Randomness in structures and computation

Zhile Jiang

PhD Dissertation

Department of Computer Science
Aarhus University
Denmark

Randomness in structures and computation

A Dissertation
presented to the Faculty of Natural Sciences
of Aarhus University
in Partial Fulfillment of the Requirements
for the PhD Degree

by
Zhile Jiang
November 2024

Abstract

In this thesis, we study three different problems where randomness plays an important role either as a powerful algorithm design tool, or as a source to generate combinatorial structures, or in the modeling the behavior of economic agents.

Randomized algorithms often yield efficient solutions with strong worst-case guarantees, outperforming deterministic methods in complex settings. A classic example is the Goemans-Williamson algorithm for MAX-CUT, which leverages semidefinite programming and randomized rounding to achieve a groundbreaking approximation ratio. Building on these foundational techniques, our first contribution extends their application to game theory, introducing a polynomial-time algorithm for computing approximate pure Nash equilibria in cut games—marking the first improvement since the work of Bhalgat, Chakraborty, and Khanna back in 2010.

In our second contribution, we investigate the behavior of random 3-SAT formulas under relaxed independence assumptions. Instead of assuming mutual independence on the formation of clauses, we consider k-wise independence and examine how this affects satisfiability. Our analysis reveals surprising asymmetries between satisfiability and unsatisfiability, providing new insights into the structural properties of random 3-SAT formulas when limited independence is introduced.

The last part of this thesis examines mechanism design under information asymmetry, where agents' private valuations introduce inherent randomness. We investigate the revenue gap—the difference between the revenue of theoretically optimal mechanisms and simpler linear pricing strategies. In the context of selling divisible items such as communication bandwidth, we employ the ex-ante relaxation technique to provide a comprehensive analysis of the revenue gap. Our findings initialize the study on the trade-offs between simplicity and optimal revenue for selling divisible goods.

Resumé

I denne afhandling undersøger vi tre forskellige problemer i algoritmedesign, teoretisk analyse og mekanismedesign, hvor tilfældighed spiller en vigtig rolle inden.

Randomiserede algoritmer giver ofte effektive løsninger med stærke sandsynlighedsmæssige garantier og overgår deterministiske metoder i komplekse situationer. Et klassisk eksempel er Goemans-Williamson-algoritmen for MAX-CUT, som anvender semidefinit programmering og randomiseret afrunding for at opnå et banebrydende approksimationsforhold. Baseret på disse fundamentale teknikker er vores første bidrag en udvidelse af deres anvendelse til spilteori, hvor vi introducerer en polynomiel-tids algoritme til at beregne tilnærmede rene Nash-ligevægte i cut-spil—den første forbedring siden resultatet af Bhalgat, Chakraborty og Khanna tilbage fra 2010.

I vores andet bidrag undersøger vi opførslen af tilfældige 3-SAT-formler under begrænsede uafhængighedsantagelser. I stedet for at antage fuld uafhængighed, betragter vi k-vis uafhængighed og analyserer, hvordan dette påvirker tilfredsstillelse af formlen. Vores analyse afslører overraskende asymmetrier mellem tilfredsstillelse og utilfredsstillelse og giver ny indsigt i de strukturelle egenskaber af tilfældige 3-SAT-formler, når begrænset uafhængighed introduceres.

Den sidste del af denne afhandling undersøger mekanismedesign under informationsasymmetri, hvor agenternes private værdisætninger introducerer iboende tilfældighed. Vi undersøger indtægtsgabet — forskellen mellem indtægten fra teoretisk optimale mekanismer og enklere lineære prissætningsstrategier. I forbindelse med salg af delbare varer, såsom kommunikationsbåndbredde, anvender vi ex-ante teknikken for at give en uddybende analyse af indtægtsgabet. Vores resultater indleder studiet af balancen mellem enkelhed og optimal indtjening ved handel med delbare varer.

Acknowledgments

First of all, I would like to express my sincere gratitude to my advisor, Ioannis Caragiannis. His commitment, patience, and insightful ideas have been a great help throughout my PhD journey. He set a high standard for mentorship, always approachable and generous with his time, making even the hardest challenges seem less daunting. From him, I learned the importance of careful thinking as well as the value of kindness and support in academia. His constant encouragement has played a key role in my development, and I am very thankful for everything I learned under his guidance.

I would also like to extend my thanks to Kristoffer Arnsfelt Hansen. His encouragement and belief in my abilities gave me the confidence to embark on this PhD journey. The same thanks goes to Peyman Afshani, Gerth Stølting Brodal, Kasper Green Larsen, Chris Schwiegelshohn. Thank all of you for introducing me to the world of theoretical computer science and showing me that this field is not as distant and unreachable as I once imagined. And I also wish to thank everyone in Computational Complexity and Game Theory group and Algorithms, Data Structures and Foundations of Machine Learning group for the great research environment. Specifically, I want to thank Karl Fehrs for being very nice officemate. I would also like to thank Nick Gravin for hosting me for my stay abroad. That was a enjoyable and inspiring stay in Shanghai.

I also want to thank Amik Raj Behera, Ping'an Cheng, Sebastian Homrighausen, Cheng Huang, Chuanwei Lin, Nidhi Rathi, Sudarshan Shyam, Tianyao Zhou for being nice friends. They made my life in Aarhus so much more enjoyable and memorable.

Finally, I want to thank my parents and my girlfriend, Wenyue Ma. Their love and support have been invaluable and irreplaceable throughout this journey.

Zhile Jiang,
Aarhus, November 14, 2024.

Contents

Part I

Overview

Chapter 1

Introduction and roadmap

Randomness undoubtedly plays a pivotal role across diverse domains of computer science. In algorithm design, randomized algorithms offer solutions to complex problems with strong performance guarantees by leveraging randomness. These algorithms are typically more efficient and simpler than their deterministic counterparts when addressing the same problems. A seminal example is the Goemans-Williamson algorithm for the MAX-CUT problem, which achieves a groundbreaking approximation ratio of approximately 0.878 through semidefinite programming and randomized rounding techniques [82]. This innovative approach has been adapted to other optimization problems such as MAX-2-SAT and MAX-DICUT [82], where the randomized rounding method continues to prove valuable. Further improvements in the algorithm for MAX-2-SAT and MAX-DICUT have been achieved by modifying the randomized rounding procedure [65, 109].

In our first work [27], we draw on semidefinite programming and randomized rounding techniques inspired by these foundational methods, extending their applications to new areas in game theory and optimization. Our polynomial-time algorithm for computing approximate pure Nash equilibrium in cut games marks the first improvement in the approximation ratio since the breakthrough result by Bhalgat, Chakraborty, and Khanna in 2010 [20]. Our application of these techniques is both non-trivial and non-standard, showcasing the power of randomized algorithms in advancing this field.

Despite the successes of randomized algorithms, the question of whether randomness inherently enhances computational power remains an open problem. A prominent conjecture suggests that any problem solvable by a probabilistic polynomial-time algorithm can also be solved by a deterministic polynomial-time one[94, 95, 122]. One supporting argument for this conjecture is the existence of derandomization techniques, which convert probabilistic algorithms into deterministic ones while preserving effectiveness, eliminating the need for randomness. For instance, the semidefinite programming algorithms and randomized rounding techniques mentioned above can be derandomized within polynomial time, meaning that deterministic polynomial-time algorithms can achieve the same approximation ratios for these problems [113]. Another classic approach to derandomization uses pairwise independence or, more generally,

k-wise independence. For example, the $1/2$-approximate randomized algorithm for the MAX-CUT problem can be derandomized using pairwise independence. In such cases, the correctness of the algorithm depends only on pairwise independence rather than full mutual independence, highlighting that mutual independence may sometimes be an unnecessarily strong assumption. Indeed, in the random graph model [9] and monotone Boolean functions [18], certain properties require only logarithmic-wise independence.

On the other hand, some real-world problems do not align as well with theoretical predictions based on mutual independence. A notable example is SAT-solvers: theoretically, the transition threshold should simplify SAT instances in most cases, but this prediction does not fully align with practical experience, indicating a gap between real-world behavior and theoretical models under the assumption of mutual independence.

To address this, our second work [35] investigates the transition threshold in random 3-SAT under k-wise independence. By relaxing the assumption of mutual independence to k-wise independence, we explore how the transition threshold shifts when only a constant k is considered. Unlike the mutual independence setting, the k-wise independent setting describes not a single probability distribution but rather a family of distributions. Our constructions and proofs focus on invariant statistics across this family, revealing a surprising asymmetry between satisfiability and unsatisfiability in this relaxed setting.

Our third work returns to the field of algorithmic game theory, specifically focusing on Bayesian mechanism design. Unlike our algorithm for computing approximate pure Nash equilibrium, where randomness serves primarily as a computational tool, in Bayesian mechanism design, randomness presents an inherent challenge that all mechanisms must navigate due to information asymmetry—where agents hold private information, such as valuations, that remains unknown to the designer [38, 89]. Even in deterministic mechanisms, like Myerson's auction [119], understanding the influence of randomness is crucial. This randomness stems from the unpredictable nature of agents' private information and strategic behavior, which can vary significantly and introduce uncertainty into a mechanism's performance. By understanding the effects of randomness, mechanism designers can better anticipate outcomes, enhance robustness, and refine mechanisms to achieve closer-to-optimal revenue, even under uncertain conditions.

This brings us to the concept of the revenue gap—the difference in expected revenue between theoretically optimal mechanisms and simpler, more practical ones [90]. While optimal mechanisms can achieve higher revenue, they often rely on assumptions about agents' behavior and private information that may not hold in large-scale, practical applications. Simpler mechanisms, though easier to implement and computationally efficient, may sacrifice revenue due to their inability to fully address the complexities introduced by information asymmetry. The revenue gap thus encapsulates the trade-off between the complexity of a mechanism and its effectiveness in managing the inherent randomness in agents' private information.

Our work [36] introduces a new setting in which a seller offers a perfectly divisible item, such as communication bandwidth, to multiple agents. In this context, we analyze the revenue gap between revenue-optimal mechanisms and linear pricing mechanisms by leveraging the potential of a canonical technique known as ex-ante relaxation [7].

Roadmap. The remainder of Part I is divided into three chapters, each offering an overview of a distinct project. Chapter 2 delves into equilibrium computation in cut games, where we introduce the background of equilibrium computation, particularly highlighting the role of cut games. A concise overview of our novel techniques is also provided. Chapter 3 shifts the focus to the random SAT problem, beginning with an exploration of the transition threshold conjecture in a uniform setting. We then discuss how things changes under our new k-wise independent setting. Chapter 4 examines the revenue gap of linear pricing in the context of selling divisible items. This chapter includes a formal definition of the new setting and mechanism, followed by an introduction to the techniques employed in our analysis. We present our findings alongside the challenges faced in this new framework. Some sections in Part I may include content that appears less directly related to our main focus, but I have included them as they provide explanations for questions that arose during my study.

Part II consists of the corresponding three publications/manuscripts:

1. Computing better approximate pure Nash equilibria in cut games via semidefinite programming, published in STOC 2023 [27].

2. On the satisfiability of random 3-SAT formulas with k-wise independent clauses, in submission (for arXiv version, see [35]).

3. Bounds on the revenue gap of linear posted pricing for selling a divisible item, in submission (for arXiv version, see [36]).

Chapter 2

Approximate pure Nash equilibrium

In 1950, John Nash introduced the concept of Nash equilibrium—a stable state in non-cooperative games (hereafter referred to as "games") where no player can improve their utility by unilaterally changing their strategy. In his seminal doctoral dissertation, Nash also proved the existence of mixed Nash equilibrium, where players use a probability distribution over actions as their strategy, in finite games [120]. This concept has since become one of the foundational pillars of game theory and has played a pivotal role in advancing modern economics.

As game theory evolved, Rosenthal applied game theory to model several real-world scenarios as *congestion games* [130]. He demonstrated the existence of pure Nash equilibrium—where each player deterministically selects a single action as their strategy-in congestion games. His proof reveals that every congestion game possesses a potential function, a powerful property that facilitates the analysis of player incentives and outcomes. Actually, congestion games are shown to be equivalent to games with a potential function, or *potential games*, whose concept is formalized by Monderer and Shapley [118].

While equilibrium concepts are generally known to exist, the question of whether Nash equilibria can be computed efficiently remains unresolved. Kamal Jain aptly noted, "If your laptop can't find it, then neither can the market," underscoring the challenge: if an equilibrium cannot be computed efficiently, its application in predicting or analyzing outcomes in large-scale practical problems becomes questionable. Unfortunately, recent results suggest significant limitations. In particular, computing a mixed Nash equilibrium, even in small games with two players, has been shown to be PPAD-hard [42, 54], settling the difficulty of finding an equilibrium from complexity perspective.

Moreover, computing approximate mixed Nash equilibrium is generally no easier than computing exact mixed Nash equilibrium [42, 54, 134]. In two-player games, approximation is feasible in quasi-polynomial time [110], but this cannot be improved to polynomial time unless PPAD-hard problems can be solved in subexponential time

[133]. This complexity underscores the formidable challenges involved in efficiently computing Nash equilibrium, especially as games scale in size and complexity.

What about pure Nash equilibria in potential games? Although the potential function guarantees convergence to a pure Nash equilibrium via best-response dynamics—where players sequentially adjust their actions to respond optimally to the current game state—this process can require an exponential number of improvement steps, as seen in other PLS-complete problems [63]. The PLS-completeness result extends to several subclasses of congestion games, such as network congestion games [63], market-sharing games [5], and cut games [135]. Regarding approximate pure Nash equilibria in congestion games, the outlook remains challenging: there is no universal polynomial-time algorithm for computing an approximate pure Nash equilibrium in general potential games unless PLS is tractable [136].

However, for some subclasses of potential games, such as matroid congestion games [5] and symmetric congestion games with similar or jump-bounded latency functions [19, 43], the number of iterations required by best-response dynamics to reach a $(1+\varepsilon)$-approximate pure Nash equilibrium is polynomially bounded. Bhalgat et al. explored various subclasses of constraint satisfaction games, including cut games and party affiliation games, without restricting player attributes [20]. They proposed a method of partitioning players into multiple layers based on their highest possible utility. Using top-down layer dynamics and a rearrangement phase, they computed a 3-approximate pure Nash equilibrium for these games in polynomial time.

A more general algorithm scheme proposed by Caragiannis et al. [30] uses similar idea of partitioning players to handle the dissimilarity among players. Unlike the rearrangement phase in [20], this scheme relies on $(1+\rho)$-best-response dynamics to ensure that each layer is visited only a limited number of times. Here, the key idea is that no coalition of players can significantly improve their contribution to the potential function by more than $1+\rho$ multiplicatively after reaching a $(1+\varepsilon_\rho)$-approximate pure Nash equilibrium (where ε_ρ is a small positive constant relative to ρ). This innovative technique provides strong adaptability, achieving a $(2+\varepsilon)$-approximate pure Nash equilibrium in congestion games with linear latency functions (non-negative coefficients) and a $d^{O(d)}$-approximate pure Nash equilibrium in congestion games with d-degree polynomial latency functions (non-negative coefficients). Later works [67, 139] improved these approximation ratios, and the scheme has been extended to weighted congestion games [26, 32] (which are not potential games, expect the case with linear latency functions [71]), constraint satisfaction games [33], and cost-sharing games [78].

Our work builds on this scheme, specifically focusing on cut games. The algorithm in [33] computes a 3-approximate pure Nash equilibrium in cut games but does not improve upon the results from [20]. We enhance this by replacing the $(1+\varepsilon_\rho)$-best-response dynamics with a subroutine that identifies coalitions of players who can substantially increase their contribution to the potential function by a factor of $1+\rho$ and guarantees a non-negligible improvement in the potential when such a coalition exists. Inspired by [65, 82], this subroutine leverages semidefinite programming and randomized rounding to solve MAX-CUT-like problems effectively. The improvement results in a polynomial-time algorithm that computes a 2.7371-approximate pure Nash equilibrium in cut games (Theorem 2.4.1).

2.1 Congestion games and pure Nash equilibrium

Traffic congestion is a daily challenge, especially for those living in large urban areas. It occurs when the traffic network becomes overloaded. If we view roads as resources, congestion can be described as an increase in travel cost when too many people attempt to use the same resources simultaneously. This increase in cost depends solely on the number of users on each road. In this context, congestion games provide a natural model for analyzing and understanding such scenarios. Formally, we can define congestion games as follows.

Definition 2.1.1 (Congestion games)**.** *A congestion game* $\mathcal{G} = ([n], E, \{S_u\}_{u\in[n]}, \{f_e\}_{e\in E})$ *consists of a finite set of n players* $[n] = \{1,2,\ldots,n\}$*, a finite set of resources E, a finite set of pure strategies* $S_u \subseteq 2^E$ *for each player* $u \in [n]$*, and a latency function* $f_e : \mathbb{N} \to \mathbb{R}$ *for each resource* $e \in E$*. The payoff function (as cost) of each player u with respect to a state* $\mathbf{s} \in \prod_{u\in[n]} S_u$ *of game* $\mathcal{G}$ *is* $p_u(\mathbf{s}) = \sum_{e\in s_i} f_e(n_e(\mathbf{s}))$*, where* $n_e(\mathbf{s}) = |\{u \in [n] \mid s_u \ni e\}|$ *denotes the number of players that choose resources e in profile* $\mathbf{s}$*. The players are cost-minimizers in the congestion game.*

Note *payoff function* may be referred to *cost function* when each player aims to minimize it (the form of congestion games is in this case), and may be referred to *utility function* when each player aims to maximize it. In such a congestion game, each player i aims to minimize their cost p_i selfishly. I.e. given a state $\mathbf{s}$, player i will minimize cost $p_i(\mathbf{s}_{-i}, s'_i)$ by choose their best strategy s'_i from their strategy set S_i. The notation $(\mathbf{s}_{-i}, s'_i)$ defines the state obtained from $\mathbf{s}$ when player i deviates to strategy s'_i unilaterally.

Assuming all players are rational optimizers, the outcome of games can be analyzed via stable states, or Nash equilibria, in which no player has an improvement move. Formally, we have the following definition.

Definition 2.1.2 (Pure Nash equilibrium)**.** *Consider a finite game* $\mathcal{G} = ([n], \{S_u\}_{u\in[n]}, \{p_u\}_{u\in[n]})$ *consisting of a set of n players* $[n] = \{1,2,\ldots,n\}$*, a finite set of pure strategies* S_u *for each player* $u \in [n]$*, and a payoff function* $p_u : \prod_{u\in[n]} S_u \to \mathbb{R}$ *for each player* $u \in [n]$*. A state* $\mathbf{s} \in \prod_{u\in[n]} S_u$ *is a pure Nash equilibrium of* $\mathcal{G}$ *iff*

$$\forall u \in [n], s'_u \in S_u \quad p_u(\mathbf{s}) \succeq p_u(s'_u, \mathbf{s}_{-u}). \tag{2.1}$$

The relation $\succeq$ *can be either* $\geq$ *for the case of maximizing the utility or* $\leq$ *for the case of minimizing the cost.*

The existence of pure Nash equilibrium is not guaranteed in general finite games, but it is not hard to prove its existence in congestion games.

Theorem 2.1.3 ([130]). *Every congestion game has at least one pure Nash equilibrium.*

Proof. Consider any congestion game $\mathscr{G} = ([n], E, \{S_u\}_{u\in[n]}, \{f_e\}_{e\in E})$ of standard form in the definition, and a function $\Phi : \prod_{u\in[n]} S_u \to \mathbb{R}$ defined as

$$\Phi(\mathbf{s}) = \sum_{e\in E} \sum_{i=1}^{n_e(\mathbf{s})} f_e(i).$$

For any state $\mathbf{s}$ and $(\mathbf{s}_{-u}, s'_u)$, we have

$$\begin{aligned}
\Phi(\mathbf{s}_{-u}, s'_u) - \Phi(\mathbf{s}) &= \sum_{e\in E} \left(\sum_{i=1}^{n_e(\mathbf{s}_{-u}, s'_u)} f_e(i) - \sum_{i=1}^{n_e(\mathbf{s})} f_e(i) \right) \\
&= \sum_{e\in s_u \triangle s'_u} \left(\sum_{i=1}^{n_e(\mathbf{s}_{-u}, s'_u)} f_e(i) - \sum_{i=1}^{n_e(\mathbf{s})} f_e(i) \right) \\
&= \sum_{e\in s'_u \setminus s_u} f_e(n_e(\mathbf{s}_{-u}, s'_u)) - \sum_{e\in s_u \setminus s'_u} f_e(n_e(\mathbf{s})) \\
&= \sum_{e\in s'_u} f_e(n_e(\mathbf{s}_{-u}, s'_u)) - \sum_{e\in s_u} f_e(n_e(\mathbf{s})) \\
&= p_u(\mathbf{s}_{-u}, s'_u) - p_u(\mathbf{s}). \qquad (2.2)
\end{aligned}$$

The first equation is by the definition of Φ. In the second equation, the notation $s_u \triangle s'_u$ represents the set consisting of all resources that are in either s_u or s'_u exclusively. The equation is due to the fact $n_e(\mathbf{s}_{-u}, s'_u) = n_e(\mathbf{s})$ which holds for any $e \in E \setminus (s_u \triangle s'_u)$. The third equation is due to the observation that the derivation of u from s_u to s'_u increases n_e by one for any resource e that is in s'_u but not in s_u, and decreases n_e by one for any resource e that is in s_u but not in s'_u. The fourth equation follows since n_e remains the same before and after the derivation of u for any resource e in both s_u and s'_u. The last equation is by the definition of the payoff function in congestion games.

Equation (2.2) implies that

$$\Phi(\mathbf{s}) \le \Phi(\mathbf{s}_{-u}, s'_u) \iff p_u(\mathbf{s}) \le p_u(\mathbf{s}_{-u}, s'_u).$$

Since Φ is a real-valued function with a finite domain, there always exists a state $\mathbf{s}^* \in \prod_{u\in[n]} S_u$ that minimizes Φ. The state $\mathbf{s}^*$ must satisfy the condition (2.1) due to the above implication and is a pure Nash equilibrium in the congestion game $\mathscr{G}$. $\square$

The property $\Phi(\mathbf{s}_{-u}, s'_u) - \Phi(\mathbf{s}) = p_u(\mathbf{s}_{-u}, s'_u) - p_u(\mathbf{s})$ whenever a player u unilaterally move their own strategy to s'_u from s_u in a state $\mathbf{s}$ is crucial. Functions with such property are named as *potential functions*. Potential games are defined as games with a potential function. Congestion games are potential games since above proof shows the existence of potential function, called Rosenthal's potential function, in congestion games. An interesting fact is that any potential game can always be represented by some congestion game, which is not obvious [118]. Formally speaking, any potential game is *isomorphic* to some congestion game. Thus, potential games and congestion games are just two different names of the same class of games.

Despite the ease of proving its existence, computing a pure Nash equilibrium in congestion games is PLS-complete in general [63], as we will discuss further in the next section. This PLS-completeness result also extends to computing a ρ-approximate pure Nash equilibrium in congestion games, where ρ can be any constant computable in polynomial time. Below, we introduce the definition of ρ-approximate pure Nash equilibrium in games where players act as either minimizers or maximizers.

Definition 2.1.4 (ρ-approximate Pure Nash equilibrium)**.** *Consider a finite game $\mathcal{G} = ([n], \{S_u\}_{u\in[n]}, \{p_u\}_{u\in[n]})$ consists of a set of n players $[n] = \{1,2,\ldots,n\}$, a finite set of pure strategies S_u for each player $u \in [n]$, and a payoff function $p_u : \prod_{u\in[n]} S_u \to \mathbb{R}$ for each player $u \in [n]$.*

If p_u is a cost function and the objective of each player is to minimize it, a state $\mathbf{s} \in \prod_{u\in[n]} S_u$ is a ρ-approximate pure Nash equilibrium of $\mathcal{G}$ iff

$$\forall u \in [n], s'_u \in S_u \quad p_u(\mathbf{s}) \le \rho \cdot p_u(s'_u, \mathbf{s}_{-u}). \tag{2.3}$$

If p_u is a utility function and the objective of each player is to maximize it, a state $\mathbf{s} \in \prod_{u\in[n]} S_u$ is a ρ-approximate pure Nash equilibrium of $\mathcal{G}$ iff

$$\forall u \in [n], s'_u \in S_u \quad p_u(\mathbf{s}) \ge \frac{1}{\rho} \cdot p_u(s'_u, \mathbf{s}_{-u}). \tag{2.4}$$

The complexity class PLS

We have mentioned the complexity class PLS multiple times, but one might ask why PLS is a relevant benchmark for discussing hardness of equilibrium computation problem. Therefore, a brief overview of PLS and its inherent challenges will be useful. PLS, or *Polynomial Local Search*, is a complexity class for local optimization problems defined in [102]. A local search problem, denoted as Π_{LS}, consists of a set of optimization problem instances defined by a neighborhood function. The objective is to find a locally optimal solution for an instance of Π_{LS}, where “local” optimality is specified by the neighborhood function. A typical neighborhood function is *flip*, commonly used for problems with Boolean variables. Consider a solution x to an optimization problem with boolean variables as a $0/1$ vector. The neighborhood function $\mathrm{flip}(x)$ defines a set of boolean vectors such that any $x' \in \mathrm{flip}(x)$ has Hamming distance of 1 to x. For problems on graphs, *swap* can be a common neighborhood function. Generally, “polynomial-time” in this context means that an improved solution in the neighborhood of the current solution can be computed in polynomial time. The formal definition follows.

Definition 2.1.5 (Polynomial-time Local Search [116])**.** *Let Π_{LS} be a local search problem and Π the underlying combinatorial optimization problem. The local search problem Π_{LS} is in the class* PLS *if $\Pi \in$* NPO *(the optimization counterpart of* NP*) and if two polynomial-time algorithms A and B exist that satisfy the following properties:*

- *For a problem instance of Π_{LS}, algorithm A returns a solution.*
- *For a problem instance of Π_{LS} and a solution, algorithm B decides whether the solution is a local optimum and, if this is not the case, it returns a better neighboring solution.*

A local search problem Π_{LS} always has an underlying combinatorial optimization problem Π (without the neighborhood function). For example, the underlying combinatorial optimization problem of computing a pure Nash equilibrium in a potential game is optimizing the potential function of the game. We usually denote a local search problem Π_{LS} as $\Pi/\textit{neighborhood}$, where Π is the underlying combinatorial optimization problem and *neighborhood* is the neighborhood function. The main difference between the local search problem and the classical combinatorial optimization problem is that the former has a structure defined by a neighborhood function and we only aim to find a locally optimal solution.

In other words, PLS contains problems that can compute an initial solution (the initial position of the computing path) and decide an improvement within the neighborhood (a move in the computing path) in polynomial time while the length of computing path can be exponentially long. However, it only means that local search method itself can not solve PLS-complete problems efficiently; then why do we believe the PLS is hard? It is true that PLS, as a subclass of TFNP where absence of solution absent itself from, is unlikely to contain any FNP-complete problem, which allows absence of solution. But researchers have several collateral evidences to believe the hardness of PLS. Some of them are:

- PLS reflects the implicit mathematical argument "Every finite directed acyclic graph has a sink". It is unlike this fundamental combinatorial principle can be translated into an efficient constructive manner, which provides polynomial-time algorithms. [126]
- PLS-complete problems cannot be solved in polynomial time if we assume the existence of one-way permutations and indistinguishability obfuscation for P/poly [92].
- TFNP is unlikely to have complete problems [114]. And since no non-syntactic complexity class between PLS and TFNP is known yet, any attempt to capture the hardness of TFNP needs to consider PLS [127].

2.2 Cut games

In this section, we introduce the MAX-CUT/flip problem, which is equivalent to computing a pure Nash equilibrium in cut games. Cut games are potential games. In a cut game, players are associated with vertices in a graph, and each player assigns a Boolean value to their vertex in order to maximize the total weight of adjacent edges connecting to vertices with different values. The formal definition is provided below.

Definition 2.2.1 (Cut Games)**.** *A cut game consists of a graph $G = (V,E)$, with $|V| = n$ and with each edge $e \in E$ having a positive weight w_e, and a finite set of n players where each player $u \in [n]$ controls a distinct vertex u on G. The pure strategies of each player are either 0 or 1, and the utility function of each player u with respect to a state $\mathbf{s} \in \{0,1\}^n$ is $f_u(\mathbf{s}) = \sum_{(u,v)\in E_u | s_u \neq s_v} w_{u,v}$, where E_u is the set of the edges incident to u.*

Cut games are a subclass of potential games. Their potential function Φ of cut games is defined as

$$\Phi(\mathbf{s}) = \sum_{\substack{(u,v)\in E \\ \text{s.t. } s_u \neq s_v}} w_{u,v}.$$

It is straightforward to observe that this function is the objective function for the MAX-CUT problem. The set of states where exactly one player changes their strategy from $\mathbf{s}$ is identical to the set defined by the neighborhood function flip. Thus, computing a pure Nash equilibrium in cut games is equivalent to the local search problem MAX-CUT/flip, well-known to be PLS-complete [135]. Cut games are also congestion games by the equivalence between congestion games and potential games [118]. Moreover, a simple explicit reduction (Reduction 2.2.2) can show PLS-hardness of computing pure Nash equilibrium in congestion games with *linear* latency functions. Thus, understanding cut games is crucial, as they serve as a bridge connecting the fundamental combinatorial optimization problem and the fundamental congestion games.

