approximation which is inverse polynomial in n is sufficient to encode the END OF THE LINE instance into a 2-player n-strategy game. In fact, the same negative result can be extended to graphical games: For any ϵ which is inverse polynomial in n, computing an ϵ-Nash equilibrium of a n-player graphical game of degree 3 is PPAD-complete. So, for both normal-form and graphical games, a fully polynomial-time approximation scheme seems unlikely. The following important question emerges at the boundary of intractability.

- *Is there a polynomial-time approximation scheme for the Nash equilibrium problem?*

We discuss this question in its full generality in Section 5.1. We also present special classes of two-player games for which there exists a polynomial-time approximation scheme, and we conclude the section with a discussion of challenges towards obtaining a polynomial-time approximation scheme for general two-player games. In Sections 5.2 through 5.9, we consider a broad and important class of games, called *anonymous games*, for which we present a polynomial-time approximation scheme.

5.1 General Games and Special Classes

The problem of computing approximate equilibria was considered by Lipton, Markakis and Mehta in [LMM03], where a quasi-polynomial-time algorithm was given for normalized normal-form games. [1] This algorithm is based upon the realization that, in every r-player game, there exists an ϵ-approximate Nash equilibrium in which all players' mixed strategies have support of size $O\left(\frac{r^2 \log(r^2 n)}{\epsilon^2}\right)$. Hence, an ϵ-approximate equilibrium can be found by exhaustive search over all mixed strategy profiles with this support size. Despite extensive research on the subject, no improvement of this result is known for general values of ϵ. For fixed values of ϵ, we have seen a se-

[1] Most of the research on computing approximate Nash equilibria has focused on normalized games; since the approximation is defined in the additive sense this decision is a reasonable one.

quence of results, computing ϵ-Nash equilibria of normalized 2-player games with $\epsilon = .5$ [DMP06], .39 [DMP07], .37 [BBM07]; the best known ϵ at the time of writing is .34 [TS07].

Our knowledge for *multiplayer games* is also quite limited. In [DP08a], we show that an ϵ-Nash equilibrium of a normalized normal-form game with two strategies per player can be computed in time $n^{O(\log\log n + \log\frac{1}{\epsilon})}$, where n is the size of the game. For graphical games of constant degree, where our hardness result from Chapter 4 comes in the picture, a similar algorithm is unlikely, since it would imply that PPAD has quasi-polynomial-time algorithms. [2] On the positive side, Elkind, Goldberg and Goldberg show that a Nash equilibrium of graphical games with maximum degree 2 and 2 strategies per player can be computed in polynomial time [EGG06]. And what is known about larger degrees? In [DP06], we describe a polynomial-time approximation scheme for normalized graphical games with a constant number of strategies per player, bounded degree, and treewidth which is at most logarithmic in the number of players. Whether this result can be extended to graphical games with super-logarithmic treewidth remains an important open problem.

Since our knowledge for general games is limited, it is natural to ask the following.

- *Are there special classes of games for which approximate equilibria can be computed efficiently?*

Recall that two-player zero-sum games are solvable exactly in polynomial time by Linear Programming [Neu28, Dan63, Kha79]. Kannan and Theobald extend this tractability result by providing a polynomial-time approximation scheme for a generalization of two-player zero-sum games, called *low-rank games* [KT07]. These are games in which the sum of the players' payoff matrices[3] is a matrix of fixed rank.

[2] Recall that, as noted before, finding an ϵ-Nash equilibrium of bounded degree graphical games remains PPAD-complete for values of ϵ scaling inverse polynomially with the number of players.

[3] In two-player games, the payoffs of the players can be described by specifying two $n \times n$ matrices R and C, where n is the number of strategies of the players, so that R_{ij} and C_{ij} is respectively the

In [DP08b], we observe that a PTAS exists for another class of two-player games, called *bounded-norm games*, in which every player's payoff matrix is the sum of a constant matrix and a matrix with bounded infinity norm. These games have been shown to be PPAD-complete [CDT06b]. Hence, our tractability result exhibits a rare class of games which are PPAD-complete to solve exactly, yet a polynomial-time approximation scheme exists for solving them approximately.

In view of these positive results for special classes of two-player games, the following question arises.

- *Is there a polynomial-time approximation scheme for general two-player games?*

It is well-known that, if a two-player game has a Nash equilibrium in which both players' strategies have support of some fixed size, then that equilibrium can be recovered in polynomial time. Indeed, all we need to do is to perform an exhaustive search over all possible supports for the two players. For the right choice of supports, the Nash equilibrium can be found by solving a linear program. However, this straightforward approach does not extend beyond fixed size supports, since in this case the number of possible supports becomes super-polynomial.

- *Is it then the case that supports of size linear in the number of strategies are hard?*

Surprisingly, we show that this not always the case [DP08b]: If a two-player game has a Nash equilibrium in which both players' strategies spread non-trivially (that is, with significant probability mass) over a linear-size subset of the strategies, then an ϵ-Nash equilibrium can be recovered in randomized polynomial time, for any ϵ. Observe that the PPAD-hard instances of two-player games constructed in Section 4.2 only have equilibria of (non-trivial) linear support. Hence, our positive result for linear supports is another case of a problem which is PPAD-complete to

payoff of the first and the second player, if the first player chooses her i-th strategy and the second player chooses her j-th strategy.

solve exactly, yet a randomized polynomial-time approximation scheme exists for solving it approximately. It also brings about the following question.

- *If neither fixed nor linear, what support sizes are hard?*

The following discussion seems to suggest that *logarithmic* size supports are hard. It turns out that our PTAS for both the case of linear size support and the class of bounded-norm games, discussed previously, is of a very special kind, called *oblivious*. This means that it looks at a fixed set of pairs of mixed strategies, by sampling a distribution over that set, and uses the input game only to determine whether the sampled pair of mixed strategies constitutes an approximate Nash equilibrium. The guarantee in our algorithms is that an approximate Nash equilibrium is sampled with inverse polynomial probability, so that only a polynomial number of samples is needed in expectation.

We show, however, that an oblivious PTAS does not exist for general two-player games [DP08b]. And here is how logarithmic support comes into play. In our proof, we define a family of 2-player n-strategy games, indexed by all subsets of strategies of about logarithmic size, with the following property: The game indexed by a subset S satisfies that, in any ϵ-Nash equilibrium, the mixed strategy of one of the players is within total variation distance $O(\epsilon)$ from the uniform distribution over S. Since there are $n^{\Theta(\log n)}$ subsets of size $\log n$, it is not hard to deduce that any oblivious algorithm should have expected running time $n^{\Omega_\epsilon(\log n)}$, that is super-polynomial. Incidentally, note that the (also oblivious) algorithm of Lipton, Markakis and Mehta [LMM03] runs in time $n^{\Theta\left(\log n/\epsilon^2\right)}$, and it works by exhaustively searching over all multisets of strategies of size $\Theta(\log n/\epsilon^2)$.

It is natural to conjecture that an important step towards obtaining a polynomial-time approximation scheme for two-player games is to understand how an approximate Nash equilibrium can be computed in the presence of an exact Nash equilibrium of logarithmic support.

5.2 Anonymous Games

In the rest of this chapter, we consider algorithms for computing approximate equilibria in a very broad and important class of games, called *anonymous games*. These are games in which the players' payoff functions, although potentially different, do not differentiate among the identities of the other players. That is, each player's payoff depends on the strategy that she chooses and only *the number* of the other players choosing each of the available strategies. An immediate example is traffic: The delay incurred by a driver depends on the number of cars on her route, but not on the identities of the drivers. Another example arises in certain auction settings where the utility of a bidder is affected by the distribution of the other bids, but not on the identities of the other bidders. In fact, many problems of interest for algorithmic game theory, such as congestion games, participation games, voting games, and certain markets and auctions, are anonymous. The reader is referred to [Mil96, Blo99, Blo05, Kal05] for recent work on the subject by economists.

Note that anonymous games are much more general than symmetric games, in which all players are identical. In fact, any normal-form game can be represented by an anonymous game as follows. Two-player games are obviously anonymous for trivial reasons. To encode a multi-player non-anonymous game into an anonymous game, we can give to each player the option of choosing a strategy belonging to any player of the original game, but, at the same time, punish a player who chooses a strategy belonging to another player. Observe that this encoding incurs only a polynomial blowup in description complexity if the starting game has a constant number of players. Hence, all hardness results from the previous chapters apply to this case.

We are going to focus instead on anonymous games with many players and a few strategies per player. Observe that, if n is the number of players and k the number of strategies, only $O(n^k)$ numbers are needed to specify the game. Hence, anonymous games are a rare case of multiplayer games that have a polynomially succinct representation — as long as the number k of strategies is fixed. Our main result is a polynomial-time approximation scheme for such games.

Our PTAS extends to several generalizations of anonymous games, for example the case in which there are a few *types* of players, and the utilities depend on how many players *of each type* play each strategy; and to the case in which we have *extended families* (disjoint graphical games of constant degree and with logarithmically many players, each with a utility depending in arbitrary (possibly non-anonymous) ways on their neighbors in the graph, in addition to their anonymous —possibly typed— interest on everybody else). This generalizations are discussed in Section 5.8. Observe that, if we allowed larger extended families, we would be able to embed in the game graphical games with super-logarithmic size, for which the intractability result of the previous chapters comes into play.

Let us conclude our introduction to anonymous games with a discussion resonating the introduction to this dissertation in Chapter 1. Algorithmic Game Theory aspires to understand the Internet and the markets it encompasses and creates, hence the study of *multiplayer* games is of central importance. We believe that our PTAS is a positive algorithmic result spanning a vast expanse in this space. Because of the tremendous analytical difficulties detailed in Sections 5.5 through 5.7, our algorithm is not practical (as we shall see, the number of strategies and the accuracy appear in the exponent of the running time). It could be, of course, the precursor of more practical algorithms; in fact, we discuss a rather efficient algorithm for the case of two strategies in Section 5.9. But, more importantly, our algorithm should be seen as compelling computational evidence that there are very extensive and important classes of common games which are free of the negative implications of our complexity result from the previous chapters.

The structure of the remaining of this chapter is the following: In Section 5.3, we define anonymous games formally and introduce some useful notation. In Section 5.4, we state our main result and discuss our proof techniques. In Section 5.5, we state our main technical lemma and show how it implies the PTAS, and, in Section 5.6, we discuss its proof, which we give in Section 5.7. In Section 5.8, we discuss extensions of our PTAS to broader classes of games, and, in Section 5.9, we discuss more efficient PTAS's.

5.3 Definitions and Notation

A (normalized) *anonymous game* is a triple $G = (n, k, \{u_i^p\})$ where $[n] = \{1, \dots, n\}$, $n \geq 2$, is a set of players, $[k] = \{1, \dots, k\}$, $k \geq 2$, is a set of strategies, and u_i^p with $p \in [n]$ and $i \in [k]$ is the utility of player p when she plays strategy i, a function mapping the set of partitions $\Pi_{n-1}^k = \{(x_1, \dots, x_k) : x_i \in \mathbb{N}_0 \text{ for all } i \in [k], \sum_{i=1}^k x_i = n-1\}$ to the interval $[0, 1]$. [4] This means that the payoff of each player depends on her own strategy and only the number of the other players choosing each of the k strategies. Let us denote by Δ_{n-1}^k the convex hull of the set Π_{n-1}^k. That is, $\Delta_{n-1}^k = \{(x_1, \dots, x_k) : x_i \geq 0 \text{ for all } i \in [k], \sum_{i=1}^k x_i = n-1\}$.

A *mixed strategy profile* is a set of n distributions $\{\delta_p \in \Delta^k\}_{p \in [n]}$, where by Δ^k we denote the $(k-1)$-dimensional simplex, or, equivalently, the set of distributions over $[k]$. In this notation, a mixed strategy profile is an ϵ-*Nash equilibrium* if, for all $p \in [n]$ and $j, j' \in [k]$,

$$E_{\delta_1, \dots, \delta_n} u_j^p(x) > E_{\delta_1, \dots, \delta_n} u_{j'}^p(x) + \epsilon \Rightarrow \delta_p(j') = 0,$$

[4] As we noted in Section 5.1, the literature on Nash approximation studies normalized games so that the approximation error is additive.

where x is drawn from Π^k_{n-1} by drawing $n-1$ random samples from $[k]$ independently according to the distributions $\delta_q, q \neq p$, and forming the induced partition. Notice the similarity to (2.2) in Chapter 2.

Similarly, a mixed strategy profile is an *ϵ-approximate Nash equilibrium* if, for all $p \in [n]$ and $j \in [k]$, $E_{\delta_1,\ldots,\delta_n} u^p_i(x) + \epsilon \geq E_{\delta_1,\ldots,\delta_n} u^p_j(x)$, where i is drawn from $[k]$ according to δ_p and x is drawn from Π^k_{n-1} as above, by drawing $n-1$ random samples from $[k]$ independently according to the distributions $\delta_q, q \neq p$, and forming the induced partition.

Our working assumptions are that n is large and k is fixed; notice that, in this case, anonymous games are *succinctly representable* [PR05], in the sense that their representation requires specifying $O(n^k)$ numbers, as opposed to the nk^n numbers required for general games. Arguably, succinct games are the only multiplayer games that are computationally meaningful; see [PR05] for an extensive discussion of this point.

5.4 A Polynomial-Time Approximation Scheme for Anonymous Games

Our main result is a PTAS for anonymous games with a few strategies, namely

Theorem 5.1. *There is a PTAS for the mixed Nash equilibrium problem for normalized anonymous games with a constant number of strategies.*

We provide the proof of the theorem in the next section, where we also describe the basic technical lemma needed for the proof. Let us give here instead some intuition about our proof techniques. The basic idea of our algorithm is extremely simple and intuitive: Instead of performing the search for an approximate Nash equilibrium over the full set of mixed strategy profiles, we restrict our attention to mixed strategies assigning to each strategy in their support probability mass which is an integer multiple of $\frac{1}{z}$, where z is a large enough natural number. We call this process *discretization*. Searching the space of discretized mixed strategy profiles can be done efficiently with dynamic programming. Indeed, there are less than $(z+1)^{k-1}$ discretized mixed strategies available to each player, so at most $n^{(z+1)^{k-1}-1}$ partitions of the number n of players into these discretized mixed strategies. And checking if there is an approximate Nash equilibrium consistent with such a partition can be done efficiently using a max-flow argument (see details in the proof of Theorem 5.1 given in Section 5.5).

The challenge, however, lies somewhere else: We need to establish that any mixed Nash equilibrium of the original game is close to a discretized mixed strategy profile. And this requires the following non-trivial approximation lemma for multinomial distributions: The distribution of the sum of n independent random unit vectors with values ranging over $\{e_1, \dots, e_k\}$, where e_i is the unit vector along dimension i of the k-dimensional Euclidean space, can be approximated by the distribution of the sum of another set of independent unit vectors whose probabilities of obtaining each value are multiples of $\frac{1}{z}$, and so that the variational distance of the two distributions depends only on z (in fact, a decreasing function of z) and the dimension k, but not on the number of vectors n. In our setting, the original random vectors correspond to the strategies of the players in a Nash equilibrium, and the discretized ones to the discretized mixed strategy profile. The total variation distance bounds the approximation error incurred by replacing the Nash equilibrium with the discretized mixed strategy profile.