Reduction 2.2.2. *Given an edge-weighted graph $G = (V,E)$, there is a congestion game $\mathcal{G}$ with linear latency functions so that any pure Nash equilibrium in $\mathcal{G}$ corresponds to a solution of* MAX-CUT*/flip on G. The definition is as follows:*

- *The player set of $\mathcal{G}$ is $[|V|]$, i.e., where each player corresponds to a vertex in V.*
- *Each edge $e \in E$ corresponds to two resources $r_e^{(0)}$ and $r_e^{(1)}$. They have the same latency function $f_{r_e^{(0)}}(x) = f_{r_e^{(1)}}(x) = w_e \cdot x$.*
- *The player corresponding to vertex u has two strategies. One is $s_u^{(0)} = \{r_e^{(0)} | e \in E_u\}$, the other is $s_u^{(1)} = \{r_e^{(1)} | e \in E_u\}$, where E_u is the set of edges indicated by u.*

2.3 An algorithmic template for computing approximate pure Nash equilibria

Our work builds on an algorithm for computing approximate pure Nash equilibria in congestion games [30]. This algorithm was later adapted for various game classes, including constraint satisfaction games [33], cost-sharing games [78], and even non-potential games such as weighted congestion games [26, 32, 80]. A common feature

ALGORITHM 2.1: The algorithm for computing approximate pure Nash equilibria in cut games

Input: A constraint satisfaction game $\mathcal{G}$ with n players
Output: A state S_{out} of $\mathcal{G}$, a real number $\rho > 1$

1: $\Delta \leftarrow f(n,\rho)$;
2: Set $U_{\min} \leftarrow \min_{j\in[n]} U_j$, $U_{\max} \leftarrow \max_{j\in[n]} U_j$, and $m \leftarrow 1 + \lfloor \log_\Delta (U_{\max}/U_{\min}) \rfloor$;
3: Partition the players into blocks B_1, B_2, ..., B_m, such that $j \in B_i \iff U_j \in \left(U_{\max}\Delta^{-i}, U_{\max}\Delta^{1-i}\right]$;
4: Set $S \leftarrow$ an arbitrary initial state;
5: Update $S \leftarrow$ MagicState$(\mathcal{G}, S, B_1, \rho)$;
6: **for** phase $i \leftarrow 1$ to $m-1$ such that $B_i \neq \emptyset$ **do**
7:     **repeat**
8:         Update $(S, change) \leftarrow$ MagicState$(\mathcal{G}, S, B_{i+1}, \rho)$;
9:         Update $S \leftarrow$ APNE$(\mathcal{G}, S, B_i, \rho)$;
10:     **until** not *change*
11: **end for**
12: **return** $S_{\text{out}} \leftarrow S$;

across all these games is the presence of either a potential function or an approximate potential function, which guarantees the existence of a so-called "magic state". In simple terms, a ρ-magic state for a group of players guarantees that the value of the potential function of any subset of the group is at least $1/\rho$ times its maximum value. We define the potential function of a subset R of players in cut games as

$$\Phi_R(\mathbf{s}) = \sum_{\substack{(u,v)\in E_R \\ \text{s.t. } s_u \neq s_v}} w_{u,v},$$

where E_R denotes a subset of E containing edges incidents to at least one vertex in R. The definition of the potential function for subsets is similar in other games, though the specifics may differ.

In Algorithm 2.1, we present an abstracted version of the algorithmic template tailored for cut games, omitting the specific implementation details for identifying the magic states. Note that the algorithm is described for the scenario where players are utility-maximizers (as in cut games); a similar adaptation can be applied for cost-minimizers, which is relevant in the context of congestion games.

The algorithm initializes parameter Δ to be polynomially related to n (line 1 where f is a polynomial function). Next (line 2-3), the players are partitioned into blocks B_1, B_2, ..., B_m based on their maximum possible utility. We use U_j to represent the maximum possible utility for agent j. By our partition, the players' maximum utilities differ by at most Δ. Starting with an arbitrary initial state (line 4), the algorithm proceeds in a series of phases (line 5-11). In phase i, players from blocks B_i and B_{i+1} are allowed to update their strategies. The players from block B_{i+1} with lower maximum utility will try to converge to a ρ-magic state (line 8). The subroutine MagicState$(\mathcal{G}, S, B, \rho)$ acts as an oracle that computes a ρ-magic state for B according to the information of game $\mathcal{G}$ and current state S without touching any other players outside B. The players from B_i, with a higher maximum utility, keep fixing themselves to a ρ-approximate pure Nash equilibrium (line 9). The subroutine APNE($\mathcal{G}, S, B, \rho$) implements this process by local search, which converges within polynomial time when players have similar maximum utility. Especially, in phase 0, we only have a single block B_1 where MagicState() is invoked (line 6).

Now, let's provide some intuition on why the algorithm converges to a ρ-approximate pure Nash equilibrium. Once the strategy of a player j in block B_i is fixed at the end of phase i, their utility is relatively high, close to their maximum possible utility. This outcome is ensured by the properties of the magic state. Any reduction in their utility afterward can only result from local search updates in subsequent phases for block B_{i+1} and later or from the calls to the magic state subroutine for block B_{i+2} and beyond.

The potential impact from strategy improvements in later phases is bounded by $O(U_{\max}\Delta^{-i})$. Similarly, the influence from the calls to the magic state subroutine is bounded by $O(n \cdot U_{\max}\Delta^{-i-1})$. By choosing Δ carefully (based on the constant hidden in the big-O notation), we can ensure that both of these influences are negligible for players in block B_i. Because this negligible impact does not disrupt the ρ-approximate pure Nash equilibrium achieved at the end of phase i, players in B_i will remain at a ρ-approximate equilibrium until the algorithm terminates.

We note that the quality of the ρ-approximate pure Nash equilibrium achieved by the algorithm is directly tied to the implementation of the MagicState() subroutine. In particular, the accuracy of the ρ-magic state plays a crucial role in ensuring that players remain close to equilibrium throughout the process. Moreover, the MagicState() subroutine is also the primary bottleneck in terms of time complexity. While the rest of the algorithm can be executed efficiently in polynomial time, the MagicState() subroutine involves verifying a property over the entire power set of players, which can be computationally intensive. Thus, optimizing this component is key to improving the overall efficiency of the algorithm.

2.4 Our results and techniques

The next theorem states our first main result.

Theorem 2.4.1 (Informal). *A* 2.7371*-approximate pure Nash equilibrium in cut games can be computed in polynomial time.*

In previous works, the oracle MagicState() was typically implemented using local search techniques. These approaches converge to a $(1+\varepsilon)$-approximate pure Nash equilibrium in polynomial time, and this $(1+\varepsilon)$-approximate equilibrium implies a magic state by an argument similar to arguments used to bound the price of anarchy. However, the quality of the magic state is limited by the same factors as the price of anarchy. In contrast, our breakthrough is presented in Theorem 2.4.1, where we implement the oracle MagicState() using a sequence of improvements to the potential function determined by semidefinite programming (SDP) and randomized rounding.

In our approach, players can simultaneously update their strategies in a single coordinated step, rather than sequentially as in local search. We refer to this coordinated update as a *global move*. However, directly applying global moves to compute an approximate pure Nash equilibrium for general cases is challenging, as convergence is not guaranteed without first partitioning the players based on their maximum possible utility. This partitioning is crucial to ensure sufficient progress and avoid the pitfalls of unbounded update sequences.

The key technique involves using a mathematical program to determine whether there exists a global move that significantly improves the potential function of a subset of players. For any given state S, we introduce the Boolean variables x_i (where $x_i = 1$ indicates that player i changes their strategy and $x_i = -1$ indicates no change) to represent whether the strategy of agent i is flipped. The total change in the potential function is expressed as:

$$X = \frac{1}{2}\left(\sum_{(i,j)\in E\setminus\mathrm{CUT}(S)} w_{i,j}\cdot(1-x_i\cdot x_j) - \sum_{(i,j)\in\mathrm{CUT}(S)} w_{i,j}\cdot(1-x_i\cdot x_j)\right).$$

Here, $\mathrm{CUT}(S)$ denotes the set of edges that are in the cut defined by state S, and $(1-x_i\cdot x_j)$ captures the XOR-condition. The potential of the subset containing all players who have flipped their strategies in state S is given by:

$$Y = \frac{1}{4}\left(\sum_{(i,j)\in\mathrm{CUT}(S)} w_{i,j}\cdot(3+x_i+x_j-x_i\cdot x_j)\right),$$

where $(3+x_i+x_j-x_i\cdot x_j)$ encodes the OR-condition. By maximizing $Z = X-\rho\cdot Y$, we can check if there is a global move that improves the potential of the flipped players by a factor of ρ. However, solving this problem with Boolean variables is computationally difficult. To address this, we relax the Boolean variables to n-dimensional unit vectors (referred to as v_i for x_i), leading to the following semidefinite program:

$$\begin{aligned}
\text{maximize}\quad & Z = \frac{1}{2}\sum_{(i,j)\in E\setminus\mathrm{CUT}(S)} w_{i,j}\cdot(1-v_i\cdot v_j)\\
& -\frac{1}{4}\sum_{(i,j)\in\mathrm{CUT}(S)} w_{i,j}\cdot(3\rho-1+(\rho-1)\cdot(\widehat{v}\cdot v_i+\widehat{v}\cdot v_j)-(\rho+1)\cdot v_i\cdot v_j)\\
\text{subject to}\quad & \widehat{v}\cdot\widehat{v}=1,\ \widehat{v}\in\mathbb{R}^n,\\
& v_i\cdot v_i = 1,\ v_i\in\mathbb{R}^n.
\end{aligned}$$

By applying randomized rounding to the SDP solution, we can determine a suitable global move. A successful rounding requires that:

$$\sum_{(i,j)\in E\setminus\mathrm{CUT}(S)} w_{i,j}\cdot\Pr[\mathrm{XOR}(i,j)] - \sum_{(i,j)\in\mathrm{CUT}(S)} w_{i,j}\cdot\Pr[\mathrm{XOR}(i,j)] > 0,$$

whenever the optimal value Z is positive so that the potential function increase on average. The event $\mathrm{XOR}(i,j)$ denotes the probability that v_i and v_j are rounded to different signs. A surfficient conditions so that the above inequality is true are:

$$\Pr[\mathrm{XOR}(i,j)] \geq \frac{A}{2}\cdot(1-v_i\cdot v_j), \tag{2.5}$$

$$\Pr[\mathrm{XOR}(i,j)] \leq \frac{A}{2}\cdot(3\rho-1+(\rho-1)\cdot(v_i\cdot\widehat{v}+v_j\cdot\widehat{v})-(\rho+1)\cdot v_i\cdot v_j). \tag{2.6}$$

for appropriately selected A and ρ. While Inequality (2.5) inequality can be easily achieved using the classic random hyperplane rounding method from [82], Inequality (2.6) is more challenging. To address this, we first adjust the vectors before applying the random hyperplane rounding, similar to the approach in [65]. Specifically, we relocate each vector v_i, which forms an angle θ with the reference vector $\widehat{v}$, to a new position v_i' in the 2-dimensional plane defined by v_i and $\widehat{v}$, at an angle $f(\theta)$ from $\widehat{v}$ (see Figure 2.1). In our rounding scheme, we use a rotation function $f(\theta)=\frac{\pi}{2}\cdot(1-\cos\theta)$. We note that the analysis of the rounding method using this rotation function is highly non-trivial and requires detailed arguments (to get the approximation ratio 2.7371).

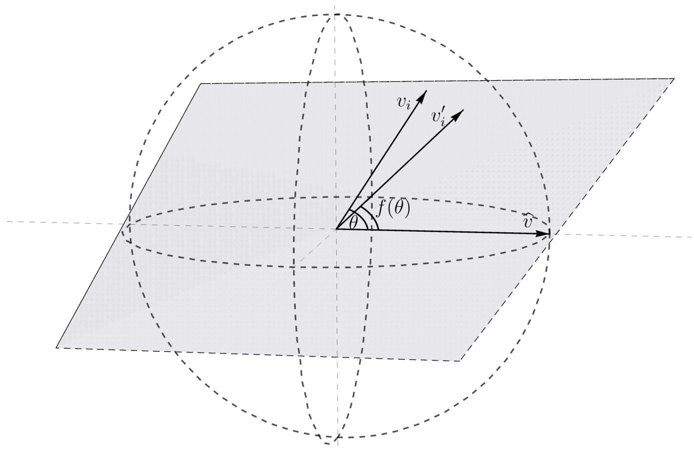

Figure 2.1: The reference vector is represented by $\widehat{v}$, v_i' represents the new position of v_i after rotating. The light gray plane represents the 2-dimensional plane defined by v_i and $\widehat{v}$.

2.5 Discussion and open problems

In this work, we introduce a novel optimization problem as a tool to support equilibrium computation, marking the first improvement in computing approximate pure Nash equilibria in cut games since 2010. This approach broadens the scope of equilibrium computation and enhances the applicability of semidefinite programming to game theory.

After improving the approximation ratio from 3 to 2.7371, a natural question arises: can we push it further? The potential lies in refining the rounding technique. As we discussed in Section 2.4, we currently round solution vectors from our semidefinite program using the rotation function $f(\theta) = \frac{\pi}{2} \cdot (1 - \cos\theta)$, which adjusts vectors based on their angles relative to the reference vector $\widehat{v}$. This approach allows us to approximate both XOR-type and OR-type terms in the objective function of our semidefinite program. However, we do not know whether this rotation function is best possible. We do know however that it provides a relatively poor approximation for XOR-type terms, despite good overall performance. Thus, it is plausible that an alternative rotation function could yield an approximation ratio below 2.7371 by more effectively balancing the performance for both term types if such a function exists.

If an improved rotation function cannot be found, studying the optimality of this function (perhaps within a family of rotation functions, as in [109]) could still be valuable. Such an investigation might reveal a limitation of our method. The general lower bound for approximate pure Nash equilibrium in cut games also remains an open question. Elsässer and Tscheuschner showed that MAX-CUT/flip on graphs with a maximum degree of five is PLS-hard [62]. Extending these results could potentially lead to a constant-factor inapproximability for pure Nash equilibrium in cut games.

Our idea of global moves appears adaptable for computing approximate pure Nash equilibria in other potential games. For instance, as noted in the conclusion of our relevant paper (Section 5.7), extending this technique to party-affiliation games would be easy. Other promising directions include game versions of the MAX-2-SAT and the MAX-DICUT problem, as the combinatorial optimization version of these problems have nice approximations via semidefinite programming. Applying our techniques to approximate pure Nash equilibria in congestion games with linear or even polynomial latency functions is another intriguing challenge. This may require introducing convex programming, which is also polynomial-time solvable[1], to formulate our global moves in congestion games with polynomial latency function. In this case, a new rounding technique would also be necessary.

[1]There are NP-hard instances of convex programming, but many subclasses of convex programming (most likely including our case) are polynomial-time solvable by interior-point methods.

Chapter 3

Random SAT with limited independence

The Boolean satisfiability problem (SAT) is a foundational challenge in theoretical computer science and complexity theory. SAT asks whether there exists an assignment of Boolean values (true or false) to variables that satisfies a given Boolean formula. Typically formulas are expressed in conjunctive normal form (CNF), where the formula is represented as a conjunction of clauses with each clause being a disjunction of literals. SAT was one of the first problems proven to be NP-complete [51], a result that has had a profound impact on the field of computational complexity. Many key developments in modern complexity theory, including the Exponential Time Hypothesis [93], rely on the inherent difficulty of solving SAT efficiently.

SAT has become a central problem for both theoretical research and practical applications. On the theoretical side, SAT serves as a benchmark problem for understanding the complexity of NP-complete problems and the limitations of algorithmic approaches. In practice, SAT solvers are widely used in areas such as hardware verification [86], model checking [48], automated planning [49], and social choice [55, 129]. Despite its NP-hardness, modern SAT solvers have demonstrated remarkable efficiency in handling large instances with millions of variables. However, there remains a significant gap between the practical performance of these solvers and the predictions made by theoretical average-case analyses [21], motivating further study of the problem under random models.

The random SAT model provides a framework for analyzing the typical-case behavior of SAT. In the standard random r-SAT model, a formula is generated by randomly selecting clauses with exactly r literals from a set of variables. The key parameter of interest is the clause-to-variable ratio, which influences the density of constraints. As the ratio increases, the formula becomes more constrained, making it harder to satisfy. One of the central questions in the study of random SAT is determining the satisfiability threshold, a critical point at which the probability of the formula being satisfiable abruptly drops from near certainty to near impossibility.

This phase transition phenomenon is a striking feature of random SAT, akin to physical transitions observed in statistical mechanics, such as the nature of the glassy state [81]. At low clause-to-variable ratios, the formula tends to be satisfiable because the constraints are relatively sparse. However, as the ratio exceeds a critical threshold, the formula becomes heavily constrained, and satisfiability sharply declines. This abrupt transition marks the boundary between "easy" and "hard" SAT instances, where instances near the threshold are often the most challenging for solvers [117]. Understanding this phase transition is crucial, as it highlights the point at which SAT instances shift from being computationally manageable to intractable, prompting the need for sophisticated algorithmic techniques.

While the random r-SAT model captures essential aspects of typical-case complexity, it relies on the assumption of uniform sampling, where each variable has an equal likelihood of appearing in clauses. However, this uniformity does not reflect the complexities observed in real-world SAT instances, which often exhibit skewed distributions and correlations between variables [12]. In practice, certain variables tend to appear more frequently, and groups of variables may co-occur in clauses due to underlying structural dependencies.

To address these practical discrepancies, researchers have extended the standard random SAT model to include non-uniform distributions. These non-uniform models aim to better reflect empirical observations, where variable frequencies may follow a power-law distribution rather than being evenly distributed. Researchers [74–76, 123] introduced a non-uniform random SAT model that allows for varying probabilities of variable appearances, capturing differences in frequency across variables. However, this model still assumes independent sampling of variables, which limits its ability to account for the correlations and community structures often seen in practical SAT instances.

Empirical studies have shown that real-world SAT instances frequently display complex dependencies, where certain variables tend to appear together in clauses, forming patterns of co-occurrence that reflect deeper structural relationships [13]. The mutual independence assumption in existing non-uniform models does not capture these interactions, creating a gap between theoretical models and the observed properties of practical instances.

To bridge this gap, we propose relaxing the independence assumption while maintaining uniform marginal distributions. Rather than assuming full independence, we introduce a richer family of probability distributions that allow for controlled dependencies among variables, offering a more realistic framework for studying random SAT. In our model, we consider families of k-wise independent distributions. Each formula in this family is generated by selecting m clauses uniformly at random from all possible sets of three literals over n variables, ensuring that the selection of clauses exhibits k-wise independence.

Our analysis explores the impact of different levels of k-wise independence on the satisfiability of the generated formulas. We find that under pairwise independence ($k = 2$), formulas become almost always unsatisfiable when the number of clauses m grows as $\Omega(n^3)$, and when the degree of independence increases to 3, satisfiable instances can still be found at similar clause densities (Theorem 3.3.1). However, as the level of independence rises further ($k \geq 4$), the threshold tightens again, with unsatisfiability occurring at $m = O(n^2)$ (Theorem 3.3.2). Moreover, we establish a general relationship between the degree of k-wise independence and the required clause density for satisfiability. For any integer $k \geq 2$, we demonstrate that any instance with $m = O(n^{1-1/k})$ clauses is satisfiable with high probability (Theorem 3.3.3). We conjecture our estimation is tight (Conjecture 3.4.1).

3.1 Transition phase in random SAT

A r-CNF formula is a conjunction of disjunctive clauses, where each clause contains exactly r literals with different variables. In the standard random r-SAT model, an instance is generated uniformly at random from all possible r-CNF formulas with m clause over (at most) n variables. Figure 3.1 illustrates the probability of satisfiability for random 3-SAT instances across different values of n and m. It is evident that the probability of a random 3-SAT formula being satisfiable drops sharply from nearly 1 to almost 0 within a narrow range of the ratio m/n. Moreover, this transition becomes increasingly abrupt as the number of variables n grows, suggesting a more pronounced and sudden shift in satisfiability as the formula size increases. This phenomenon, known as *phase transition* in random r-SAT, has been well-documented through empirical studies [79, 84, 117]. The widely accepted *satisfiability conjecture* states that for each r, there exists a constant α_r such that random r-SAT instances with fewer than $(\alpha_r - o(1)) \cdot n$ clauses are almost always satisfiable, while those with more than $(\alpha_k + o(1)) \cdot n$ clauses are almost always unsatisfiable as n tends to infinity. The interval $[\alpha_r - o(1), \alpha_r + o(1)]$ is referred to as the *transition window*, and the constant α_r is known as the *satisfiability threshold*.

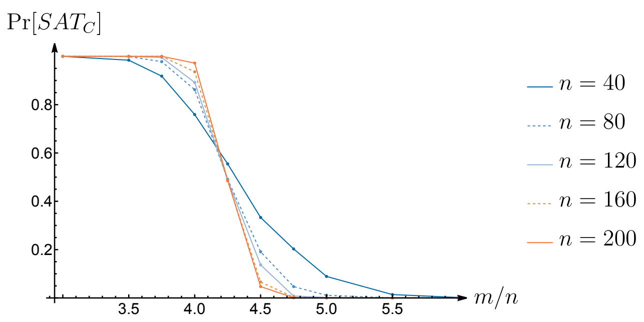

Figure 3.1: The empirical behavior of random 3-SAT instances across different number of variables n and ratio m/n. We can see between there is as shape transition between $m/n = 4.0$ and $m/n = 4.5$. This is within our predication since the experimental prediction of α_3 is 4.26 [52, 115]

The satisfiability conjecture has been proven for both $r = 2$ and $r \to \infty$. For random 2-SAT, it has been established that the threshold is $\alpha_2 = 1$ [114], with a transition window of $[n - \Theta(n^{1/3}), n + \Theta(n^{1/3})]$ [23]. When k is sufficiently large, the conjecture has also been confirmed [50, 57, 106], where the threshold α_k is approximately $2^r \log 2 - \frac{1}{2}(1+\log 2)$. However, the conjecture remains unresolved for smaller values of r. In this range, most of the research has focused on the case $r = 3$. We summarize the historical progress and state-of-the-art results on the lower bounds (Table 3.1) and the upper bounds (Table 3.2).

2/3	Chvátal and Reed [46]
1.63	Broder et al. [25]
3.003	Frieze and Suen [77]
3.145	Achlioptas [1]
3.26	Achlioptas and Sorkin[4]
3.52	Hajiaghayi and Sorkin [88] and Kaporis et al.[104]

Table 3.1: Lower bounds for α_3

5.191	Franco and Paull [72]
5.081	El Maftouhi and Fernandez de la Vega [112]
4.762	Kamath et al. [103]
4.643	Dubois and Boufkhand [58]
4.602	Kirousis et al. [106]
4.596	Janson et al. [97]
4.506	Dubois et al. [59]
4.49	Diáz et al. [56]

Table 3.2: Upper bounds for α_3

The role of mutual independence All the aforementioned works rely on the assumption of mutual independence among clauses, which is implied by the uniform distribution over all possible instances. To see the importance of mutual independence, we show a toy upper bound of α_r.

Proposition 3.1.1. *Let $\alpha = \ln 2/\ln\left(1+\frac{1}{2^r-1}\right)$, a random r-SAT instance C with n variables and $m > \alpha \cdot n$ clauses is satisfiable with probability of 0 when $n \to \infty$.*

Proof. The probability of any fixed assignment σ from $\{0,1\}^n$ is a solution to the instance is

$$\Pr[\sigma \text{ satisfies } C] = \left(1-\frac{1}{2^k}\right)^m,$$

since the assignment satisfies a single clause with probability of $1-\frac{1}{2^r}$ and there are m independent clauses. Let X be the number of solutions to the instance. We can calculate its expectation as

$$\mathsf{E}[X] = 2^n \cdot \left(1-\frac{1}{2^r}\right)^m.$$

By Markov's inequality, we have

$$\Pr[C \text{ is satisfiable}] = \Pr[X \geq 1] \geq \mathsf{E}[X] = \left(2\cdot\left(1-\frac{1}{2^r}\right)^{\frac{m}{n}}\right)^n.$$

As n goes to ∞, the probability that the instance is satisfiable approaches to 0 if $2\cdot\left(1-\frac{1}{2^r}\right)^{\frac{m}{n}} < 1$, which is implied by $m/n > \alpha = \ln 2/\ln\left(1+\frac{1}{2^r-1}\right)$ $\square$

In the proof, we use the first moment method, where estimating the expected number of solutions is a key component. This estimation is straightforward only in the case of mutual independence. Notably, this challenge is not limited to our toy proof but also arises in many previous works, such as [3, 50, 57, 112]. In these studies, estimating the expectation is unavoidable, as they rely on either the first moment method (for upper bounds) or the second moment method (for lower bounds), both of which require accurate expectation calculations.

3.2 k-clause independent random 3-SAT[1]

We present several definitions about our k-wise independent relaxation as a foundation before stating our results. We represent a 3-SAT instance as a multiset of clauses and denote the multiset as C. Given the number of variables n, let $\mathscr{T}(n)$ be the set of all possible clauses, i.e., all triplets with literals of different variables. We will usually omit the dependency of $\mathscr{T}$ on n, when the value of n is clear from the context.

Unlike in the case of mutual independence where the distribution is uniquely identified (denoted as $\mathscr{D}_{\text{Ind.}}$), we consider all distribution that satisfies the following definition.

Definition 3.2.1 (k-clause independent random 3-SAT(from [35]). *A distribution $\mathscr{D}$ for selecting a random 3-SAT instance is k-clause independent if the set of any fixed k clauses $S \subseteq C$ is distributed uniformly at random over all k-tuples of possible clauses, i.e.,*

$$\Pr_{C\sim\mathscr{D}}[c_i = t_i, \forall i \in [k]] = \Pr_{C\sim\mathscr{D}_{Ind.}}[c_i = t_i, \forall i \in [k]] = \prod_{i\in[k]} \Pr[c_i = t_i].$$

for every $c_1,\ldots,c_k \in C$ and $t_1,\ldots,t_k \in \mathscr{T}$. We denote the family of all k-clause independent distributions with m clauses over n variables as $\mathscr{F}_k(n,m)$.

By definition, the probability of any event A_S that depends only on a subset S of at most k clauses in the k-clause independent distribution $\mathscr{D}$ is the same as for $\mathscr{D}_{\text{Ind.}}$. I.e., the expectations of the indicator function $\mathbb{I}[A_S]$ are the same for $\mathscr{D}$ and $\mathscr{D}_{\text{Ind.}}$. Hence, by linearity of expectation, any statistic that involves at most k clauses must be the same for $\mathscr{D}$ and $\mathscr{D}_{\text{Ind.}}$.

We would like to understand the largest (resp. smallest) possible number of clauses for a random 3-SAT formula to be almost surely satisfiable (resp. unsatisfiable) under *any* k-clause independent distribution. Using $SAT(C)$ to denote the event that the SAT instance C is satisfiable, we define the following satisfiability "thresholds".

Definition 3.2.2 (Upper satisfiability threshold). *A function $m^*(n)$ is the asymptotic upper satisfiability threshold function $\mathrm{UST}_k(n)$ for k-clause independent distributions*

[1] In this section, we use definitions and notation that appears in our manuscript [35]. Introducing these technical concepts here is necessary to make the rest of this chapter understandable.

if and only if

$$\forall m(n) = \omega(m^*(n)), \forall \mathscr{D} \in \mathscr{F}_k(n, m(n)), \Pr_{C \sim \mathscr{D}}[SAT(C)] = o(1),$$

and

$$\forall m(n) = o(m^*(n)), \exists \mathscr{D} \in \mathscr{F}_k(n, m(n)), \Pr_{C \sim \mathscr{D}}[SAT(C)] = 1 - o(1).$$

Definition 3.2.3 (Lower satisfiability threshold)**.** *A function $m^*(n)$ is the asymptotic lower satisfiability threshold function $\mathsf{LST}(n)$ for k-clause independent distributions if and only if*

$$\forall m(n) = o(m^*(n)), \forall \mathscr{D} \in \mathscr{F}_k(n, m(n)), \Pr_{C \sim \mathscr{D}}[SAT(C)] = 1 - o(1),$$

and

$$\forall m(n) = \omega(m^*(n)), \exists \mathscr{D} \in \mathscr{F}_k(n, m(n)), \Pr_{C \sim \mathscr{D}}[SAT(C)] = o(1).$$

The definitions are slightly different from the definitions in our manuscript [35], since here we focus on the asymptotic behavior and ignore any constant in this introduction.

3.3 Our results and techniques

In this work, we explore the satisfiability thresholds of random 3-SAT formulas under the assumption of k-wise independence, extending the analysis beyond the classical mutually independent model. Our investigation focuses on both the upper and lower satisfiability thresholds, revealing new insights into the behavior of random 3-SAT instances when the independence assumption is relaxed.

We begin by examining small degrees of independence, specifically, $k = 2$ and $k = 3$. For these cases, we establish the tight upper satisfiability thresholds.

Theorem 3.3.1. *$\mathsf{UST}_2(n) = \mathsf{UST}_3(n) = \Theta(n^3)$ for random 3-SAT.*

Theorem 3.3.1 demonstrates that a formula requires nearly all possible clauses to ensure unsatisfiability under 2- or 3-wise independence over clauses. A key technical challenge in proving this result lies in constructing a 3-clause independent distribution with many clauses where the formula remains satisfiable with high probability. We overcome this by employing a construction based on formulas where each clause has either one or three literal(s) satisfied by a *planted* truth assignment. We call such formulas 3-XOR formulas. Specifically, our construction is a distribution that samples an instance in two phases. In the first phase, we randomly decide whether the instance should be satisfiable or not. If it should be satisfiable, we will draw a 3-XOR formula based on a randomly planted truth assignment in the second phase. Otherwise, the instance will be drawn from sub-distributions where the structures that cannot appear in the 3-XOR formulas occur frequently.

Our most technically involved result shows the upper bound of the upper satisfiability threshold in the case $k = 4$.

Theorem 3.3.2. $UST_4(n) = O(n^2)$ *for random* 3-SAT.

Theorem 3.3.2 requires a more sophisticated analysis. We introduce a bipartite multigraph representation and use statistical measures. In our bipartite multigraph, each clause corresponds to three disjoint egdes. We focus on the number of $K_{2,2}$ subgraphs in our bipartite multigraph, which always involve four edges supported by four different clauses. By our definition of 4-clause independent distributions, the expected number of $K_{2,2}$ is the same as in the mutual independent case. We lower bound the number considering both cases when the 3-SAT instance is satisfiable or not by extremal combinatorial arguments. The general lower bound of the number of $K_{2,2}$ subgraphs is slightly smaller than its expectation. However, when the graph corresponds to a satisfiable instance, the lower bound is much larger than the expectation. Therefore, the probability cannot be large in order to maintain the expectation guaranteed by a 4-wise independent distribution.

On the lower satisfiability threshold side, we prove a much more general result.

Theorem 3.3.3. *For any integer* $k \geq 2$*,* $LST_k(n) = \Omega(n^{1-\frac{1}{k}})$ *for random* 3-SAT.

The analysis of LST is easier. For any k-clause independent distribution, we consider the uniform hypergraph representing how the variables of a random instance generated from this distribution are placed into clauses and show that this hypergraph is unlikely to contain Berge-cycles with high probability.

3.4 Discussion and open problems

We believe that our lower bounds in Theorem 3.3.3 are tight.

Conjecture 3.4.1. *For any integer* $k \geq 2$*,* $LST_k(n) = \Theta(n^{1-\frac{1}{k}})$ *for random* 3-SAT.