The approximation lemma needed in our proof can be interpreted as constructing a surprisingly sparse cover of the set of multinomial-sum distributions under the total variation distance. Covers have been considered extensively in the literature of approximation algorithms, but we know of no non-trivial result working in the set of multinomial-sum distributions or producing a cover of the required sparsity to achieve a polynomial-time approximation scheme for the Nash equilibrium in anonymous games. In the next section, we state the precise approximation result that we need and show how it can be used to derive a PTAS for anonymous games with a constant number of strategies. In Section 5.6, we discuss the challenges in establishing this result, and the full proof is given in Section 5.7.

5.5 An Approximation Theorem for Multinomial Distributions

Before stating our result, let us define the *total variation distance* between two distributions $\mathbb{P}$ and $\mathbb{Q}$ over a finite set $\mathcal{A}$ as

$$||\mathbb{P} - \mathbb{Q}||_{TV} = \frac{1}{2}\sum_{\alpha \in \mathcal{A}} |\mathbb{P}(\alpha) - \mathbb{Q}(\alpha)|.$$

Similarly, if X and Y are two random variables ranging over a finite set, their total variation distance, denoted

$$||X \; - \; Y||_{TV},$$

is defined to be the total variation distance between their distributions. Our approximation result is the following.

Theorem 5.2. *Let $\{p_i \in \Delta^k\}_{i\in[n]}$, and let $\{\mathcal{X}_i \in \mathbb{R}^k\}_{i\in[n]}$ be a set of independent k-dimensional random unit vectors such that, for all $i \in [n]$, $\ell \in [k]$, $\Pr[\mathcal{X}_i = e_\ell] = p_{i,\ell}$, where e_ℓ is the unit vector along dimension ℓ; also, let $z > 0$ be an integer. Then there exists another set of probability vectors $\{\widehat{p}_i \in \Delta^k\}_{i\in[n]}$ such that*

1. $|\widehat{p}_{i,\ell} - p_{i,\ell}| = O\left(\frac{1}{z}\right)$, *for all* $i \in [n], \ell \in [k]$;

2. $\widehat{p}_{i,\ell}$ *is an integer multiple of* $\frac{1}{2^k}\frac{1}{z}$, *for all* $i \in [n], \ell \in [k]$;

3. *if* $p_{i,\ell} = 0$*, then* $\widehat{p}_{i,\ell} = 0$*, for all* $i \in [n], \ell \in [k]$*;*

4. *if* $\{\widehat{\mathcal{X}}_i \in \mathbb{R}^k\}_{i\in[n]}$ *is a set of independent random unit vectors such that* $\Pr[\widehat{\mathcal{X}}_i = e_\ell] = \widehat{p}_{i,\ell}$*, for all* $i \in [n]$*,* $\ell \in [k]$*, then*

$$\left|\left|\sum_i \mathcal{X}_i - \sum_i \widehat{\mathcal{X}}_i\right|\right|_{TV} = O\left(f(k)\frac{\log z}{z^{1/5}}\right) \tag{5.1}$$

and, moreover, for all $j \in [n]$*,*

$$\left|\left|\sum_{i\neq j} \mathcal{X}_i - \sum_{i\neq j} \widehat{\mathcal{X}}_i\right|\right|_{TV} = O\left(f(k)\frac{\log z}{z^{1/5}}\right), \tag{5.2}$$

where $f(k)$ *is an exponential function of* k *estimated in the proof.*

In other words, there is a way to quantize any set of n independent random vectors into another set of n independent random vectors, whose probabilities of obtaining each value are integer multiples of $\epsilon \in [0,1]$, so that the total variation distance between the distribution of the sum of the vectors before and after the quantization is bounded by $O(f(k)2^{k/6}\epsilon^{1/6})$. The important, and perhaps surprising, property of this bound is the lack of dependence on the number n of random vectors. From this, the proof of Theorem 5.1 follows.

Proof of Theorem 5.1: Consider a mixed Nash equilibrium $(p_1, \ldots, p_n)$ of the game. We claim that the mixed strategy profile $(\widehat{p}_1, \ldots, \widehat{p}_n)$ specified by Theorem 5.2 constitutes a $O(f(k)z^{-\frac{1}{6}})$-Nash equilibrium. Indeed, for every player $i \in [n]$ and every pure strategy $m \in [k]$ for that player, let us track down the change in the expected utility of the player for playing strategy m when the distribution over Π^k_{n-1} defined by the $\{p_j\}_{j\neq i}$ is replaced by the distribution defined by the $\{\widehat{p}_j\}_{j\neq i}$. It is not hard to see that the absolute change is bounded by the total variation distance between the distributions of the random vectors $\sum_{j\neq i} \mathcal{X}_j$ and $\sum_{j\neq i} \widehat{\mathcal{X}}_j$, where $\{\mathcal{X}_j\}_{j\neq i}$

are independent random vectors distributed according to the distributions $\{p_j\}_{j \neq i}$ and, similarly, $\{\widehat{\mathcal{X}}_j\}_{j \neq i}$ are independent random vectors distributed according to the distributions $\{\widehat{p}_j\}_{j \neq i}$. [5] Hence, by Theorem 5.2, the change in the utility of the player is at most $O(f(k)z^{-\frac{1}{6}})$, which implies that the $\widehat{p}_i$'s constitute an $O(f(k)z^{-\frac{1}{6}})$-Nash equilibrium of the game. If we take $z = \left(\frac{f(k)}{\epsilon}\right)^6$, this is a δ-Nash equilibrium, for $\delta = O(\epsilon)$.

From the previous discussion it follows that there exists a mixed strategy profile $\{\widehat{p}_i\}_i$ which is of the very special kind described by Property 2 in the statement of Theorem 5.2 and constitutes a δ-Nash equilibrium of the given game, if we choose $z = \left(\frac{f(k)}{\epsilon}\right)^6$. The problem is, of course, that we do not know such a mixed strategy profile and, moreover, we cannot afford to do exhaustive search over all mixed strategy profiles satisfying Property 2, since there is an exponential number of those. We do instead the following search which is guaranteed to find a δ-Nash equilibrium.

Notice first that there are at most $(2^k z)^k = 2^{k^2}\left(\frac{f(k)}{\epsilon}\right)^{6k} =: K$ "quantized" mixed strategies with each probability being a multiple of $\frac{1}{2^k}\frac{1}{z}$, $z = \left(\frac{f(k)}{\epsilon}\right)^6$. Let $\mathcal{K}$ be the set of such quantized mixed strategies. We start our algorithm by guessing the partition of the number n of players into quantized mixed strategies; let $\theta = \{\theta_\sigma\}_{\sigma \in \mathcal{K}}$ be the partition, where θ_σ represents the number of players choosing the discretized mixed strategy $\sigma \in \mathcal{K}$. Now we only need to determine if there exists an assignment of mixed strategies to the players in $[n]$, with θ_σ of them playing mixed strategy $\sigma \in \mathcal{K}$, so that the corresponding mixed strategy profile is a δ-Nash equilibrium. To answer this question it is enough to solve the following *max-flow* problem. Let us consider the bipartite graph $([n], \mathcal{K}, E)$ with edge set E defined as follows: $(i, \sigma) \in E$, for $i \in [n]$ and $\sigma \in \mathcal{K}$, if $\theta_\sigma > 0$ and σ is a δ-best response for player i, if the partition of the other players into the mixed strategies in $\mathcal{K}$ is the partition θ, with

[5] The proof of this bound is similar to the derivation of the bound (3.14) in the proof of Lemma 3.32, using also that the game is anonymous and normalized, i.e., all utilities lie in $[0, 1]$.

one unit subtracted from θ_σ. [6] Note that to define E expected payoff computations are required. By straightforward dynamic programming, the expected utility of player i for playing pure strategy $s \in [k]$ given the mixed strategies of the other players can be computed with $O(kn^k)$ operations on numbers with at most $b(n,z,k) := \lceil 1 + n(k + \log_2 z) + \log_2(1/u_{\min}) \rceil$ bits, where $u_{\min}$ is the smallest non-zero payoff value of the game. [7] To conclude the construction of the max-flow instance we add a source node u connected to all the left hand side nodes and a sink node v connected to all the right hand side nodes. We set the capacity of the edge (σ, v) equal to θ_σ, for all $\sigma \in \mathcal{K}$, and the capacity of all other edges equal to 1. If the max-flow from u to v has value n then there is a way to assign discretized mixed strategies to the players so that θ_σ of them play mixed strategy $\sigma \in \mathcal{K}$ and the resulting mixed strategy profile is a δ-Nash equilibrium (details omitted). There are at most $(n+1)^{K-1}$ possible guesses for θ; hence, the search takes overall time

$$O\left((nKk^2n^kb(n,z,k) + p(n+K+2)) \cdot (n+1)^{K-1}\right),$$

where $p(n+K+2)$ is the time needed to find an integral maximum flow in a graph with $n+K+2$ nodes and edge-weights encoded with at most $\lceil \log_2 n \rceil$ bits. Hence, the overall time is

$$n^{O\left(2^{k^2}\left(\frac{f(k)}{\epsilon}\right)^{6k}\right)} \cdot \log_2(1/u_{\min}).$$

□

[6]For our discussion, a mixed strategy σ of player i is a δ-*best response* to a set of mixed strategies for the other players iff the expected payoff of player i for playing any pure strategy s in the support of σ is no more than δ worse than her expected payoff for playing any pure strategy s'.

[7]To compute a bound on the number of bits required for the expected utility computations, note that the expected utility is positive, cannot exceed 1, and its smallest possible non-zero value is at least $(\frac{1}{2^k}\frac{1}{z})^n u_{\min}$, since the mixed strategies of all players are from the set $\mathcal{K}$.

5.6 Discussion of Proof Techniques

Observe first that, from a technical perspective, the $k = 2$ case of Theorem 5.2 is inherently different than the $k > 2$ case. Indeed, when $k = 2$, knowledge of the number of players who selected their first strategy determines the whole partition of the number of players into strategies; therefore, in this case the probabilistic experiment is in some sense *one-dimensional*. On the other hand, when $k > 2$, knowledge of the number of "balls in a bin", that is, the number of players who selected a particular strategy, does not provide full information about the number of balls in the other bins. This complication would be quite benign if the vectors $\mathcal{X}_i$ were identically distributed, since in this case the number of balls in a bin would at least characterize precisely the probability distribution of the number of balls in the other bins (as a multinomial distribution with one bin less and the bin-probabilities appropriately renormalized). But, in our case, the vectors $\mathcal{X}_i$ are not identically distributed. Hence, already for $k = 3$ the problem is fundamentally different than the $k = 2$ case.

Indeed, it turns out that obtaining the result for the $k = 2$ case is easier. Here is the intuition: If the expectation of every $\mathcal{X}_i$ at the first bin was small, their sum would be distributed like a Poisson distribution (marginally at that bin); if the expectation of every $\mathcal{X}_i$ was large, the sum would be distributed like a (discretized) Normal distribution. [8] So, to establish the result we can do the following (see [DP07] for details): First, we cluster the $\mathcal{X}_i$'s into those with small and those with large expectation at the first bin, and then we discretize the $\mathcal{X}_i$'s separately in the two clusters in such a way that the sum of their expectations (within each cluster) is preserved to within the discretization accuracy. To show the closeness in total variation distance between the sum of the $\mathcal{X}_i$'s before and after the discretization, we compare instead

[8] Comparing, in terms of variational distance, a sum of independent Bernoulli random variables to a Poisson or a Normal distribution is an important problem in probability theory. The approximations we use are obtained by applications of *Stein's method* [BC05, BHJ92, R07].

the Poisson or Normal distributions (depending on the cluster) which approximate the sum of the $\mathcal{X}_i$'s: For the "small cluster", we compare the Poisson distributions approximating the sum of the $\mathcal{X}_i$'s before and after the discretization. For the "large cluster", we compare the Normals approximating the sum of the $\mathcal{X}_i$'s before and after the discretization.

One would imagine that a similar technique, i.e., approximating by a multidimensional Poisson or Normal distribution, would work for the $k > 2$ case. Comparing a sum of multinomial random variables to a multidimensional Poisson or Normal distribution is a little harder in many dimensions (see the discussion in [Bar05]), but almost optimal bounds *are* known for both the multidimensional Poisson [Bar05, Roo98] and the multidimensional Normal [Bha75, G9̈1] approximations. Nevertheless, these results by themselves are not sufficient for our setting: Approximating by a multidimensional Normal performs very poorly at the coordinates where the vectors have small expectations, and approximating by a multidimensional Poisson fails at the coordinates where the vectors have large expectations. And in our case, it could very well be that the sum of the $\mathcal{X}_i$'s is distributed like a multidimensional Poisson distribution in a subset of the coordinates and like a multidimensional Normal in the complement (those coordinates where the $\mathcal{X}_i$'s have respectively small or large expectations). What we really need, instead, is a multidimensional approximation result that combines the multidimensional Poisson and Normal approximations in the same picture; and such a result is not known.

Our approach instead is very indirect. We define an alternative way of sampling the vectors $\mathcal{X}_i$ which consists of performing a random walk on a binary decision tree and performing a probabilistic choice between two strategies at the leaves of the tree (Sections 5.7.1 and 5.7.2). The random vectors are then clustered so that, within a cluster, all vectors share the same decision tree (Section 5.7.3), and the rounding, performed separately for every cluster, consists of discretizing the probabilities for the probabilistic experiments at the leaves of the tree (Section 5.7.4). The rounding is done in such a way that, if all vectors $\mathcal{X}_i$ were to end up at the same leaf after walking on the decision tree, then the one-dimensional result described above would apply for the (binary) probabilistic choice that the vectors are facing at the leaf. However, the random walks will not all end up at the same leaf with high probability. To remedy this, we define a coupling between the random walks of the original and the discretized vectors for which, in the typical case, the probabilistic experiments that the original vectors are running at every leaf of the tree are very "similar" to the experiments that the discretized vectors are running. That is, our coupling guarantees that, with high probability over the random walks, the total variation distance between the choices (as random variables) that are to be made by the original vectors at every leaf of the decision tree and the choices (again as random variables) that are to be made by the discretized vectors is very small. The coupling of the random walks is defined in Section 5.7.5, and a quantification of the similarity of the leaf experiments under this coupling is given in Section 5.7.6.

For a discussion about why naive approaches such as *rounding to the closest discrete distribution* or *randomized rounding* do not appear useful, even for the $k = 2$ case, see Section 3.1 of [DP07].

5.7 Proof of the Multinomial Approximation Theorem

5.7.1 The Trickle-Down Process

Consider the mixed strategy p_i of player i. The crux of our argument is an alternative way to sample from this distribution, based on the so-called *trickle-down process*, defined next.

TDP — Trickle-Down Process

Input: (S, p), where $S = \{i_1, \ldots, i_m\} \subseteq [k]$ is a set of strategies and p a probability distribution $p(i_j) > 0 : j = 1, \ldots, m$. We assume that the elements of S are ordered $i_1, \ldots, i_m$ in such a way that (a) $p(i_2)$ is the largest of the $p(i_j)$'s and (b) for $2 \neq j < j' \neq 2$, $p(i_j) \leq p(i_{j'})$. That is, the largest probability is second, and, other than that, the probabilities are sorted in non-decreasing order (ties broken lexicographically).

if $|S| \leq 2$ stop; **else** apply the *partition and double operation*:

1. let $\ell^* < m$ be the (unique) index such that $\sum_{\ell<\ell^*} p(i_\ell) \leq \frac{1}{2}$ and $\sum_{\ell>\ell^*} p(i_\ell) < \frac{1}{2}$;
2. Define the two sets $S_L = \{i_\ell : \ell \leq \ell^*\}$ and $S_R = \{i_\ell : \ell \geq \ell^*\}$
3. Define the probability distribution p_L such that, for all $\ell < \ell^*$, $p_L(i_\ell) = 2p(i_\ell)$. Also, let $t := 1 - \sum_{\ell=1}^{\ell^*-1} p_L(i_\ell)$; if $t = 0$, then remove ℓ^* from S_L, otherwise set $p_L(i_{\ell^*}) = t$. Similarly, define the probability distribution p_R such that $p_R(i_\ell) = 2p(i_\ell)$, for all $\ell > \ell^*$ and $p_R(i_{\ell^*}) = 1 - \sum_{\ell^*+1}^{m} p_R(i_\ell)$. Notice that, because of the way we have ordered the strategies in S, i_{ℓ^*} is neither the first nor the last element of S in our ordering, and hence $2 \leq |S_L|, |S_R| < |S|$.
4. call **TDP**(S_L, p_L); call **TDP**(S_R, p_R);

That is, TDP splits the support of the mixed strategy of a player into a tree of finer and finer sets of strategies, with all leaves having just two strategies. At each level the two sets in which the set of strategies is split overlap in at most one strategy (whose probability mass is divided between its two copies). The two sets then have probabilities adding up to $1/2$, but then the probabilities are multiplied by 2, so that each node of the tree represents a distribution.

5.7.2 An Alternative Sampling of the Random Vectors

Let p_i be the mixed strategy of player i, and $\mathcal{S}_i$ be its support. [9] The execution of $\mathbf{TDP}(\mathcal{S}_i, p_i)$ defines a rooted binary tree T_i with node set V_i and set of leaves ∂T_i. Each node $v \in V_i$ is identified with a pair $(S_v, p_{i,v})$, where $S_v \subseteq [k]$ is a set of strategies and $p_{i,v}$ is a distribution over S_v. Based on this tree, we define the following alternative way to sample $\mathcal{X}_i$:

SAMPLING $\mathcal{X}_i$

1. (*Stage 1*) Perform a random walk from the root of the tree T_i to the leaves, where, at every non-leaf node, the left or right child is chosen with probability 1/2; let $\Phi_i \in \partial T_i$ be the (random) leaf chosen by the random walk;

2. (*Stage 2*) Let (S, p) be the label assigned to the leaf Φ_i, where $S = \{\ell_1, \ell_2\}$; set $\mathcal{X}_i = e_{\ell_1}$, with probability $p(\ell_1)$, and $\mathcal{X}_i = e_{\ell_2}$, with probability $p(\ell_2)$.

The following lemma, whose straightforward proof we omit, states that this is indeed an alternative sampling of the mixed strategy of player i.

Lemma 5.3. *For all $i \in [n]$, the process* SAMPLING $\mathcal{X}_i$ *outputs $\mathcal{X}_i = e_\ell$ with probability $p_{i,\ell}$, for all $\ell \in [k]$.*

5.7.3 Clustering the Random Vectors

We use the process **TDP** to cluster the random vectors of the set $\{\mathcal{X}_i\}_{i\in[n]}$, by defining a cell for every possible tree structure. More formally, for some $\alpha > 0$ to be determined later in the proof,

Definition 5.4 (Cell Definition)**.** Two vectors $\mathcal{X}_i$ and $\mathcal{X}_j$ belong to the same cell if

[9] In this section and the following two sections we assume that $|\mathcal{S}_i| > 1$; if not, we set $\widehat{p}_i = p_i$, and all claims we make in Sections 5.7.5 and 5.7.6 are trivially satisfied.

- there exists a tree isomorphism $f_{i,j} : V_i \to V_j$ between the trees T_i and T_j such that, for all $u \in V_i$, $v \in V_j$, if $f_{i,j}(u) = v$, then $S_u = S_v$, and in fact the elements of S_u and S_v are ordered the same way by $p_{i,u}$ and $p_{j,v}$.

- if $u \in \partial T_i$, $v = f_{i,j}(u) \in \partial T_j$, and $\ell^* \in S_u = S_v$ is the strategy with the smallest probability mass for both $p_{i,u}$ and $p_{j,v}$, then either $p_{i,u}(\ell^*), p_{j,v}(\ell^*) \leq \frac{\lfloor z^\alpha \rfloor}{z}$ or $p_{i,u}(\ell^*), p_{j,v}(\ell^*) > \frac{\lfloor z^\alpha \rfloor}{z}$; the leaf is called *Type A leaf* in the first case, *Type B leaf* in the second case.

It is easy to see that the total number of cells is bounded by a function of k only, call it $g(k)$. The following claim provides an estimate of $g(k)$.

Claim 5.5. *Any tree resulting from TDP has at most $k-1$ leaves, and the total number of cells is bounded by $g(k) = k^{k^2} 2^{k-1} 2^k k!$.*

5.7.4 Discretization within a Cell of the Clustering

Recall that our goal is to "discretize" the probabilities in the distribution of the $\mathcal{X}_i$'s. We will do this separately in every cell of our clustering. In particular, supposing that $\{\mathcal{X}_i\}_{i \in \mathcal{I}}$ is the set of vectors falling in a particular cell, for some index set $\mathcal{I}$, we will define a set of "discretized" vectors $\{\widehat{\mathcal{X}}_i\}_{i \in \mathcal{I}}$ in such a way that, for $h(k) = k2^k$, and for all $j \in \mathcal{I}$,

$$\left|\left|\sum_{i \in \mathcal{I}} \mathcal{X}_i - \sum_{i \in \mathcal{I}} \widehat{\mathcal{X}}_i\right|\right|_{TV} = O(h(k) \log z \cdot z^{-1/5}); \tag{5.3}$$

$$\left|\left|\sum_{i \in \mathcal{I} \setminus \{j\}} \mathcal{X}_i - \sum_{i \in \mathcal{I} \setminus \{j\}} \widehat{\mathcal{X}}_i\right|\right|_{TV} = O(h(k) \log z \cdot z^{-1/5}). \tag{5.4}$$

We establish these bounds in Section 5.7.5. Using the bound on the number of cells in Claim 5.5, an easy application of the coupling lemma implies the bounds shown in (5.1) and (5.2) for $f(k) := h(k) \cdot g(k)$, thus concluding the proof of Theorem 5.2.

We shall henceforth concentrate on a particular cell containing the vectors $\{\mathcal{X}_i\}_{i \in \mathcal{I}}$, for some $\mathcal{I} \subseteq [n]$. Since the trees $\{T_i\}_{i \in \mathcal{I}}$ are isomorphic, for notational convenience we shall denote all those trees by T. To define the vectors $\{\widehat{\mathcal{X}}_i\}_{i \in \mathcal{I}}$ we must provide, for all $i \in \mathcal{I}$, a distribution $\widehat{p}_i : [k] \to [0,1]$ such that $\Pr[\widehat{\mathcal{X}}_i = e_\ell] = \widehat{p}_i(\ell)$, for all $\ell \in [k]$. To do this, we assign to all $\{\widehat{\mathcal{X}}_i\}_{i \in \mathcal{I}}$ the tree T and then, for every leaf $v \in \partial T$ and $i \in \mathcal{I}$, define a distribution $\widehat{p}_{i,v}$ over the two-element ordered set S_v, by the Rounding process below. Then the distribution $\widehat{p}_i$ is implicitly defined as $\widehat{p}_i(\ell) = \sum_{v \in \partial T : \ell \in S_v} 2^{-depth_T(v)} \widehat{p}_{i,v}(\ell)$.

Rounding: for all $v \in \partial T$ with $S_v = \{\ell_1, \ell_2\}$, $\ell_1, \ell_2 \in [k]$ do the following

1. find a set of probabilities $\{p_{i,\ell_1}\}_{i \in \mathcal{I}}$ with the following properties
   - for all $i \in \mathcal{I}$, $|p_{i,\ell_1} - p_{i,v}(\ell_1)| \leq \frac{1}{z}$;
   - for all $i \in \mathcal{I}$, p_{i,ℓ_1} is an integer multiple of $\frac{1}{z}$;
   - $\left|\sum_{i \in \mathcal{I}} p_{i,\ell_1} - \sum_{i \in \mathcal{I}} p_{i,v}(\ell_1)\right| \leq \frac{1}{z}$;
2. for all $i \in \mathcal{I}$, set $\widehat{p}_{i,v}(\ell_1) := p_{i,\ell_1}$, $\widehat{p}_{i,v}(\ell_2) := 1 - p_{i,\ell_1}$;

Finding the set of probabilities required by Step 1 of the Rounding process is straightforward and the details are omitted (see [DP07], Section 3.3 for a way to do so). It is now easy to check that the set of probability vectors $\{\widehat{p}_i\}_{i \in \mathcal{I}}$ satisfies Properties 1, 2 and 3 of Theorem 5.2.

5.7.5 Coupling within a Cell of the Clustering

We are now coming to the main part of the proof: Showing that the variational distance between the original and the discretized distribution within a cell depends only on z and k. We will only argue that our discretization satisfies (5.3); the proof of (5.4) is identical.

Before proceeding let us introduce some notation. Specifically,

- let $\Phi_i \in \partial T$ be the leaf chosen by Stage 1 of the process SAMPLING $\mathcal{X}_i$ and $\hat{\Phi}_i \in \partial T$ the leaf chosen by Stage 1 of SAMPLING $\widehat{\mathcal{X}}_i$;
- let $\Phi = (\Phi_i)_{i \in \mathcal{I}}$ and let G denote the distribution of Φ; similarly, let $\hat{\Phi} = (\hat{\Phi}_i)_{i \in \mathcal{I}}$ and let $\widehat{G}$ denote the distribution of $\hat{\Phi}$.

Moreover, for all $v \in \partial T$, with $S_v = \{\ell_1, \ell_2\}$ and ordering (ℓ_1, ℓ_2),

- let $\mathcal{I}_v \subseteq \mathcal{I}$ be the (random) index set such that $i \in \mathcal{I}_v$ iff $i \in \mathcal{I} \wedge \Phi_i = v$ and, similarly, let $\widehat{\mathcal{I}}_v \subseteq \mathcal{I}$ be the (random) index set such that $i \in \widehat{\mathcal{I}}_v$ iff $i \in \mathcal{I} \wedge \hat{\Phi}_i = v$;
- let $\mathcal{J}_{v,1}, \mathcal{J}_{v,2} \subseteq \mathcal{I}_v$ be the (random) index sets such $i \in \mathcal{J}_{v,1}$ iff $i \in \mathcal{I}_v \wedge \mathcal{X}_i = e_{\ell_1}$ and $i \in \mathcal{J}_{v,2}$ iff $i \in \mathcal{I}_v \wedge \mathcal{X}_i = e_{\ell_2}$;
- let $T_{v,1} = |\mathcal{J}_{v,1}|$, $T_{v,2} = |\mathcal{J}_{v,2}|$ and let F_v denote the distribution of $T_{v,1}$;
- let $T := ((T_{v,1}, T_{v,2}))_{v \in \partial T}$ and let F denote the distribution of T;
- let $\widehat{\mathcal{J}}_{v,1}$, $\widehat{\mathcal{J}}_{v,2}$, $\widehat{T}_{v,1}$, $\widehat{T}_{v,2}$, $\widehat{T}$, $\widehat{F}_v$, $\widehat{F}$ be defined similarly.

The following is easy to see, so we postpone its proof to the appendix.

Claim 5.6. *For all $\theta \in (\partial T)^{\mathcal{I}}$, $G(\theta) = \widehat{G}(\theta)$.*

Since G and $\widehat{G}$ are the same distribution we will henceforth denote that distribution by G. The following lemma is sufficient to conclude the proof of Theorem 5.2.

Lemma 5.7. *There exists a value of α, used in the definition of the cells, such that, for all $v \in \partial T$,*

$$G\left(\theta : ||F_v(\cdot|\Phi = \theta) - \widehat{F}_v(\cdot|\hat{\Phi} = \theta)||_{TV} \leq O\left(\frac{2^k \log z}{z^{1/5}}\right)\right) \geq 1 - \frac{4}{z^{1/3}},$$

where $F_v(\cdot|\Phi)$ denotes the conditional probability distribution of $T_{v,1}$ given Φ and, similarly, $\widehat{F}_v(\cdot|\hat{\Phi})$ denotes the conditional probability distribution of $\widehat{T}_{v,1}$ given $\hat{\Phi}$.

Lemma 5.7 states roughly that, for all $v \in \partial T$, with probability at least $1 - \frac{4}{z^{1/3}}$ over the choices made by Stage 1 of processes $\{\textsc{Sampling } \mathcal{X}_i\}_{i \in \mathcal{I}}$ and $\{\textsc{Sampling } \widehat{\mathcal{X}}_i\}_{i \in \mathcal{I}}$ — assuming that these processes are coupled to make the same decisions in Stage 1 — the total variation distance between the conditional distribution of $T_{v,1}$ and $\widehat{T}_{v,1}$ is bounded by $O\left(\frac{2^k \log z}{z^{1/5}}\right)$.

To complete the proof, note first that Lemma 5.7 implies via a union bound that

$$G\left(\theta : \forall v \in \partial T, ||F_v(\cdot|\Phi = \theta) - \widehat{F}_v(\cdot|\hat{\Phi} = \theta)||_{TV} \leq O\left(\frac{2^k \log z}{z^{1/5}}\right)\right) \geq 1 - O(kz^{-1/3}), \tag{5.5}$$

since by Claim 5.5 the number of leaves is at most $k-1$. Now suppose that for a given value of $\theta \in (\partial T)^{\mathcal{I}}$ it holds that

$$\forall v \in \partial T, ||F_v(\cdot|\Phi = \theta) - \widehat{F}_v(\cdot|\hat{\Phi} = \theta)||_{TV} \leq O\left(\frac{2^k \log z}{z^{1/5}}\right). \tag{5.6}$$

Note that the variables $\{T_{v,1}\}_{v \in \partial T}$ are conditionally independent given Φ, and, similarly, the variables $\{\widehat{T}_{v,1}\}_{v \in \partial T}$ are conditionally independent given $\hat{\Phi}$. This by the coupling lemma, Claim 5.5 and (5.6) implies that

$$||F(\cdot|\Phi = \theta) - \widehat{F}(\cdot|\hat{\Phi} = \theta)||_{TV} \leq O\left(k\frac{2^k \log z}{z^{1/5}}\right),$$

where we also used the fact that, if $\Phi = \hat{\Phi} = \theta$, then $|\mathcal{I}_v| = |\widehat{\mathcal{I}}_v|$, for all $v \in \partial T$.

Hence, (5.5) implies that

$$G\left(\theta : ||F(\cdot|\Phi = \theta) - \widehat{F}(\cdot|\hat{\Phi} = \theta)||_{TV} \leq O\left(k\frac{2^k \log z}{z^{1/5}}\right)\right) \geq 1 - O(kz^{-1/3}). \tag{5.7}$$

All that remains is to shift the bound (5.7) to the unconditional space. The following lemma establishes this reduction. Its proof is postponed to the appendix.