Our reasoning here is as follows: A 3-SAT instance C is naturally represented by a 3-uniform hypergraph G, where the variables of the instance serve as the nodes. If there exists a dense subgraph H within G, i.e. with $|E(H)|/|V(H)| \geq 6$, we can use a similar argument as in the proof of Proposition 3.1.1, to bound the probability that C is satisfiable, given that the literals are chosen uniformly at random. The presence of a dense subgraph H would typically violate the k-wise independence assumption only if there exists a Berge-cycle of size at most k (due to our arguments proving Theorem 3.3.3). Fortunately, constructions of dense hypergraphs without small cycles have been well-established (e.g., [61]). Thus, if we can carefully balance the statistics for all structures involving at most k variables, we can successfully achieve the desired construction. This does not look hard since the constructions of H require only a constant number of nodes, meaning that the statistics are very close to the requirements.

In fact, we have verified that such a construction is feasible for the case $k = 2$. We present this result without proof.

Theorem 3.4.2. $LST_2(n) = \Theta(\sqrt{n})$ *for* 3-SAT.

The advantage in this case is that we only need to consider four distinct structures that can occur between two clauses: (1) the clauses are disjoint, (2) they share exactly one variable, (3) they share exactly two variables, or (4) they share all the three variables. Our construction is analogous to the one used for proving Theorem 3.3.1. To ensure that the instance is unsatisfiable by embedding a dense subgraph, structures of type (3) and (4) appear slightly less frequently than required by a 2-clause independent distribution. However, sunflower structures, which can only include pairs on clauses of type (3) or (4), can be drawn with a low probability, which suffices to balance the overall statistics.

Another open problem concerns the behavior of UST as the degree of independence k increases. While we have shown that LST approaches linear growth with increasing k, the question remains: what is the behavior of UST in this context? It is important to note that statistics involving only k-wise independence seem to serve as a polynomial-size proofs of unsatisfiability. Consequently, it is unlikely that $\mathtt{UST}_k$ can decrease to linear, given the $\Omega(n^{3/2-\varepsilon})$ lower bound for sub-exponential size unsatisfiability proofs [17]. In fact, the best-known algorithm that generates efficiently computable proofs of unsatisfiability only works for random 3-SAT instances with $m = \Omega(n^2/\log n)$ clauses [16]. Thus, it would be a breakthrough to prove a bound for $\mathtt{UST}_k$ below $\Omega(n^2/\log n)$ for some constant k.

Additionally, our setting can be extended to general random r-SAT. The techniques we developed appear to be adaptable, particularly for random 2-SAT, where we might be able to achieve a complete characterization for both LST and UST.

Chapter 4

Revenue gap for selling a divisible item

Revenue maximization is a primary objective in algorithmic mechanism design, particularly in settings where platforms, sellers, or institutions interact with self-interested agents [1]. The core challenge lies in creating mechanisms incentivizing agents to act truthfully while optimizing the seller's or platform's revenue. These mechanisms have broad applications, notably in auction design, online advertising, and digital marketplaces, where effective pricing and allocation strategies can directly impact overall revenue.

A foundational result in this area is Myerson's auction, introduced by Roger Myerson in 1981 [119]. This auction mechanism for single-item sales with independent, private valuations achieves optimal revenue and incentive compatibility, ensuring that bidders are motivated to reveal their true valuations. By carefully structuring payment rules, Myerson's auction maximizes revenue while making truth-telling a dominant strategy for participants. This result has significantly influenced subsequent research, highlighting conditions under which optimal revenue can be achieved without sacrificing truthful bidding.

The optimality of Myerson's auction is well-established in single-parameter environments, where each agent has a single private valuation. In these cases, the mechanism leverages simple, well-defined reserve pricing to extract maximum revenue effectively. However, extending these principles to multi-parameter environments—where agents have complex, multi-dimensional preferences across multiple items or attributes—introduces significant challenges. In multi-parameter settings, agents' valuations are influenced by combinations of items, creating exponentially larger strategy spaces and intricate interdependencies [128]. This complexity makes it difficult to design mechanisms that maintain both incentive compatibility and revenue optimality, often requiring new techniques that go beyond the straightforward application of Myerson's framework.

[1] We use the terms buyers and bidders to refer such agents depending on the context. For example, the term bidders is often used when the mechanism is an auction.

As algorithmic mechanism design has evolved, classic principles have been adapted to meet the demands of digital markets, where interactions occur at high speeds, involve large numbers of participants, and are often automated. In digital platforms—such as online marketplaces and ad auctions—mechanisms must be both revenue-maximizing and computationally efficient, able to handle vast numbers of bids and participants in real time. These environments introduce new challenges, such as balancing computational efficiency with revenue outcomes, managing high-frequency interactions, and addressing large-scale strategic behavior. Consequently, traditional approaches have been extended to accommodate dynamic pricing, personalized recommendations, and continuous resource allocation, especially as digital markets frequently involve divisible goods like communication bandwidth and cloud computing resources.

To address the complexities inherent in digital markets, recent research has investigated trade-offs between "simple" and "optimal" mechanisms, exploring how much revenue might be sacrificed in exchange for simplicity and practicality [7, 90, 98–100]. For instance, posted pricing mechanisms, where prices are set in advance and agents select items accordingly, are easier to implement in digital marketplaces compared to optimal auctions. Although these simpler mechanisms may yield slightly lower revenue, they offer significant advantages in terms of computational efficiency and scalability.

Building on these foundations, this chapter introduces a new setting involving the sale of a perfectly divisible item, such as communication bandwidth or cloud computing resources. Our model is also quite general as a mathematical framework, encompassing several variants of the single indivisible item setting as special cases. Within this context of divisible items, we examine the revenue gap between the optimal mechanism and linear posted pricing mechanisms.

The main challenge lies in the lack of detailed knowledge about the optimal mechanism in this multi-parameter environment. To address this challenge, we leverage the technique of *ex-ante relaxation*, a powerful tool initially used for analyzing combinatorial auctions in [6]. Subsequently, Alaei et al. adapted this technique to improve the analysis of the revenue gap between Myerson's auction and anonymous pricing in the single indivisible item setting [7]. It has also been extended to a k-unit setting, where the seller offers k identical copies to unit-demand buyers [100]. By applying a black-box reduction to the result in [7], we demonstrate that the revenue gap depends solely on a natural parameter of the agents' possible valuations (Theorem 4.3.4). Through a detailed and involved analysis, we further refine this dependency, reducing it to a logarithmic factor, albeit at the cost of introducing a logarithmic dependence on the number of buyers (Theorem 4.3.5).

Additionally, we establish a tight revenue gap for the single-buyer case in our new setting (Theorem 4.3.3). It is worth noting that this result highlights the increased complexity of the divisible item setting, as anonymous pricing is the optimal mechanism in the single-buyer scenario for the indivisible item setting. Our result in Theorem 4.3.3 shows that this does not hold for divisible items.

4.1 Selling a divisible item

Consider the scenario with a perfectly divisible item that is to be sold to n agents. A valuation function $v_i : [0,1] \to \mathbb{R}_{\geq 0}$ describes that agent i gets value $v_i(x)$ if exactly a fraction of x of the item is allocated to her. It is a non-decreasing concave function independently drawn according to the publicly-known probability distribution $\mathbf{F}_i$ of agent i.

Selling the item is done through a mechanism $M = (\mathbf{x}, \mathbf{p})$, consisting of an allocation rule $\mathbf{x} = \{x_i\}_{i\in[n]}$ indicating the fraction of the item each agent gets and a payment rule $\mathbf{p} = \{p_i\}_{i\in[n]}$ indicating how much each agent should pay. Both $\mathbf{x}$ and $\mathbf{p}$ depend on the distribution of valuation functions and the bids from the agents. Denoting by $\mathbf{F}$ the joint probability distribution, and by $\mathbf{b}$ the bid vector of the agents, we use $x_i(\mathbf{F}, \mathbf{b})$ and $p_i(\mathbf{F}, \mathbf{b})$ to refer to the fraction allocated to agent i and the payment to the mechanism. Agents are utility maximizers, aiming to maximize the difference between the value they get and the required payment.

In our setting, the counterpart to anonymous pricing in the classic single indivisible item scenario is linear posted pricing. Our work specifically focuses on analyzing linear posted pricing.

Linear posted pricing

Linear pricing is a very simple mechanism which the payment from each agent is proportional to the allocated fraction to her. Specifically, in such a mechanism, the seller posts the price $p(\mathbf{F})$ for the whole item based on the joint distribution $\mathbf{F}$. Then the buyers come sequentially, buy a part of the item, and pay $p(\mathbf{F})$ times the fraction of the item they get. The mechanism terminates when the good is sold out or when all buyers have considered buying without exhausting the item. Ideally, agent i would like to buy a fraction of $y_i^*(v_i, p(\mathbf{F})) = {v'}_i^{-1}(p(\mathbf{F}))$ of the item, since v_i' is a non-increasing function and the utility can be written as $\int_0^{y_i^*}(v_i'(z) - p(\mathbf{F}))dz$.

However, the supply is limited to one. Thus, the amount that agent i is able to buy depends on the order in which the agents act. Fixing the ordering π, we use the notation $j \prec_\pi i$ to indicate that agent j comes before agent i in the order π. The fraction of the item agent i is willing to and able to buy is

$$y_i(v_i, p(\mathbf{F}), \pi) = \begin{cases} y_i^*(v_i, p(\mathbf{F})), & \sum_{j\preceq_\pi i} y_j^*(v_j, p(\mathbf{F})) \leq 1 \\ 1 - \sum_{j\prec_\pi i} y_j^*(v_j, p(\mathbf{F})), & \sum_{j\prec_\pi i} y_j^*(v_j, p(\mathbf{F})) \leq 1 < \sum_{j\preceq_\pi i} y_j^*(v_j, p(\mathbf{F})) \\ 0, & \sum_{j\prec_\pi i} y_j^*(v_j, p(\mathbf{F})) > 1 \end{cases}.$$

We use $\pi(\mathbf{F})$ later in the allocation rules and payment rules since the seller may decide the order based on the joint distribution $\mathbf{F}$. Since the price and the order only depend on the joint distribution, there is no reason for agents to report anything besides their true valuation. Given this, the seller will allocate the item using the reported valuations as the true valuations. Formally, the allocation rule is defined as

$$x_i(\mathbf{F}, \mathbf{b}) = y_i(b_i, p(\mathbf{F}), \pi(\mathbf{F})),$$

while the payment rule is

$$p_i(\mathbf{F},\mathbf{b}) = p(\mathbf{F}) \cdot y_i(b_i, p(\mathbf{F}), \pi(\mathbf{F})).$$

The seller's objective is to maximize the expected revenue, i.e., the expectation of total payment

$$\begin{aligned}
\mathsf{E}_{\mathbf{v}\sim\mathbf{F}}[\sum_{i\in[n]} p_i(\mathbf{F},\mathbf{b})] &= \mathsf{E}_{\mathbf{v}\sim\mathbf{F}}[p(\mathbf{F}) \cdot \sum_{i\in[n]} y_i(b_i, p(\mathbf{F}), \pi(\mathbf{F}))] \\
&= p(\mathbf{F}) \cdot \mathsf{E}_{\mathbf{v}\sim\mathbf{F}}[\sum_{i\in[n]} y_i(b_i, p(\mathbf{F}), \pi(\mathbf{F}))] \\
&= p(\mathbf{F}) \cdot \mathsf{E}_{\mathbf{v}\sim\mathbf{F}}[\sum_{i\in[n]} y_i(v_i, p(\mathbf{F}), \pi(\mathbf{F}))] \\
&= p(\mathbf{F}) \cdot \mathsf{E}_{\mathbf{v}\sim\mathbf{F}}[\min\{1, \sum_{i\in[n]} y_i^*(v_i, p(\mathbf{F}))\}].
\end{aligned}$$

The first equation is by definition of the payment rule. The second equation is due to the linearity of expectation. The third equation holds since agents have no incentive to report anything besides their true valuation in this mechanism as discussed above. The last equation is by definition of y_i. Thus, the best linear pricing satisfies $p(\mathbf{F}) \in \arg\max_{p\in\mathbb{R}_{\geq 0}} p \cdot \mathsf{E}_{\mathbf{v}\sim\mathbf{F}}[\min\{1, \sum_{i\in[n]} y_i^*(v_i, p)]$. Notice that the revenue from the best linear pricing does not depend on the ordering.

4.2 Revenue gap

Revenue gap is a quantity describing how closely a mechanism approximates the revenue-optimal mechanism. Let $R_M(\mathbf{F})$ be the revenue achieved by a mechanism M given the joint valuation distribution $\mathbf{F}$, and OPT be the revenue-optimal mechanism. The revenue gap of a mechanism M is then $\sup_{\mathbf{F}\in\mathscr{F}} \frac{R_{OPT}(\mathbf{F})}{R_M(\mathbf{F})}$, where $\mathscr{F}$ is a set of all valid valuation distributions.

Estimating the revenue gap is a well-studied problem in single-parameter settings. In the classical indivisible single-item setting, it is known that the revenue gap of anonymous pricing is 2.62 [98], where anonymous pricing refers to setting a common price to the item for all potential buyers. For other mechanisms (e.g., discriminating pricing, second pricing auction with reserve, etc.), the revenue gap remains not tight, but it is known to be constant [99]. In the indivisible k-unit setting, Jin et al. [100] have recently shown that the revenue gap of anonymous pricing is $\Theta(\log k)$ if the buyers are unit-demand and the items are identical.

However, the setting we consider is more general and multi-parameter. The challenge here is that we do not know the revenue-optimal mechanism for selling a perfectly divisible item. Fortunately, the so-called ex-ante relaxation [7] provides a useful tool allowing us to analyze the revenue gap without knowing the details of the revenue-optimal mechanism.

To explain the ex-ante relaxation, we need to get back into the indivisible item setting, where each buyer $i \in [n]$ draws valuation v_i to the item from a publicly-known valuation distribution $\mathbf{F}_i$ (we denote its cumulative distribution function as F_i). Note that the valuation v_i is simply a scalar, unlike in our setting where each buyer draws a valuation function. We will introduce anonymous pricing and Myerson's auction to explain why ex-ante relaxation is a useful tool.

Anonymous pricing. In the single-indivisible item setting, an anonymous pricing mechanism selects a price p in advance, potentially based on the valuation distributions. If there is at least one buyer whose valuation exceeds p, the seller will sell the item to one such buyer chosen arbitrarily. The revenue obtained from anonymous pricing with price p can be expressed as

$$R_{AP(p)}(\mathbf{F}) = p \cdot \Pr_{\mathbf{v}\sim\mathbf{F}}[\exists i \in [n] \text{ s.t. } v_i > p] = p \cdot (1 - \prod_{i\in[n]} F_i(p)).$$

We use $R_{AP}(\mathbf{F})$ to denote the revenue achieved by the best possible anonymous pricing for the instance $\mathbf{F}$.

Myerson's auction. Myerson's auction is well-known as the revenue-optimal mechanism in the single-parameter environment. At the core of Myerson's auction is the concept of adjusting bidders' valuations using a virtual valuation function, which accounts for the underlying valuation distributions. The virtual valuation function is defined as

$$\phi_i(v) = 1 - \frac{v - F_i(v)}{f_i(v)},$$

where f_i is the probability density function of buyer i's valuation distribution. Notice that people normally assume that ϕ_i is monotone non-decreasing on v, this property is called *regularity*. Valuation distributions satisfying regularity are called regular distributions.

The auction allocates the item to the bidder with the highest non-negative virtual valuation $\phi_i(v_i)$. This requires running a second-price auction using the virtual bids, ensuring dominant strategy incentive compatibility (i.e., it is always a best strategy for each bidder to truthfully report their valuation as bid). The revenue from Myerson's auction can be expressed by the expected social welfare taking into the virtual valuation, i.e.,

$$R_{OPT}(\mathbf{F}) = \mathop{\mathsf{E}}_{\mathbf{v}\sim\mathbf{F}}\left[\max_{i\in[n]} \phi_i(v_i^+)\right].$$

In general, there is no closed-form expression for this revenue due to the complexity introduced by the virtual valuation function. This makes analyzing the optimal revenue quite challenging, often requiring sophisticated methods and intricate arguments, such as those found in [98]. By contrast, the ex-ante relaxation provides a simpler approach that yields a good approximation, i.e. it upper-bounds the revenue gap by $e \approx 2.72$ [7] while the tight revenue gap is 2.62 by [98].

Ex-ante relaxation. The intuition behind the ex-ante relaxation is that any mechanism, including the revenue-optimal one, effectively results in an "average" allocation based on the valuation distribution. This allocation is specified by the probabilities of selling the item to the buyers. Given the valuation distribution, these selling probabilities determine the corresponding selling prices for the buyers. By optimizing the selling probabilities, we can compute an upper bound on the optimal revenue, expressed as

$$R_{OPT}(\mathbf{F}) \le R_{EX}(\mathbf{F}) = \max \sum_{i\in[n]} q_i \cdot F_i^{-1}(1-q_i)$$

Thus, the revenue gap is bounded by $\sup_{\mathbf{F}\in\mathscr{F}} \frac{R_{EX}(\mathbf{F})}{R_{AP}(\mathbf{F})}$. When we impose the condition $R_{AP}(\mathbf{F}) \le 1$, the revenue gap is characterized by $\sup_{\mathbf{F}\in\mathscr{F}} R_{EX}(\mathbf{F})$. The full ex-ante relaxation problem can then be formulated as follows:

$$\begin{aligned} \text{maximize} & \sum_{i\in[n]} q_i \cdot F_i^{-1}(1-q_i) \\ \text{subject to} & \sum_{i\in[n]} q_i \le 1, \forall i \in [n] \\ & q_i \ge 0 \\ & p \cdot \left(1 - \prod_{i\in[n]} F_i(p)\right) \le 1, \forall p \ge 1 \end{aligned} \tag{4.1}$$

The first two constraints ensure that $\mathbf{q}$ forms a valid probability distribution. The third constraint limits the revenue achieved by the optimal anonymous pricing. It is important to note that the ex-ante relaxation does not require any specific information about the optimal mechanism, making it particularly useful since the details of the optimal mechanism are unknown in our setting.

4.3 Our techniques and results

The first contribution of this work is the introduction of a new parameter, the *initial marginal over total value* (IMOTV for short), which plays a central role in determining the revenue gap in the divisible item setting. The following example illustrates that the revenue gap depends on the newly defined parameter. In fact, this example provides a lower bound for the revenue gap between the optimal mechanism and linear pricing.

Proposition 4.3.1. *Let $\kappa \ge 1$, define the valuation function:*

$$v_\kappa(x) = \begin{cases} \kappa \cdot x & x \in [0, \frac{1}{\kappa\cdot\rho}] \\ 1 + \frac{\ln x}{\rho} & x \in [\frac{1}{\kappa\cdot\rho}, 1] \end{cases},$$

where we choose $\rho \ge 1$ as the root of the equation $\rho - \ln\rho = 1 + \ln\kappa$. For any $\kappa \ge 1$, we can construct an instance with a single buyer whose deterministic valuation function is v_κ. In this case, the revenue gap is at least $\Omega(\ln\kappa)$.

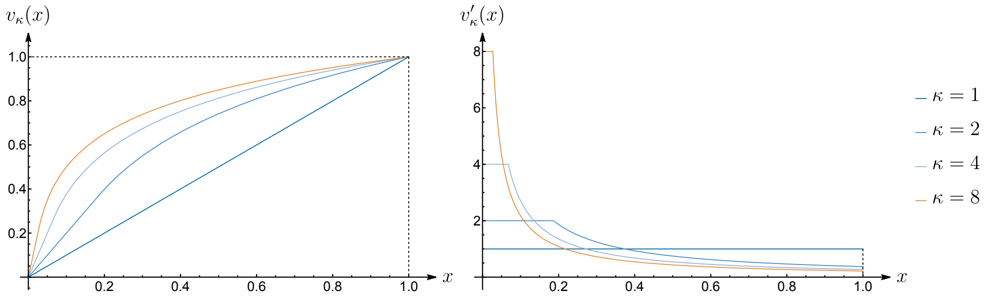

Figure 4.1: The figure shows how v_κ changes when κ increases. It is obvious that when $\kappa = 1$, it is a linear valuation function. When κ increases, the valuation becomes more concave (see the left subfigure). And by looking at the derivative, it is straightforward that every linear pricing yields the same revenue when κ is fixed (see the right subfigure).

In this example, the revenue-optimal mechanism is achieved by setting the pricing function $p(x) = v_\kappa(x)$, yielding an optimal revenue of $v_\kappa(1) = 1$. For any linear pricing with $p \geq \kappa$, the revenue is zero. When $p = \kappa$, the revenue obtained is $1/\rho$. For prices in the range $1/\rho \leq p < \kappa$, the revenue remains $1/\rho$. For any price $p < 1/\rho$, the revenue is simply p, which does not exceed $1/\rho$. Thus, the revenue gap is at least $\rho = \Omega(\ln \kappa)$.

This construction reveals that the parameter κ, which equals to the ratio of the derivative of v_κ at point 0 over the value of v_κ at point 1, governs the revenue gap. To generalize this concept for instances with multiple buyers and random valuations, we formally define the notion of IMOTV as follows.

Definition 4.3.2 (Initial marginal over total value (IMOTV))**.** *Given a n-agent instance where each agent $i \in [n]$ draws their valuation function v_i from distribution $\mathbf{F}_i$, the initial marginal over total value (IMOTV for short) κ is defined as*

$$\kappa = \max_{i\in[n]} \max_{v_i \in supp(\mathbf{F}_i)} \frac{v_i'(0)}{v_i(1)}.$$

After determining the importance of IMOTV, the first challenge to us upper bound the optimal revenue. As we introduce in the last section, the ex-ante relaxation is our natural choice. In our case, the ex-ante relaxation can be formulated as follows:

$$\begin{aligned} \text{maximize} \sum_{i\in[n]} \int_0^1 & q_i(x) \cdot F_{i,x}^{-1}(1 - q_i(x))dx \qquad (4.2) \\ \text{subject to} \sum_{i\in[n]} \int_0^1 & q_i(x)dx \leq 1 \\ & q_i(x) \in [0,1], \forall i \in [n], x \in [0,1] \\ & p \cdot \mathsf{E}[\min\{1, \sum_{i\in[n]} y_i^*(v_i, p)\}] \leq 1, \forall p \geq 1 \end{aligned}$$

The formulation of this mathematical program is closely related to (4.1). Conceptually, we can think of each point (i,x) as representing an independent agent. We use $F_{i,x}$ to denote the cumulative distribution function of the derivative of the valuation function $v_i'(x)$. The first constraint sets a limit on the expected quantity sold, ensuring that the total allocation does not exceed the available supply. The second constraint guarantees that selling probability $q_i(x)$ is a valid probability. The final constraint restricts the achievable revenue to that of the optimal linear pricing. However, Program (4.2) only provides an upper bound on the optimal revenue when each agent i consistently has non-negative utility across all points $x \in [0,1]$. This condition does not always hold. To resolve this issue, we introduce *shortsighted* agents who seek to maximize their utility while ensuring it remains non-negative at every point in the fraction of the item they get. Figure 4.2 illustrates a counterexample where the condition is violated and how shortsighted agents behave. Thanks to the concavity of the valuation functions, this additional assumption does not impact the revenue achieved by any linear pricing.

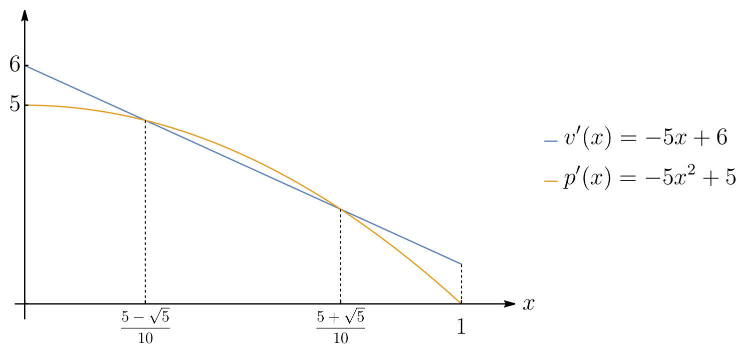

Figure 4.2: The figure illustrates the case that the seller sets the pricing function as $p(x) = -\frac{5}{3}x^2 + 5x$ and the buyer holds a deterministic valuation function $v(x) = -\frac{5}{2}x^2 + 6x$. If the buyer maximizes their utility globally, the utility of $1/6$ is achieved by taking the whole item. However, the points $x \in (\frac{5-\sqrt{5}}{10}, \frac{5+\sqrt{5}}{10})$ will contribute negatively to the utility. Thus, a shortsighted agent will buy up to $\frac{5-\sqrt{5}}{10}$ and will get a lower utility.

Under the assumption of shortsighted agent, the example in Proposition 4.3.1 represents the worst-case scenario for the revenue gap (Theorem 4.3.3).

Theorem 4.3.3. *The revenue gap of linear posted pricing for a single agent with random concave valuations of IMOTV at most κ is $\Theta(\ln \kappa)$.*

In this case, Program (4.2) can be simplified as

$$\begin{aligned} &\text{maximize} \int_0^1 q(x) \cdot F_x^{-1}(1-q(x))dx \\ &\text{subject to} \int_0^1 q(x)dx \leq 1 \end{aligned} \tag{4.3}$$

$$q(x) \in [0,1], \forall x \in [0,1]$$
$$p \cdot \mathsf{E}[y^*(v,p)] \leq 1, \forall p \geq 1$$

The core idea of our proof involves reducing the instance with random valuations to an equivalent instance with a deterministic valuation, specifically defined as:

$$v(x) = \int_0^x q^*(z) \cdot F_z^{-1}(1-q(x))dz,$$

where $q^*(x)$ is the optimal solution to Program (4.3). The function $v(x)$ retains concavity and is a valid valuation function. Moreover, the revenue gap in this newly constructed instance is guaranteed to be larger than that of the original instance. With these observations in place, Theorem 4.3.3 follows by demonstrating that IMOTV of the new instance is polynomially related to the IMOTV of the original instance.

Our first result on the revenue gap for the case of multiple agents (Theorem 4.3.4) is achieved by a black-boxed reduction to the result in [7]. Since that result [7] assumes regularity, we also need to add regularity assumption to $F_{i,0}$ for each agent $i \in [n]$ in Program (4.2). This result demonstrates that the revenue gap is independent on the number of agents.

Theorem 4.3.4. *The revenue gap of linear posted pricing for multiple agents with random concave valuations of IMOTV at most κ and with derivatives that follow regular probability distributions at* 0 *is at most* $O(\kappa^2)$.

To strengthen the result of Theorem 4.3.4, we extend the regularity assumption to every point and delve deeper into the analysis of Program (4.2) using the obvious property that the selling probability q_i is monotone non-increasing. Drawing inspiration from the techniques in [100], we establish Theorem 4.3.5. However, adapting these methods is challenging. A key difficulty was in attempting to bound the maximum value of $F_{i,x}^{-1}(1-q_i(x))$ in the optimal solution by a polynomial of κ. Despite our efforts, we were unable to achieve this, necessitating the introduction of a dependence on n.

Theorem 4.3.5. *The revenue gap of linear posted pricing for n agents with random concave valuations of IMOTV at most κ and with derivatives that follow regular probability distributions is at most* $O(\ln \kappa + \ln n)$.

4.4 Discussion and open problems

In this work, we investigate the effectiveness of linear pricing for selling a perfectly divisible item. Our findings open up several avenues for further exploration. One of the key open questions is determining the exact revenue gap for linear pricing. In narrowing the gap between our upper bound of $O\left(\max\left\{\ln \kappa + \ln n, \kappa^2\right\}\right)$ and our lower bound of $\Omega(\ln \kappa)$, we conjecture the tight revenue gap is $\Theta(\ln \kappa)$. Additionally, the assumption of shortsighted agents might not be appealing to all researchers, making it worthwhile to explore how to relax it.

The broader area of mechanism design in this new setting also offers promising directions. Notably, the observed revenue gap for linear pricing primarily arises due to the concavity of valuation functions. It would be worthwhile to investigate whether a concave anonymous pricing function could achieve a constant-factor revenue approximation. The ultimate goal in revenue maximization is to characterize the revenue-optimal mechanism in this setting. However, from a computational perspective, dealing with continuous valuation functions is a big challenging. A natural follow-up problem is whether discretizing the valuations would allow for the computation of the revenue-optimal mechanism in polynomial time.

An extra open problem arising from an observation regarding the proof of the $O(\kappa^2)$ result leads us back to the indivisible item setting. In this proof, each agent's valuation distribution only needs to be regular at a single point, which reveals an interesting outcome in the context of selling a single indivisible item. Specifically, consider a scenario in which a seller offers an indivisible item to n agents. Each agent $i \in [n]$ has a private valuation for the item, drawn from a publicly known distribution F_i. We define the range ratio r of this instance as

$$r = \max_{i \in [n]} \frac{\sup \operatorname{supp}(F_i)}{\inf \operatorname{supp}(F_i)}.$$

Our result immediately establishes an $O(r^2)$ revenue gap between Myerson's auction and the anonymous pricing mechanism in the single indivisible item setting *without* the regularity assumption. It is true that $O(r^2)$, or even $O(r)$, is a trivial bound, but things would be interesting if we could go below $O(r)$. Notably, this new revenue gap does not contradict the $\Omega(\ln n)$ or $\Omega(n)$ lower bound of the revenue gap in [39] and [7], as in their constructions r is at least n^2. In other words, when the range ratio r is dominated by the number of agents n, the revenue gap depends primarily on r. This observation raises an intriguing question: can we improve the upper bound to sublinear? We currently conjecture that the revenue gap between Myerson's auction and the anonymous pricing mechanism is at most $O(\ln r)$.

[2] A minor adjustment is required for the construction in [7] to guarantee that r is well-defined.

Part II

Publications

Chapter 5

Better Approximate Pure Nash Equilibria in Cut Games

Cut games are among the most fundamental strategic games in algorithmic game theory. It is well-known that computing an exact pure Nash equilibrium in these games is PLS-hard, so research has focused on computing approximate equilibria. We present a polynomial-time algorithm that computes 2.7371-approximate pure Nash equilibria in cut games. This is the first improvement to the previously best-known bound of 3, due to the work of Bhalgat, Chakraborty, and Khanna from EC 2010. Our algorithm is based on a general recipe proposed by Caragiannis, Fanelli, Gravin, and Skopalik from FOCS 2011 and applied on several potential games since then. The first novelty of our work is the introduction of a phase that can identify subsets of players who can simultaneously improve their utilities considerably. This is done via semidefinite programming and randomized rounding. In particular, a negative objective value to the semidefinite program guarantees that no such considerable improvement is possible for a given set of players. Otherwise, randomized rounding of the SDP solution is used to identify a set of players who can simultaneously improve their strategies considerably and allows the algorithm to make progress. The way rounding is performed is another important novelty of our work. Here, we exploit an idea that dates back to a paper by Feige and Goemans from 1995, but we take it to an extreme that has not been analyzed before.

5.1 Introduction

Understanding the computational aspects of equilibria is a key goal of algorithmic game theory. In this direction, we consider the fundamental class of cut games. In a cut game, each node of an edge-weighted graph is controlled by a distinct player. The players aim to build a cut of the graph in a non-cooperative way. Each player has two strategies, to put the node she controls in one of the two sides of the cut, and aims to maximize her utility, defined as the total weight of edges in the cut that are incident to the node the player controls. A pure Nash equilibrium (or, simply, an equilibrium) is a state of the game in which no player has any incentive to unilaterally change her strategy in order to improve her utility.

Cut games are potential games. The total weight of the edges in the cut is a potential function, with the following remarkable property: for every two states that differ in the strategy of a single player, the difference in the value of the potential function is equal to the difference in the utility of the deviating player. Thus, a sequence of best-response moves, consisting of players who change their strategies and improve their utilities, is guaranteed to lead to a state that locally maximizes the potential function and is, thus, an equilibrium. So, finding an equilibrium is equivalent to computing a locally-maximum cut in a graph, i.e., a cut that cannot be improved by moving a single node from one side of the cut to the other.

Unfortunately, computing a locally-maximum cut in a graph is a PLS-hard problem [135]. In light of this, approximate pure Nash equilibria seem to be a reasonable compromise for cut games; in general, a ρ-approximate pure Nash equilibrium is a state of a strategic game in which no player deviation can increase her utility by a factor of more than ρ. Among other results, Bhalgat et al. [20] present a polynomial-time algorithm for computing a 3-approximate equilibrium in cut games.

At first glance, two algorithmic ideas seem relevant for tackling the challenge of computing approximate pure Nash equilibria. The first one is to follow a sequence of deviations by players who improve their utility by a factor of more than ρ. The existence of a potential function guarantees that this process will eventually converge to a ρ-approximate equilibrium. Unfortunately, such a sequence can be exponentially long, as Bhalgat et al. [20] have shown specifically for cut games (and for small approximation factors). The second one is to exploit an approximation algorithm for the problem of maximizing the potential function. For cut games, this could involve the celebrated algorithm of Goemans and Williamson [82] for MAX-CUT or, more generally, excellent approximations of local maxima using the techniques of Orlin et al. [124]. Unfortunately, approximations of the potential function and approximate equilibria are unrelated notions. So, the algorithm in [20] exploits the structure of cut games to define restricted subgames in which sequences of player deviations are applied separately. This approach leads to 3-approximate equilibria in cut games and is also applicable to the broader class of constraint satisfaction games.

Our results and techniques

In this paper, we present the first improvement to the result of Bhalgat et al. [20], showing how to compute efficiently a 2.7371-approximate equilibrium in a cut game. Our result follows a general algorithmic recipe proposed by Caragiannis et al. [30] for congestion games. Applications of this recipe to constraint satisfaction games [33] has improved many of the approximation bounds presented in [20] but not the one for cut games (even though it has rediscovered a 3-approximation). Unlike [20], the algorithms in [30, 33] identify polynomially-long sequences of coordinated player deviations that eventually lead to an approximate equilibrium. Our improvement is possible by introducing a *global move step* in which a set of players change their strategies simultaneously. Such a set of players is identified by solving a semidefinite program and rounding its solution. To the best of our knowledge, this is the first application of semidefinite programming to the computation of approximate pure Nash equilibria. Essentially, our paper blends the two algorithmic approaches mentioned in the previous paragraph for the first time.

The general structure of the algorithm in [33] is as follows. Players are classified into blocks, so that players within the same block are polynomially-related in terms of their maximum utility. Then, a set of phases is executed. In each phase, the players in two consecutive blocks are allowed to move. The players in the (heavy) block of higher maximum utility play (roughly) 3-moves, i.e., they change their strategy whenever they can improve their utility by a factor of more than 3. The players in the (light) block of lower maximum utility play $(1+\varepsilon)$-moves (for some tiny positive ε). At the end of the phase, the strategies of the players in the heavy block are irrevocably decided. The crucial property of the algorithm is that the light players of a phase are the heavy ones in the next one. Due to the moves of the light players, the state reached at the end of a phase has *low stretch* (of at most 3) for the light players, meaning that there is no subset of them who could simultaneously change their strategy and improve their contribution to the potential function by a factor of more than 3. This guarantees that the number of 3-moves by the heavy players in the next phase will be polynomially-bounded and, furthermore, the effect that these moves have to players whose strategies were irrevocably decided in previous phases is minimal. So, all players stay close to a 3-approximate equilibrium at the end of the whole process.

Our algorithm follows the same general structure, even though it coordinates more carefully the moves of the players in different blocks within a phase. The improvement comes from a different implementation of the moves of the light players within a phase. Here, we focus explicitly on the problem of detecting whether there exists a set of light players who can improve their contribution to the potential by a factor of ρ by changing their strategies simultaneously. Essentially, the algorithm of [33] solves this problem for $\rho = 3$ using local search (the analog of best-response moves by the light players). Instead, we obtain the better approximation of 2.7371 by formulating this problem as a semidefinite program. Non-positive objective values of the semidefinite program guarantee that the current cut has low stretch for the light players. Positive objective values guarantee that by rounding the SDP solution appropriately, we obtain a global move which improves the potential considerably and makes progress, similar to the progress an $(1+\varepsilon)$-move by a single light player makes in the algorithm of [33].

Rounding the SDP solution is very challenging. The SDP objective has a mixture of XOR-type terms, requiring that the two vectors corresponding to the endpoints of an edge that is not in the current cut are opposite, and OR-type terms requiring that one of the endpoints of an edge in the current cut coincides with a special vector the SDP is using. Even though the XOR-type terms have non-negative coefficients and standard hyperplane rounding could yield excellent approximations for the particular terms (like in [82]), the OR-type terms have negative coefficients, making standard hyperplane rounding disastrous for them. So, we instead use an idea that originates from Feige and Goemans [65], which has inspired much follow-up work on approximating MAX2-SAT and related problems, but we consider it in an extreme that has not been considered before. In particular, the vectors in the SDP solution are first rotated in the 2-dimensional plane they define with the special vector and, then, hyperplane rounding is performed. We use the rotation function $f(\theta) = \frac{\pi}{2}(1-\cos\theta)$, meaning that a vector at an angle of θ from the special vector is relocated at an angle $f(\theta)$ from it. This gives a rather poor $1/2$-approximation to the XOR-type terms but approximates well the OR-type constraints.

The approximation ratio we obtain is the solution of a quadratic equation; the exact value is $\frac{1+2\sqrt{13}}{3} \approx 2.7371$. This is extremely rare for the approximation factors obtained from SDP-based algorithms for combinatorial optimization problems. To prove the properties of rounding, we need to prove two inequalities for triplets of 3-dimensional vectors, which turn out to depend on two variables only: the angles the special vector forms with the two other vectors involved in a term of the SDP objective. As it is often the case with the analysis of SDP-based approximation algorithms (e.g., in [109]), we have verified one of these inequalities by extensive numerical computations. These imply that the worst-case bound is obtained for a restriction of the parameters to more tractable cases, for which we present a formal analysis.

Related work

Even though research in algorithmic game theory gained popularity after the seminal papers by Koutsoupias and Papadimitriou [107], Nisan and Ronan [121], and Roughgarden and Tardos [131] (the conference versions appeared in 1999, 1999, and 2000, respectively), research in computational aspects of pure Nash equilibria had started earlier, with the introduction of the class PLS —standing for polynomial local search— by Johnson, Papadimitriou, and Yannakakis [102] and subsequent (mainly negative) results on computing locally-optimum solutions for combinatorial optimization problems. Mostly related to the topic of the current paper is the paper by Schaeffer and Yannakakis [135]. Expressed in game-theoretic terms, they proved that computing a pure Nash equilibrium in a cut game is a PLS-hard problem.

The connection of cut games and local search should not come as a surprise. Cut games are potential games [118]. They admit a potential function defined over the states of the game so that for any two states differing in the strategy of a single player, the difference in the potential values between the two states and the difference in the utility of the deviating player have the same sign. The class of potential games also includes congestion games [63, 118, 130], network design games [11], constraint satisfaction games [20, 33], market sharing games [83], and more. Most of these games have the additional property that the two differences mentioned above not only have the same sign but they are actually equal. Such potential functions are usually called exact [63]. For cut games, the total weight of edges that cross the cut is an exact potential function.

Early attempts on computing equilibria in potential games have aimed to follow a best-response dynamics, i.e., a sequence of player moves, in which the utility of the deviating player increases by at least a $1+\varepsilon$ factor. This has been proved useful in a few cases —e.g., in restricted cases of congestion games [5, 19, 43]— to compute $(1+\varepsilon)$-approximate equilibria. Such approaches work nicely if the players are similar, in the sense that their moves result in comparable utility increases. Simple arguments, which relate the increase in the potential function with the minimum increase in the utility of a deviating player can be used to bound the running time (i.e., the length of the best-response dynamics to the approximate equilibrium) in terms of the game size and $1/\varepsilon$. However, players rarely have such similarities. In a cut game, we may have players controlling nodes that have incident edges of high weight and others which have only incident edges of negligible weight. Then, such approaches cannot compute approximate equilibria in polynomial time, even though they can be used to compute states of high social welfare, e.g., see [14, 15, 22, 44].

The most successful recipe for computing approximate equilibria in potential games has been proposed by Caragiannis et al. [30]. Their approach has yielded 2-approximate equilibria for linear congestion games, and $d^{\mathcal{O}(d)}$-approximate equilibria for congestion games with polynomial latency functions of degree d. These results have been improved in [67, 139]. More importantly, the recipe of [30] has been applied to other potential games such as constraint satisfaction [33] and cost sharing games [78], as well as to non-potential games such as weighted congestion games in [26, 32, 80]. These last results have been possible by either approximating the original game by a potential game in which the algorithm is applied, or by introducing an approximate potential function to keep track of best-response moves in the original games. In all applications of the recipe of [30], the algorithm identifies a sequence of deviations by the players, either in the original game or in its approximation. The current paper is the first one to extend the recipe with global moves (of simultaneous player deviations).

Our global moves are implemented via semidefinite programming and randomized rounding. Starting with the celebrated paper of Goemans and Williamson [82], these tools have been proved very useful in approximating combinatorial optimization problems such as MAX-CUT (the problem of computing a cut of maximum total weight) and MAX-2SAT (the problem of computing a Boolean assignment that maximizes the total weight of satisfied clauses in a formula in conjunctive normal form), among many others. The classical semidefinite program for MAX-CUT uses a unit vector v_j for each graph node j, to represent the side of the cut in which to node is put. The objective is just the sum of the terms $\frac{1}{2}(1-v_j\cdot v_k)$ multiplied by the weight of edge (j,k) over all edges in the input graph. Notice that the quantity $\frac{1}{2}(1-v_j\cdot v_k)$ is a XOR-type term, taking the value 1 if the two vectors are opposite and 0 if the two vectors coincide (corresponding to nodes in different sides or at the same side of the cut, respectively). The solution of the SDP, which may inevitably consist of vectors in many different directions, is rounded by picking a random hyperplane crossing the vectors' origin, and putting the nodes at the left or the right of the cut, according to side of the hyperplane the corresponding unit vectors are lying. In this way, Goemans and

Williamson [82] improved the trivial $1/2$-approximation for MAX-CUT to 0.8785.

Modeling OR-type terms (like those required by MAX-2SAT) requires an extra variable, a reference (or special) vector $\widehat{v}$. Then, using the vector v_j as (an approximation to) the value of a Boolean variable, the OR-type term $\frac{1}{4}\cdot(3+v_j\cdot\widehat{v}+v_k\cdot\widehat{v}-v_j\cdot v_k)$ takes the value 1 if at least one of the vectors v_j and v_k coincides with the reference vector $\widehat{v}$. Feige and Goemans [65] improved the approximation bound of [82] for MAX-2SAT by exploiting the existence of the reference vector $\widehat{v}$ to rotate all vectors appropriately before rounding them using a random hyperplane. Rotation relocates each vector v_j forming an angle θ with the reference vector at a new position in the 2-dimensional plane defined by v_j and $\widehat{v}$, at angle $f(\theta)$ from $\widehat{v}$. The rotation function f they use is a linear combination between the no-rotation function $f(\theta)=\theta$ and the rotation function $f(\theta)=\frac{\pi}{2}\cdot(1-\cos\theta)$ that we also consider here. An extensive discussion of this approach with additional details is given in [141]. Lewin et al. [109] present the best known bound of 0.9401 for MAX-2SAT; in their paper, they discuss several alternatives for rounding the solution of SDPs for MAX-2SAT. Finally, we remark that even though SDPs and rounding have been used to handle negative coefficients successfully, e.g., in approximating the cut-norm of a matrix [8] or in maximizing quadratic programs [37], those formulation are very far from ours and the corresponding techniques do not seem to be applicable in our case.

Roadmap

The rest of the paper is structured as follows. We begin with preliminary definitions and notation in Section 5.2. Our algorithm is presented in Section 5.3, which concludes with the statement of our main result (Theorem 5.3.1). Then, Section 5.4 presents the properties of the global-moves routine of our algorithm. We put everything together and prove Theorem 5.3.1 in Section 5.5, while the proof of Lemma 5.4.2, which states the properties of randomized rounding is deferred to Section 5.6. We conclude in Section 5.7.

5.2 Preliminaries

A *cut game* is defined by an edge-weighted complete n-node graph $G=(V,E)$. Each edge e of the graph has a non-negative weight w_e. We overload this notation and write w_{jk} to denote the weight of the edge (j,k) and $w(A)=\sum_{e\in A} w_e$ to denote the total weight of the edges in set A. There are n players, each corresponding to a distinct node of G. The players aim to collectively but non-cooperatively build a *cut* of the graph, i.e., a partition of its nodes into two sets. Player $j\in\mathcal{N}$ has two different strategies: putting her node i at the *left* or at the *right* side of the cut. A *state* of the game is represented by an n-vector $S=(s_1,s_2,...,s_n)$, indicating that player j uses strategy s_j. Given a state S, we denote by $\mathrm{CUT}(S)$ the set of edges whose endpoints correspond to players that select different strategies in S. For a subset of players $R\subseteq\mathcal{N}$, we denote by $\mathrm{CUT}_R(S)$ the subset of $\mathrm{CUT}(S)$ that consists of edges with at least one endpoint corresponding to a player in R. With some abuse of notation, we simplify $\mathrm{CUT}_{\{j\}}(S)$

to $\mathrm{CUT}_j(S)$. The *utility* of player j is the total weight of the edges incident to her node, i.e., $u_j(S) = w(\mathrm{CUT}_j(S))$. We also denote by E_R the set of edges that are incident to at least one player in R and simplify $E_{\{j\}}$ to E_j. We denote by U_j the *maximum utility* that player j may have, i.e., $U_j = w(E_j)$.

We use a standard game-theoretic notation. Given a state $S = (s_1, s_2, ..., s_n)$ and a player $j \in \mathcal{N}$, we denote by (S_{-j}, s'_j) the state obtained by S when player j unilaterally changes her strategy from s_j to its complement s'_j. This is an *improvement move* (or, simply, a *move*) for player j if her utility increases, i.e., if $u_j(S_{-j}, s'_j) > u_j(S)$. We call it a ρ*-move* when the utility of player j increases by more than a factor of $\rho > 1$, i.e., $u_j(S_{-j}, s'_j) > \rho \cdot u_j(S)$. A state S is a *pure Nash equilibrium* (or, simply, an *equilibrium*) if no player has a move to make. Similarly, S is a ρ*-approximate (pure Nash) equilibrium* if no player has any ρ-move to make.

Given a state S, we denote by $\Phi(S)$ the total weight of the edges in $\mathrm{CUT}(S)$, i.e., $\Phi(S) = w(\mathrm{CUT}(S))$. The function Φ is a *potential function*, having the remarkable property that for every two states S and (S_{-j}, s'_j) that differ only in the strategy of player j, the difference in the potential function is equal to the difference of the utility of player j, i.e., $\Phi(S) - \Phi(S_{-j}, s'_j) = u_j(S) - u_j(S_{-j}, s'_j)$.

Following [33], we will often consider sequences of moves in which only players from a certain subset $R \subseteq \mathcal{N}$ are moving. These can be thought of as moves in a *subgame* defined on graph G among the players in R only, in which the nodes that are not controlled by players in R are *frozen* to a fixed placement in one of the sides of the cut. We represent states of a subgame in the same way we do for the original game using n-vectors of strategies, including both the strategies of the players in R and the frozen strategies of the players in $\mathcal{N} \setminus R$.

The next two claims follow easily by our definitions and are used repeatedly in our proofs. Claim 5.2.1 shows that the subgames of a cut game are also potential games and Claim 5.2.2 gives basic facts about the utility of players.

Claim 5.2.1. *Consider a cut game $\mathcal{G}$ with a set $\mathcal{N}$ of players. For every subgame of $\mathcal{G}$ defined among a subset $R \subseteq \mathcal{N}$ of players, the function Φ_R, defined as $\Phi_R(S) = w(CUT_R(S))$ for each state S of the subgame, is an exact potential function.*

Proof. Consider a subgame of $\mathcal{G}$ defined by a subset $R \subseteq \mathcal{N}$ of players and let S and (S_{-j}, s'_j) be two states of the subgame differing in the strategies of player $j \in R$. By definition, we have

$$\begin{aligned}\Phi_R(S) - \Phi_R(S_{-j}, s'_j) &= w(\mathrm{CUT}_R(S)) - w(\mathrm{CUT}_R(S_{-j}, s'_j)) \\ &= w(\mathrm{CUT}_j(S)) - w(\mathrm{CUT}_j(S_{-j}, s'_j)) \\ &= u_j(S) - u_j(S_{-j}, s'_j),\end{aligned}$$

as desired. $\square$

Claim 5.2.2. *Consider a cut game $\mathcal{G}$ with a set $\mathcal{N}$ of players, and let $j \in \mathcal{N}$ be a player and S a state of $\mathcal{G}$. If player j performs a p-move, her utility and the potential*

Φ_R for every subgame involving her (i.e., $j \in R \subseteq \mathcal{N}$) increase by more than $\frac{p-1}{p+1} \cdot U_j$. If player j does not have any p-move to make at state S, then $u_j(S) \geq \frac{1}{p+1} \cdot U_j$.

Proof. Assume that player j performs a p-move, reaching the state (S_{-j}, s'_j) from S, after changing her strategy from s_j to s'_j. Then, the increase in player j's utility during the p-move is

$$\begin{aligned}
u_j(S_{-j}, s'_j) - u_j(S) &> (p-1) \cdot u_j(S) \\
&= \frac{p-1}{2} \cdot u_j(S) + \frac{p-1}{2} \cdot (U_j - u_j(S_{-j}, s'_j)) \\
&= \frac{p-1}{2} \cdot U_j - \frac{p-1}{2} \cdot (u_j(S_{-j}, s'_j) - u_j(S)),
\end{aligned}$$

i.e., $u_j(S_{-j}, s'_j) - u_j(S) \geq \frac{p-1}{p+1} \cdot U_j$. Since, by Claim 5.2.1, Φ_R is an exact potential for any subset of players R that contains player j, we have that $\Phi_R(S_{-j}, s'_j) - \Phi_R(S) \geq \frac{p-1}{p+1} \cdot U_j$ as well.

Now assume that player j has no p-move to make at state S. We have

$$U_j = u_j(S) + u_j(S_{-j}, s'_j) \leq (p+1) \cdot u_j(S).$$

The claim follows. □

Given two states S and T of game $\mathcal{G}$ and a possibly empty subset of players $R \subseteq \mathcal{N}$, we denote by (S_{-R}, T_R) the state in which each player of set R uses her strategy in state T and each player in set $\mathcal{N} \setminus R$ uses her strategy in state S. In particular, we denote by (S_{-R}, S'_R) the state in which each player of set R uses the complement of her strategy in state S and each player of set $\mathcal{N} \setminus R$ uses her strategy in S. We refer to the ratio $\frac{\Phi_R(S_{-R}, S'_R)}{\Phi_R(S)}$ as the *stretch* of state S for the subgame defined by the subset of players $R \subseteq \mathcal{N}$.

5.3 The algorithm

Our algorithm (see its pseudocode as Algorithm 5.1) has a simple general structure. It uses several constants: a tolerance parameter $\varepsilon \in (0, 1/4]$ and the constants $\rho = \frac{1+2\sqrt{13}}{3} \approx 2.7371$, $\sigma = \rho + \varepsilon/3$, and $\tau = \rho + 2\varepsilon/3$. The algorithm takes as input a cut game $\mathcal{G}$ with n players and returns a state S_{out}. As we will show, the algorithm terminates after polynomially many steps in terms of n and $1/\varepsilon$ and the state S_{out} is a $(\rho + \varepsilon)$-approximate pure Nash equilibrium, with high probability.

The algorithm initializes parameter Δ to be polynomially related to n and $1/\varepsilon$ (line 1). Then, in lines 2 and 3, it implicitly partitions the players into blocks B_1, B_2, ..., B_m according to their maximum utility. Denoting by $U_{\max}$ the maximum utility among all players, block B_i consists of the players with maximum utility in the interval $(U_{\max}\Delta^{-i}, U_{\max}\Delta^{1-i}]$. By the definition of Δ, the players in the same block have polynomially related maximum utilities.

ALGORITHM 5.1: The algorithm for computing approximate pure Nash equilibria in cut games

Input: A cut game $\mathscr{G}$ with a set $\mathscr{N}$ of n players
Output: A state S_{out} of $\mathscr{G}$
1: $\Delta \leftarrow 480n/\varepsilon^2$;
2: Set $U_{\min} \leftarrow \min_{j\in\mathscr{N}} U_j$, $U_{\max} \leftarrow \max_{j\in\mathscr{N}} U_j$, and $m \leftarrow 1 + \lfloor \log_\Delta (U_{\max}/U_{\min}) \rfloor$;
3: (Implicitly) partition the players into blocks $B_1, B_2, ..., B_m$, such that $j \in B_i$ implies that $U_j \in \left(U_{\max}\Delta^{-i}, U_{\max}\Delta^{1-i}\right]$;
4: Set $S \leftarrow$ an arbitrary initial state;
5: Update $S \leftarrow$ GLOBAL$(\mathscr{G}, S, B_1)$;
6: **for** phase $i \leftarrow 1$ to $m-1$ such that $B_i \neq \emptyset$ **do**
7:     **repeat**
8:         Update $(S, ch1) \leftarrow$ LOCAL$(\mathscr{G}, S, B_i, \tau)$;
9:         Update $(S, ch2) \leftarrow$ GLOBAL$(\mathscr{G}, S, B_{i+1})$;
10:     **until** $ch1 =$ **false and** $ch2 =$ **false**
11: **end for**
12: Update $S \leftarrow$ LOCAL$(\mathscr{G}, S, B_m, \tau)$;
13: **return** $S_{\text{out}} \leftarrow S$;

The lines 4-12 are the main body of the algorithm. The algorithm utilizes two routines, namely LOCAL() and GLOBAL(), implementing a set of local and global moves among the players in a particular block. The computation is divided into phases. Starting with an arbitrary initial state (in line 4), phase 0 (line 5) executes global moves for the players in block B_1. Then, for $i = 1, ..., m-1$, phase i repeatedly executes local moves for the players in block B_i and global moves for the players in block B_{i+1} until the execution of these routines does not produce new states. Finally, the algorithm runs phase m, which executes local moves for the players in block B_m. The state produced after the execution of phase m is returned as the output state S_{out}.

The routine LOCAL() (see Algorithm 5.2) takes as input the cut game $\mathscr{G}$, a current state S, a subset of players B, and a parameter $p > 1$; on exit, it returns an output state and a flag denoting whether the output state is different than the input one. LOCAL() repeatedly performs p-moves as long as there is a player in set B that has such a p-move to make.

ALGORITHM 5.2: The routine LOCAL()

Input: A cut game $\mathscr{G}$ with a set $\mathscr{N}$ of n players, a state S, a subset $B \subseteq \mathscr{N}$ of players, and a parameter $p > 1$
Output: A state of $\mathscr{G}$ and a Boolean flag
Set *change* $\leftarrow$ **false**;
while there is a player $j \in B$ that satisfies $u_j(S_{-j}, s'_j) > p \cdot u_j(S)$ **do**
    Update $S \leftarrow (S_{-j}, s'_j)$;
    change $\leftarrow$ **true**;
end while
return $(S, change)$;

The routine GLOBAL() is the main novelty of our algorithm compared to the algorithm in [33]. It calls the routines SDP() and ROUND(). When executed on input a cut game $\mathcal{G}$ with a set $\mathcal{N}$ of n players, state S, and the subset of players $B \subseteq \mathcal{N}$, the routine SDP() solves the following semidefinite program:

$$\begin{aligned}
\text{maximize} \quad & Z = \frac{1}{2} \cdot \sum_{(j,k)\in E\setminus \mathrm{CUT}_B(S)} w_{jk} \cdot (1 - v_j \cdot v_k) \\
& -\frac{1}{4} \cdot \sum_{(j,k)\in \mathrm{CUT}_B(S)} w_{jk} \cdot (3\sigma - 1 + (\sigma - 1) \cdot \widehat{v} \cdot v_j \\
& \quad +(\sigma - 1) \cdot \widehat{v} \cdot v_k - (\sigma + 1) \cdot v_j \cdot v_k) \\
\text{subject to} \quad & \sum_{(j,k)\in \mathrm{CUT}_B(S)} w_{jk} \cdot (3 + v_j \cdot \widehat{v} + v_k \cdot \widehat{v} - v_j \cdot v_k) \geq \min_{j\in B} U_j \\
& \widehat{v} \cdot \widehat{v} = 1, \widehat{v} \in \mathbb{R}^n \\
& v_j \cdot v_j = 1, v_j \in \mathbb{R}^n, \forall j \in B \\
& v_j = -\widehat{v}, \forall j \in \mathcal{N} \setminus B
\end{aligned}$$

The output of SDP() is the matrix $\mathbf{v}$ containing as columns the unit vectors v_j for $j \in \mathcal{N}$ and $\widehat{v}$, and the objective value Z.

At a high level, the semidefinite program aims to identify a set of players $R \subseteq B$ so that the stretch of the current state for the subgame defined by the players in R is at least σ. The special vector $\widehat{v}$ is used as a basis for the orientation of the remaining vectors. The vectors v_j for $j \in \mathcal{N} \setminus B$ are set to $-\widehat{v}$, indicating that the players in $\mathcal{N} \setminus B$ are not considered for inclusion in the set R. For $j \in B$, vector v_j indicates whether player j belongs to set R (when $v_j = \widehat{v}$) or not (when $v_j = -\widehat{v}$). Of course, the vectors in a solution of the semidefinite program can be much different than these two extreme ones. As we will see, if the semidefinite program has non-positive objective value, no set R of players defining a subgame of high stretch at the current state exists, and this provides a useful condition for our algorithm. Otherwise, a subsequent appropriately performed (randomized) rounding of the SDP solution will yield a set of simultaneous (global) moves by players that increase the potential considerably.

This is done by the routine ROUND(), which takes as input a cut game $\mathcal{G}$ with a set $\mathcal{N}$ of n players, a state S, a subset of players $B \subseteq \mathcal{N}$, and a matrix $\mathbf{v}$ having vectors v_j for each player $j \in B$ and the special vector $\widehat{v}$ as columns. ROUND() performs $\frac{23040n^2}{\varepsilon^3} \cdot \ln\left(\frac{414720n^4}{\varepsilon^4}\right)$ iterations. In each of them, it decides a set $R \subseteq \mathcal{N}$ of players by rounding the set of unit vectors $\mathbf{v}$. Rounding is performed as follows. First, for $j \in B$, each vector v_j is rotated to produce the unit vector v'_j as follows: v'_j is the unit vector in $\mathbb{R}^n$ that lies on the plane defined by vectors v_j and $\widehat{v}$ and is at angle $\frac{\pi}{2} \cdot (1 - v_j \cdot \widehat{v})$ from vector $\widehat{v}$ and at angle $|\arccos(v_j \cdot \widehat{v}) - \frac{\pi}{2} \cdot (1 - v_j \cdot \widehat{v})|$ from vector v_j. Then, a random unit vector $r \in \mathbb{R}^n$ is picked. For $j \in B$, player j is included in set R if the inner products $v'_j \cdot r$ and $\widehat{v} \cdot r$ have the same sign. Informally, we can view vector r as the unit normal vector of a random hyperplane; then, player j is included in set R if the rotated vector v'_j and vector $\widehat{v}$ are on the same side of the hyperplane. ROUND()

keeps the set R of players in the rounding iteration that produces the highest potential increase $\Phi_R(S_{-R}, S'_R) - \Phi_R(S)$ and returns state (S_{-R}, S'_R) as output.

Now, the routine GLOBAL() (see Algorithm 5.3) takes as input the cut game $\mathscr{G}$, a state S, and a subset of players B; on exit, it returns an output state and a flag indicating whether the output state is different than the input one. GLOBAL() repeatedly runs the routing SDP(), as long as the objective value of the semidefinite program it solves is positive. Then, it executes the routine ROUND() to round the solution of the semidefinite program and decide, in this way, the set of players who will change their strategy simultaneously. Essentially, this implements a global move. Prior to executing SDP(), the routine LOCAL() is executed to make sure that the input state to SDP() is an approximate equilibrium for the players in B.

ALGORITHM 5.3: The routine GLOBAL()

Input: A cut game $\mathscr{G}$ with a set $\mathscr{N}$ of n players, a state S, and subset $B \subseteq \mathscr{N}$ of players
Output: A state of $\mathscr{G}$ and a Boolean flag

Set *change* $\leftarrow$ **false**;
Update $(S, change) \leftarrow$ LOCAL$(\mathscr{G}, S, B, \tau)$
Set $(\mathbf{v}, Z) \leftarrow$ SDP$(\mathscr{G}, S, B)$;
while $Z > 0$ **do**
    Update $S \leftarrow$ ROUND$(\mathscr{G}, S, B, \mathbf{v})$;
    Update *change* $\leftarrow$ **true**;
    Update $S \leftarrow$ LOCAL$(\mathscr{G}, S, B, \tau)$;
    Set $(\mathbf{v}, Z) \leftarrow$ SDP$(\mathscr{G}, S, B)$;
end while
return $(S, change)$;

This completes the detailed description of our algorithm. We will prove the following statement.