Lemma 5.8. (5.7) *implies*

$$||F - \widehat{F}||_{TV} \leq O\left(k\frac{2^k \log z}{z^{1/5}}\right). \tag{5.8}$$

Note that (5.8) implies easily (5.3), which completes the proof of our main result.

5.7.6 Total Variation Distance within a Leaf

To conclude the proof of Theorem 5.2, it remains to show Lemma 5.7. Roughly speaking, the proof consists of showing that, with high probability over the random walks performed in Stage 1 of SAMPLING, the one-dimensional experiment occurring at a particular leaf v of the tree is similar in both the original and the discretized distribution. The similarity is quantified by Lemmas 5.12 and 5.13 for leaves of type A and B respectively. Then, Lemmas 5.9, 5.10 and 5.11 establish that, if the experiments are sufficiently similar, they can be coupled so that their outcomes agree with high probability.

More precisely, let $v \in \partial T$, $\mathcal{S}_v = \{\ell_1, \ell_2\}$, and suppose the ordering (ℓ_1, ℓ_2). Also, let us denote $\ell_v^* = \ell_1$ and define the following functions

- $\mu_v(\theta) := \sum_{i:\theta_i=v} p_{i,v}(\ell_v^*)$;
- $\widehat{\mu}_v(\hat{\theta}) := \sum_{i:\hat{\theta}_i=v} \widehat{p}_{i,v}(\ell_v^*)$.

Note that the random variable $\mu_v(\Phi)$ represents the total probability mass that is placed on the strategy ℓ_v^* after the Stage 1 of the SAMPLING process is completed for all vectors $\mathcal{X}_i$, $i \in \mathcal{I}$. Conditioned on the outcome of Stage 1 of SAMPLING for the vectors $\{\mathcal{X}_i\}_{i\in\mathcal{I}}$, $\mu_v(\Phi)$ is the expected number of the vectors from $\mathcal{I}_v$ that will select strategy ℓ_v^* in Stage 2 of SAMPLING. Similarly, conditioned on the outcome of Stage 1 of SAMPLING for the vectors $\{\widehat{\mathcal{X}}_i\}_{i\in\mathcal{I}}$, $\widehat{\mu}_v(\widehat{\Phi})$ is the expected number of the vectors from $\widehat{\mathcal{I}}_v$ that will select strategy ℓ_v^* in Stage 2 of SAMPLING.

Intuitively, if we can couple the choices made by the random vectors $\mathcal{X}_i$, $i \in \mathcal{I}$, in Stage 1 of SAMPLING with the choices made by the random vectors $\widehat{\mathcal{X}}_i$, $i \in \mathcal{I}$, in Stage 1 of SAMPLING in such a way that, with overwhelming probability, $\mu_v(\Phi)$ and $\widehat{\mu}_v(\widehat{\Phi})$ are close, then also the conditional distributions $F_v(\cdot|\Phi)$, $\widehat{F}_v(\cdot|\hat{\Phi})$ should be close in total variation distance. The goal of this section is to make this intuition rigorous. We do this in 2 steps by showing the following.

1. The choices made in Stage 1 of SAMPLING can be coupled so that the absolute difference $|\mu_v(\Phi) - \widehat{\mu}_v(\widehat{\Phi})|$ is small with high probability. (Lemmas 5.12 and 5.13.)

2. If the absolute difference $|\mu_v(\theta) - \widehat{\mu}_v(\widehat{\theta})|$ is sufficiently small, then so is the total variation distance $||F_v(\cdot|\Phi = \theta) - \widehat{F}_v(\cdot|\hat{\Phi} = \theta)||_{TV}$. (Lemmas 5.9, 5.10, and 5.11.)

We start with Step 2 of the above program. We use different arguments depending on whether v is a Type A or Type B leaf. Let $\partial T = Ł_A \sqcup Ł_B$, where $Ł_A$ is the set of type A leaves of the cell and $Ł_B$ the set of type B leaves of the cell. For some constant β to be decided later, we show the following lemmas.

Lemma 5.9. *For some $\theta \in (\partial T)^{\mathcal{I}}$ and $v \in Ł_A$ suppose that*

$$|\mu_v(\theta) - \mathcal{E}[\mu_v(\Phi)]| \leq z^{(\alpha-1)/2}\sqrt{\mathcal{E}[\mu_v(\Phi)] \log z} \tag{5.9}$$

$$\left|\widehat{\mu}_v(\theta) - \mathcal{E}[\widehat{\mu}_v(\hat{\Phi})]\right| \leq z^{(\alpha-1)/2}\sqrt{\mathcal{E}[\widehat{\mu}_v(\hat{\Phi})] \log z} \tag{5.10}$$

then

$$||F_v(\cdot|\Phi = \theta) - \widehat{F}_v(\cdot|\hat{\Phi} = \theta)||_{TV} \leq O\left(\frac{\sqrt{\log z}}{z^{(1-\alpha)/2}}\right).$$

Lemma 5.10. *For some $\theta \in (\partial T)^{\mathcal{I}}$ and $v \in Ł_B$ suppose that*

$$n_v(\theta) := |\{i : \theta_i = v\}| \geq z^{\beta}, \tag{5.11}$$

$$|\mu_v(\theta) - \widehat{\mu}_v(\theta)| \leq \frac{1}{z} + \frac{\sqrt{\log z}}{z}\sqrt{|\mathcal{I}|} \tag{5.12}$$

and

$$|n_v(\theta) - 2^{-depth_T(v)}|\mathcal{I}|| \leq \sqrt{3\log z}\sqrt{2^{-depth_T(v)}|\mathcal{I}|} \tag{5.13}$$

then

$$||F_v(\cdot|\Phi = \theta) - \widehat{F}_v(\cdot|\hat{\Phi} = \theta)||_{TV} \\ \leq O\left(\frac{2^{\frac{depth_T(v)}{2}}\sqrt{\log z}}{z^{\frac{1+\alpha}{2}}}\right) + O\left(\frac{2^{\frac{depth_T(v)}{2}}\log z}{z^{\frac{\alpha+\beta+1}{2}}}\right) + O(z^{-\alpha}) + O(z^{-(\frac{\alpha+\beta-1}{2})}).$$

Lemma 5.11. *For some $\theta \in (\partial T)^{\mathcal{I}}$ and $v \in Ł_B$ suppose that*

$$n_v(\theta) := |\{i : \theta_i = v\}| \leq z^{\beta} \tag{5.14}$$

then

$$||F_v(\cdot|\Phi = \theta) - \widehat{F}_v(\cdot|\hat{\Phi} = \theta)||_{TV} \leq O(z^{-(1-\beta)}).$$

The proof of Lemma 5.11 follows from a coupling argument similar to that used in the proof of Lemma 3.13 in [DP07] and is omitted. The proofs of Lemmas 5.9 and 5.10 can be found respectively in Sections A.1 and A.2 of the appendix.

Lemma 5.9 provides conditions which, if satisfied by some θ at a leaf of Type A, then the conditional distributions $F_v(\cdot|\Phi = \theta)$ and $\widehat{F}_v(\cdot|\hat{\Phi} = \theta)$ are close in total variation distance. Similarly, Lemmas 5.10 and 5.11 provide conditions for the leaves of Type B. The following lemmas state that these conditions are satisfied with high probability. Their proof is given in Section A.3 of the appendix.

Lemma 5.12. *Let* $v \in Ł_A$*. Then*

$$G\left(\begin{array}{l}\theta : |\mu_v(\theta) - \mathcal{E}[\mu_v(\Phi)]| \leq \frac{\sqrt{\log z}}{z^{(1-\alpha)/2}}\sqrt{\mathcal{E}[\mu_v(\Phi)]} \\ \quad \wedge \left|\widehat{\mu}_v(\theta) - \mathcal{E}[\widehat{\mu}_v(\hat{\Phi})]\right| \leq \frac{\sqrt{\log z}}{z^{(1-\alpha)/2}}\sqrt{\mathcal{E}[\widehat{\mu}_v(\hat{\Phi})]}\end{array}\right) \geq 1 - 4z^{-1/3}. \tag{5.15}$$

Lemma 5.13. *Let* $v \in Ł_B$*. Then*

$$G\left(\begin{array}{l}\theta : |\mu_v(\theta) - \widehat{\mu}_v(\theta)| \leq \frac{1+\sqrt{|\mathcal{I}|\log z}}{z} \\ \quad \wedge \quad |n_v(\theta) - 2^{-depth_T(v)}|\mathcal{I}|| \leq \sqrt{3\log z}\sqrt{2^{-depth_T(v)}|\mathcal{I}|}\end{array}\right) \geq 1 - \frac{4}{z^{1/2}}. \tag{5.16}$$

Setting $\alpha = \frac{3}{5}$ and $\beta = \frac{4}{5}$ and combining the above, we have that, regardless of whether $v \in Ł_A$ or $v \in Ł_B$,

$$G\left(\theta : ||F_v(\cdot|\Phi = \theta) - \widehat{F}_v(\cdot|\hat{\Phi} = \theta)||_{TV} \leq O\left(\frac{2^k \log z}{z^{1/5}}\right)\right) \geq 1 - \frac{4}{z^{1/3}},$$

where we used that $depth_T(v) \leq k$ as implied by Claim 5.5.

5.8 Extensions

Returning to our algorithm (Theorem 5.1), there are several directions in which it can be immediately generalized. To give an idea of the possibilities, let us define a *semi-anonymous game* to be a game in which

- the players are partitioned into a fixed number of *types;*
- there is another partition of the players into an arbitrary number of disjoint graphical games of size $O(\log n)$, where n is the total number of players, and bounded degree called *extended families*;

and the utility of each player depends on (a) his/her own strategy; (b) the overall number of other players of each type playing each strategy; and (c) it also depends, in an arbitrary way, on the strategy choices of neighboring nodes in his/her own extended family. The following result, which is only indicative of the applicability of our approach, can be shown by extending the discretization method via dynamic programming (details omitted):

Theorem 5.14. *There is a PTAS for semi-anonymous games with a fixed number of strategies.*

5.9 Towards Efficient Polynomial-Time Approximation Schemes

The polynomial-time approximation scheme presented in the previous sections computes an ϵ-Nash equilibrium of an anonymous game with n players and k strategies in time

$$O\left(n^{f(k,\epsilon)}\right),$$

where f is polynomial in $\frac{1}{\epsilon}$, but super-exponential in k. Hence, despite its theoretical efficiency for bounded ϵ and k, the algorithm is not efficient in practice. Even for the simpler case of 2 strategies per player, a more efficient implementation of our algorithm, given in [DP07], runs in time

$$O\left(n^{O(1/\epsilon^2)}\right),$$

which is still not practical.

So, are there more efficient algorithms for computing Nash equilibria in anonymous games? In recent work [Das08], we present an efficient polynomial-time approximation scheme for the case of two strategies per player, with running time

$$\text{poly}(n) \cdot (1/\epsilon)^{O(1/\epsilon^2)}.$$

This scales with the number of players as a polynomial of fixed degree, independent of ϵ. The improved running time is based on a better understanding of certain structural properties of approximate Nash equilibria. In particular, we show that, for any integer z, there exists an ϵ-approximate Nash equilibrium with $\epsilon = O(1/z)$, in which

(a) either, at most $z^3 = O((1/\epsilon)^3)$ players use randomized strategies, and their strategies are integer multiples of $1/z^2$; [10]

(b) or, all players who randomize choose the same mixed strategy which is an integer multiple of $\frac{1}{zn}$.

To derive the above characterization, we study mixed strategy profiles in the neighborhood of a Nash equilibrium. We establish that there always exists a mixed strategy profile in this neighborhood which is of one of the types (a) or (b) described above and, moreover, satisfies the Nash equilibrium conditions to within an additive ϵ, hence corresponding to an ϵ-approximate equilibrium. Given this structural result, an ϵ-approximate equilibrium can be found by dynamic programming.

We feel that a more sophisticated analysis could establish similar structural properties for the approximate equilibria of multi-strategy anonymous games, thus extending our efficient PTAS to anonymous games with any fixed number of strategies. Also, a more refined structural characterization of approximate Nash equilibria could give more efficient algorithms, even a fully polynomial-time approximation scheme.

[10] Note that, since every player has 2 strategies, a mixed strategy is just a number in $[0, 1]$.

Chapter 6

Conclusions and Open Problems

Motivated by the importance of Game Theory for the study of large systems of strategic agents, such as the Internet, as well as social and biological systems, we investigated whether Nash equilibria are efficiently computable. The significance of this question is the following: The concept of the Nash equilibrium is one of Game Theory's most important frameworks for behavior prediction. For its predictions then to be plausible it is crucial that it is efficiently computable; because, if it is not, it would not be reasonable to expect that a group of strategic agents would be able to discover it in every situation and behave as it prescribes.

Since by Nash's theorem a Nash equilibrium is guaranteed to exist, to characterize the complexity of finding an equilibrium, we turned to the complexity theory of total search problems in NP. For the case of two-player games, the Nash equilibrium problem was known to belong to PPAD, defined to be the class of total search problems whose totality is certified by a parity argument in directed graphs [Pap94b]. We extended this result to all games; however, because in multi-player games all equilibria may be irrational, we analyzed the problem of finding an ϵ-Nash equilibrium (for ϵ specified in the input), which we called Nash. Our main result was that Nash is PPAD-complete. Since finding Brouwer fixed points of continuous and piecewise-linear functions is also PPAD-complete [Pap94b], our result implies the polynomial-time equivalence of these problems.

The class PPAD is not known to be inside P. Hence, our result raises questions about the plausibility of the Nash equilibrium as a concept of behavior prediction. To make this critique more solid, it is important to answer the following question.

- *What is the true complexity of the class PPAD?*

Showing P≠PPAD would imply that P≠NP, since PPAD is a subset of the search problems in NP. Hence, a proof of this statement should involve major breakthroughs in complexity theory. Therefore, a less ambitious goal would be to provide conditional hardness results for the complexity of PPAD, possibly under cryptographic assumptions such as the hardness of factoring an integer. In view of Shor's algorithm for factoring [Sho97] a reduction in the opposite direction would imply an interesting (conditional) separation between deterministic and quantum polynomial time. In the same spirit, it is interesting to investigate the following.

- *What is the relation of PPAD to other classes at the boundary of P and NP and to problems in cryptography?*

Apart from the parity argument in directed graphs giving rise to the class PPAD, there are several other (non-constructive) arguments for showing existence in combinatorics. And, as discussed in Chapter 1, each of these arguments naturally defines a class of total search problems in NP. The ones considered by Papadimitriou in [Pap94b] include the *existence of sinks in DAGs* (which corresponds to the class PLS), the *pigeonhole principle* (which gives rise to the class PPP), and the *parity argument on undirected graphs* (which defines the class PPA). The complexity classes thus defined provide a rich framework for characterizing the computational hardness of total search problems in NP and may also be helpful for the systematic classification of cryptographic problems. After all, most cryptographic primitives, such as the prime factorization, the discrete logarithm, and the existence of short vectors in lattices, correspond to total search problems. Evidence of hardness for these problems could be of great significance for cryptography. For an extensive discussion of this point, see, e.g., the work of Ajtai [Ajt96], where also lattice problems are defined from non-constructive existence arguments in PPP; unfortunately, none of these arguments are known to be PPP-complete.