Theorem 5.3.1. *Let $\varepsilon \in (0, 1/4]$. On input a cut game $\mathscr{G}$ with n players, Algorithm 5.1 computes a $(\rho + \varepsilon)$-approximate pure Nash equilibrium S_{out} in time poly$(n, 1/\varepsilon)$ with probability at least $1 - 1/n$, where $\rho = \frac{1+2\sqrt{13}}{3} \approx 2.7371$.*

5.4 Properties of global moves

We devote this section to proving two important properties of routines SDP() and ROUND(). In our analysis, we denote the upper boundary of block B_i by W_i and by W_{m+1} the lower boundary of block B_m, i.e., $W_i = U_{\max}\Delta^{1-i}$ for $i = 1, 2, ..., m+1$. So, the players of block B_i are those with maximum utility $U_j \in (W_{i+1}, W_i]$.

The next lemma guarantees that, whenever routine GLOBAL() terminates, the state returned has stretch at most σ for any subgame defined by subsets of players in block B_{i+1}.

Lemma 5.4.1. *Consider the execution of routine SDP() during phase $i \geq 0$, i.e., SDP() takes as input the cut game $\mathscr{G}$, the block B_{i+1} as the set of players B, and a* *state S that is a τ-approximate pure Nash equilibrium for the players in B_{i+1}. If the objective value Z of the semidefinite program that SDP() returns is non-positive, then for every set of players $R \subseteq B_{i+1}$, it holds that $\sigma \cdot \Phi_R(S) \geq \Phi_R(S_{-R}, S'_R)$.*

Proof. Assume otherwise and let R be a non-empty subset of B_{i+1} such that

$$\sigma \cdot \Phi_R(S) < \Phi_R(S_{-R}, S'_R). \tag{5.1}$$

Let $\widehat{v}$ be an arbitrary unit vector and set $v_j = \widehat{v}$ for $j \in R$ and $v_j = -\widehat{v}$ for $j \in \mathcal{N} \setminus R$. Since $R \neq \emptyset$, there is at least one player $j^* \in R$ such that $v_{j^*} = \widehat{v}$. Then, $3 + v_{j^*} \cdot \widehat{v} + v_k \cdot \widehat{v} - v_{j^*} \cdot v_k = 4$ for every $k \in \mathcal{N} \setminus \{j^*\}$ and

$$\begin{aligned} \sum_{(j,k) \in \mathrm{CUT}_{B_{i+1}}(S)} w_{jk} \cdot (3 + v_j \cdot \widehat{v} + v_k \cdot \widehat{v} - v_j \cdot v_k) &\geq 4 \cdot w(\mathrm{CUT}_{j^*}(S)) \\ &\geq U_{j^*} \geq \min_{j \in B_{i+1}} U_j, \end{aligned}$$

implying that $\mathbf{v}$ is a feasible solution to the semidefinite program used in the call of SDP$(\mathcal{G}, S, B_{i+1})$. The second last inequality follows using Claim 5.2.2, since the τ-approximate (and, consequently, the 3-approximate) equilibrium condition is satisfied for player j^*.

Now, observe that edge (j,k) belongs to $\mathrm{CUT}_R(S)$ if it belongs to $\mathrm{CUT}_{B_{i+1}}(S)$ and at least one of the players j and k belongs to R. Also, the quantity $\frac{1}{4} \cdot (3 + v_j \cdot \widehat{v} + v_k \cdot \widehat{v} - v_j \cdot v_k)$ is equal to 1 if one of the players j and k belongs to R and is equal to 0, otherwise. Thus,

$$\Phi_R(S) = \sum_{(j,k) \in \mathrm{CUT}_R(S)} w_{jk} = \frac{1}{4} \cdot \sum_{(j,k) \in \mathrm{CUT}_{B_{i+1}}(S)} w_{jk} \cdot (3 + v_j \cdot \widehat{v} + v_k \cdot \widehat{v} - v_j \cdot v_k). \tag{5.2}$$

Furthermore, observe that edge (j,k) belongs to $\mathrm{CUT}_R(S_{-R}, S'_R)$ if it belongs to $\mathrm{CUT}_{B_{i+1}}(S)$ and both players j and k belong to R, or it does not belong to $\mathrm{CUT}_{B_{i+1}}(S)$ and exactly one of players j and k belongs to R. The quantity $\frac{1}{4} \cdot (1 + v_j \cdot \widehat{v} + v_k \cdot \widehat{v} + v_j \cdot v_k)$ is equal to 1 if both players j and k belong to R and is equal to 0, otherwise. Also, the quantity $\frac{1}{2} \cdot (1 - v_j \cdot v_k)$ is equal to 1 if exactly one of the players j and k belongs to R and is equal to 0, otherwise. Thus,

$$\begin{aligned} \Phi_R(S_{-R}, S'_R) = &\sum_{(j,k) \in \mathrm{CUT}_R(S_{-R}, S'_R)} w_{jk} \\ = &\frac{1}{2} \cdot \sum_{(j,k) \in E \setminus \mathrm{CUT}_{B_{i+1}}(S)} w_{jk} \cdot (1 - v_j \cdot v_k) \\ &+ \frac{1}{4} \cdot \sum_{(j,k) \in \mathrm{CUT}_{B_{i+1}}(S)} w_{jk} \cdot (1 + v_j \cdot \widehat{v} + v_k \cdot \widehat{v} + v_j \cdot v_k). \end{aligned} \tag{5.3}$$

Using equations (5.2) and (5.3) and the definition of the semidefinite program used in the call of SDP$(\mathcal{G}, S, B_{i+1})$, our assumption (5.1) yields

$$0 < \Phi_R(S_{-R}, S'_R) - \sigma \cdot \Phi_R(S)$$

$$
\begin{aligned}
&= \frac{1}{2} \cdot \sum_{(j,k) \in E \setminus \mathrm{CUT}_{B_{i+1}}(S)} w_{jk} \cdot (1 - v_j \cdot v_k) \\
&\quad - \frac{1}{4} \cdot \sum_{(j,k) \in \mathrm{CUT}_{B_{i+1}}(S)} w_{jk} \cdot (3\sigma - 1 + (\sigma - 1) \cdot v_j \cdot \widehat{v} \\
&\quad + (\sigma - 1) \cdot v_k \cdot \widehat{v} - (\sigma + 1) \cdot v_j \cdot v_k) \\
&= Z,
\end{aligned}
$$

which contradicts the condition on Z. The lemma follows. □

We now aim to prove (in Lemma 5.4.3 below) that, whenever the routine ROUND() is executed on the set of vectors produced by a previous execution of routine SDP(), the potential increases considerably. To do so, we will use an important property of each rounding iteration, which we state in Lemma 5.4.2; its proof is deferred to Section 5.6. We denote by $\mathrm{XOR}(j,k)$ the event that the random hyperplane with unit normal vector r separates the rotated vectors v'_j and v'_k, i.e., the quantities $v'_j \cdot r$ and $v'_j \cdot r$ have different signs.

Lemma 5.4.2. *Let v_j and v_k be unit vectors in $\mathbb{R}^n$ and let v'_j and v'_k be the corresponding rotated vectors with respect to another unit vector $\widehat{v} \in \mathbb{R}^n$. Then,*

$$\Pr[XOR(j,k)] \geq \frac{1}{4} \cdot (1 - v_j \cdot v_k) \tag{5.4}$$

and

$$\Pr[XOR(j,k)] \leq \frac{1}{8} \cdot (3\rho - 1 + (\rho - 1) \cdot v_j \cdot \widehat{v} + (\rho - 1) \cdot v_k \cdot \widehat{v} - (\rho + 1) \cdot v_j \cdot v_k). \tag{5.5}$$

Lemma 5.4.3. *Consider the execution of routine SDP$(\mathcal{G}, S, B_{i+1})$ during phase $i \geq 0$. If the solution* $\mathbf{v}$ *to the semidefinite program computed by SDP$(\mathcal{G}, S, B_{i+1})$ has objective value $Z > 0$, then the probability that the subsequent call to ROUND$(\mathcal{G}, S, B_{i+1}, v)$ fails to identify a set of players $R \subseteq B_{i+1}$ so that $\Phi_R(S_{-R}, S'_R) - \Phi_R(S) \geq \frac{\varepsilon}{48} \cdot W_{i+2}$ is at most $\frac{\varepsilon^4}{414720 n^4}$.*

Proof. Consider the execution of routine ROUND() in phase i and a rounding iteration. Let $R \subseteq B_{i+1}$ be the set of players selected. Observe that the increase in the potential Φ_R from state S to state (S_{-R}, S'_R) is equal to the total weight of the edges (j,k) that do not belong to $\mathrm{CUT}_{B_{i+1}}(S)$ and exactly one of the players j and k change strategy, minus the total weight of the edges (j,k) that belong to $\mathrm{CUT}_{B_{i+1}}(S)$ and exactly one of the players j and k change strategy. Using this observation and Lemma 5.4.2, we have

$$
\begin{aligned}
&\mathsf{E}[\Phi_R(S_{-R}, S'_R) - \Phi_R(S)] \\
&= \sum_{(j,k) \in E \setminus \mathrm{CUT}_{B_{i+1}}(S)} w_{jk} \cdot \Pr[\mathrm{XOR}(j,k)] - \sum_{(j,k) \in \mathrm{CUT}_{B_{i+1}}(S)} w_{jk} \cdot \Pr[\mathrm{XOR}(j,k)]
\end{aligned}
$$

$$\geq \frac{1}{4} \cdot \sum_{(j,k)\in E\setminus \mathrm{CUT}_{B_{i+1}}(S)} w_{jk} \cdot (1 - v_j \cdot v_k)$$
$$- \frac{1}{8} \cdot \sum_{(j,k)\in \mathrm{CUT}_{B_{i+1}}(S)} w_{jk} \cdot (3\rho - 1 + (\rho-1)\cdot v_j \cdot \widehat{v} + (\rho-1)\cdot v_k \cdot \widehat{v} - (\rho+1)\cdot v_j \cdot v_k)$$
$$= \frac{\sigma - \rho}{8} \cdot \sum_{(j,k)\in \mathrm{CUT}_{B_{i+1}}(S)} w_{jk} \cdot (3 + v_j \cdot \widehat{v} + v_k \cdot \widehat{v} - v_j \cdot v_k) + \frac{Z}{2}$$
$$> \frac{\varepsilon}{24} \cdot W_{i+2}.$$

The last inequality follows by condition $Z > 0$, the constraint of the semidefinite program and the definition of W_{i+2}, and the fact that $\sigma - \rho = \varepsilon/3$.

Now, observe that the random variable $\sum_{j\in B_{i+1}} U_j - \Phi_R(S_{-R}, S'_R) + \Phi_R(S)$ is always non-negative. Applying the Markov inequality on it, we get

$$\Pr\left[\Phi_R(S_{-R}, S'_R) - \Phi_R(S) \leq \frac{\varepsilon}{48} \cdot W_{i+2}\right]$$
$$= \Pr\left[\sum_{j\in B_{i+1}} U_j - \Phi_R(S_{-R}, S'_R) + \Phi_R(S) \geq \sum_{j\in B_{i+1}} U_j - \frac{\varepsilon}{48} \cdot W_{i+2}\right]$$
$$\leq \frac{\sum_{j\in B_{i+1}} U_j - \mathsf{E}[\Phi_R(S_{-R}, S'_R) - \Phi_R(S)]}{\sum_{j\in B_{i+1}} U_j - \frac{\varepsilon}{48} \cdot W_{i+2}}$$
$$< \frac{\sum_{j\in B_{i+1}} U_j - \frac{\varepsilon}{24} \cdot W_{i+2}}{\sum_{j\in B_{i+1}} U_j - \frac{\varepsilon}{48} \cdot W_{i+2}} \leq 1 - \frac{\frac{\varepsilon}{48} \cdot W_{i+2}}{\sum_{j\in B_{i+1}} U_j} \leq 1 - \frac{\varepsilon \cdot W_{i+2}}{48n \cdot W_{i+1}} = 1 - \frac{\varepsilon^3}{23040n^2}.$$

I.e., the probability that $\Phi_R(S_{-R}, S'_R) - \Phi_R(S) \leq \frac{\varepsilon}{48} \cdot W_{i+2}$ in a rounding iteration is at most $1 - \frac{\varepsilon^3}{23040n^2}$. Thus, the probability that the potential increase is at most $\frac{\varepsilon}{48} \cdot W_{i+2}$ after all the $\frac{23040n^2}{\varepsilon^3} \cdot \ln\left(\frac{414720n^4}{\varepsilon^4}\right)$ rounding iterations in an execution of ROUND() is at most

$$\left(1 - \frac{\varepsilon^3}{23040n^2}\right)^{\frac{23040n^2}{\varepsilon^3} \cdot \ln\left(\frac{414720n^4}{\varepsilon^4}\right)} \leq \frac{\varepsilon^4}{414720n^4}.$$

The last inequality follows by the property $(1 - 1/t)^t \leq 1/e$. □

5.5 Proof of Theorem 5.3.1

We now show how to use the main properties of global moves from Section 5.4, as well as three additional claims (Claims 5.5.1, 5.5.2, and 5.5.3) to prove Theorem 5.3.1. In our proof, we denote by S^i the state reached at the end of phase $i \geq 0$, i.e., $S_{\mathrm{out}} = S^m$. We also denote by R_i the set of players that changed their strategy during phase i.

For every phase $i > 0$, we define the following set of edges:

$$A_i = \{(j,k) \in \mathrm{CUT}_{R_i}(S^{i-1}) : j,k \in R_i \cap B_i\}$$

$$
\begin{aligned}
C_i &= \{(j,k) \in \mathrm{CUT}_{R_i}(S^{i-1}) : j,k \in R_i \cap B_{i+1}\} \\
D_i &= \{(j,k) \in \mathrm{CUT}_{R_i}(S^{i-1}) : j \in R_i \cap B_i, k \in R_i \cap B_{i+1}\} \\
F_i &= \{(j,k) \notin \mathrm{CUT}_{R_i}(S^{i-1}) : j \in R_i \cap B_i, k \in R_i \cap B_{i+1}\} \\
H_i &= \{(j,k) \in \mathrm{CUT}_{R_i}(S^{i-1}) : j \in R_i \cap B_i, k \notin R_i\} \\
I_i &= \{(j,k) \notin \mathrm{CUT}_{R_i}(S^{i-1}) : j \in R_i \cap B_i, k \notin R_i\} \\
J_i &= \{(j,k) \in \mathrm{CUT}_{R_i}(S^{i-1}) : j \in R_i \cap B_{i+1}, k \notin R_i\} \\
K_i &= \{(j,k) \notin \mathrm{CUT}_{R_i}(S^{i-1}) : j \in R_i \cap B_{i+1}, k \notin R_i\}
\end{aligned}
$$

We will express several properties maintained by our algorithm in terms of the weights of the above edge sets. The first one uses the definition of block B_{i+1}.

Claim 5.5.1. *For every phase $i > 0$, it holds $w(C_i) + w(D_i) + w(F_i) + w(J_i) + w(K_i) \leq n \cdot W_{i+1}$.*

Proof. Indeed, observe that the edges in sets C_i, D_i, F_i, J_i, and K_i have at least one of their endpoints corresponding to a player in block B_{i+1}. The inequality follows since the total weight in edges incident to (the at most n) nodes corresponding to players in block B_{i+1} is at most W_{i+1}. □

The next claim uses the main property guaranteed by the last call of routine SDP() in phase $i-1$.

Claim 5.5.2. *For every phase $i > 0$, it holds $-(\sigma - 1) \cdot w(A_i) - \sigma \cdot w(D_i) - \sigma \cdot w(H_i) + w(F_i) + w(I_i) \leq 0$.*

Proof. By the definition of routine GLOBAL(), in the last execution of routine SDP() in phase $i-1$, the objective value of the semidefinite program returned is non-positive. Then, by applying Lemma 5.4.1 for the set of players $R_i \cap B_i$, we get

$$\Phi_{R_i \cap B_i}(S^{i-1}_{-R_i \cap B_i}, S^{i}_{R_i \cap B_i}) \leq \sigma \cdot \Phi_{R_i \cap B_i}(S^{i-1}). \tag{5.6}$$

By the definition of the edge sets, notice that $\mathrm{CUT}_{R_i \cap B_i}(S^{i-1}) = A_i \cup D_i \cup H_i$ and, hence,

$$\Phi_{R_i \cap B_i}(S^{i-1}) = w(A_i) + w(D_i) + w(H_i). \tag{5.7}$$

Furthermore, $\mathrm{CUT}_{R_i \cap B_i}(S^{i-1}_{-R_i \cap B_i}, S^{i}_{R_i \cap B_i}) = A_i \cup F_i \cup I_i$ and, hence,

$$\Phi_{R_i \cap B_i}(S^{i-1}_{-R_i \cap B_i}, S^{i}_{R_i \cap B_i}) = w(A_i) + w(F_i) + w(I_i). \tag{5.8}$$

The claim follows after using equations (5.7) and (5.8) to substitute the potential values in (5.6), and rearranging. □

Finally, our third claim follows by the fact that the players in block $R_i \cap B_i$ make τ-moves in phase i.

Claim 5.5.3. *For every phase $i > 0$, it holds $(\tau - 1) \cdot w(A_i) + (\tau - 1) \cdot w(D_i) + \tau \cdot w(H_i) - w(I_i) + \tau \cdot w(J_i) - \tau \cdot w(K_i) < 0$.*

Proof. Consider player $j \in R_i \cap B_i$ and let L_j be the set of edges incident to node j that belong to the cut after the last τ-move of player j in phase i. The increase in the utility of player j after this τ-move is higher than $(1 - 1/\tau) \cdot w(L_j)$. Now, since Φ_{R_i} is an exact potential (by Claim 5.2.1), the increase of the potential value due to the last τ-move of player j is equal to the increase of her utility. Thus, the total increase of the potential between states S^{i-1} and S^i is higher than the total utility increase in the last τ-moves of the players in $R_i \cap B_i$, i.e.,

$$\Phi_{R_i}(S^i) - \Phi_{R_i}(S^{i-1}) > \sum_{j \in R_i \cap B_i} (1 - 1/\tau) \cdot w(L_j) \geq (1 - 1/\tau) \cdot w\left(\cup_{j \in R_i \cap B_i} L_j\right). \quad (5.9)$$

Now, notice that $\mathrm{CUT}_{R_i}(S^i) = A_i \cup C_i \cup D_i \cup I_i \cup K_i$, $\mathrm{CUT}_{R_i}(S^{i-1}) = A_i \cup C_i \cup D_i \cup H_i \cup J_i$, and $\cup_{j \in R_i \cap B_i} L_j \supseteq \mathrm{CUT}_{R_i \cap B_i}(S^i) = A_i \cup D_i \cup I_i$. Hence,

$$\Phi_{R_i}(S^i) = w(A_i) + w(C_i) + w(D_i) + w(I_i) + w(K_i), \quad (5.10)$$

$$\Phi_{R_i}(S^{i-1}) = w(A_i) + w(C_i) + w(D_i) + w(H_i) + w(J_i), \quad (5.11)$$

and

$$w\left(\cup_{j \in R_i \cap B_i} L_j\right) \geq w(A_i) + w(D_i) + w(I_i). \quad (5.12)$$

Using equations (5.10), (5.11), and (5.12) to substitute the potential values and $w\left(\cup_{j \in R_i \cap B_i} L_j\right)$ in (5.9), we get

$$w(I_i) + w(K_i) - w(H_i) - w(J_i) > (1 - 1/\tau) \cdot (w(A_i) + w(D_i) + w(I_i)).$$

The claim follows by multiplying both sides of this last inequality by τ and rearranging. □

We are now ready to prove Theorem 5.3.1. We do so in Lemmas 5.5.4 and 5.5.5 below. Assuming that all calls of routine ROUND() are successful in identifying a set of players whose change of strategies that increase the potential considerably, Lemma 5.5.4 uses Claims 5.5.1, 5.5.2, and 5.5.3 to show that every player stays at a $(\rho + \varepsilon)$-equilibrium after the phase in which its strategy is irrevocably decided.

Lemma 5.5.4. *If no call to routine ROUND() ever fails, the state S_{out} is a $(\rho + \varepsilon)$-approximate pure Nash equilibrium.*

Proof. Let j be a player of block B_i. Notice that, if $i = m$, player j has no τ-move (and, hence, no $(\rho + \varepsilon)$-move) to make after the execution of the last phase. So, in the following, we assume that $0 < i \leq m - 1$.

Let M_j be the set of edges in $\mathrm{CUT}_j(S^i)$ whose other endpoint corresponds to a player in blocks B_{i+1}, ..., B_{m-1}. We will first bound $w(M_j)$. Notice that $M_j \subseteq \cup_{t > i}(H_t \cup J_t)$. Hence,

$$w(M_j) \leq \sum_{t > i} (w(H_t) + w(J_t)). \quad (5.13)$$

Summing the inequalities in the statements of Claims 5.5.1, 5.5.2, and 5.5.3, after multiplying the inequality in Claim 5.5.1 by τ, we get

$$(\tau-\sigma)\cdot w(A_t)+\tau\cdot w(C_t)+(2\tau-\sigma-1)\cdot w(D_t)+(\tau+1)\cdot w(F_t)$$
$$+(\tau-\sigma)\cdot w(H_t)+2\tau\cdot w(J_t)\leq \tau\cdot n\cdot W_{t+1}.$$

Notice that the LHS is lower-bounded by $\frac{\varepsilon}{3}\cdot(w(H_t)+w(J_t))$. Thus,

$$w(H_t)+w(J_t)\leq 9n\cdot W_{t+1}/\varepsilon.$$

Now, inequality (5.13) yields

$$w(M_j)\leq\frac{9n}{\varepsilon}\cdot\sum_{t>i}W_{t+1}\leq\frac{9n}{\varepsilon}\cdot W_{i+2}\sum_{t=i+1}^{\infty}\Delta^{i+1-t}\leq 10n\cdot W_{i+2}/\varepsilon. \tag{5.14}$$

Now, notice that the edges in M_j are those which can be removed from the cut due to moves by players in phases $i+1$, ..., m, and thus decrease the utility of player j by up to $w(M_j)$. I.e.,

$$u_j(S^m)\geq u_j(S^i)-w(M_j). \tag{5.15}$$

Furthermore, the edges in M_j are those which are not included in $\text{CUT}_j(S^i_{-j},s'_j)$ and thus do not contribute to the utility player j would have by changing her strategy. Moves by players in phases $i+1$, ..., m may increase the utility player j would have by deviating by at most $w(M_j)$, i.e.,

$$u_j(S^i_{-j},s'_j)\geq u_j(S^m_{-j},s'_j)-w(M_j). \tag{5.16}$$

We are ready to prove that player j has no $(\rho+\varepsilon)$-move to make after the execution of the last phase of the algorithm, using inequalities (5.14), (5.15), and (5.16), and the fact that player j had no τ-move to make at the end of phase i. We have

$$\begin{aligned}(\rho+\varepsilon)\cdot u_j(S^m) &\geq (\rho+\varepsilon)\cdot u_j(S^i)-10(\rho+\varepsilon)n\cdot W_{i+2}/\varepsilon\\ &\geq \tau\cdot u_j(S^i)-30n\cdot W_{i+2}/\varepsilon+(\rho+\varepsilon-\tau)\cdot u_j(S^i)\\ &\geq u_j(S^i_{-j},s'_j)-30n\cdot W_{i+2}/\varepsilon+\frac{\varepsilon}{3}\cdot u_j(S^i)\\ &\geq u_j(S^m_{-j},s'_j)-40n\cdot W_{i+2}/\varepsilon+\frac{\varepsilon}{3}\cdot u_j(S^i).\end{aligned} \tag{5.17}$$

Now, since player j has no τ-move at state S^i, Claim 5.2.2 implies that $u_j(S^i)\geq W_{i+1}/4$ and, by the definition of the blocks, $u_j(S^i)\geq 120n\cdot W_{i+2}/\varepsilon^2$. Then, equation (5.17) yields $(\rho+\varepsilon)\cdot u_j(S^m)\geq u_j(S^m_{-j},s'_j)$, as desired. □

Also, assuming that all calls of routine ROUND() are successful, Lemma 5.5.5 uses Claims 5.5.1, 5.5.2, and 5.5.3 to bound the total increase of the potential Φ_{R_i} in phase i and Claim 5.2.2 and Lemma 5.4.3 to show that this increase happens after a small number of local and global moves.

Lemma 5.5.5. *If no call to routine ROUND() ever fails, the algorithm terminates after performing at most $\mathscr{O}(n^2/\varepsilon)$ τ-moves, at most $\mathscr{O}(n^3/\varepsilon^4)$ calls to routine SDP(), and at most $\frac{414720n^3}{\varepsilon^4}$ calls to routine ROUND().*

Proof. We will first bound the increase in the potential Φ_{R_i} of phase $i > 0$, proving that

$$\Phi_{R_i}(S^i) - \Phi_{R_i}(S^{i-1}) \leq 18n \cdot W_{i+1}/\varepsilon. \tag{5.18}$$

Observe that $\mathrm{CUT}_{R_i}(S^{i-1}) = A_i \cup C_i \cup D_i \cup H_i \cup J_i$ and $\mathrm{CUT}_{R_i}(S^i) = A_i \cup C_i \cup D_i \cup I_i \cup K_i$. Hence, $\Phi_{R_i}(S^{i-1}) = w(A_i) + w(C_i) + w(D_i) + w(H_i) + w(J_i)$, $\Phi_{R_i}(S^i) = w(A_i) + w(C_i) + w(D_i) + w(I_i) + w(K_i)$ and, thus,

$$\Phi_{R_i}(S^i) - \Phi_{R_i}(S^{i-1}) = -w(H_i) + w(I_i) - w(J_i) + w(K_i). \tag{5.19}$$

We now multiply the inequalities in Claims 5.5.1, 5.5.2, and 5.5.3 by $\frac{\sigma(\tau-1)}{\tau-\sigma}$, $\frac{\tau-1}{\tau-\sigma}$, and $\frac{\sigma-1}{\tau-\sigma}$, respectively, and sum them up. We obtain

$$\begin{aligned}&\frac{\sigma(\tau-1)}{\tau-\sigma} \cdot w(C_i) + \frac{(\sigma-1)(\tau-1)}{\tau-\sigma} \cdot w(D_i) + \frac{(\sigma+1)(\tau-1)}{\tau-\sigma} \cdot w(F_i) - w(H_i)\\ &+ w(I_i) + \frac{2\tau\sigma-\tau-\sigma}{\tau-\sigma} \cdot w(J_i) + w(K_i) \leq \frac{\sigma(\tau-1)}{\tau-\sigma} \cdot n \cdot W_{i+1} \leq 18n \cdot W_{i+1}/\varepsilon.\end{aligned}$$

The proof of inequality (5.18) follows by observing that all coefficients in the RHS of this last inequality besides that of term $w(H_i)$ are positive and thus the RHS is lower-bounded by the LHS of (5.19).

By Claim 5.2.2, we have that a τ-move increases the potential by at least $W_{i+1}/4$. By the bound on the potential increase in the whole phase i from (5.18), we obtain that no more than $72n/\varepsilon$ τ-moves take place in phase i. The bound on the total number of τ-moves follows since there are at most n phases containing some player.

Similarly, by Lemma 5.4.3, we have that a successful call to routine ROUND($\mathscr{G}, S, B_{i+1}$) at phase $i > 0$ increases the potential by at least $\varepsilon \cdot W_{i+2}/48$. By the bound on the potential increase in the whole phase i from (5.18) and the relation $W_{i+1}/W_{i+2} = \Delta = 480n/\varepsilon^2$, we obtain that no more than $414720n^2/\varepsilon^4$ calls to routine ROUND($\mathscr{G}, S, B_{i+1}$) take place in phase i. Again, the bound on the total number of calls to ROUND() follows since there are at most n phases containing some player.

Regarding the number of calls to routine SDP() within a phase, we claim that this is at most the number of τ-moves plus twice the number of calls to routine ROUND() plus 1. Indeed, in each execution of the routine GLOBAL() in which ROUND() is executed $t > 0$ times, there are at most $t + 1 \leq 2t$ executions of SDP(). Also, in each execution of the routing GLOBAL() in which ROUND() is not executed, the number of executions of GLOBAL() (and, consequently, of SDP()) is at most the number of τ-moves, besides in the last round of the phase, where we do not have any τ-move but there is one SDP() call. Hence, a bound of $\mathscr{O}(n^2/\varepsilon^4)$ on the number of SDP() calls in a phase follows by the bounds on the number of τ-moves and ROUND() calls in a phase. The bound of $\mathcal{O}(n^3/\varepsilon^4)$ on the total number of executions of SDP() follows since there are at most n phases containing some player. □

To complete the proof of Theorem 5.3.1 after having proved Lemmas 5.5.4 and 5.5.5, it suffices to show that the probability that a call to routine ROUND() fails to identify a set of players whose change of strategies increases the potential considerably is at most $1/n$. This can be done by a simple application of the union bound: there are at most $\frac{414720n^3}{\varepsilon^4}$ executions of routine ROUND() and, by Lemma 5.4.3, each of them fails with probability at most $\frac{\varepsilon^4}{414720n^4}$.

5.6 Properties of randomized rounding

In this section, we prove the two inequalities of Lemma 5.4.2. Let $\theta_j, \theta_k \in [0,\pi]$ be the angles vectors v_j and v_k form with vector $\widehat{v}$ and ϕ be the rotation angle vector v_k forms with the two-dimensional plane defined by vectors v_j and $\widehat{v}$. Without loss of generality, we can assume that $\theta_j \geq \theta_k$ and, furthermore, that vectors $\widehat{v}$, v_j, and v_k are lying in three dimensions and are defined as $\widehat{v} = (1,0,0)$, $v_j = (\cos\theta_j, \sin\theta_j, 0)$, and $v_k = (\cos\theta_k, \sin\theta_k \cdot \cos\phi, \sin\theta_k \cdot \sin\phi)$. Denoting the rotation function by f (i.e., $f(\theta) = \frac{\pi}{2}\cdot(1-\cos\theta)$ for $\theta \in [0,\pi]$), we have that the rotated vectors of v_j and v_k are $v'_j = (\cos f(\theta_j), \sin f(\theta_j), 0)$ and $v'_k = (\cos f(\theta_k), \sin f(\theta_k)\cdot\cos\phi, \sin f(\theta_k)\cdot\sin\phi)$, respectively. These definitions yield

$$v'_j \cdot v'_k = \cos f(\theta_j)\cdot\cos f(\theta_k) + \sin f(\theta_j)\cdot\sin f(\theta_k)\cdot\cos\phi \tag{5.20}$$

and

$$v_j \cdot v_k = \cos\theta_j \cdot \cos\theta_k + \sin\theta_j \cdot \sin\theta_k \cdot \cos\phi. \tag{5.21}$$

The probability that a random hyperplane separates the rotated vectors v'_j and v'_k is proportional to the angle between them, i.e., $\Pr[\mathrm{XOR}(j,k)] = \frac{\arccos(v'_j\cdot v'_k)}{\pi}$.

Proof of Inequality (5.4) in Lemma 5.4.2

To prove inequality (5.4), it suffices to prove that

$$\frac{\arccos(v'_j \cdot v'_k)}{\pi} \geq \frac{1 - v_j \cdot v_k}{4},$$

or, equivalently,

$$v'_j \cdot v'_k - \cos\left(\frac{\pi}{4}\cdot(1 - v_j\cdot v_k)\right) \leq 0. \tag{5.22}$$

By (5.20) and (5.21), and by setting $g(x,y,\phi) = \cos x\cdot\cos y + \sin x\cdot\sin y\cdot\cos\phi$, the LHS of (5.22) becomes

$$g(f(\theta_j), f(\theta_k), \phi) - \cos\left(\frac{\pi}{4}\cdot(1 - g(\theta_j,\theta_k,\phi))\right).$$

Observe that g is a linear function in $\cos\phi$, taking values in $[-1,1]$. To see why, observe that $g(x,y,\phi) \le g(x,y,0) = \cos(x-y)$ and $g(x,y,\phi) \ge g(x,y,\pi) = \cos(x+y)$. Now the derivative of the LHS of (5.22) with respect to $\cos\phi$ is

$$\frac{\partial g(f(\theta_j), f(\theta_k), \phi)}{\partial \cos\phi} - \frac{\partial g(\theta_j, \theta_k, \phi)}{\partial \cos\phi} \cdot \frac{\pi}{4} \cdot \sin\left(\frac{\pi}{4}(1 - g(\theta_j, \theta_k, \phi))\right),$$

which is increasing in $\cos\phi$. To see why, observe that the derivative of function $g(x,y,\phi)$ with respect to $\cos\phi$ is equal to $\sin x \cdot \sin y$, which is non-negative when $x,y \in [0,\pi]$, and the quantity inside the sin is decreasing in $\cos\phi$, taking values between 0 and $\pi/2$. We conclude that the LHS of (5.22) is convex in $\cos\phi$. Thus, it is maximized when either $\cos\phi = 1$ or $\cos\phi = -1$ and it suffices to prove inequality (5.22) for $\cos\phi \in \{-1,1\}$. For $\cos\phi = 1$, we have $g(x,y,\phi) = \cos(x-y)$, the LHS of (5.22) becomes

$$\cos(f(\theta_j) - f(\theta_k)) - \cos\left(\frac{\pi}{4} \cdot (1 - \cos(\theta_j - \theta_k))\right),$$

and (5.22) is equivalent to

$$|f(\theta_j) - f(\theta_k)| - \frac{\pi}{4} \cdot (1 - \cos(\theta_j - \theta_k)) \ge 0. \tag{5.23}$$

For $\cos\phi = -1$, we have $g(x,y,\phi) = \cos(x+y)$, the LHS of (5.22) becomes

$$\cos(f(\theta_j) + f(\theta_k)) - \cos\left(\frac{\pi}{4} \cdot (1 - \cos(\theta_j + \theta_k))\right),$$

and (5.22) is equivalent to

$$f(\theta_j) + f(\theta_k) - \frac{\pi}{4} \cdot (1 - \cos(\theta_j + \theta_k)) \ge 0. \tag{5.24}$$

We will complete the proof by showing that both (5.23) and (5.24) are true for the particular rotation function $f(\theta) = \frac{\pi}{2} \cdot (1 - \cos(\theta))$ we use. Without loss of generality, assume that $0 \le \theta_k \le \theta_j = \theta_k + x \le \pi$ with $x \in [0, \pi - \theta_k]$. Then, using our specific rotation function, the LHS of inequality (5.23) can be written as

$$\begin{aligned}\frac{\pi}{4} \cdot (2\cos\theta_k - 2\cos(\theta_k + x) - 1 + \cos x) &= \frac{\pi}{2} \cdot \left(2\sin\left(\theta_k + \frac{x}{2}\right) \cdot \sin\frac{x}{2} - \sin^2\frac{x}{2}\right) \\ &\ge \frac{\pi}{2} \cdot \sin^2\frac{x}{2} \ge 0,\end{aligned}$$

which proves inequality (5.23). The first inequality follows since our assumption for θ_k and x implies that $\frac{x}{2} \le \theta_k + \frac{x}{2} \le \pi - \frac{x}{2}$, which yields $\sin\left(\theta_k + \frac{x}{2}\right) \ge \sin\frac{x}{2}$.

The LHS of inequality (5.24) becomes

$$\begin{aligned}&\frac{\pi}{2} \cdot (2 - \cos(\theta_k + x) - \cos\theta_k) - \frac{\pi}{4} \cdot (1 - \cos(2\theta_k + x)) \\ &= \frac{\pi}{2} \cdot \left(1 - 2\cos\left(\theta_k + \frac{x}{2}\right) \cdot \cos\frac{x}{2} + \cos^2\left(\theta_k + \frac{x}{2}\right)\right).\end{aligned}$$

Since $\cos\frac{x}{2}$ takes values in $[-1,1]$, the quantity inside this last parenthesis is between the values $\left(1 - \cos\left(\theta_k + \frac{x}{2}\right)\right)^2$ and $\left(1 + \cos\left(\theta_k + \frac{x}{2}\right)\right)^2$, which are both clearly non-negative. The proof of inequality (5.24) follows. □

Proof of Inequality (5.5) in Lemma 5.4.2

To prove inequality (5.5), it suffices to prove that

$$\frac{\arccos(v'_j \cdot v'_k)}{\pi} - \frac{1}{8} \cdot (3\rho - 1 + (\rho - 1) \cdot v_j \cdot \widehat{v} + (\rho - 1) \cdot v_k \cdot \widehat{v} - (\rho + 1) \cdot v_j \cdot v_k) \leq 0. \tag{5.25}$$

We have verified (5.25) in its general form by extensive numerical computations using Mathematica (version 13.0) and explain how we do so later in this section. Until then, we present a formal analysis for the simpler case where the vectors v_j, v_k, and $\widehat{v}$ lie on the same 2-dimensional plane. This includes the values of parameters θ_j, θ_k, and ϕ for which inequality (5.25) is tight and has allowed us to compute the approximation factor ρ explicitly, as a function of a quadratic equation. We distinguish between two cases, depending on whether ϕ is equal to 0 or π.

Formal proof of inequality (5.25) when $\phi = 0$. In this case, equations (5.20) and (5.21) yield

$$v'_j \cdot v'_k = \cos(f(\theta_j) - f(\theta_k)) = \cos\left(\frac{\pi}{2} \cdot (\cos\theta_k - \cos\theta_j)\right)$$

and $v_j \cdot v_k = \cos(\theta_j - \theta_k)$, respectively, and the LHS of (5.25) becomes

$$\begin{aligned}
&\frac{1}{2} \cdot (\cos\theta_k - \cos\theta_j) - \frac{3\rho - 1}{8} - \frac{\rho - 1}{8} \cdot (\cos\theta_j + \cos\theta_k) + \frac{\rho + 1}{8} \cdot \cos(\theta_j - \theta_k) \\
&= -\frac{\rho}{2} + \sin\left(\frac{\theta_j - \theta_k}{2}\right) \cdot \sin\left(\frac{\theta_j + \theta_k}{2}\right) - \frac{\rho - 1}{4} \cdot \cos\left(\frac{\theta_j - \theta_k}{2}\right) \cdot \cos\left(\frac{\theta_j + \theta_k}{2}\right) \\
&\quad + \frac{\rho + 1}{4} \cdot \cos^2\left(\frac{\theta_j - \theta_k}{2}\right).
\end{aligned} \tag{5.26}$$

Applying the inequality $x \cdot y \leq c \cdot x^2 + \frac{1}{4c} \cdot y^2$ for $c = \frac{1+\sqrt{13}}{4}$, $x = \sin\left(\frac{\theta_j - \theta_k}{2}\right)$, and $y = \sin\left(\frac{\theta_j + \theta_k}{2}\right)$, we obtain

$$\begin{aligned}
&\sin\left(\frac{\theta_j - \theta_k}{2}\right) \cdot \sin\left(\frac{\theta_j + \theta_k}{2}\right) \\
&\leq \frac{1 + \sqrt{13}}{4} \cdot \sin^2\left(\frac{\theta_j - \theta_k}{2}\right) + \frac{\sqrt{13} - 1}{12} \cdot \sin^2\left(\frac{\theta_j + \theta_k}{2}\right) \\
&= \frac{1 + 2\sqrt{13}}{6} - \frac{1 + \sqrt{13}}{4} \cdot \cos^2\left(\frac{\theta_j - \theta_k}{2}\right) - \frac{\sqrt{13} - 1}{12} \cdot \cos^2\left(\frac{\theta_j + \theta_k}{2}\right).
\end{aligned}$$

Using this inequality and the fact that $\rho = \frac{1+2\sqrt{13}}{3}$, the RHS of equation (5.26) is at most

$$-\frac{\rho}{2} + \frac{1 + 2\sqrt{13}}{6} - \frac{\sqrt{13} - 1}{12} \cdot \cos^2\left(\frac{\theta_j + \theta_k}{2}\right)$$

$$
\begin{aligned}
&-\frac{\rho-1}{4}\cdot\cos\left(\frac{\theta_j+\theta_k}{2}\right)\cdot\cos\left(\frac{\theta_j-\theta_k}{2}\right)\\
&-\left(\frac{1+\sqrt{13}}{4}-\frac{\rho+1}{4}\right)\cdot\cos^2\left(\frac{\theta_j-\theta_k}{2}\right)\\
&=-\frac{\sqrt{13}-1}{12}\cdot\left(\cos\left(\frac{\theta_j+\theta_k}{2}\right)+\cos\left(\frac{\theta_j-\theta_k}{2}\right)\right)^2,
\end{aligned}
$$

which is clearly non-positive, completing the proof.

We remark that this last RHS expression is equal to 0 only when $\theta_j=\pi$. In this case, the RHS of (5.26) becomes

$$
\begin{aligned}
&-\frac{\rho}{2}+\cos^2\left(\frac{\theta_k}{2}\right)+\frac{\rho-1}{4}\cdot\sin^2\left(\frac{\theta_k}{2}\right)+\frac{\rho+1}{4}\cdot\sin^2\left(\frac{\theta_k}{2}\right)\\
&=\frac{2-\rho}{2}\cdot\left(1-\sin^2\left(\frac{\theta_k}{2}\right)\right),
\end{aligned}
$$

implying that (π,π) is the only pair of values for θ_j and θ_k for which inequality (5.25) is tight.

Formal proof of inequality (5.25) when $\phi=\pi$. Then, equations (5.20) and (5.21) yield

$$
v'_j\cdot v'_k=\cos\left(f(\theta_j)+f(\theta_k)\right)=\cos\left(\pi-\frac{\pi}{2}\cdot(\cos\theta_j+\cos\theta_k)\right).
$$

and $v_j\cdot v_k=\cos(\theta_j+\theta_k)$, respectively. Notice that

$$
\frac{\arccos(v'_j\cdot v'_k)}{\pi}=1-\frac{1}{2}\cdot|\cos\theta_j+\cos\theta_k|.
$$

So, we distinguish between two subcases, depending on whether $\cos\theta_j+\cos\theta_k$ is positive or not (and, consequently, on whether $\theta_j+\theta_k$ is lower than π or not).

If $\cos\theta_j+\cos\theta_k>0$ (and $\theta_j+\theta_k<\pi$), then $\frac{\arccos(v'_j\cdot v'_k)}{\pi}=1-\frac{1}{2}\cdot(\cos\theta_j+\cos\theta_k)$ and the LHS of inequality (5.25) becomes

$$
\begin{aligned}
&\frac{9-3\rho}{8}-\frac{\rho+3}{8}\cdot(\cos\theta_j+\cos\theta_k)+\frac{\rho+1}{8}\cdot\cos(\theta_j+\theta_k)\\
&=\frac{2-\rho}{2}-\frac{\rho+3}{4}\cdot\cos\left(\frac{\theta_j-\theta_k}{2}\right)\cdot\cos\left(\frac{\theta_j+\theta_k}{2}\right)+\frac{\rho+1}{4}\cos^2\left(\frac{\theta_j+\theta_k}{2}\right)\\
&\le\frac{2-\rho}{2}-\frac{1}{2}\cdot\cos^2\left(\frac{\theta_j+\theta_k}{2}\right)<0.
\end{aligned}
$$

The first inequality follows since $\cos\left(\frac{\theta_j+\theta_k}{2}\right)>0$ and $\theta_k\ge 0$ which implies $\cos\left(\frac{\theta_j-\theta_k}{2}\right)\ge\cos\left(\frac{\theta_j+\theta_k}{2}\right)$.

Otherwise, if $\cos\theta_j + \cos\theta_k \le 0$ (and $\theta_j + \theta_k \ge \pi$), then $\frac{\arccos(v'_j \cdot v'_k)}{\pi} = 1 + \frac{1}{2} \cdot (\cos\theta_j + \cos\theta_k)$ and the LHS of inequality (5.25) becomes

$$\begin{aligned}
&\frac{9-3\rho}{8} + \frac{5-\rho}{8} \cdot (\cos\theta_j + \cos\theta_k) + \frac{\rho+1}{8} \cdot \cos(\theta_j + \theta_k) \\
&= \frac{2-\rho}{2} + \frac{5-\rho}{4} \cdot \cos\left(\frac{\theta_j - \theta_k}{2}\right) \cdot \cos\left(\frac{\theta_j + \theta_k}{2}\right) + \frac{\rho+1}{4} \cos^2\left(\frac{\theta_j + \theta_k}{2}\right) \\
&\le \frac{2-\rho}{2} + \frac{\rho-2}{2} \cdot \cos^2\left(\frac{\theta_j + \theta_k}{2}\right) \le 0.
\end{aligned}$$

The first inequality follows since $\cos\left(\frac{\theta_j+\theta_k}{2}\right) \le 0$ and $\cos\left(\frac{\theta_j-\theta_k}{2}\right) \ge -\cos\left(\frac{\theta_j+\theta_k}{2}\right)$. To see why this last inequality is true, observe that the assumption $\theta_j \in [0,\pi]$ implies that $\frac{\theta_j-\theta_k}{2} \le \pi - \frac{\theta_j+\theta_k}{2}$ and, hence, $\cos\left(\frac{\theta_j-\theta_k}{2}\right) \ge \cos\left(\pi - \frac{\theta_j+\theta_k}{2}\right) = -\cos\left(\frac{\theta_j+\theta_k}{2}\right)$.

Notice that, again, the last inequality is tight only for the pair of values (π,π) for θ_j and θ_k.

Verifying inequality (5.25) numerically. We now consider the general case. Define the function

$$\begin{aligned}
\lambda(x,y,z) = {} & \frac{\arccos(\cos f(x) \cdot \cos f(y) + z \cdot \sin f(x) \cdot \sin f(y))}{\pi} - \frac{3\rho - 1}{8} \\
& - \frac{\rho-1}{8} \cdot \cos x - \frac{\rho-1}{8} \cdot \cos y + \frac{\rho+1}{8} \cdot \cos x \cdot \cos y + \frac{\rho+1}{8} \cdot z \cdot \sin x \cdot \sin y,
\end{aligned}$$

for $x \in [0,\pi]$, $y \in [0,\pi]$, and $z \in [L,H]$ with $L = \frac{-\cos f(x)\cos f(y) - 1}{\sin f(x) \sin f(y)}$ and $H = \frac{-\cos f(x)\cos f(y)+1}{\sin f(x)\sin f(y)}$. Notice that $[-1,1]$ is a subinterval of $[L,H]$. Now, observe that $\lambda(\theta_j, \theta_k, \cos\phi)$ is identical to the LHS of inequality (5.25), after substituting $v'_j \cdot v'_k$ and $v_j \cdot v_k$ using (5.20) and (5.21), respectively. The derivative of $\lambda(x,y,z)$ with respect to z is

$$\begin{aligned}
&\frac{\partial \lambda(x,y,z)}{\partial z} = \\
&\quad - \frac{\sin f(x) \cdot \sin f(y)}{\pi \cdot \sqrt{1 - (\cos f(x) \cdot \cos f(y) + z \cdot \sin f(x) \cdot \sin f(y))^2}} + \frac{\rho+1}{8} \cdot \sin x \cdot \sin y,
\end{aligned}$$

and has two roots:

$$z_1(x,y) = -\frac{\cos f(x) \cos f(y)}{\sin f(x) \sin f(y)} - \frac{1}{\sin f(x) \cdot \sin f(y)} \cdot \sqrt{1 - \left(\frac{8 \sin f(x) \sin f(y)}{\pi(\rho+1) \cdot \sin x \sin y}\right)^2}$$

and

$$z_2(x,y) = -\frac{\cos f(x) \cos f(y)}{\sin f(x) \sin f(y)} + \frac{1}{\sin f(x) \cdot \sin f(y)} \cdot \sqrt{1 - \left(\frac{8 \sin f(x) \sin f(y)}{\pi(\rho+1) \cdot \sin x \sin y}\right)^2}.$$

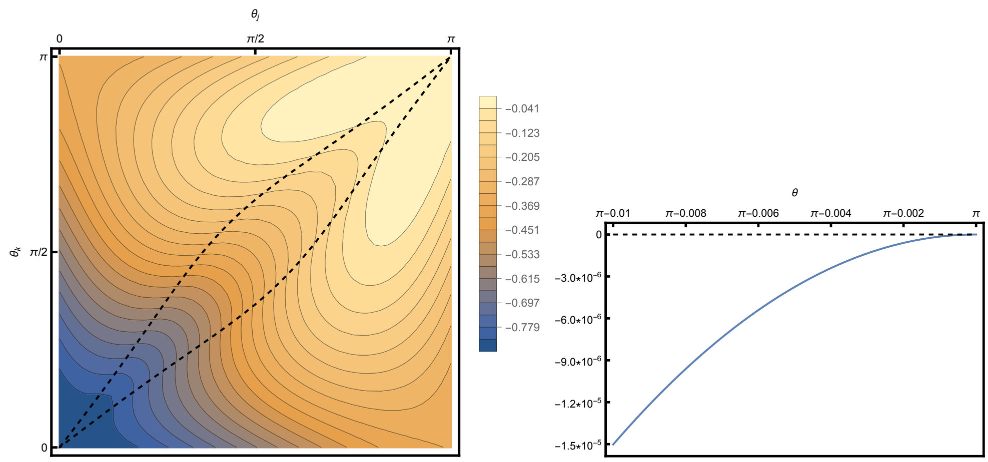

Figure 5.1: Numerical evaluation of the quantity $\max_{\phi\in[0,\pi]}\lambda(\theta_j,\theta_k,\cos\phi)$ for all pairs $(\theta_j,\theta_k)\in[0,\pi]^2$ (left) and pairs (θ_j,θ_k) with $\theta_j=\theta_k=\theta\in[0,\pi]$ (right).

By inspecting the derivative, we can see that the function $\lambda(x,y,z)$ is decreasing as z takes values in $[L,z_1(x,y)]$, has a local minimum at $z_1(x,y)$, increases as z takes values in $[z_1(x,y),z_2(x,y)]$, has a local maximum at $z_2(x,y)$, and decreases in $[z_2(x,y),H]$. To prove inequality (5.25), it suffices to show that

$$\max_{\theta_j,\theta_k,\phi\in[0,\pi]}\lambda(\theta_j,\theta_k,\cos\phi)\leq 0. \tag{5.27}$$

Given the formal bounds on $\lambda(\theta_j,\theta_k,1)$ and $\lambda(\theta_j,\theta_k,-1)$ in our analysis above, we complete the proof of (5.27) by showing through extensive numerical computations using Mathematica that the quantity

$$\max_{\substack{\theta_j,\theta_k\in[0,\pi]:\\ z_2(\theta_j,\theta_k)\in[-1,1]}}\lambda(\theta_j,\theta_k,z_2(\theta_j,\theta_k)) \tag{5.28}$$

is upper-bounded by 0. The contour plot at the left of Figure 5.1 shows the values of the quantity $\max_{\phi\in[0,\pi]}\lambda(\theta_j,\theta_k,\cos\phi)$ for all pairs $(\theta_j,\theta_k)\in[0,\pi]^2$. The values in the area defined by the dashed curves are those computed using (5.28); the remaining values are the maximum between $\lambda(\theta_j,\theta_k,1)$ and $\lambda(\theta_j,\theta_k,-1)$. In this way, we verify that (5.27) is true with the inequality being strict everywhere besides the single point (π,π). Recall that the formal analysis above has verified that the pair (π,π) is the only point at which inequality (5.25) is tight when vectors v_j, v_k, and $\widehat{v}$ lie on the same 2-dimensional plane. The plot at the right of Figure 5.1 depicts the value of $\lambda(\theta_j,\theta_k,z_2(\theta_j,\theta_k))$ for $\theta_j=\theta_k=\theta$ being very close to π.

5.7 Extensions and open problems

We have shown how to compute 2.7371-approximate pure Nash equilibria in cut games in polynomial time, providing the first improvement to the 3-approximation bound of Bhalgat et al. [20]. With a few minor modifications, our algorithm works for party affiliation games as well. These are generalizations of cut games, in which an edge can be either *enemy* or *friend*, giving utility to the players controlling its endpoints when they lie on different sides or the same side of the cut, respectively.

As future work, it would be interesting to explore whether other rotation functions, possibly combined with more sophisticated rounding techniques (like those discussed in [109]), can give further improved results. How close to almost exact equilibria can we go? Even though we have made progress on this question here, the gap is still large. Exploring the inapproximability of equilibria for cut games (possibly, by strengthening the PLS-hardness results for local MAX-CUT in bounded-degree graphs; see, e.g., [62]) is another problem that deserves investigation.

Chapter 6

Random 3-SAT formulas with k-wise independent clauses

The problem of identifying the satisfiability threshold of random 3-SAT formulas has received a lot of attention during the last decades and has inspired the study of other threshold phenomena in random combinatorial structures. The classical assumption in this line of research is that, for a given set of n Boolean variables, each clause is drawn uniformly at random among all sets of three literals from these variables, independently from other clauses. Here, we keep the uniform distribution of each clause, but deviate significantly from the independence assumption and consider richer families of probability distributions. For integer parameters n, m, and k, we denote by $\mathscr{F}_k(n,m)$ the family of probability distributions that produce formulas with m clauses, each selected uniformly at random from all sets of three literals from the n variables, so that the clauses are k-wise independent. Our aim is to make general statements about the satisfiability or unsatisfiability of formulas produced by distributions in $\mathscr{F}_k(n,m)$ for different values of the parameters n, m, and k.

Our technical results are as follows: First, all probability distributions in $\mathscr{F}_2(n,m)$ with $m \in \Omega(n^3)$ return unsatisfiable formulas with high probability. This result is tight. We show that there exists a probability distribution $\mathscr{D} \in \mathscr{F}_3(n,m)$ with $m \in O(n^3)$ so that a random formula drawn from $\mathscr{D}$ is almost always satisfiable. In contrast, for $m = \Omega(n^2)$, any probability distribution $\mathscr{D} \in \mathscr{F}_4(n,m)$ returns an unsatisfiable formula with high probability. This is our most surprising and technically involved result. Finally, for any integer $k \geq 2$, any probability distribution $\mathscr{D} \in \mathscr{F}_k(n,m)$ with $m = O(n^{1-1/k})$ returns a satisfiable formula with high probability.

6.1 Introduction

Satisfiability of propositional formulas (SAT) is one of the most renowned problems in theoretical computer science. It appeared in the first lists of NP-complete problems independently proposed by Cook and Levin, and is pivotal for many developments in modern complexity theory. Today, many lower bounds on the running time of algorithms rely on the Exponential Time Hypothesis for solving SAT [24, 53, 93, 96]. On the practical side, SAT solvers are frequently deployed in hardware circuit design, model checking, program verification, automated planning and scheduling, as well as in solving real-life instantiations of combinatorial optimization problems such as FCC spectrum auctions. Modern SAT solvers often find solutions to large industrial instances with thousands or even millions of variables despite the NP-hardness of the problem. However, there is still a large discrepancy between the performance of SAT solvers on those instances and theoretical average-case predictions, which have been studied in great depth under the line of research on *random SAT*.

Random SAT.

A j-CNF formula ϕ over n variables is composed of m OR-clauses, each containing exactly j literals of j different variables. In the most commonly studied random SAT model, a formula ϕ is generated uniformly at random from all possible j-CNF formulas over n variables and m clauses. The most prominent theoretical question related to random SAT is to identify the satisfiability threshold r_j such that $\lim_{n\to\infty} \mathbf{Pr}[\phi$ is satisfiable$]$ is equal to 0 when $m/n > r_j$, and equal to 1 when $m/n < r_j$. It has been established [46] that 2-SAT has $r_2 = 1$, and its phase transition window [23] is $m \in [n - \Theta(n^{1/3}), n + \Theta(n^{1/3})]$. For $j \geq 3$, the asymptotic j-SAT threshold was shown to be $2^j \log 2 - \frac{1}{2}(1 + \log 2) \pm o_j(1)$ as $j \to \infty$ [50] (improving previous results from [3]), while for large enough j the exact value of r_j was determined in [57]. However, the question of identifying r_j for small values of j remains open. In particular, random 3-SAT has attracted a lot of attention. For the lower bound part, it has been shown in a series of papers [1, 25, 46, 77, 88, 104] that $r_3 \geq 3.52$ (the currently best known bound is due to [88, 104]). The upper bound part is studied by [56, 59, 72, 103]; the currently best known bound is $r_3 < 4.49$ due to [56]. The estimate $r_3 \approx 4.26$ was derived from numerical experiments [115] (see also [41, 117]).

A more recent line of work [74–76, 123] extends the standard model of random j-SAT to non-uniform distributions. Their motivation comes from the empirical observation that, in practice, CNF formulas often have rather different frequencies/probabilities for the n variables to appear in each clause (following a power-law distribution instead of a uniform one). Namely, Friedrich and Rothenberger [75] proposed a non-uniform random model, where the literals $\{x_i, \overline{x_i}\}_{i\in[n]}$ are selected independently at random in each clause c of the random j-CNF with $\mathbf{Pr}[x_i \in c] = \mathbf{Pr}[\overline{x_i} \in c] = p_i$ and where probabilities $\mathbf{p} = (p_i)_{i\in[n]}$ may vary across different variables. They find satisfiability threshold $r_2(\mathbf{p})$ of non-uniform random 2-SAT for certain regimes depending on $\mathbf{p}$. However, the non-uniform model of [75] does not capture the community biases/correlations (i.e., the fact that certain variables are more likely to appear together in a clause), which are often observed in practice [13]. This leads us to the question of whether it is possible to relax the strong independence assumption in the existing random SAT literature.

Relaxation of independence.

We first observe that it does not make much sense to study distributions of SAT formulas with arbitrary correlations over the clauses. Indeed, by allowing correlation between several clauses, one may enforce that the random formula ϕ contains large fixed sub-formulas corresponding to NP-hard SAT variants. This would be at odds with our goal of studying average-case complexity. Therefore, we must keep a certain degree of independence in the distribution of instances. We propose to consider the relaxation of mutual independence over m clauses in a random formula ϕ *to k-wise independence* for a small constant k. To keep the new model tractable, we focus on 3-SAT and uniform distribution of literals within each clause. I.e., we assume that (i) every 3-OR-clause c of a random 3-CNF formula ϕ has three literals of three distinct variables drawn uniformly at random among all such triplets of literals and that (ii) given this marginal distribution of each clause $c \sim F_{\text{uni.}}$, the distribution $\mathscr{D}$ over the clause set C in ϕ is only k-wise independent instead of the mutually independent distribution $\mathscr{D}_{\text{Ind.}} = (F_{\text{uni.}})^{\otimes m}$ in the standard model. This is a natural generalization that has been considered in a number of different settings but, to the best of our knowledge, not in the context of random SAT. Note that the smaller k is, the bigger the set of possible distributions $\mathscr{D}$. Furthermore, for small values of k, a k-wise independent distribution $\mathscr{D}$ can still capture a large class of dependencies among clauses but at the same time does not allow correlation between any k-tuples of clauses. In mathematical terms, the family of discrete k-wise independent distributions naturally appears when we map the set of distributions to the set of their low-degree moments. Specifically, if a distribution $\mathscr{D}$ is supported on the n-dimensional binary cube[1] $\mathsf{supp}(D) = \{-1,1\}^n$, then all its moments of degree up to k can be described as $\mu(D) = (\mathbf{E}[\prod_{i \in S} x_i])_{|S| \le k}$. As low-degree moments (basically, the image of μ) are extremely important in statistical analysis, it is equally important to study the kernel of the aforementioned mapping, which exactly corresponds to the family of k-wise independent distributions. Let us provide additional justifications of our framework by discussing some of the theoretical work on random 3SAT and on other settings with similar k-wise independence relaxation.

Pseudo-randomness. Historically, the k-wise relaxation of independence has been actively used in the literature on derandomization and pseudo-randomness, as it allows to significantly reduce the amount of random bits needed to generate random objects. For example, Alon and Nussboim [9] consider random Erdős-Rényi graphs and examine the minimal degree k of independence needed to achieve a variety of graph properties and statistics (such as connectivity, existence of perfect matchings, existence of Hamiltonian cycles, clique and chromatic numbers, etc.) that match those in the mutually independent case. Benjamini et al. [18] consider similar questions for monotone boolean functions. The motivation in [9] comes from the fact that there are efficient constructions of k-wise independent distributions with "low degree of independence" (say

[1] Similar moment functions can be defined for any distribution with discrete marginals.

$k = O(\log n)$) that utilize only polylog(n) random bits, i.e., much fewer than the polynomial number of random bits required to generate mutually independent distributions. While some of this motivation can be applied to our setting of random 3-SAT, it is a conceptually different story. Indeed, the perspective of pseudo-random generation is through the lenses of "probability theory", where one controls the distributions and can simply choose one that satisfies necessary conditions such as, e.g., $(\log n)$-wise independence. On the other hand, our motivation stems from "statistics", as our ideal model should have a reasonable fit to empirical observations. So, we would like to use as minimal assumptions as possible and study small (constant) degrees of independence.