Our PPAD-completeness result for the Nash equilibrium problem is a worst-case result. But it could be the case that it does not apply for "situations arising in practice." To understand whether this is the case, we should investigate the following.

- *Are there broad and important classes of games for which Nash equilibria can be computed efficiently?*

- *Are there natural random ensembles of games for which the Nash equilibrium problem is tractable with high probability?*

As discussed in previous chapters, two-player zero-sum games are solvable exactly in polynomial time [Neu28, Dan63, Kha79], and these are essentially the only interesting games for which strong tractability results are known. For several other special classes of games, such as win-lose games and two-player symmetric games, intractability persists (see Section 4.3). An intriguing open problem at the boundary of intractability is the following.

- *What is the complexity of three-player symmetric games?*

In the realm of random ensembles, Bárány, Vempala and Vetta describe a simple distribution over two-player games that can be solved exactly in polynomial time with high probability [BVV07]. However, their ensemble — assuming that every utility value is independently and identically distributed with respect to the others — is rather special. On the other hand, finding a Nash equilibrium in the smoothed complexity model (whereby every entry of the input game is perturbed with random noise) is still PPAD-complete [CDT06a]. Whether there are natural and practically appealing random ensembles of games for which the Nash equilibrium problem is tractable remains an important open problem.

Shifting away from exact computation, we discussed in Chapter 5 that computing ϵ-Nash equilibria for values of ϵ scaling inverse-polynomially in the size of the game remains PPAD-complete [CDT06a]; hence a fully polynomial-time approximation scheme is out of the question unless PPAD is equal to P. At the boundary of intractability, the following emerges as an important open problem.

- *Is there a polynomial-time approximation scheme for the Nash equilibrium problem?*

Despite much research on the subject, the question remains open even for the case of two-player games. In Chapter 5, we discussed special classes of two-player games for which a PTAS exists, such as low-rank games and bounded-norm games. It is important to consider the general case, as well as special classes for which a polynomial-time approximation scheme exists. It is also important to consider whether polynomial-time approximation schemes exist for graphical games. In [DP06], we describe a polynomial-time approximation scheme for normalized graphical games with a constant number of strategies per player, bounded degree, and treewidth which is at most logarithmic in the number of players. Going beyond logarithmic treewidth would be rather important.

- *Is there a polynomial-time approximation scheme for graphical games of bounded degree, but super-logarithmic treewidth?*

Another case worth studying is when the degree is super-logarithmic. In this case, our PPAD-completeness result for graphical games does not apply. Is it then the case that the problem is easier? In [DP08a], we show that an ϵ-Nash equilibrium of a normalized normal-form game with two strategies per player can be computed in time $n^{O(\log\log n+\log\frac{1}{\epsilon})}$, where n is the size of the game. It is intriguing to examine if this result can be improved.

- *Is there a polynomial-time approximation scheme for multi-player normal-form games with two strategies per player?*

Finally, it is important to study other broad and important classes of games which can be approximated efficiently. In Chapter 5, we considered anonymous games and presented a polynomial-time approximation scheme for the case of many players and a constant number of strategies per player. If the number of players is constant, and the number of strategies scales, the problem becomes PPAD-complete. The following questions arise.

- *Is the Nash equilibrium problem in multi-player anonymous games with a constant number of strategies per player PPAD-complete? More generally, what tradeoff between number of strategies and players renders the problem PPAD-complete?*

- *Does our PTAS for multi-player anonymous games extend to a super-constant number of strategies per player?*

- *Is there a fully polynomial-time approximation scheme for multi-player anonymous games with a constant number of strategies per player? If not, is there an efficient PTAS?*

In [Das08], we make progress towards answering the last question by providing an efficient PTAS for the case of two-strategy anonymous games. We believe that our techniques should extend to any constant number of strategies and could possibly also provide a fully polynomial-time approximation scheme. Regarding the first couple of questions, let us note that Papadimitriou and Roughgarden provide in [PR05] an algorithm for multi-player symmetric games — a special case of anonymous games — which remains efficient up to about a logarithmic number of strategies per player. We believe that a similar result should be possible for anonymous games.

Bibliography

[Ajt96] Miklós Ajtai. Generating Hard Instances of Lattice Problems (Extended Abstract). In *the 28th Annual ACM Symposium on Theory of Computing (STOC)*, 1996.

[AKV05] Timothy G. Abbott, Daniel Kane, and Paul Valiant. On the Complexity of Two-Player Win-Lose Games. In *the 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2005.

[AR86] Donald J. Albers and Constance Reid. An Interview with George B. Dantzig: The Father of Linear Programming. *The College Mathematics Journal*, 17(4):293–314, 1986.

[AV07] Ilan Adler and Sushil Verma. The Linear Complementarity Problem, Lemke Algorithm, Perturbation, and the Complexity Class PPAD. *Manuscript*, 2007.

[Bar05] Andrew D. Barbour. *Multivariate Poisson-Binomial Approximation Using Stein's Method*. A. D. Barbour and L. H. Y. Chen (editors), Lecture Notes Series No. 4, Institute for Mathematical Sciences, National University of Singapore, Singapore University Press and World Scientific, 2005.

[BBM07] Hartwig Bosse, Jaroslaw Byrka, and Evangelos Markakis. New Algorithms for Approximate Nash Equilibria in Bimatrix Games. In *the 3rd International Workshop on Internet and Network Economics (WINE)*, 2007.

[BC05] Andrew D. Barbour and Louis H. Y. Chen. *An Introduction to Stein's Method*. A. D. Barbour and L. H. Y. Chen (editors), Lecture Notes Series No. 4, Institute for Mathematical Sciences, National University of Singapore, Singapore University Press and World Scientific, 2005.

[BCE+98] Paul Beame, Stephen A. Cook, Jeff Edmonds, Russell Impagliazzo, and Toniann Pitassi. The Relative Complexity of NP Search Problems. *Journal of Computer and System Sciences*, 57(1):3–19, 1998.

[BCI+08] Christian Borgs, Jennifer T. Chayes, Nicole Immorlica, Adam Tauman Kalai, Vahab S. Mirrokni, and Christos H. Papadimitriou. The Myth of the Folk Theorem. In *the 40th Annual ACM Symposium on Theory of Computing (STOC)*, 2008.

[Bha75] Rabindra N. Bhattacharya. On Errors of Normal Approximation. *The Annals of Probability*, 3(5):815–828, 1975.

[BHJ92] Andrew D. Barbour, Lars Holst, and Svante Janson. *Poisson Approximation*. Oxford University Press, 1992.

[BL06] Andrew D. Barbour and Torgny Lindvall. Translated Poisson Approximation for Markov Chains. *Journal of Theoretical Probability*, 19(3):609–630, 2006.

[Blo99] Matthias Blonski. Anonymous Games with Binary Actions. *Games and Economic Behavior*, 28(2):171–180, 1999.

[Blo05] Matthias Blonski. The Women of Cairo: Equilibria in Large Anonymous Games. *Journal of Mathematical Economics*, 41(3):253–264, 2005.

[BN50] George W. Brown and John von Neumann. Solutions of Games by Differential Equations. In *H. W. Kuhn and A. W. Tucker (editors), Contributions to the Theory of Games*, volume 1, pages 73–79. Princeton University Press, 1950.

[Bub79] Valentinas Bubelis. On Equilibria in Finite Games. *International Journal of Game Theory*, 8(2):65–79, 1979.

[BVV07] Imre Bárány, Santosh Vempala, and Adrian Vetta. Nash equilibria in random games. *Random Structures and Algorithms*, 31(4):391–405, 2007.

[CD05] Xi Chen and Xiaotie Deng. 3-Nash is PPAD-Complete. *Electronic Colloquium on Computational Complexity (ECCC)*, (134), 2005.

[CD06] Xi Chen and Xiaotie Deng. Settling the Complexity of Two-Player Nash Equilibrium. In *the 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2006.

[CDT06a] Xi Chen, Xiaotie Deng, and Shang-Hua Teng. Computing Nash Equilibria: Approximation and Smoothed Complexity. In *the 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2006.

[CDT06b] Xi Chen, Xiaotie Deng, and Shang-Hua Teng. Sparse Games Are Hard. In *the 2nd International Workshop on Internet and Network Economics (WINE)*, 2006.

[CKK+00] Gerard J. Chang, Wen-Tsai Ke, David Kuo, Daphne Der-Fen Liu, and Roger K. Yeh. On $l(d, 1)$-Labelings of Graphs. *Discrete Mathematics*, 220(1-3):57–66, 2000.

[Con92] Anne Condon. The Complexity of Stochastic Games. *Information and Computation*, 96(2):203–224, 1992.

[CS03] Vincent Conitzer and Tuomas Sandholm. Complexity Results about Nash Equilibria. In *the 18th International Joint Conference on Artificial Intelligence (IJCAI)*, 2003.

[CSVY06] Bruno Codenotti, Amin Saberi, Kasturi R. Varadarajan, and Yinyu Ye. Leontief Economies Encode Nonzero Sum Two-Player Games. In *the 17th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2006.

[CT06] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, 2006.

[CTV07] Xi Chen, Shang-Hua Teng, and Paul Valiant. The Approximation Complexity of Win-Lose Games. In *the 18th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2007.

[Dan63] George B. Dantzig. *Linear Programming and Extensions*. Princeton University Press, 1963.

[Das08] Constantinos Daskalakis. An Efficient PTAS for Two-Strategy Anonymous Games. *Manuscript*, 2008.

[DFP06] Constantinos Daskalakis, Alex Fabrikant, and Christos H. Papadimitriou. The Game World Is Flat: The Complexity of Nash Equilibria in Succinct Games. In *the 33rd International Colloquium on Automata, Languages and Programming (ICALP)*, 2006.

[DGP06] Constantinos Daskalakis, Paul W. Goldberg, and Christos H. Papadimitriou. The Complexity of Computing a Nash Equilibrium. In *the 38th Annual ACM Symposium on Theory of Computing (STOC)*, 2006.

[DMP06] Constantinos Daskalakis, Aranyak Mehta, and Christos H. Papadimitriou. A Note on Approximate Nash Equilibria. In *the 2nd International Workshop on Internet and Network Economics (WINE)*, 2006.

[DMP07] Constantinos Daskalakis, Aranyak Mehta, and Christos H. Papadimitriou. Progress in Approximate Nash Equilibria. In *8th ACM Conference on Electronic Commerce (EC)*, 2007.

[DP05] Constantinos Daskalakis and Christos H. Papadimitriou. Three-Player Games Are Hard. *Electronic Colloquium on Computational Complexity (ECCC)*, (139), 2005.

[DP06] Constantinos Daskalakis and Christos H. Papadimitriou. Computing Pure Nash Equilibria in Graphical Games via Markov Random Fields. In *the 7th ACM Conference on Electronic Commerce (EC)*, 2006.

[DP07] Constantinos Daskalakis and Christos H. Papadimitriou. Computing Equilibria in Anonymous Games. In *the 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2007.

[DP08a] Constantinos Daskalakis and Christos H. Papadimitriou. Discretized Multinomial Distributions and Nash Equilibria in Anonymous Games. In *the 49th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2008.

[DP08b] Constantinos Daskalakis and Christos H. Papadimitriou. On Oblivious PTAS for Nash Equilibrium. *Manuscript*, 2008.

[Eav72] B. Curtis Eaves. Homotopies for Computation of Fixed Points. *Mathematical Programming*, 3:1–22, 1972.

[EGG06] Edith Elkind, Leslie Ann Goldberg, and Paul W. Goldberg. Nash Equilibria in Graphical Games on Trees Revisited. In *the 7th ACM Conference on Electronic Commerce (EC)*, 2006.

[EY07] Kousha Etessami and Mihalis Yannakakis. On the Complexity of Nash Equilibria and Other Fixed Points (Extended Abstract). In *the 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2007.

[FL99] Drew Fudenberg and David K. Levine. *The Theory of Learning in Games*. The MIT Press, 1999.

[FNS07] Tomás Feder, Hamid Nazerzadeh, and Amin Saberi. Approximating Nash Equilibria Using Small-Support Strategies. In *the 8th ACM Conference on Electronic Commerce (EC)*, 2007.

[FPT04] Alex Fabrikant, Christos H. Papadimitriou, and Kunal Talwar. The Complexity of Pure Nash Equilibria. In *the 36th Annual ACM Symposium on Theory of Computing (STOC)*, 2004.

[G̈91] Friedrich Götze. On the Rate of Convergence in the Multivariate CLT. *The Annals of Probability*, 19(2):724–739, 1991.

[GGJ76] M. R. Garey, Ronald L. Graham, and David S. Johnson. Some NP-Complete Geometric Problems. In *the 8th Annual ACM Symposium on Theory of Computing (STOC)*, 1976.

[GKT50] David Gale, Harold W. Kuhn, and Albert W. Tucker. On Symmetric Games. In *H. W. Kuhn and A. W. Tucker (editors), Contributions to the Theory of Games*, volume 1, pages 81–87. Princeton University Press, 1950.

[GLL73] Crisostomo B. Garcia, Carlton E. Lemke, and Hans-Jakob Lüthi. Simplicial Approximation of an Equilibrium Point of Noncooperative N-Person Games. *Mathematical Programming*, 4:227–260, 1973.

[GP06] Paul W. Goldberg and Christos H. Papadimitriou. Reducibility Among Equilibrium Problems. In *the 38th Annual ACM Symposium on Theory of Computing (STOC)*, 2006.

[GZ89] Itzhak Gilboa and Eitan Zemel. Nash and Correlated Equilibria: Some Complexity Considerations. *Games and Economic Behavior*, 1(1):80–93, 1989.

[HPV89] Michael D. Hirsch, Christos H. Papadimitriou, and Stephen A. Vavasis. Exponential Lower Bounds for Finding Brouwer Fixed Points. *Journal of Complexity*, 5(4):379–416, 1989.

[JPY88] David S. Johnson, Christos H. Papadimitriou, and Mihalis Yannakakis. How Easy is Local Search? *Journal of Computer and System Sciences*, 37(1):79–100, 1988.

[Kal05] Ehud Kalai. Partially-Specified Large Games. In *the 1st International Workshop on Internet and Network Economics (WINE)*, 2005.

[Kha79] Leonid G. Khachiyan. A Polynomial Algorithm in Linear Programming. *Soviet Mathematics Doklady*, 20(1):191–194, 1979.

[KKM29] Bronislaw Knaster, Casimir Kuratowski, and Stefan Mazurkiewicz. Ein Beweis des Fixpunktsatzes für n-dimensionale Simplexe. *Fundamenta Mathematicae*, 14:132–137, 1929.

[KLS01] Michael J. Kearns, Michael L. Littman, and Satinder P. Singh. Graphical Models for Game Theory. In *the 17th Conference in Uncertainty in Artificial Intelligence (UAI)*, 2001.

[KPS06] Spyros C. Kontogiannis, Panagiota N. Panagopoulou, and Paul G. Spirakis. Polynomial Algorithms for Approximating Nash Equilibria of Bimatrix Games. In *the 2nd International Workshop on Internet and Network Economics (WINE)*, 2006.