Refutability of 3-SAT. While the research on lower bounds for random 3-SAT often comes up with certain simple heuristics that efficiently find a satisfying assignment (see, e.g., the surveys by Achlioptas [2] and Flaxman [70]), it is extremely hard to find an efficient refutation of a unsatisfiable 3-SAT formula. Indeed, the common approach to refute a given SAT formula is proof in resolution. Chvatal and Szemeredi [47] first showed that a random 3-CNF formula with $m = \Theta(n)$ clauses (which is almost surely unsatisfiable) almost surely admits only exponential size proof in resolution. Later, Ben-Sasson and Wigderson [17] derived similar result for much larger $m = O(n^{3/2-\varepsilon})$. On the positive side, [73] gave the first polynomial time algorithm via spectral techniques that almost surely[2] refutes a random 3-SAT formula with $m = n^{3/2+\varepsilon}$ clauses. The best known bound on m is due to Feige and Ofek [66] who proved that, for a sufficiently large constant c, random 3-SAT formulas with $m = c \cdot n^{3/2}$ clauses can be almost surely refuted in polynomial time using another spectral graph algorithm. We note that a similar situation (extremely high probability of unsatisfiability for a random formula and inability to efficiently confirm it) is unlikely to happen in our k-clause independent model for constant k. Indeed, the main proof approach for dealing with arbitrary k-wise independent distribution is to define a k-wise statistic, which differentiates any satisfiable formula from a typical unsatisfiable one.

Testing k-wise independence. The property of k-wise independence of a distribution with n components can be tested using $n^{O(k)} = poly(n)$ many samples in polynomial time, when k is a constant [10, 132]. This is a useful property to have, as it allows one to verify with only polynomially many instances of random 3-SAT, whether these instances conform to k-wise independence or not.

Robust mechanism design. A recent line of work in robust mechanism design also considers families of k-wise independent Bayesian priors in single and multi-unit auctions [34, 60, 85, 87]. Their motivation is similar to ours, as they also rely on the statistical point of view to justify the extension of the results for

[2]Refutation in this case is an algorithm with one-sided error: it always refutes the formula correctly by producing certain certificates, or says that the formula might be correct.

mutual independent priors typically assumed in Bayesian mechanism design to k-wise independent ones.

Problem formulation

We consider random 3-CNF formulas with n variables generated from a distribution $\mathscr{D}$ over m clauses, where the mutual independence assumption over clauses is relaxed to k-wise independence. We use the term k-clause independence to refer to such distributions. We denote such families of distributions by $\mathscr{F}_k(n,m)$, where each $\mathscr{D} \in \mathscr{F}_k(n,m)$ has identical marginals uniformly distributed over all possible OR-clauses and those marginals are only assumed to be k-wise independent in $\mathscr{D}$. We would like to understand the following question for small values of k:

> How does the satisfiability threshold r_3 of random 3-SAT formulas behave under any k-clause independent distribution $\mathscr{D} \in \mathscr{F}_k(n,m)$?

As the distribution $\mathscr{D}$ is not unique, there might be a large gap between lower and upper estimates of r_3. To this end, we formally define the *lower satisfiability threshold* $\texttt{LST}_k(n)$ as an upper bound on m, such that a random formula ϕ drawn from a distribution in $\mathscr{F}_k(n,m)$ with $m \leq \texttt{LST}_k(n)$ clauses has $\mathbf{Pr}[\phi \text{ is satisfiable}] \geq \frac{2}{3}$. Similarly, the *upper satisfiability threshold* $\texttt{UST}_k(n)$ is a lower bound on m, such that the random formula ϕ with $m \geq \texttt{UST}_k(n)$ clauses has $\mathbf{Pr}[\phi \text{ is satisfiable}] \leq \frac{1}{3}$. What kind of bounds on upper $\texttt{UST}_k(n)$ and lower $\texttt{LST}_k(n)$ thresholds should we expect?

Reasonable expectations. The condition $\mathscr{D} \in \mathscr{F}_k(n,m)$ only says something about configurations of at most k clauses and does not put any other restrictions on the random formula $\phi \sim \mathscr{D}$. As the degree of independence k is a small constant, any argument that gives bounds on $\texttt{LST}_k$ or $\texttt{UST}_k$ can only rely on statistics of at most a constant number of clauses. Hence, it is rather likely that bounds on $\texttt{LST}_k$ and $\texttt{UST}_k$ come together with efficient procedures of, respectively, finding a satisfying assignment for a random formula ϕ, or certifying that ϕ is not satisfiable. Hence, given the prior work on random 3-SAT for $\mathscr{D}_{\text{Ind.}} \in \mathscr{F}_k(n,m)$, we get the following picture:

Upper satisfiability threshold. The best known result for refuting 3-CNF formulas efficiently is due to Feige and Ofek [66], who show how to do it only for a fairly large number of clauses $m = c \cdot n^{3/2}$. Furthermore, for any smaller number of clauses $m = O(n^{3/2-\varepsilon})$, a random 3-CNF formula is likely to have only exponential in n proof size for any unsatisfiability proof in resolution [17]. Hence, it is out of reach to aim for a better bound on $\texttt{UST}_k(n)$ than $O(n^{3/2})$ while relying only on k-wise independence for some constant k. In fact, the best known positive result on *efficiently computable proofs* of unsatisfiability in resolution is due to Beame et al. [16], who show that an ordered DLL algorithm executed on a random 3-SAT instance with $m = \Omega(n^2/\log n)$ clauses terminates in polynomial time.

Lower satisfiability threshold. As the proofs for the lower bounds on r_3 often establish simple procedures that find satisfying assignments with high probability, it is still possible that $\mathtt{LST}_k(n)$ is of similar order $\Theta(n)$ as the lower bounds on r_3 for $\mathscr{D}_{\text{Ind.}}$. Thus, the most ambitious result would be to show that $\mathtt{LST}_k(n) \le c_k \cdot n$ for constant c_k that increases with k. A more modest goal is to aim for $\mathtt{LST}_k(n) = o(n)$ for a constant k, where $\mathtt{LST}_k(n) \to \Theta(n)$ as $k \to +\infty$.

Our results

We obtain the following bounds on the lower and upper satisfiability thresholds $\mathtt{LST}_k(n)$ and $\mathtt{UST}_k(n)$ for various values of k.

Lower satisfiability thresholds. We show (in Theorem 6.6.1) that $\mathtt{LST}_k(n) \ge \Omega(n^{1-1/k})$ for any $k \ge 2$. I.e., any k-clause independent random formula is satisfiable with high probability if it contains at most $O(n^{1-1/k})$ clauses. The argument is simple: for any k-clause independent distribution, we look at the 3-uniform hypergraph that corresponds to the variables of a random formula ϕ produced according to this distribution, and argue that this graph does not have Berge-cycles, with high probability. We also provide an informal justification that this bound is asymptotically tight, i.e., that $\mathtt{LST}_k(n) = O(n^{1-1/k})$. Specifically, we outline a plausible approach for constructing a k-clause independent distribution with $m = O(n^{1-1/k})$ clauses such that most of its formulas are unsatisfiable. Our approach is built upon existing constructions of dense hyper-graphs with large girth. It is interesting to note that, in both the proof of the $\mathtt{LST}_k(n) = \Omega(n^{1-1/k})$ result and the approach for showing that $\mathtt{LST}_k(n) = O(n^{1-1/k})$, we only need to consider variables and can completely ignore the distribution over the literals.

Upper satisfiability thresholds. We first consider small degrees of independence, i.e., $k \in \{2,3\}$. In both cases, we show that $\mathtt{UST}_k(n) = \Theta(n^3)$, meaning that one needs almost all possible clauses in a 3-clause (as well as 2-clause) independent formula to ensure that it is unsatisfiable (see Theorem 6.3.1 and Theorem 6.4.1). The most nontrivial part is to construct the distribution $\mathscr{D} \in \mathscr{F}_3(n,m)$ with $m = \Theta(n^3)$ and $\mathbf{Pr}[\phi \text{ is satisfiable}] \ge \frac{2}{3}$. Our construction is based on “3-XOR formulas” (i.e., OR-clauses that have either one or three literals that are satisfied by a randomly planted truth assignment), which aligns well with the intuition developed in previous work [64, 66]. The main technical difficulty is to ensure k-clause independence by adding a small fraction of unsatisfiable instances and checking all 3-wise statistics.

Our most exciting and technically involved result (see Theorem 6.5.1) is our proof that $\mathtt{UST}_4(n) = O(n^2)$, i.e., a random formula $\phi \sim \mathscr{D}$ with $m = O(n^2)$ clauses is unsatisfiable with large probability for any 4-clause independent distribution $\mathscr{D} \in \mathscr{F}_4(n,m)$. It is worth noting that such a bound is much harder to get under the 4-wise independence assumption than in the case of a mutually independent distribution $\mathscr{D}_{\text{Ind.}}$. Indeed, Feige and Ofek [66] describe a very simple refutation algorithm for $m = \Theta(n^2)$ that fixes a variable x and considers all clauses containing x or $\bar{x}$ (there will be $\Theta(n)$ such clauses in expectation). Then, after deleting x (or $\bar{x}$), one can reduce the problem to the refutation of the respective random 2-CNF sub-instance, which can be easily verified in polynomial time and has a low satisfiability threshold of $r_2 = 1$. This simple approach obviously fails for 4-clause independent distributions. We instead construct a bipartite multigraph $G(\phi)$ between pairs of distinct literals on one side and all singleton literals on the other, in which every OR-clause in ϕ corresponds to three different edges. We then carefully examine the statistic $\kappa(\phi)$ that counts $K_{2,2}$ subgraphs in $G(\phi)$ for a random $\phi \sim \mathscr{D}$. We find that the expected value of $\kappa(\phi)$ for random ϕ is only slightly larger than its absolute minimal value, while at the same time $\kappa(\phi)$ is significantly larger than its expectation when ϕ is satisfiable. Our argument bears certain similarities with the argument in [66], which also looked at intersections of two literals between pairs of clauses but used the 3-XOR principle and a differently constructed non-bipartite graph.

Roadmap

The rest of the paper is structured as follows. We begin with preliminary definitions and notation in Section 6.2. Then, we warm up with our tight bounds on the upper satisfiability threshold $\mathtt{UST}_2(n)$ in Section 6.3. Our lower bound on $\mathtt{UST}_3(n)$ is proved in Section 6.4 while our upper bound on $\mathtt{UST}_4(n)$ follows in Section 6.5. Section 6.6 is devoted to the study of the lower satisfiability threshold.

6.2 Preliminaries

Let $x_1, x_2, \cdots, x_n$ be n boolean variables. A literal ℓ is a boolean variable or the negation of it. For convenience, we usually represent a literal as a variable-sign pair, i.e. $\ell = (x_i, s)$, where s is the positive sign + if the literal is a boolean variable and the negative sign - if it is its negation. We define $\Sigma(n) = \{+,-\}^n$. An instance of the *Satisfiability* problem in conjunctive normal form (or *SAT instance*, for short) is a boolean formula over a subset of the n variables which is a conjunction of disjunctive clauses, each clause containing literals with different variables. In a 3-SAT instance, every clause has exactly 3 literals. Each SAT instance C can be described as a multiset of clauses. We denote the size of an instance C as $m = |C|$.

Given the number of variables n, let $X(n)$ be the set of variables. Let $T(n)$ be the set of all unordered triplets of variables $T(n) = \{(x_i, x_j, x_k) \mid x_i, x_j, x_k \in X(n), i < j < k\}$ and $\mathscr{T}(n)$ be the set of all possible clauses, i.e., all triplets with literals of different variables. We will usually omit the dependency of X, T, and $\mathscr{T}$ on n, when the value of n is clear from the context. To refer to the variable (the set of variables) or the sign (the set of signs) of a literal (a clause $c \in C$), we define operators $\mathrm{Var}(\cdot)$ and $\mathrm{Sign}(\cdot)$, so that if, e.g., $c = ((x_1,+),(x_2,-),(x_3,+))$, then $\mathrm{Var}(c) = (x_1, x_2, x_3)$ and $\mathrm{Sign}(c) = (+,-,+)$.

A *truth assignment* is a vector $\sigma \in \{0,1\}^n$, where 1 or 0 at the σ_i coordinate corresponds to the "true" or "false" value of x_i, respectively. A truth assignment σ is *satisfying* for an instance C when each clause $c \in C$ has at least one true literal under

σ. An instance C is *satisfiable* if there exists a satisfying assignment; otherwise, C is *unsatisfiable*.

The random 3-SAT model assumes that the instance is drawn from a probability distribution over all possible 3-SAT instances with m clauses and n variables. The main object of study in such a model is the probability of such a random instance being satisfiable as a function of n and m.

Random 3-SAT without mutual independence

In the standard random 3-SAT model, each clause is drawn uniformly at random among all clauses in $\mathscr{T}(n)$, independently from the other clauses. A constructive definition is given by Model 6.1. We denote the distribution over instances generated by Model 6.1 as $\mathscr{D}_{\text{Ind.}}$.

MODEL 6.1: Selects a 3-SAT instance uniformly at random from all possible instances

Input: Integers $n \geq 3$ and $m \geq 1$.
Output: A 3-SAT instance with n variables and m clauses.

```
C ← ∅;
for l ← 1,...,m do
    pick a literal triplet (ℓ1, ℓ2, ℓ3) uniformly at random from 𝒯(n);
    c ← (ℓ1, ℓ2, ℓ3);
    C ← C ∪ {c};
end for
return C
```

Our main focus will be on k-wise relaxations of the independent distribution.

Definition 6.2.1 (k-clause independent random SAT)**.** *A distribution $\mathscr{D}$ for selecting a random SAT instance is k-clause independent if the set of any fixed k clauses $S \subseteq C$ is distributed uniformly at random over all k-tuples of possible clauses, i.e.,*

$$\Pr_{C\sim\mathscr{D}}\left[c_i = t_i, \forall i \in [k]\right] = \Pr_{C\sim\mathscr{D}_{Ind.}}\left[c_i = t_i, \forall i \in [k]\right] = \prod_{i\in[k]} \Pr\left[c_i = t_i\right].$$

for every $c_1, \ldots, c_k \in C$ and $t_1, \ldots, t_k \in \mathscr{T}$. We denote the family of all k-clause independent distributions with m clauses over n variables as $\mathscr{F}_k(n,m)$.

We remark that, in contrast to the random 3-SAT model (Model 6.1), which defines a single distribution for given n and m, the family $\mathscr{F}_k(n,m)$ contains many different distributions.

By definition, the probability of any event A_S that depends only on a subset S of at most k clauses in the k-clause independent distribution $\mathscr{D}$ is the same as for $\mathscr{D}_{\text{Ind.}}$. I.e., the expectations of the indicator function $\mathbb{I}[A_S]$ are the same for $\mathscr{D}$ and $\mathscr{D}_{\text{Ind.}}$. Hence, by linearity of expectation, any statistic that involves only $k' \leq k$ clauses must be the same for $\mathscr{D}$ and $\mathscr{D}_{\text{Ind.}}$, i.e.,

$$\mathop{\mathbf{E}}_{C\sim\mathscr{D}}\left[\sum_{\substack{S\subseteq C,\\|S|=k'}}\mathbb{I}[A_S]\right]=\mathop{\mathbf{E}}_{C\sim\mathscr{D}_{\text{Ind.}}}\left[\sum_{\substack{S\subseteq C,\\|S|=k'}}\mathbb{I}[A_S]\right]. \tag{6.1}$$

We would like to understand what are the largest/smallest possible number of clauses for a random 3-SAT formula to be almost surely satisfiable/unsatisfiable under any k-clause independent distribution. Using $SAT(C)$ to denote the event that the SAT instance C is satisfiable, we define the following satisfiability "thresholds".

Definition 6.2.2 (Upper satisfiability threshold)**.** *The upper satisfiability threshold function $\mathtt{UST}_k(n)$ is defined as follows. For integer $n\geq 3$, $\mathtt{UST}_k(n)$ is the minimum integer m such that*

$$\mathop{\mathbf{Pr}}_{C\sim\mathscr{D}}[SAT(C)]\leq 1/3,\ \ \forall\mathscr{D}\in\mathscr{F}_k(n,m).$$

Definition 6.2.3 (Lower satisfiability threshold)**.** *The lower satisfiability threshold function $\mathtt{LST}_k(n)$ is defined as follows. For integer $n\geq 3$, $\mathtt{LST}_k(n)$ is the maximum integer m such that*

$$\mathop{\mathbf{Pr}}_{C\sim\mathscr{D}}[\neg SAT(C)]\leq 1/3,\ \ \forall\mathscr{D}\in\mathscr{F}_k(n,m).$$

Extending the line of research on the standard random 3-SAT model, we would like to have as tight estimates of $\mathtt{UST}_k(n)$ and $\mathtt{LST}_k(n)$ as possible.

6.3 Tight bounds for the upper satisfiability threshold $\mathtt{UST}_2(n)$

We begin with a technical warm up and prove asymptotically tight bounds on the upper satisfiability threshold of 2-clause independent random 3-SAT.

Theorem 6.3.1. $\mathtt{UST}_2(n)=\Theta(n^3)$.

Theorem 6.3.1 follows by the next two lemmas. Lemma 6.3.2 provides an upper bound on $\mathtt{UST}_2(n)$. In the proof, we introduce the technique that we will use later in Section 6.5 to get a much more involved upper bound on $\mathtt{UST}_4(n)$.

Lemma 6.3.2. *For any $\mathscr{D}\in\mathscr{F}_2(n,m)$ with $m\geq 56\binom{n}{3}$, $\mathbf{Pr}_{C\sim\mathscr{D}}[SAT(C)]\leq 56\binom{n}{3}/m$.*

Proof. Let $\xi(C)\stackrel{\text{def}}{=\!=}\sum_{c_i,c_j\in C}\mathbb{I}[c_i=c_j]$ be the number of identical clause pairs in the instance C. On the one hand, Equation (6.1) implies that the expectation of $\xi(C)$ is the same when C is drawn from a 2-clause independent distribution and $\mathscr{D}_{\text{Ind.}}$. On the other hand, if an instance C has a satisfying assignment, at least $1/8$ of possible clauses from $\mathscr{T}$ must not appear in C, which means that the value of $\xi(C)$ is significantly higher than its expectation. These two observations will allow us to get the desired bound on the probability $\mathbf{Pr}_{C\sim\mathscr{D}}[SAT(C)]$.

Let $\mathscr{D} \in \mathscr{F}_2(n,m)$ and define $p \stackrel{\text{def}}{=} \mathbf{Pr}_{C\sim\mathscr{D}}[SAT(C)]$. We shall derive two lower bounds on the value the random variable $\xi(C)$ can get: an unconditional lower bound $\mathtt{LB}[\xi]$ (not far from $\mathbf{E}_{C\sim\mathscr{D}}[\xi(C)]$) and a lower bound $\mathtt{LB}[\xi \mid SAT]$ on $\xi(C)$ when C is satisfiable (this will be significantly larger than $\mathbf{E}_{C\sim\mathscr{D}}[\xi(C)]$). We note that

$$\begin{aligned}\mathbf{E}_{C\sim\mathscr{D}}[\xi(C)] &= \mathbf{Pr}[SAT(C)]\cdot\mathbf{E}[\xi(C) \mid SAT(C)] \\ &\quad + \mathbf{Pr}[\neg SAT(C)]\cdot\mathbf{E}[\xi(C) \mid \neg SAT(C)] \\ &\geq p\cdot\mathtt{LB}[\xi \mid SAT] + (1-p)\cdot\mathtt{LB}[\xi]. \end{aligned} \tag{6.2}$$

We next derive $\mathbf{E}_{C\sim\mathscr{D}}[\xi(C)]$, $\mathtt{LB}[\xi]$, and $\mathtt{LB}[\xi \mid SAT]$. We denote as $M \stackrel{\text{def}}{=} |\mathscr{T}| = 8\binom{n}{3}$ the number of possible clauses and $\lambda \stackrel{\text{def}}{=} m/M > 1$. First, we observe that

$$\begin{aligned}\mathbf{E}_{C\sim\mathscr{D}}[\xi(C)] = \sum_{c_i,c_j\in C}\mathbf{E}\left[\mathbb{I}[c_i = c_j]\right] &= \frac{m^2-m}{2}\cdot\mathbf{Pr}[c_i = c_j] \\ &= \frac{m^2-m}{2\cdot M} = m\cdot\frac{\lambda\cdot M-1}{2\cdot M}. \end{aligned} \tag{6.3}$$

To derive the lower bounds $\mathtt{LB}[\xi]$ and $\mathtt{LB}[\xi \mid SAT]$, we observe that any given clause type $t\in\mathscr{T}$ contributes $\binom{d_t}{2}$ pairs to $\xi(C)$, where $d_t \stackrel{\text{def}}{=} \sum_{c\in C}\mathbb{I}[c=t]$ is the number of type t clauses in C. I.e.,

$$\xi(C) = \sum_{t\in\mathscr{T}}\frac{d_t^2-d_t}{2}, \qquad \text{where } \sum_{t\in\mathscr{T}} d_t = m. \tag{6.4}$$

The minimum of $\xi(C) = \frac{1}{2}\sum d_t^2 - \frac{m}{2}$ under the constraint $\sum d_t = m$ is achieved when all the d_t variables are equal, i.e., equal to $\frac{m}{|\mathscr{T}|} = \lambda$. Thus, we get the following lower bound $\mathtt{LB}[\xi]$ on $\xi(C)$:

$$\xi(C) \geq \mathtt{LB}[\xi] \stackrel{\text{def}}{=} \frac{1}{2}\sum_{t\in\mathscr{T}}\lambda^2 - \frac{m}{2} = m\cdot\frac{\lambda-1}{2}. \tag{6.5}$$

To derive the lower bound $\mathtt{LB}[\xi \mid SAT]$ on $\xi(C)$ for a satisfiable formula C, we note that at least a $\frac{1}{8}$-fraction of the clause types are not present in C. I.e., at least $\frac{M}{8}$ of the d_t variables have value equal to 0 in (6.4). Similarly to (6.5), the minimum value of $\xi(C)$ is achieved when all the remaining $\frac{7M}{8}$ d_t variables are equal to each other, getting the value $\frac{8\cdot m}{7\cdot M}$. That is,

$$\xi(C) \geq \mathtt{LB}[\xi \mid SAT] \stackrel{\text{def}}{=} \frac{7\cdot M}{8}\cdot\left(\frac{8\lambda}{7}\right)^2 - \frac{m}{2} = m\cdot\frac{\frac{8}{7}\lambda-1}{2}. \tag{6.6}$$

We plug the bounds (6.3),(6.5), and (6.6) into (6.2) to get

$$m\cdot\frac{\lambda\cdot M-1}{2\cdot M} \geq p\cdot m\cdot\frac{\frac{8}{7}\lambda-1}{2} + (1-p)\cdot m\cdot\frac{\lambda-1}{2}.$$

After simple algebraic transformation, this is equivalent to the inequality $1 - \frac{1}{M} \geq \frac{p}{7}\lambda$, which implies that $p \leq \frac{7 \cdot M}{m} = 56\binom{n}{3}/m$. $\square$

To lower-bound $\mathtt{UST}_2(n)$, we use a specific 2-clause independent distribution, which is depicted as Model 6.2. The next lemma proves the correctness of our construction.

Lemma 6.3.3. *Model 6.2 defines a* 2*-clause independent probability distribution that generates 3-SAT instances of size* $\Omega(n^3)$ *that are satisfiable with probability at least* $1 - O(n^{-3})$.

MODEL 6.2: Selects a 3-SAT instance according to a 2-clause independent distribution

---

Input: Integer $n \geq 3$.
Output: A 3-SAT instance C with n variables and $m = \binom{n}{3}$ clauses.

- With probability $1 - \binom{n}{3}^{-1}$, construct a satisfiable instance C
  1. Match $m = \binom{n}{3}$ clauses to all different $T(n)$ variable triplets uniformly at random;
  2. Pick a random truth assignment $\sigma \sim \mathtt{Uni}[\{0,1\}^n]$;
  3. For each clause $c \in C$ matched to the variable triplet $(x_i, x_j, x_k) \in T(n)$
     - Pick a random sign triplet (s_1, s_2, s_3) of the same parity with $(\sigma_i, \sigma_j, \sigma_k)$ (i.e., $\mathbb{I}[s_1 = \texttt{-}] + \mathbb{I}[s_2 = \texttt{-}] + \mathbb{I}[s_3 = \texttt{-}] + \sigma_i + \sigma_j + \sigma_k = 0 \mod 2$);
     - Let clause $c \leftarrow ((x_i, s_1), (x_j, s_2), (x_k, s_3))$;
- With probability $\binom{n}{3}^{-1}$, sample C from distribution $\mathscr{D}_{\mathtt{uni-var}}$ defined as follows:
  1. Pick a random **single** variable triplet $(x_i, x_j, x_k) \sim \mathtt{Uni}[T(n)]$;
  2. For each clause $c \in C$
     - Pick a random sign triplet $(s_1, s_2, s_3) \sim \mathtt{Uni}[\{\texttt{+}, \texttt{-}\}^3]$;
     - Let clause $c \leftarrow ((x_i, s_1), (x_j, s_2), (x_k, s_3))$;

Proof. By its definition, with probability $1 - \binom{n}{3}^{-1}$, Model 6.2 returns a 3-SAT instance that has a planted satisfying assignment σ. Indeed, the condition $\mathbb{I}[s_1 = \texttt{-}] + \mathbb{I}[s_2 = \texttt{-}] + \mathbb{I}[s_3 = \texttt{-}] + \sigma_i + \sigma_j + \sigma_k = 0 \mod 2$ ensures that the assignment $x_i = \sigma_i, x_j = \sigma_j, x_k = \sigma_k$ satisfies clause $c = ((x_i, s_1), (x_j, s_2), (x_k, s_3))$. Hence, using $\mathscr{D}$ to denote the distribution according to which Model 6.2 selects the 3-SAT instance, we have $\mathbf{Pr}_{C \sim \mathscr{D}}[SAT(C)] \geq 1 - \binom{n}{3}^{-1}$.

We are only left to verify that the distribution $\mathscr{D}$ is 2-clause independent. We first consider the distribution $\mathrm{Var}(c)$ of variable triplets for $c \in C$ with $C \sim \mathscr{D}$ and notice that it is pair-wise independent. Indeed, the probability of drawing a pair of identical variable triplets in a 2-clause independent distribution is $\binom{n}{3}^{-1}$, while the probability of drawing two different variable triplets is $1 - \binom{n}{3}^{-1}$. We exactly match these probabilities in our construction, as we pick a pair of non-identical variable triplets uniformly at random with probability $1-\binom{n}{3}^{-1}$ and with probability $\binom{n}{3}^{-1}$ generate a pair of clauses with identical set of variables.

For the distribution of sign triplets, we first note that distribution $\mathscr{D}_{\texttt{uni-var}}$ is a uniform pair-wise independent distribution for each single triplet of variables (x_i, x_j, x_k). Second, we notice that when variable triplets are different in a pair of clauses c_1, c_2, then each of these clauses has at least one unique variable that does not appear in another clause (i.e., $x_i \in \mathrm{Var}(c_1)$ and $x_i \notin \mathrm{Var}(c_2)$, while $x_\ell \in \mathrm{Var}(c_2)$ and $x_\ell \notin \mathrm{Var}(c_1)$). As we pick in our construction the satisfying assignment σ uniformly at random, the parity condition for the sign triplet in clause c_2 does not affect the sign of x_ℓ variable for any fixed sign triplet of c_1. Hence, we get all combinations of sign triplets in c_1 and c_2 (with a fixed set of variables) with the same probability. Therefore, our construction is 2-clause independent. □

6.4 A tight lower bound for $\mathtt{UST}_3(n)$

Clearly, Lemma 6.3.2 also provides an upper bound on the upper satisfiability threshold $\mathtt{UST}_3(n)$. The proof of the next statement follows by presenting a matching lower bound through Model 6.3.

Theorem 6.4.1. $UST_3(n) = \Theta(n^3)$.

Proof. We give an explicit construction (see Model 6.3) of a 3-clause independent distribution $\mathscr{D} \in \mathscr{F}_3(n,m)$ with n variables and (an even number of) $m \le \frac{1}{3}\cdot\binom{n}{3}$ clauses, such that $\mathbf{Pr}_{C\sim\mathscr{D}}[SAT(C)] \ge 2/3 - O(n^{-6})$. Our construction in Model 6.3 follows the same pattern as in Model 6.2, but uses an additional step and minor modifications to ensure 3-clause independence.[3]

Now, we need to differentiate between three types of situations for a triplet of clauses: type I, when all three clauses have pair-wise distinct sets of variables; type II, when exactly one pair of clauses share the same variable triplet; type III, when all three clauses have the same set of variables. Notice that

$$\mathop{\mathbf{Pr}}_{\mathscr{D}_{\text{Ind.}}}[I] = 1 - 3\cdot\binom{n}{3}^{-1}, \qquad \mathop{\mathbf{Pr}}_{\mathscr{D}_{\text{Ind.}}}[II] = 3\cdot\binom{n}{3}^{-1} - \binom{n}{3}^{-2}, \qquad \mathop{\mathbf{Pr}}_{\mathscr{D}_{\text{Ind.}}}[III] = \binom{n}{3}^{-2}.$$

The distribution $\mathscr{D}$ used by Model 6.3 consists of three distributions: $\mathscr{D}_1$ (sampled with probability $1-p-q$), $\mathscr{D}_2$ (sampled with probability p), and $\mathscr{D}_3 = \mathscr{D}_{\texttt{uni-var}}$ (sampled with probability q). The parameters p and q are chosen so that

$$\mathop{\mathbf{Pr}}_{C\sim\mathscr{D}}[II] = p\cdot\mathop{\mathbf{Pr}}_{\mathscr{D}_2}[II] = p\cdot\frac{3}{m-1} = \mathop{\mathbf{Pr}}_{\mathscr{D}_{\text{Ind.}}}[II]; \qquad \mathop{\mathbf{Pr}}_{C\sim\mathscr{D}}[III] = q = \mathop{\mathbf{Pr}}_{\mathscr{D}_{\text{Ind.}}}[III].$$

[3]The first and third block of Model 6.3 correspond to the two blocks of Model 6.2. The only difference in the first block is that the matching of C is not to all the variable triplets in $T(n)$ but only to m of them.