[KT07] Ravi Kannan and Thorsten Theobald. Games of Fixed Rank: a Hierarchy of Bimatrix Games. In *the 18th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2007.

[LH64] Carlton E. Lemke and Joseph T. Howson, Jr. Equilibrium Points of Bimatrix Games. *Journal of the Society for Industrial and Applied Mathematics*, 12(2):413–423, 1964.

[LM04] Richard J. Lipton and Evangelos Markakis. Nash Equilibria via Polynomial Equations. In *6th Latin American Symposium (LATIN)*, 2004.

[LMM03] Richard J. Lipton, Evangelos Markakis, and Aranyak Mehta. Playing Large Games Using Simple Strategies. In *the 4th ACM Conference on Electronic Commerce (EC)*, 2003.

[LT79] Gerard van der Laan and Adolphus Johannes Jan Talman. A Restart Algorithm for Computing Fixed Points Without an Extra Dimension. *Mathematical Programming*, 17(1):74–84, 1979.

[LT82] Gerard van der Laan and Adolphus Johannes Jan Talman. On the Computation of Fixed Points in the Product Space of Unit Simplices and an

Application to Noncooperative N Person Games. *Mathematics of Operations Research*, 7(1):1–13, 1982.

[Mil96] Igal Milchtaich. Congestion Games with Player-Specific Payoff Functions. *Games and Economic Behavior*, 13:111–124, 1996.

[MP91] Nimrod Megiddo and Christos H. Papadimitriou. On Total Functions, Existence Theorems and Computational Complexity. *Theoretical Computer Science*, 81(2):317–324, 1991.

[Nas51] John Nash. Non-Cooperative Games. *Annals of Mathematics*, 54(2):286–295, 1951.

[Neu28] John von Neumann. Zur Theorie der Gesellshaftsspiele. *Mathematische Annalen*, 100:295–320, 1928.

[NM44] John von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, 1944.

[OR94] Martin J. Osborne and Ariel Rubinstein. *A Course in Game Theory*. MIT Press, 1994.

[Pap77] Christos H. Papadimitriou. The Euclidean Traveling Salesman Problem is NP-Complete. *Theoretical Computer Science*, 4(3):237–244, 1977.

[Pap94a] Christos H. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994.

[Pap94b] Christos H. Papadimitriou. On the Complexity of the Parity Argument and Other Inefficient Proofs of Existence. *Journal of Computer and System Sciences*, 48(3):498–532, 1994.

[PR05] Christos H. Papadimitriou and Tim Roughgarden. Computing Equilibria in Multi-Player Games. In *the 16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 82–91, 2005.

[R0̈7] Adrian Röllin. Translated Poisson Approximation Using Exchangeable Pair Couplings. *Annals of Applied Probability*, 17:1596–1614, 2007.

[Roo98] Bero Roos. Metric Multivariate Poisson Approximation of the Generalized Multinomial Distribution. *Theory of Probability and its Applications*, 43(2):306–315, 1998.

[Ros71] Joachim Rosenmüller. On a Generalization of the Lemke-Howson Algorithm to Noncooperative N-Person Games. *SIAM Journal on Applied Mathematics*, 21(1):73–79, 1971.

[Ros73] Robert W. Rosenthal. A Class of Games Possessing Pure-Strategy Nash Equilibria. *International Journal of Game Theory*, 2(1):65–67, 1973.

[Rub79] Ariel Rubinstein. Equilibrium in Supergames with the Overtaking Criterion. *Journal of Economic Theory*, 21:1–9, 1979.

[Sca67] Herbert E. Scarf. The Approximation of Fixed Points of a Continuous Mapping. *SIAM Journal on Applied Mathematics*, 15(5):1328–1343, 1967.

[Sho97] Peter Shor. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. *SIAM Journal of Computing*, 26(5):1484–1509, 1997.

[SS04] Rahul Savani and Bernhard von Stengel. Exponentially Many Steps for Finding a Nash Equilibrium in a Bimatrix Game. In *the 45th Symposium on Foundations of Computer Science (FOCS)*, 2004.

[SV06] Grant Schoenebeck and Salil P. Vadhan. The Computational Complexity of Nash Equilibria in Concisely Represented Games. In *the 7th ACM Conference on Electronic Commerce (EC)*, 2006.

[TS07] Haralampos Tsaknakis and Paul G. Spirakis. An Optimization Approach for Approximate Nash Equilibria. In *the 3rd International Workshop on Internet and Network Economics (WINE)*, 2007.

[Wil71] Robert Wilson. Computing Equilibria of N-Person Games. *SIAM Journal on Applied Mathematics*, 21(1):80–87, 1971.

Appendix A

Skipped Proofs

Proof of Claim 5.5: That a tree resulting from TDP has $k-1$ leaves follows by induction: It is true when $k=2$, and for general k, the left subtree has j strategies and thus, by induction, $j-1$ leaves, and the right subtree has at most $k+1-j$ strategies and $k-j$ leaves; adding we get the result.

To estimate the number of cells, let us fix the set of strategies and their ordering at the root of the tree (thus the result of the calculation will have to be multiplied by $2^k k!$) and then count the number of trees that could be output by TDP. Suppose that the root has cardinality m and that the children of the root are assigned sets of sizes j and $m+1-j$ (or, in the event of no duplication, $m-j$), respectively. If $j=2$, then a duplication has to have happened and, for the ordering of the strategies at the left child of the root, there are at most 2 possibilities depending on whether the "divided strategy" is still the largest at the left side; similarly, for the right side there are $m-1$ possibilities: either the divided strategy is still the largest at the right side, or it is not in which case it has to be inserted at the correct place in the ordering and the last strategy of the right side must be moved to the second place. If $j>2$, similar considerations show that there are at most $j-1$ possibilities for the left side and 1 possibility for the right side. It follows that the number of trees is bounded from above by the solution $T(k)$ of the recurrence

$$T(n) = 2\,T(2) \cdot (n-1)T(n-1) \\ + \sum_{j=3}^{n-1} (j-1)T(j) \cdot \max\{T(n-j), T(n+1-j)\}.$$

with T(2)=1. It follows that the total number of trees can be upper-bounded by the function k^{k^2}. Taking into account that there are $2^k k!$ choices for the set of strategies and their ordering at the root of the tree, and that each leaf can be of either Type A, or of Type B, it follows that the total number of cells is bounded by $g(k) = k^{k^2} 2^{k-1} 2^k k!$. $\square$

Proof of Claim 5.6: The proof follows by a straightforward coupling argument. Indeed, for all $i \in \mathcal{I}$, let us couple the choices made by Stage 1 of SAMPLING $\mathcal{X}_i$ and SAMPLING $\widehat{\mathcal{X}}_i$ so that the random leaf $\Phi_i \in \partial T$ chosen by SAMPLING $\mathcal{X}_i$ and the random leaf $\hat{\Phi}_i \in \partial T$ chosen by SAMPLING $\widehat{\mathcal{X}}_i$ are equal, that is, for all $i \in \mathcal{I}$, in the joint probability space $\Pr[\Phi_i = \hat{\Phi}_i] = 1$; the existence of such a coupling is straightforward since Stage 1 of both SAMPLING $\mathcal{X}_i$ and SAMPLING $\widehat{\mathcal{X}}_i$ is the same random walk on T. $\square$

Proof of Lemma 5.8: Let us denote by

$$Good = \{\theta | \theta \in (\partial T)^{\mathcal{I}} : ||F(\cdot|\Phi = \theta) - \widehat{F}(\cdot|\hat{\Phi} = \theta)||_{TV} \leq O\left(k \frac{2^k \log z}{z^{1/5}}\right),$$

and let $Bad = (\partial T)^{\mathcal{I}} - Good$. By Equation (5.7) if follows that $G(Bad) \leq O(kz^{-1/3})$.

$$
\begin{aligned}
||T - \widehat{T}||_{TV} &= \frac{1}{2}\sum_t |F(t) - \widehat{F}(t)| \\
&= \frac{1}{2}\sum_t \left|\sum_\theta F(t|\Phi=\theta)G(\Phi=\theta) - \sum_\theta \widehat{F}(t|\hat{\Phi}=\theta)\widehat{G}(\hat{\Phi}=\theta)\right| \\
&= \frac{1}{2}\sum_t \left|\sum_\theta (F(t|\Phi=\theta) - \widehat{F}(t|\hat{\Phi}=\theta))G(\theta)\right| \qquad \Big(\text{using } G(\theta)=\widehat{G}(\theta), \forall\theta\Big) \\
&\le \frac{1}{2}\sum_t\sum_\theta \left|F(t|\Phi=\theta) - \widehat{F}(t|\hat{\Phi}=\theta)\right| G(\theta) \\
&= \frac{1}{2}\sum_t\sum_{\theta\in Good} \left|F(t|\Phi=\theta) - \widehat{F}(t|\hat{\Phi}=\theta)\right| G(\theta) \\
&\qquad + \frac{1}{2}\sum_t\sum_{\theta\in Bad} \left|F(t|\Phi=\theta) - \widehat{F}(t|\hat{\Phi}=\theta)\right| G(\theta) \\
&\le \sum_{\theta\in Good} G(\theta)\left(\frac{1}{2}\sum_t \left|F(t|\Phi=\theta) - \widehat{F}(t|\hat{\Phi}=\theta)\right|\right) \\
&\qquad + \sum_{\theta\in Bad} G(\theta)\left(\frac{1}{2}\sum_t \left|F(t|\Phi=\theta) - \widehat{F}(t|\hat{\Phi}=\theta)\right|\right) \\
&\le \sum_{\theta\in Good} G(\theta)\cdot O\left(k\frac{2^k\log z}{z^{1/5}}\right) + \sum_{\theta\in Bad} G(\theta) \\
&\le O\left(k\frac{2^k\log z}{z^{1/5}}\right) + O(kz^{-1/3}).
\end{aligned}
$$

□

A.1 Proof of Lemma 5.9

Proof. By the assumption it follows that

$$
\begin{aligned}
&|\mu_v(\theta) - \widehat{\mu}_v(\theta)| \le \\
&\quad \left|\mathcal{E}[\mu_v(\Phi)] - \mathcal{E}[\widehat{\mu}_v(\hat{\Phi})]\right| + z^{(\alpha-1)/2}\sqrt{\mathcal{E}[\mu_v(\Phi)]\log z} + z^{(\alpha-1)/2}\sqrt{\mathcal{E}[\widehat{\mu}_v(\hat{\Phi})]\log z}.
\end{aligned}
$$

Moreover, note that

$$\mathcal{E}[\mu_v(\Phi)] = 2^{-depth_T(v)} \sum_{i \in \mathcal{I}} p_{i,v}(\ell_v^*)$$

and, similarly,

$$\mathcal{E}[\widehat{\mu}_v(\hat{\Phi})] = 2^{-depth_T(v)} \sum_{i \in \mathcal{I}} \widehat{p}_{i,v}(\ell_v^*).$$

By the definition of the Rounding procedure it follows that

$$|\mathcal{E}[\mu_v(\Phi)] - \mathcal{E}[\widehat{\mu}_v(\hat{\Phi})]| \leq 2^{-depth_T(v)} \frac{1}{z}.$$

Hence it follows that

$$|\mu_v(\theta) - \widehat{\mu}_v(\theta)| \leq 2^{-depth_T(v)} \frac{1}{z} + \frac{2\sqrt{\log z}}{z^{(1-\alpha)/2}} \sqrt{\max \{\mathcal{E}[\mu_v(\Phi), \mathcal{E}[\widehat{\mu}_v(\hat{\Phi})]\}]}. \tag{A.1}$$

Let $\mathcal{N}_v(\theta) := \{i : \theta_i = v\}$, $n_v = |\mathcal{N}_v|$. Conditioned on $\Phi = \theta$, the distribution of $T_{v,1}$ is the sum of n_v independent Bernoulli random variables $\{Z_i\}_{i \in \mathcal{N}_v}$ with expectations $\mathcal{E}[Z_i] = p_{i,v}(\ell_v^*) \leq \frac{\lfloor z^\alpha \rfloor}{z}$. Similarly, conditioned on $\hat{\Phi} = \theta$, the distribution of $\widehat{T}_{v,1}$ is the sum of n_v independent Bernoulli random variables $\{\widehat{Z}_i\}_{i \in \mathcal{N}_v}$ with expectations $\mathcal{E}[\widehat{Z}_i] = \widehat{p}_{i,v}(\ell_v^*) \leq \frac{\lfloor z^\alpha \rfloor}{z}$. Note that

$$\mathcal{E}\left[\sum_{i \in \mathcal{N}_v} Z_i\right] = \mu_v(\theta)$$

and, similarly,

$$\mathcal{E}\left[\sum_{i \in \mathcal{N}_v} \widehat{Z}_i\right] = \widehat{\mu}_v(\theta).$$

Without loss of generality, let us assume that $\mathcal{E}[\mu_v(\Phi)] \geq \mathcal{E}[\widehat{\mu}_v(\hat{\Phi})]$. Let us further distinguish two cases for some constant $\tau < 1 - \alpha$ to be decided later

Case 1: $\mathcal{E}[\mu_v(\Phi)] \leq \frac{1}{z^\tau}$.

From (5.9) it follows that,

$$\mu_v(\theta) \leq \mathcal{E}[\mu_v(\Phi)] + z^{(\alpha-1)/2}\sqrt{\mathcal{E}[\mu_v(\Phi)] \log z} \leq \frac{1}{z^\tau} + \frac{\sqrt{\log z}}{z^{(\tau+1-\alpha)/2}} =: g(z).$$

Similarly, because $\mathcal{E}[\widehat{\mu}_v(\hat{\Phi})] \leq \mathcal{E}[\mu_v(\Phi)] \leq \frac{1}{z^\tau}$, $\widehat{\mu}_v(\theta) \leq g(z)$.

By Markov's inequality, $\Pr_{\Phi=\theta}[\sum_{i \in \mathcal{N}_v} Z_i \geq 1] \leq \frac{\mu_v(\theta)}{1} \leq g(z)$ and, similarly, $\Pr_{\hat{\Phi}=\theta}[\sum_{i \in \mathcal{N}_v} \widehat{Z}_i \geq 1] \leq g(z)$. Hence,

$$\begin{aligned}\left|\Pr{}_{\Phi=\theta}\left[\sum_{i \in \mathcal{N}_v} Z_i = 0\right] - \Pr{}_{\hat{\Phi}=\theta}\left[\sum_{i \in \mathcal{N}_v} \widehat{Z}_i = 0\right]\right| \\ = \left|\Pr{}_{\Phi=\theta}\left[\sum_{i \in \mathcal{N}_v} Z_i \geq 1\right] - \Pr{}_{\hat{\Phi}=\theta}\left[\sum_{i \in \mathcal{N}_v} \widehat{Z}_i \geq 1\right]\right| \leq 2g(z).\end{aligned}$$

It then follows easily that

$$||F_v(\cdot|\Phi = \theta) - \widehat{F}_v(\cdot|\hat{\Phi} = \theta)||_{TV} \leq 4g(z) = 4 \cdot \left(\frac{1}{z^\tau} + \frac{\sqrt{\log z}}{z^{(\tau+1-\alpha)/2}}\right). \tag{A.2}$$

Case 2: $\mathcal{E}[\mu_v(\Phi)] \geq \frac{1}{z^\tau}$.

The following claim was proven in [DP07] (Lemma 3.9).

Claim A.1. *For any set of independent Bernoulli random variables $\{Z_i\}_i$ with expectations $\mathcal{E}[Z_i] \leq \frac{\lfloor z^\alpha \rfloor}{z}$,*

$$\left\|\sum_i Z_i - Poisson\left(\mathcal{E}\left(\sum_i Z_i\right)\right)\right\|_{TV} \leq \frac{1}{z^{1-\alpha}}.$$

By application of this lemma it follows that

$$\left\|\sum_{i\in\mathcal{N}_v} Z_i - Poisson(\mu_v(\theta))\right\|_{TV} \leq \frac{1}{z^{1-\alpha}}, \tag{A.3}$$

$$\left\|\sum_{i\in\mathcal{N}_v} \widehat{Z}_i - Poisson(\widehat{\mu}_v(\theta))\right\|_{TV} \leq \frac{1}{z^{1-\alpha}}. \tag{A.4}$$

We study next the distance between the two Poisson distributions. We use the following lemma whose proof is postponed till later in this section.

Lemma A.2. *If* $\lambda = \lambda_0 + D$ *for some* $D > 0$, $\lambda_0 > 0$,

$$\|Poisson(\lambda) - Poisson(\lambda_0)\|_{TV} \leq D\sqrt{\frac{2}{\lambda_0}}.$$

An application of Lemma A.2 gives

$$\|Poisson(\mu_v(\theta)) - Poisson(\widehat{\mu}_v(\theta))\|_{TV} \leq |\mu_v(\theta) - \widehat{\mu}_v(\theta)|\sqrt{\frac{2}{\min\{\mu_v(\theta), \widehat{\mu}_v(\theta)\}}}. \tag{A.5}$$

We conclude with the following lemma proved in the end of this section.

Lemma A.3. *From* (5.9), (5.10), (A.1) *and the assumption* $\mathcal{E}[\mu_v(\Phi)] \geq \frac{1}{z^\tau}$, *it follows that*

$$|\mu_v(\theta) - \widehat{\mu}_v(\theta)|\sqrt{\frac{2}{\min\{\mu_v(\theta), \widehat{\mu}_v(\theta)\}}} \leq \sqrt{72\frac{\log z}{z^{1-\alpha}}}.$$

Combining (A.3), (A.4), (A.5) and Lemma A.3 we get

$$\left\|\sum_{i\in\mathcal{N}_v} Z_i - \sum_{i\in\mathcal{N}_v} \widehat{Z}_i\right\|_{TV} \leq \frac{2}{z^{1-\alpha}} + \sqrt{72\frac{\log z}{z^{1-\alpha}}} = O\left(\frac{\sqrt{\log z}}{z^{(1-\alpha)/2}}\right),$$

which implies

$$||F_v(\cdot|\Phi=\theta) - \widehat{F}_v(\cdot|\hat{\Phi}=\theta)||_{TV} \le O\left(\frac{\sqrt{\log z}}{z^{(1-\alpha)/2}}\right). \tag{A.6}$$

Taking $\tau > (1-\alpha)/2$, we get from (A.2), (A.6) that in both cases

$$||F_v(\cdot|\Phi=\theta) - \widehat{F}_v(\cdot|\hat{\Phi}=\theta)||_{TV} \le O\left(\frac{\sqrt{\log z}}{z^{(1-\alpha)/2}}\right). \tag{A.7}$$

□

Proof of lemma A.2: We make use of the following lemmas.

Lemma A.4. *If $\lambda, \lambda_0 > 0$, the Kullback-Leibler divergence between $Poisson(\lambda_0)$ and $Poisson(\lambda)$ is given by*

$$\Delta_{KL}(Poisson(\lambda)||Poisson(\lambda_0)) = \lambda\left(1 - \frac{\lambda_0}{\lambda} + \frac{\lambda_0}{\lambda}\log\frac{\lambda_0}{\lambda}\right).$$

Lemma A.5 (e.g. [CT06])**.** *If P and Q are probability measures on the same measure space and P is absolutely continuous with respect to Q then*

$$\|P - Q\|_{TV} \le \sqrt{2\Delta_{KL}(P||Q)}.$$

By simple calculus we have that

$$\Delta_{KL}(Poisson(\lambda)||Poisson(\lambda_0)) = \lambda\left(1 - \frac{\lambda_0}{\lambda} + \frac{\lambda_0}{\lambda}\log\frac{\lambda_0}{\lambda}\right) \le \frac{D^2}{\lambda_0}.$$

Then by Lemma A.5 it follows that

$$\|Poisson(\lambda) - Poisson(\lambda_0)\|_{TV} \le D\sqrt{\frac{2}{\lambda_0}}.$$

□

Proof of lemma A.3: From (A.1) and the assumption $\mathcal{E}[\mu_v(\Phi)] \geq \mathcal{E}[\widehat{\mu}_v(\hat{\Phi})]$ we have

$$|\mu_v(\theta) - \widehat{\mu}_v(\theta)|^2 \leq \frac{1}{z^2} + \frac{4 \log z}{z^{1-\alpha}} \mathcal{E}[\mu_v(\Phi)] + 4\frac{1}{z}\frac{\sqrt{\log z}}{z^{(1-\alpha)/2}}\sqrt{\mathcal{E}[\mu_v(\Phi)]}.$$

From the assumption $\mathcal{E}[\mu_v(\Phi)] \geq \frac{1}{z^\tau}$ it follows

$$\mathcal{E}[\mu_v(\Phi)] = \sqrt{\mathcal{E}[\mu_v(\Phi)]}\sqrt{\mathcal{E}[\mu_v(\Phi)]} \tag{A.8}$$

$$\geq \frac{1}{z^{\tau/2}}\sqrt{\mathcal{E}[\mu_v(\Phi)]}. \tag{A.9}$$

Since $\tau < 1 - \alpha$, it follows that, for sufficiently large z which only depends on α and τ, $\frac{1}{z^{\tau/2}} \geq \frac{2\sqrt{\log z}}{z^{(1-\alpha)/2}}$. Hence,

$$\mathcal{E}[\mu_v(\Phi)] \geq \frac{2\sqrt{\log z}}{z^{(1-\alpha)/2}}\sqrt{\mathcal{E}[\mu_v(\Phi)]},$$

which together with (5.9) implies

$$\mu_v(\theta) \geq \mathcal{E}[\mu_v(\Phi)] - z^{(\alpha-1)/2}\sqrt{\mathcal{E}[\mu_v(\Phi)] \log z} \geq \frac{1}{2}\mathcal{E}[\mu_v(\Phi)] \tag{A.10}$$

Similarly, starting from $\mathcal{E}[\widehat{\mu}_v(\hat{\Phi})] \geq \mathcal{E}[\mu_v(\Phi)] - \frac{1}{z} \geq \frac{1}{z^\tau} - \frac{1}{z}$, it can be shown that for sufficiently large z

$$\widehat{\mu}_v(\theta) \geq \frac{1}{2}\mathcal{E}[\widehat{\mu}_v(\hat{\Phi})]. \tag{A.11}$$

From (A.10), (A.11) it follows that

$$\min\{\mu_v(\theta), \widehat{\mu}_v(\theta)\} \geq \frac{1}{2}\min\{\mathcal{E}[\mu_v(\Phi)], \mathcal{E}[\widehat{\mu}_v(\hat{\Phi})]\} = \frac{1}{2}\mathcal{E}[\widehat{\mu}_v(\Phi)] \geq \frac{1}{2}\mathcal{E}[\mu_v(\Phi)] - \frac{1}{2z} \geq \frac{1}{4}\mathcal{E}[\mu_v(\Phi)],$$

where we used that $\mathcal{E}[\mu_v(\Phi)] \geq \frac{1}{z^\tau} \geq \frac{2}{z}$ for sufficiently large z, since $\tau < 1 - \alpha$. Combining the above we get

$$\begin{aligned}
\frac{2|\mu_v(\theta) - \widehat{\mu}_v(\theta)|^2}{\min\{\mu_v(\theta), \widehat{\mu}_v(\theta)\}} &\leq 2\frac{\frac{1}{z^2} + \frac{4\log z}{z^{1-\alpha}}\mathcal{E}[\mu_v(\Phi)] + 4\frac{1}{z}\frac{\sqrt{\log z}}{z^{(1-\alpha)/2}}\sqrt{\mathcal{E}[\mu_v(\Phi)]}}{\frac{1}{4}\mathcal{E}[\mu_v(\Phi)]} \\
&\leq 8\frac{1}{z^2\mathcal{E}[\mu_v(\Phi)]} + 32\frac{\log z}{z^{1-\alpha}} + 32\frac{\sqrt{\log z}}{z^{1+(1-\alpha)/2}\sqrt{\mathcal{E}[\mu_v(\Phi)]}} \\
&\leq 8\frac{z^\tau}{z^2} + 32\frac{\log z}{z^{1-\alpha}} + 32\frac{z^{\tau/2}\sqrt{\log z}}{z^{1+(1-\alpha)/2}} \\
&\leq 8\frac{1}{z^{2-\tau}} + 32\frac{\log z}{z^{1-\alpha}} + 32\frac{\sqrt{\log z}}{z^{(3-\alpha-\tau)/2}} \\
&\leq 72\frac{\log z}{z^{1-\alpha}},
\end{aligned}$$

since $2 - \tau > 1 - \alpha$ and $(3 - \alpha - \tau)/2 > 1 - \alpha$, assuming sufficiently large z. □

A.2 Proof of Lemma 5.10

Proof. We will derive our bound by approximating with the *translated Poisson distribution*, which is defined next.

Definition A.6 ([Rö07])**.** *We say that an integer random variable Y has a* translated Poisson distribution *with paremeters μ and σ^2 and write*

$$Ł(Y) = TP(\mu, \sigma^2)$$

if $Ł(Y - \lfloor \mu - \sigma^2 \rfloor) = Poisson(\sigma^2 + \{\mu - \sigma^2\})$, where $\{\mu - \sigma^2\}$ represents the fractional part of $\mu - \sigma^2$.

The following lemma provides a bound for the total variation distance between two translated Poisson distributions with different parameters.

Lemma A.7 ([BL06]). *Let $\mu_1, \mu_2 \in \mathbb{R}$ and $\sigma_1^2, \sigma_2^2 \in \mathbb{R}_+ \setminus \{0\}$ be such that $\lfloor \mu_1 - \sigma_1^2 \rfloor \leq \lfloor \mu_2 - \sigma_2^2 \rfloor$. Then*

$$\left|\left|TP(\mu_1, \sigma_1^2) - TP(\mu_2, \sigma_2^2)\right|\right|_{TV} \leq \frac{|\mu_1 - \mu_2|}{\sigma_1} + \frac{|\sigma_1^2 - \sigma_2^2| + 1}{\sigma_1^2}.$$

The following lemma was proven in [DP07] (Lemma 3.14).

Lemma A.8. *Let $z > 0$ be some integer and $\{Z_i\}_{i=1}^m$, where $m \geq z^\beta$, be any set of independent Bernoulli random variables with expectations $\mathcal{E}[Z_i] \in \left[\frac{\lfloor z^\alpha \rfloor}{z}, \frac{1}{2}\right]$. Let $\mu_1 = \sum_{i=1}^m \mathcal{E}[Z_i]$ and $\sigma_1^2 = \sum_{i=1}^m \mathcal{E}[Z_i](1 - \mathcal{E}[Z_i])$. Then*

$$\left\| \sum_{i=1}^m Z_i - TP\left(\mu_1, \sigma_1^2\right) \right\|_{TV} \leq O\left(z^{-\frac{\alpha+\beta-1}{2}}\right).$$

Let $\mathcal{N}_v(\theta) := \{i : \theta_i = v\}$, $n_v(\theta) = |\mathcal{N}_v(\theta)|$. Conditioned on $\Phi = \theta$, the distribution of $T_{v,1}$ is the sum of $n_v(\theta)$ independent Bernoulli random variables $\{Z_i\}_{i \in \mathcal{N}_v(\theta)}$ with expectations $\mathcal{E}[Z_i] = p_{i,v}(\ell_v^*)$. Similarly, conditioned on $\hat{\Phi} = \theta$, the distribution of $\widehat{T}_{v,1}$ is the sum of $n_v(\theta)$ independent Bernoulli random variables $\{\widehat{Z}_i\}_{i \in \mathcal{N}_v(\theta)}$ with expectations $\mathcal{E}[\widehat{Z}_i] = \widehat{p}_{i,v}(\ell_v^*)$. Note that

$$\sum_{i \in \mathcal{N}_v(\theta)} \mathcal{E}\left[Z_i\right] = \mu_v(\theta)$$

and, similarly,

$$\sum_{i \in \mathcal{N}_v(\theta)} \mathcal{E}\left[\widehat{Z}_i\right] = \widehat{\mu}_v(\theta).$$

Setting $\mu_1 := \mu_v(\theta)$, $\mu_2 := \widehat{\mu}_v(\theta)$ and

$$\sigma_1^2 = \sum_{i \in \mathcal{N}_v(\theta)} \mathcal{E}\left[Z_i\right](1 - \mathcal{E}\left[Z_i\right]),$$

$$\sigma_2^2 = \sum_{i \in \mathcal{N}_v(\theta)} \mathcal{E}\left[\widehat{Z}_i\right](1 - \mathcal{E}\left[\widehat{Z}_i\right]),$$

we have from Lemma A.8 that

$$\left\| \sum_{i \in \mathcal{N}_v(\theta)} Z_i - TP\left(\mu_1, \sigma_1^2\right) \right\|_{TV} \leq O\left(z^{-\frac{\alpha+\beta-1}{2}}\right). \tag{A.12}$$

$$\left\| \sum_{i \in \mathcal{N}_v(\theta)} \widehat{Z}_i - TP\left(\mu_2, \sigma_2^2\right) \right\|_{TV} \leq O\left(z^{-\frac{\alpha+\beta-1}{2}}\right). \tag{A.13}$$

It remains to bound the total variation distance between the translated poisson distributions using Lemma A.7. Without loss of generality let us assume $\lfloor \mu_1 - \sigma_1^2 \rfloor \leq \lfloor \mu_2 - \sigma_2^2 \rfloor$. Note that

$$\sigma_1^2 = \sum_{i \in \mathcal{N}_v(\theta)} \mathcal{E}\left[Z_i\right](1 - \mathcal{E}\left[Z_i\right]) \geq n_v(\theta) \frac{\lfloor z^\alpha \rfloor}{z} \left(1 - \frac{\lfloor z^\alpha \rfloor}{z}\right) \geq \frac{1}{2} n_v(\theta) \frac{\lfloor z^\alpha \rfloor}{z},$$

where the last inequality holds for values of z which are larger than some function of constant α. Also,

$$\begin{aligned}
|\sigma_1^2 - \sigma_2^2| &\leq \sum_{i \in \mathcal{N}_v(\theta)} \left| \mathcal{E}\left[Z_i\right](1 - \mathcal{E}\left[Z_i\right]) - \mathcal{E}\left[\widehat{Z}_i\right](1 - \mathcal{E}\left[\widehat{Z}_i\right]) \right| \\
&= \sum_{i \in \mathcal{N}_v(\theta)} |p_{i,v}(\ell_v^*)(1 - p_{i,v}(\ell_v^*)) - \widehat{p}_{i,v}(\ell_v^*)(1 - \widehat{p}_{i,v}(\ell_v^*))| \\
&= \sum_{i \in \mathcal{N}_v(\theta)} \left(|p_{i,v}(\ell_v^*) - \widehat{p}_{i,v}(\ell_v^*)| + \left|p_{i,v}^2(\ell_v^*) - \widehat{p}_{i,v}^2(\ell_v^*)\right|\right) \\
&\leq \sum_{i \in \mathcal{N}_v(\theta)} \frac{3}{z} \qquad \left(\text{using } |p_{i,v}(\ell_v^*) - \widehat{p}_{i,v}(\ell_v^*)| \leq \frac{1}{z}\right) \\
&\leq \frac{3 n_v(\theta)}{z}.
\end{aligned}$$

Using the above and Lemma A.7 we have that

$$\begin{aligned}\left|\left|TP(\mu_1,\sigma_1^2)-TP(\mu_2,\sigma_2^2)\right|\right| &\leq \frac{|\mu_1-\mu_2|}{\sigma_1}+\frac{|\sigma_1^2-\sigma_2^2|}{\sigma_1^2}+\frac{1}{\sigma_1^2}\\ &\leq \frac{|\mu_1-\mu_2|}{\sigma_1}+\frac{\frac{3n_v(\theta)}{z}}{\frac{1}{2}n_v(\theta)\frac{\lfloor z^\alpha\rfloor}{z}}+\frac{1}{\frac{1}{2}n_v(\theta)\frac{\lfloor z^\alpha\rfloor}{z}}\\ &\leq \frac{|\mu_1-\mu_2|}{\sigma_1}+O(z^{-\alpha})+\frac{1}{\frac{1}{2}z^\beta\frac{\lfloor z^\alpha\rfloor}{z}}\\ &\leq \frac{|\mu_1-\mu_2|}{\sigma_1}+O(z^{-\alpha})+O(z^{-(\alpha+\beta-1)}).\end{aligned}$$

To bound the ratio $\frac{|\mu_1-\mu_2|}{\sigma_1}$ we distinguish the following cases:

- $\sqrt{3\log z}\sqrt{2^{-depth_T(v)}}\sqrt{|\mathcal{I}|}\leq \frac{1}{2}2^{-depth_T(v)}|\mathcal{I}|$: Combining this inequality with (5.13) we get that

$$|\mathcal{I}|\leq 2^{1+depth_T(v)}n_v(\theta).$$

Hence,

$$\begin{aligned}\frac{|\mu_1-\mu_2|}{\sigma_1} &\leq \frac{\frac{1}{z}+\frac{\sqrt{\log z}}{z}\sqrt{|\mathcal{I}|}}{\sqrt{\frac{1}{2}n_v(\theta)\frac{\lfloor z^\alpha\rfloor}{z}}}\\ &\leq \frac{\frac{1}{z}+\frac{\sqrt{\log z}}{z}\sqrt{2^{1+depth_T(v)}n_v(\theta)}}{\sqrt{\frac{1}{2}n_v(\theta)\frac{\lfloor z^\alpha\rfloor}{z}}}\\ &= O\left(\frac{1}{z^{\frac{\alpha+\beta+1}{2}}}\right)+O\left(\frac{2^{\frac{depth_T(v)}{2}}\sqrt{\log z}}{z^{\frac{1+\alpha}{2}}}\right)\end{aligned}$$

- $\sqrt{3\log z}\sqrt{2^{-depth_T(v)}}\sqrt{|\mathcal{I}|}> \frac{1}{2}2^{-depth_T(v)}|\mathcal{I}|$: It follows that

$$|\mathcal{I}|<12\ 2^{depth_T(v)}\log z.$$

Hence,

$$\begin{aligned}
\frac{|\mu_1-\mu_2|}{\sigma_1} &\leq \frac{\frac{1}{z}+\frac{\sqrt{\log z}}{z}\sqrt{|\mathcal{I}|}}{\sqrt{\frac{1}{2}n_v(\theta)\frac{\lfloor z^\alpha \rfloor}{z}}} \\
&\leq \frac{\frac{1}{z}+\frac{\sqrt{\log z}}{z}\sqrt{12\ 2^{depth_T(v)}\log z}}{\sqrt{\frac{1}{2}n_v(\theta)\frac{\lfloor z^\alpha \rfloor}{z}}} \\
&= O\left(\frac{1}{z^{\frac{\alpha+\beta+1}{2}}}\right) + O\left(\frac{2^{\frac{depth_T(v)}{2}}\log z}{z^{\frac{\alpha+\beta+1}{2}}}\right)
\end{aligned}$$

Combining the above, it follows that

$$\begin{aligned}
\left|\left|TP(\mu_1,\sigma_1^2)-TP(\mu_2,\sigma_2^2)\right|\right| &\leq \frac{|\mu_1-\mu_2|}{\sigma_1}+\frac{|\sigma_1^2-\sigma_2^2|+1}{\sigma_1^2} \\
&\leq O\left(\frac{1}{z^{\frac{\alpha+\beta+1}{2}}}\right)+O\left(\frac{2^{\frac{depth_T(v)}{2}}\sqrt{\log z}}{z^{\frac{1+\alpha}{2}}}\right) \\
&\quad + O\left(\frac{2^{\frac{depth_T(v)}{2}}\log z}{z^{\frac{\alpha+\beta+1}{2}}}\right)+O(z^{-\alpha})+O(z^{-(\alpha+\beta-1)}) \\
&\leq O\left(\frac{2^{\frac{depth_T(v)}{2}}\sqrt{\log z}}{z^{\frac{1+\alpha}{2}}}\right) \\
&\quad + O\left(\frac{2^{\frac{depth_T(v)}{2}}\log z}{z^{\frac{\alpha+\beta+1}{2}}}\right)+O(z^{-\alpha})+O(z^{-(\alpha+\beta-1)}).
\end{aligned}$$

Combining the above with (A.12) and (A.13) we get

$$\begin{aligned}
\left\|\sum_{i\in\mathcal{N}_v(\theta)} Z_i - \sum_{i\in\mathcal{N}_v(\theta)} \widehat{Z}_i\right\|_{TV} &\leq O\left(\frac{2^{\frac{depth_T(v)}{2}}\sqrt{\log z}}{z^{\frac{1+\alpha}{2}}}\right)+O\left(\frac{2^{\frac{depth_T(v)}{2}}\log z}{z^{\frac{\alpha+\beta+1}{2}}}\right) \\
&\quad + O(z^{-\alpha})+O(z^{-(\frac{\alpha+\beta-1}{2})}).
\end{aligned}$$

□

A.3 Concentration of the Leaf Experiments

The following lemmas constitute the last piece of the puzzle and complete the proof of Lemma 5.7. They roughly state that, after the random walk in Stage 1 of the processes Sampling is performed, the experiments that will take place in Stage 2 of the processes Sampling are similar with high probability.

Proof of Lemma 5.12: Note that

$$\mu_v(\Phi) = \sum_{i \in \mathcal{I}} \Omega_i =: \Omega,$$

where $\{\Omega_i\}_i$ are independent random variables defined as

$$\Omega_i = \begin{cases} p_{i,v}(\ell_v^*), & \text{with probability } 2^{-depth_T(v)} \\ 0, & \text{with probability } 1 - 2^{-depth_T(v)}. \end{cases}$$

We apply the following version of Chernoff/Hoeffding bounds to the random variables $\Omega_i' := z^{1-\alpha}\Omega_i \in [0,1]$.

Lemma A.9 (Chernoff/Hoeffding). *Let $Z_1, \ldots, Z_m$ be independent random variables with $Z_i \in [0,1]$, for all i. Then, if $Z = \sum_{i=1}^n Z_i$ and $\gamma \in (0,1)$,*

$$\Pr[|Z - \mathcal{E}[Z]| \geq \gamma \mathcal{E}[Z]] \leq 2\exp(-\gamma^2 \mathcal{E}[Z]/3).$$

Letting $\Omega' = \sum_{i \in I} \Omega_i'$ and applying the above lemma with $\gamma := \sqrt{\frac{1}{\mathcal{E}[\Omega']} \log z}$, it follows that

$$\Pr\left[\left|\Omega' - \mathcal{E}[\Omega']\right| \geq \sqrt{\mathcal{E}[\Omega'] \log z}\right] \leq 2z^{-1/3},$$

which in turn implies

$$\Pr\left[|\Omega - \mathcal{E}[\Omega]| \geq z^{(\alpha-1)/2}\sqrt{\mathcal{E}[\Omega] \log z}\right] \leq 2z^{-1/3},$$

or, equivalently,

$$\Pr\left[|\mu_v(\Phi) - \mathcal{E}[\mu_v(\Phi)]| \geq z^{(\alpha-1)/2}\sqrt{\mathcal{E}[\mu_v(\Phi)]\log z}\right] \leq 2z^{-1/3}.$$

Similarly, it can be derived that

$$\Pr\left[\left|\widehat{\mu}_v(\hat{\Phi}) - \mathcal{E}[\widehat{\mu}_v(\hat{\Phi})]\right| \geq z^{(\alpha-1)/2}\sqrt{\mathcal{E}[\widehat{\mu}_v(\hat{\Phi})]\log z}\right] \leq 2z^{-1/3}.$$

Let us consider the joint probability space which makes $\Phi = \hat{\Phi}$ with probability 1; this space exists since as we observed above $G(\theta) = \widehat{G}(\theta), \forall\theta$. By a union bound for this space

$$\Pr\left[\begin{array}{l} |\mu_v(\Phi) - \mathcal{E}[\mu_v(\Phi)]| \geq \frac{\sqrt{\log z}}{z^{(1-\alpha)/2}}\sqrt{\mathcal{E}[\mu_v(\Phi)]} \\ \quad \vee \left|\widehat{\mu}_v(\hat{\Phi}) - \mathcal{E}[\widehat{\mu}_v(\hat{\Phi})]\right| \geq \frac{\sqrt{\log z}}{z^{(1-\alpha)/2}}\sqrt{\mathcal{E}[\widehat{\mu}_v(\hat{\Phi})]} \end{array}\right] \leq 4z^{-1/3}.$$

which implies

$$G\left(\begin{array}{l} \theta : |\mu_v(\theta) - \mathcal{E}[\mu_v(\Phi)]| \leq \frac{\sqrt{\log z}}{z^{(1-\alpha)/2}}\sqrt{\mathcal{E}[\mu_v(\Phi)]} \\ \quad \wedge \left|\widehat{\mu}_v(\theta) - \mathcal{E}[\widehat{\mu}_v(\hat{\Phi})]\right| \leq \frac{\sqrt{\log z}}{z^{(1-\alpha)/2}}\sqrt{\mathcal{E}[\widehat{\mu}_v(\hat{\Phi})]} \end{array}\right) \geq 1 - 4z^{-1/3}.$$

□

Proof of Lemma 5.13: Suppose that the random variables Φ and $\hat{\Phi}$ are coupled so that, with probability 1, $\Phi = \hat{\Phi}$. Then

$$\mu_v(\Phi) - \widehat{\mu}_v(\hat{\Phi}) = \sum_{i \in \mathcal{I}} \Omega_i =: \Omega,$$

where $\{\Omega_i\}_i$ are independent random variables defined as

$$\Omega_i = \begin{cases} p_{i,v}(\ell_v^*) - \widehat{p}_{i,v}(\ell_v^*), & \text{with probability } 2^{-depth_T(v)} \\ 0, & \text{with probability } 1 - 2^{-depth_T(v)}. \end{cases}$$

We apply Hoeffding's inequality to the random variables Ω_i.

Lemma A.10 (Hoeffding's Inequality)**.** *Let $X_1, \ldots, X_n$ be independent random variables. Assume that, for all i, $\Pr[X_i \in [a_i, b_i]] = 1$. Then, for $t > 0$:*

$$\Pr\left[\sum_i X_i - \mathcal{E}\left[\sum_i X_i\right] \geq t\right] \leq \exp\left(-\frac{2t^2}{\sum_{i=1}^n (b_i - a_i)^2}\right).$$

Applying the above lemma we get

$$\Pr\left[|\Omega - \mathcal{E}\left[\Omega\right]| \geq t\right] \leq 2\exp\left(-\frac{2t^2}{|\mathcal{I}|\frac{4}{z^2}}\right),$$

since, for all $i \in \mathcal{I}$, $|p_{i,v}(\ell_v^*) - \widehat{p}_{i,v}(\ell_v^*)| \leq \frac{1}{z}$. Setting $t = \sqrt{\log z}\sqrt{|\mathcal{I}|}\frac{1}{z}$ we get

$$\Pr\left[|\Omega - \mathcal{E}\left[\Omega\right]| \geq \sqrt{\log z}\sqrt{|\mathcal{I}|}\frac{1}{z}\right] \leq 2\frac{1}{z^{1/2}}.$$

Note that

$$|\mathcal{E}[\Omega]| = |\sum_{i \in \mathcal{I}} \mathcal{E}[\Omega_i]| = |2^{-depth_T(v)} \sum_{i \in \mathcal{I}} (p_{i,v}(\ell_v^*) - \widehat{p}_{i,v}(\ell_v^*))| \leq \frac{1}{z}.$$

It follows from the above that

$$\Pr\left[|\Omega| \leq \frac{1}{z} + \sqrt{\log z}\sqrt{|\mathcal{I}|}\frac{1}{z}\right] \geq 1 - 2\frac{1}{z^{1/2}},$$

which gives immediately that

$$G\left(\theta : |\mu_v(\theta) - \widehat{\mu}_v(\theta)| \leq \frac{1}{z} + \frac{\sqrt{\log z}}{z}\sqrt{|\mathcal{I}|}\right) \geq 1 - \frac{2}{z^{1/2}}.$$

Moreover, an easy application of Lemma A.9 gives

$$G\left(\theta : |n_v(\theta) - 2^{-depth_T(v)}|\mathcal{I}|| \leq \sqrt{3\log z}\sqrt{2^{-depth_T(v)}|\mathcal{I}|}\right) \geq 1 - \frac{2}{z}. \tag{A.14}$$

Indeed, let $T_i = 1_{\Phi_i = v}$. Then $n_v(\Phi) = \sum_{i\in\mathcal{I}} T_i$ and $\mathcal{E}[\sum_{i\in\mathcal{I}} T_i] = 2^{-depth_T(v)}|\mathcal{I}|$. Applying Lemma A.9 with $\gamma = \sqrt{\frac{3\log z}{2^{-depth_T(v)}|\mathcal{I}|}}$ we get

$$\Pr\left[\left|\sum_{i\in\mathcal{I}} T_i - \mathcal{E}\left[\sum_{i\in\mathcal{I}} T_i\right]\right| \geq \sqrt{3\log z}\sqrt{2^{-depth_T(v)}|\mathcal{I}|}\right] \leq \frac{2}{z},$$

which implies

$$\Pr\left[|n_v(\Phi) - 2^{-depth_T(v)}|\mathcal{I}|| \leq \sqrt{3\log z}\sqrt{2^{-depth_T(v)}|\mathcal{I}|}\right] \geq 1 - \frac{2}{z}.$$

□