MODEL 6.3: Selects a 3-SAT instance according to a 3-clause independent distribution

Input: Integer $n \geq 3$ and even integer $m \leq \frac{1}{3} \cdot \binom{n}{3}$.
Output: A 3-SAT instance C with n variables and m clauses.
Set $p \stackrel{\text{def}}{=\!=} (m-1) \cdot \left(\binom{n}{3}^{-1} - \frac{1}{3}\binom{n}{3}^{-2}\right)$ and $q \stackrel{\text{def}}{=\!=} \binom{n}{3}^{-2}$;

- With probability $1-p-q$, construct a satisfiable instance C
  1. Match C clauses to m different variable triplets in $T(n)$ uniformly at random;
  2. Pick a random truth assignment $\sigma \sim \mathtt{Uni}[\{0,1\}^n]$;
  3. For each clause $c \in C$ matched to the variable triplet $(x_i, x_j, x_k) \in T(n)$
     - Pick a random sign triplet (s_1, s_2, s_3) of the same parity with $(\sigma_i, \sigma_j, \sigma_k)$ (i.e., $\mathbb{I}[s_1 = \text{-}] + \mathbb{I}[s_2 = \text{-}] + \mathbb{I}[s_3 = \text{-}] + \sigma_i + \sigma_j + \sigma_k = 0 \mod 2$);
     - Let clause $c \leftarrow ((x_i, s_1), (x_j, s_2), (x_k, s_3))$;
- With probability p, construct an instance C with $m/2$ different variable triplets
  1. Uniformly at random match C to $\frac{m}{2}$ different variable triplets in $T(n)$ (exactly 2 clauses in C per one variable triplet);
  2. For each clause $c \in C$ assigned to the variable triplet $(x_i, x_j, x_k) \in T(n)$
     - Pick a random sign triplet $(s_1, s_2, s_3) \sim \mathtt{Uni}[\{+,\text{-}\}^3]$;
     - Let clause $c \leftarrow ((x_i, s_1), (x_j, s_2), (x_k, s_3))$;
- With probability q, sample C from distribution $C \sim \mathscr{D}_{\mathtt{uni\text{-}var}}$ as follows:
  1. Pick a random single variable triplet $(x_i, x_j, x_k) \sim \mathtt{Uni}[T(n)]$;
  2. For each clause $c \in C$
     - Pick a random sign triplet $(s_1, s_2, s_3) \sim \mathtt{Uni}[\{+,\text{-}\}^3]$;
     - Let clause $c \leftarrow ((x_i, s_1), (x_j, s_2), (x_k, s_3))$;

I.e., we perfectly match the probabilities of the events I, II, and III for 3-SAT instances produced by distributions $\mathscr{D}$ and $\mathscr{D}_{\text{Ind.}}$. Note that, similarly to Model 6.2, the generated 3-SAT instance is always satisfiable when $C \sim \mathscr{D}_1$. The numbers p and q in the definition of Model 6.2 are chosen so that $p + q \leq \frac{1}{3} + O(n^{-6})$ for $m \leq \frac{1}{3} \cdot \binom{n}{3}$. Hence, $\mathbf{Pr}_{C \sim \mathscr{D}}[SAT(C)] \geq 2/3 - O(n^{-6})$.

It remains to verify that distribution $\mathscr{D}$ is indeed 3-clause independent. Let us fix any three clauses $c_1, c_2, c_3 \in C$ with $C \sim \mathscr{D}$. We shall separately study the distributions of variable triplets and sign triples of (c_1, c_2, c_3). To this end, we define $\text{Var}(c_i) \stackrel{\text{def}}{=\!=} \widehat{T_i} \in T(n)$ and $\text{Sign}(c_i) \stackrel{\text{def}}{=\!=} \widehat{\rho}_i \in \{+,\text{-}\}^3$ for each $i \in [3]$. Let $T_1, T_2, T_3 \in T(n)$ and $\rho_1, \rho_2, \rho_3 \in \{+,\text{-}\}^3$ be respectively any three fixed variable triplets and any three fixed sign triplets. Since we perfectly match the probabilities of types I, II, III for variable triples with the respective probabilities in $\mathscr{D}_{\text{Ind.}}$, and we pick variable triplets uniformly at random in each $\mathscr{D}_1, \mathscr{D}_2$, and $\mathscr{D}_3$, we have

$$\Pr_{C\sim\mathscr{D}}\left[\widehat{T}_i = T_i,\ i\in[3]\right] = \Pr_{C\sim\mathscr{D}_{\text{Ind.}}}\left[T_i,\ i\in[3]\right].$$

Thus, the respective distribution of variable triples generated by distribution $\mathscr{D}$ is 3-wise independent. For the distribution of sign triplets, we note that we sample signs independently from the variable triplets and uniformly over $\{+,-\}^3$ in either of $\mathscr{D}_2$ and $\mathscr{D}_3$. Hence, it only remains to show that the distribution of sign triplets in $\mathscr{D}_1$ is uniform 3-wise independent for any feasible triplets of variables (T_1,T_2,T_3) such that $T_1 \neq T_2, T_2 \neq T_3$, and $T_1 \neq T_3$. We prove this by considering two cases. First, when variable triples T_1, T_2, T_3 have at least 5 different variables among them.

Claim 6.4.2. *When $|T_1\cup T_2\cup T_3| \geq 5$ and $T_1 \neq T_2$, $T_2 \neq T_3$, $T_1 \neq T_3$, we have*

$$\Pr_{C\sim\mathscr{D}_1}\left[\widehat{\rho}_i = \rho_i,\ i\in[3]\ \middle|\ \widehat{T}_i = T_i,\ i\in[3]\right] = \Pr_{C\sim\mathscr{D}_{Ind.}}\left[\rho_i,\ i\in[3]\right].$$

Proof. In this case, at least one of the variable triplets T_1, T_2, or T_3 must have a variable that is not present in the other two triplets. Indeed, if we count appearances of the variables in T_1,T_2,T_3, we get $3\cdot 3$. On the other hand, if each variable in $T_1\cup T_2\cup T_3$ appears at least twice, our count must be at least $2\cdot 5 > 9$ – a contradiction. Hence, we may assume, without loss of generality, that variable triplet T_3 has a unique variable $z\in T_3$ such that $z\notin T_1,T_2$.

As we choose the satisfying assignment σ in $\mathscr{D}_1$ uniformly at random, $\sigma(z)\in\{0,1\}$ equally likely for any fixed assignment of variables in $T_1\cup T_2$. Therefore, the choice of sign triplet for T_3 for any choice of sign triplets in T_1 and T_2 is uniform over $\{+,-\}^3$. This means that the choice of sign triplet for c_3 in $\mathscr{D}_1$ is uniform and independent of the clauses c_1 and c_2. By the same reasoning as in the proof of Lemma 6.3.3 for Model 6.2, the sign triplets in c_1 and c_2 are uniform over $\{+,-\}^3$ and pairwise independent in $\mathscr{D}_1$. □

Next, we consider the case when T_1, T_2, and T_3 have exactly 4 variables among them.

Claim 6.4.3. *When $|T_1\cup T_2\cup T_3| = 4$ and $T_1 \neq T_2$, $T_2 \neq T_3$, $T_1 \neq T_3$, we have*

$$\Pr_{C\sim\mathscr{D}_1}\left[\widehat{\rho}_i = \rho_i,\ i\in[3]\ \middle|\ \widehat{T}_i = T_i,\ i\in[3]\right] = \Pr_{C\sim\mathscr{D}_{Ind.}}\left[\rho_i,\ i\in[3]\right].$$

Proof. Let $T_1\cup T_2\cup T_3 = \{x_1,x_2,x_3,x_4\}$. We shall prove a stronger statement that four sign triplets $\widehat{\rho}_i \in \{+,-\}^3$ for $i\in[4]$ with $C\sim\mathscr{D}_1$ are 4-wise independent when their respective variable triplets T_1,T_2,T_3,T_4 are all four different subsets of $\{x_1,x_2,x_3,x_4\}$. To this end, consider a system of 4 equations with 4 variables over the finite field $\mathbb{F}_2$

$$\begin{cases} x_1\oplus x_2\oplus x_3 = v_1 & \text{corresponds to } \widehat{T}_1,\widehat{\rho}_1\\ x_1\oplus x_2\oplus x_4 = v_2 & \text{corresponds to } \widehat{T}_2,\widehat{\rho}_2\\ x_1\oplus x_3\oplus x_4 = v_3 & \text{corresponds to } \widehat{T}_3,\widehat{\rho}_3\\ x_2\oplus x_3\oplus x_4 = v_4 & \text{corresponds to } \widehat{T}_4,\widehat{\rho}_4. \end{cases} \tag{6.7}$$

System (6.7) always has a unique solution for any $v_1, v_2, v_3, v_4 \in \{0,1\}$. I.e., for any choice of sign triplets $(\rho_i)_{i\in[4]}$, there is a unique truth assignment σ restricted to variables x_1,x_2,x_3,x_4 that satisfies the parity constraint from our construction of $\mathscr{D}_1$ (for each $\rho_i = (s_1^i, s_2^i, s_3^i)$, $i \in [4]$, we set $v_i = \mathbb{I}\left[s_1^i = \text{-}\right] \oplus \mathbb{I}\left[s_2^i = \text{-}\right] \oplus \mathbb{I}\left[s_3^i = \text{-}\right]$ and the restriction of the truth assignment σ to variables x_1,x_2,x_3,x_4 is the solution to the corresponding system of linear equations (6.7)). As we choose the satisfying assignment σ uniformly at random in $\mathscr{D}_1$, the sign triplets $(\widehat{\rho}_i)_{i\in[4]}$ are independent and uniformly distributed over $\{+,\text{-}\}^3$. □

The above two cases in Claims 6.4.2 and 6.4.3 cover all the possibilities for the variable triplets generated in $\mathscr{D}_1$, as we always have $\widehat{T}_i \neq \widehat{T}_j$ for different clauses in $\mathscr{D}_1$, which concludes the proof of 3-clause independence of distribution $\mathscr{D}$. Since our construction has size $m = \Omega(n^3)$, and given that $\mathtt{UB}_3(n) \leq \mathtt{UB}_2(n) = O(n^3)$, Theorem 6.4.1 follows. □

6.5 An upper bound for $\mathtt{UST}_4(n)$

Our next result is rather surprising as it indicates that 4-wise independence allows for a steep decrease in the upper satisfiability threshold compared to 2- and 3-wise independence.

Theorem 6.5.1. $UST_4(n) = O(n^2)$.

Proof. We will prove the theorem by showing that for any positive integers n and m and any 4-clause independent probability distribution $\mathscr{D} \in \mathscr{F}_4(n,m)$, it holds $\mathbf{Pr}_{C\sim\mathscr{D}}[SAT(C)] \leq O\left(\max\left\{\frac{n^2}{m}, \frac{1}{\sqrt{n}}\right\}\right)$. The claim is obvious for $n < 10$. We will assume that $n \geq 10$ and $m \leq \sqrt{10}\cdot n^{5/2}$, and will show that $\mathbf{Pr}_{C\sim\mathscr{D}}[SAT(C)] \leq \frac{4288\cdot n^2}{m}$ for every 4-clause independent probability distribution $\mathscr{D} \in \mathscr{F}_4(n,m)$. Note that for $m > \sqrt{10}\cdot n^{5/2}$, the probability bound of $O\left(\frac{1}{\sqrt{n}}\right)$ follows by selecting uniformly at random a subset of $\sqrt{10}\cdot n^{5/2}$ clauses.

We will use a graph representation of 3-SAT instances defined as follows. Given a 3-SAT instance C consisting of m clauses over n variables, the bipartite multigraph $G(C) = (L\cup R, E)$ has a node corresponding to each (unordered) pair of literals $\{\ell_1, \ell_2\}$ from different variables at the left node side L and a node corresponding to each literal ℓ at the right node side R. Hence, $|L| = 4\binom{n}{2}$ and $|R| = 2n$. For every clause $c = (\ell_1, \ell_2, \ell_3)$ of C, $G(C)$ has the three edges between the node corresponding to the pair of literals (ℓ_i, ℓ_j) and the node corresponding to literal ℓ_{6-i-j} for $(i,j) \in \{(1,2),(1,3),(2,3)\}$.

The main proof idea of Theorem 6.5.1 is to analyse the statistic $\kappa(C)$ defined as the number of distinct $K_{2,2}$ subgraphs in graph $G(C)$. Namely, we first derive an upper bound on the expectation $\mathbf{E}_{C\sim\mathscr{D}}[\kappa(C)]$ (see Lemma 6.5.3) when C is drawn from the 4-clause independent probability distribution $\mathscr{D} \in \mathscr{F}_4(n,m)$. We then give two lower bounds on the value of the random variable $\kappa(C)$ by considering an underlying simple subgraph of $G(C)$. Note that the underlying simple subgraph corresponds to a smaller instance $\widetilde{C}$, in which we remove all repeated clauses. As we show in Lemma 6.5.7, this does not significantly reduce the size of the instance. Our first lower bound (Lemma 6.5.5) on $\kappa(\widetilde{C})$ holds for any instance $\widetilde{C}$ and is very close to the upper bound on the expectation of $\kappa(C)$. On the other hand, when $\widetilde{C}$ is satisfiable, we manage to give a significantly stronger lower bound on $\kappa(\widetilde{C})$ in Lemma 6.5.6. We conclude the proof of Theorem 6.5.1 by relating all these upper and lower bounds with $\mathbf{E}_{C\sim\mathscr{D}}[\kappa(C)]$ in Lemma 6.5.8.

Upper-bounding $\mathbf{E}_{C\sim\mathscr{D}}[\kappa(C)]$. A $K_{2,2}$ subgraph in graph $G(C)$ is formed by edges corresponding to four clauses. In the next simple Lemma 6.5.2, we describe necessary and sufficient conditions for a set of four clauses to create one or two $K_{2,2}$ subgraphs in $G(C)$. The expected number of $K_{2,2}$ subgraphs in $G(C)$ can be calculated by linearity of expectation: we simply estimate the probability that a set of 4 clauses corresponds to $K_{2,2}$ in $G(C)$ for any 4-wise independent distribution $\mathscr{D} \in \mathscr{F}_4(n,m)$, which is the same as for $\mathscr{D}_{\text{Ind.}}$ distribution. We apply Lemma 6.5.2 to get a lower bound (a tight estimate of) on the expected number of $K_{2,2}$ subgraphs in Lemma 6.5.3.

Lemma 6.5.2. *Any set of four clauses in C satisfies one of the following* 3 *cases.*

1. *Form exactly one $K_{2,2}$ subgraph in $G(C)$ and there is an ordering of the clauses (c_1,c_2,c_3,c_4) such that c_4 and c_1 share no literals; c_2 shares two literals with c_1 and one literal with c_4; and c_3 has the literals of c_1 and c_4 that do not appear in c_2.*

2. *Form exactly two $K_{2,2}$ subgraphs and there is an ordering of the clauses (c_1,c_2,c_3,c_4) such that c_4 shares one literal with c_1; c_2 has the common literal of c_1 and c_4, one more literal of c_1, and one more literal of c_4; and c_3 shares the common literal of c_1 and c_4, and has the literals of c_1 and c_4 that do not belong to c_2.*

3. *Do not form any $K_{2,2}$ subgraphs in $G(C)$.*

Proof. Consider a $K_{2,2}$ subgraph H of $G(C)$ with the four edges e_1, e_2, e_3, and e_4 as in Figure 6.1a and assume that they correspond to clauses c_1, c_2, c_3, and c_4, respectively. Then, the two right nodes in H correspond to different literals, e.g., g and h. The two left nodes in H may correspond to either two disjoint pairs of literals, e.g. $\{a,b\}$ and $\{d,f\}$ (see Figure 6.1b) or to two pairs sharing a literal, e.g., $\{a,b\}$ and $\{a,d\}$ (see Figure 6.1c). So, using the letters a, b, d, f, g, h to denote distinct literals, we distinguish between the two cases:

- **Case 1.** $c_1 = (a,b,g)$, $c_2 = (a,b,h)$, $c_3 = (d,f,g)$, and $c_4 = (d,f,h)$. Then, the twelve edges corresponding to c_1, c_2, c_3, and c_4 in $G(C)$ are depicted in Figure 6.1e. We observe that the $K_{2,2}$ of Figure 6.1b is the only $K_{2,2}$ subgraph defined by c_1, c_2, c_3, and c_4.

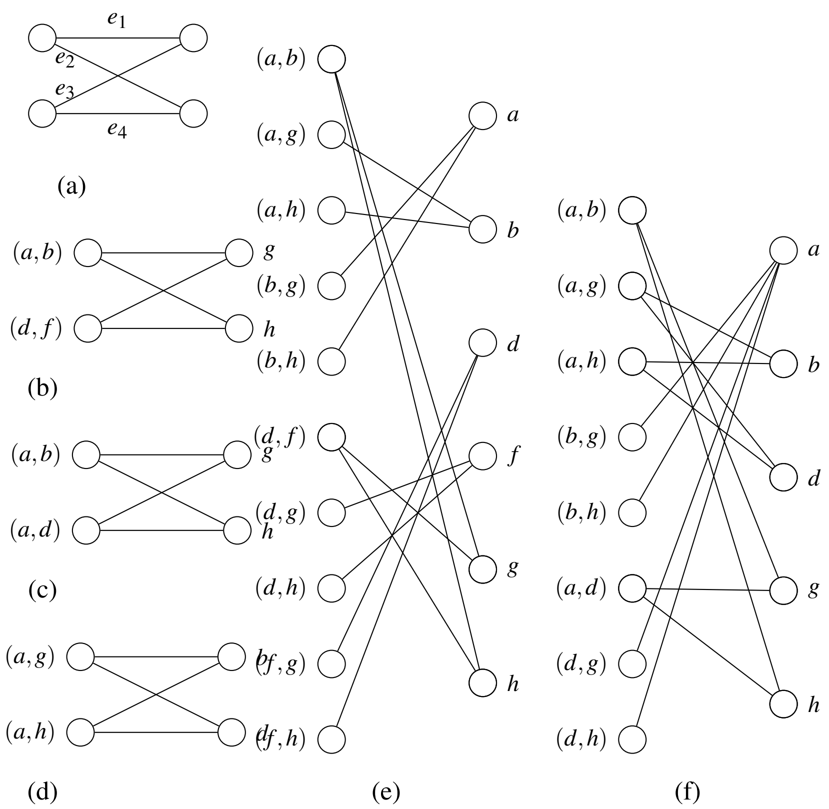

Figure 6.1: Possible structures formed by four clauses in the corresponding bipartite multi-graph. In subfigure (e), there is one $K_{2,2}$ subgraph formed by nodes (a,b), (d,g), g, and h and their incident edges. In subfigure (f), there are two $K_{2,2}$ subgraphs; one formed by the nodes (a,b), (a,d), g, and h and their incident edges and one formed by the nodes (a,g), (a,h), b, and d and their incident edges.

- **Case 2.** $c_1 = (a,b,g)$, $c_2 = (a,b,h)$, $c_3 = (a,d,g)$, and $c_4 = (a,d,h)$. Now, the twelve edges corresponding to c_1, c_2, c_3, and c_4 in $G(C)$ are depicted in Figure 6.1f. Observe that they form two $K_{2,2}$ subgraphs in $G(C)$ (the two depicted in Figures 6.1c and 6.1d).

These are the only cases in which four clauses can form $K_{2,2}$ subgraphs in graph $G(C)$. $\square$

Lemma 6.5.3. *For any $\mathscr{D} \in \mathscr{F}_4(n,m)$, it holds that* $\mathbf{E}_{C\sim\mathscr{D}}[\kappa(C)] \leq \frac{81m^4}{64n^6} + \frac{729m^3}{32n^4}$.

Proof. Let p_1 and p_2 denote the probabilities that a set of specific four clauses of $C \sim \mathscr{D}$ respectively falls into case 1 and 2 from Lemma 6.5.2 (as $\mathscr{D} \in \mathscr{F}_4(n,m)$ the probabilities p_1 and p_2 are the same for any four clauses). By linearity of expectation and Lemma 6.5.2 we have

$$\mathop{\mathbf{E}}_{C\sim\mathscr{D}}[\kappa(C)] = \binom{m}{4}\cdot(p_1+2\cdot p_2). \tag{6.8}$$

To bound p_1, consider an arbitrary ordering (c_1,c_2,c_3,c_4) of the four clauses (there are $4! = 24$ orderings). Given c_1 and c_4 have different literals (happens with probability close to 1), there are 9 ways to select two literals from c_1 and one literal from c_4 in c_2 and then only one way to select the remaining literals in c_3. Note that there are exactly 4 orderings (c_1,c_2,c_3,c_4), (c_2,c_1,c_4,c_3), (c_3,c_4,c_1,c_2), and (c_4,c_1,c_4,c_3) of $\{c_1,c_2,c_3,c_4\}$ that meet the conditions of Case 1 from Lemma 6.5.2. As we can assume without loss of generality that these four clauses are selected independently (due to 4-clause independence of $\mathscr{D}$), we have

$$p_1 \le 4!\cdot\frac{9}{8\binom{n}{3}}\cdot\frac{1}{8\binom{n}{3}}\cdot\frac{1}{4} = \frac{243}{8n^2(n-1)^2(n-2)^2} \le \frac{243}{8n^6}+\frac{1215}{4n^7}. \tag{6.9}$$

The last inequality follows after verifying that $\frac{1}{(n-1)^2(n-2)^2} \le \frac{1}{n^4}+\frac{10}{n^5}$ when $n \ge 10$.

We now bound probability p_2 that the four clauses meet the conditions of Case 2 from Lemma 6.5.2. Let us fix an ordering (c_1,c_2,c_3,c_4) of the four clauses ($4! = 24$ orderings in total). For a given clause c_1 there are 3 choices of the shared literal between c_1 and c_4, i.e., the probability that c_4 meets the conditions of Case 2 is at most $\frac{3\cdot 4\binom{n-1}{2}}{8\binom{n}{3}}$. For fixed c_1 and c_4, there are only four choices for c_2; and for given c_1, c_2, and c_4 there is a unique choice for c_3. There are exactly 4 orderings (c_1,c_2,c_3,c_4), (c_2,c_1,c_4,c_3), (c_3,c_4,c_1,c_2), and (c_4,c_1,c_4,c_3) of $\{c_1,c_2,c_3,c_4\}$ that meet conditions of Case 2. As we can assume without loss of generality that these four clauses are selected independently (due to 4-clause independence of $\mathscr{D}$), we have

$$p_2 \le \frac{4!}{4}\cdot\frac{3\cdot 4\binom{n-1}{2}}{8\binom{n}{3}}\cdot\frac{4}{8\binom{n}{3}}\cdot\frac{1}{8\binom{n}{3}} = \frac{243}{4n^3(n-1)^2(n-2)^2} \le \frac{243}{2n^7}, \tag{6.10}$$

where the last inequality holds, as $(n-1)^2(n-2)^2 \ge \frac{n^4}{2}$ for $n \ge 10$. Finally, by substituting (6.9) and (6.10) into (6.8) we get

$$\mathop{\mathbf{E}}_{C\sim\mathscr{D}}[\kappa(C)] \le \frac{81m^4}{64n^6}+\frac{729m^4}{32n^7} \le \frac{81m^4}{64n^6}+\frac{729m^3}{32n^4},$$

as desired (the last inequality follows, since $m \le \sqrt{10}\cdot n^{5/2} \le n^3$). □

Lower-bounding $\mathbf{E}_{C\sim\mathscr{D}}[\kappa(C)]$. For a 3-SAT instance C with m clauses over n variables, let $\widetilde{C}$ be the maximal subset of clauses in C with removed duplicates (identical clauses). The (simple) graph $G(\widetilde{C})$ is the underlying simple subgraph of the multi-graph $G(C)$. We denote by $\widetilde{m}$ the number of clauses in $\widetilde{C}$. We next obtain two lower bounds on $\kappa(\widetilde{C})$ in terms of n and $\widetilde{m}$: (i) for arbitrary $\widetilde{C}$ in Lemmas 6.5.5 and better bound (ii) for the case when $\widetilde{C}$ is SAT. These bounds only exploit the structure of the graph $G(\widetilde{C})$ and thus do not depend on any properties of $\mathscr{D}$. We then show in Lemma 6.5.7 how to relate expectation of $\widetilde{m}$ ($\widetilde{m} \leq m$) to m. Finally, we combine all these bounds in Lemma 6.5.8 to relate $\mathbf{E}_{C\sim\mathscr{D}}[\kappa(C)]$ with $\mathbf{Pr}_{C\sim\mathscr{D}}[SAT(C)]$.

Let us now define some notation used in the proof of both Lemmas 6.5.5 and 6.5.6. We denote by R_2 the set of all pairs of distinct nodes in the right node set R of graph $G(C) = (R \cup L, E)$. Thus, $|R_2| = \binom{2n}{2} > |L|$. Then, for a pair $(v, v') \in R_2$, let $B_{v,v'}$ denote the set of nodes in L that are adjacent to both nodes v and v' in R.

In the proofs of Lemmas 6.5.5 and 6.5.6, we extensively use the following observation.

Claim 6.5.4. *Let $x_1, \ldots, x_k \geq 0$. Then $\sum_{i=1}^k x_i^2 \geq \frac{1}{k}\left(\sum_{i=1}^k x_i\right)^2$.*

We are ready to prove our first lower bound on the number $\kappa(\widetilde{C})$.

Lemma 6.5.5. *Let $\widetilde{C}$ be an instance over n variables which consists of $\widetilde{m}$ distinct clauses. Then,*

$$\kappa(\widetilde{C}) \geq \frac{81\widetilde{m}^4}{64n^6} - \frac{27\widetilde{m}^3}{8n^4}.$$

Proof. Observe that the lemma trivially holds if $\widetilde{m} \leq 8n^2/3$, as the RHS of the inequality in its statement is non-positive. In the following, we consider the case $\widetilde{m} > 8n^2/3$. Notice that this implies that $\widetilde{m} > 4|R_2|/3$ and $\widetilde{m} > 4|L|/3$; these facts will be useful later in the proof.

Let v and v' be two (different) nodes in R side of $G(\widetilde{C})$. The number of different $K_{2,2}$ subgraphs in $G(\widetilde{C})$ with vertices v and v' is $\binom{|B_{v,v'}|}{2}$. Thus, using Claim 6.5.4, $\kappa(\widetilde{C})$ can be bounded as

$$\begin{aligned}\kappa(\widetilde{C}) &= \sum_{(v,v')\in R_2} \binom{|B_{v,v'}|}{2} = \frac{1}{2}\sum_{(v,v')\in R_2} |B_{v,v'}|^2 - \frac{1}{2}\sum_{(v,v')\in R_2} |B_{v,v'}| \\ &\geq \frac{1}{2|R_2|}\left(\sum_{(v,v')\in R_2} |B_{v,v'}|\right)^2 - \frac{1}{2}\sum_{(v,v')\in R_2} |B_{v,v'}|. \end{aligned} \tag{6.11}$$

Now, notice that the quantity $\sum_{(v,v')\in R_2} |B_{v,v'}|$ is the total number of distinct pairs of different nodes in R that the nodes in L have. Thus, denoting the degree of node $u \in L$ by d_u (recall that $\sum_{u\in L} d_u = 3\widetilde{m}$), we get

$$\begin{aligned}\sum_{(v,v')\in R_2} |B_{v,v'}| &= \sum_{u\in L}\binom{d_u}{2} = \frac{1}{2}\sum_{u\in L} d_u^2 - \frac{1}{2}\sum_{u\in L} d_u \\ &\geq \frac{1}{2|L|}\left(\sum_{u\in L} d_u\right)^2 - \frac{1}{2}\sum_{u\in L} d_u = \frac{9\widetilde{m}^2}{2|L|} - \frac{3\widetilde{m}}{2}. \end{aligned} \tag{6.12}$$

The inequality follows by Claim 6.5.4. We now aim to substitute $\sum_{(v,v')\in R_2} |B_{v,v'}|$ with $\frac{9\widetilde{m}^2}{2|L|} - \frac{3\widetilde{m}}{2}$ at the RHS of Equation (6.11) and get the desired bound on $\kappa(\widetilde{C})$. However, before doing so, we need to make sure that the RHS of Equation (6.11) is increasing for the range of values Equation (6.12) defines for $\sum_{(v,v')\in R_2}|B_{v,v'}|$. Indeed, we have

$$\frac{9\widetilde{m}^2}{2|L|}-\frac{3\widetilde{m}}{2}=\frac{3\widetilde{m}}{2}\left(\frac{3\widetilde{m}}{|L|}-1\right)>6|R_2|\geq|R_2|/2. \tag{6.13}$$

The inequality follows by the facts $\widetilde{m}>4|R_2|/3$ (and, hence, $\frac{3\widetilde{m}}{2}\geq 2|R_2|$) and $\widetilde{m}>4|L|/3$ (and, hence, $\frac{3\widetilde{m}}{|L|}-1>3$). Now, observe that the function $\frac{1}{2|R_2|}x^2-\frac{1}{2}x$ is increasing for $x\geq|R_2|/2$. Thus, due to Equation (6.12) and by our observation in Equation (6.13), Equation (6.11) yields

$$\begin{aligned}
\kappa(\widetilde{C}) &\geq \frac{1}{2|R_2|}\left(\frac{9\widetilde{m}^2}{2|L|}-\frac{3\widetilde{m}}{2}\right)^2-\frac{1}{2}\left(\frac{9\widetilde{m}^2}{2|L|}-\frac{3\widetilde{m}}{2}\right)\\
&=\frac{81\widetilde{m}^4}{8|R_2||L|^2}\left(1-\frac{|L|}{3\widetilde{m}}\right)^2-\frac{9\widetilde{m}^2}{4|L|}+\frac{3\widetilde{m}}{4}\\
&\geq\frac{81\widetilde{m}^4}{8|R_2||L|^2}-\frac{27\widetilde{m}^3}{4|R_2||L|}-\frac{9\widetilde{m}^2}{4|L|}\geq\frac{81\widetilde{m}^4}{8|R_2||L|^2}-\frac{27\widetilde{m}^3}{2|R_2||L|}\\
&=\frac{27\widetilde{m}^3}{2|R_2||L|}\left(\frac{3\widetilde{m}}{4|L|}-1\right)\geq\frac{27\widetilde{m}^3}{8n^4}\left(\frac{3\widetilde{m}}{8n^2}-1\right)=\frac{81\widetilde{m}^4}{64n^6}-\frac{27\widetilde{m}^3}{8n^4},
\end{aligned}$$

as desired. The second inequality follows by the property $(1-x)^2\geq 1-2x$. The third one follows since $\widetilde{m}>4|R_2|/3>|R_2|/3$ (and, hence, $\frac{9\widetilde{m}^2}{4|L|}\leq\frac{27\widetilde{m}^3}{4|R_2||L|}$). The last inequality follows by the facts $|R_2|=\binom{2n}{2}<2n^2$ and $|L|=4\binom{n}{2}<2n^2$. □

We now present a second lower bound on the number $\kappa(\widetilde{C})$, specifically for the case of a satisfiable instance C.

Lemma 6.5.6. *Let C be a statisfiable instance with n variables and $\widetilde{m}$ distinct clauses. Then*

$$\kappa(\widetilde{C})\geq\frac{82\widetilde{m}^4}{64n^6}-\frac{123\widetilde{m}^3}{16n^4}.$$

Proof. Observe that the lemma trivially holds if $\widetilde{m}\leq 6n^2$, as the RHS of the inequality in its statement is non-positive. In the following, we consider the case $\widetilde{m}>6n^2$.

We begin with some additional definitions. Consider an assignment σ satisfying C (and, thus, satisfying $\widetilde{C}$). Let L_S be the subset of L consisting of nodes identified by two literals, at least one of which is true according to σ. Let $L_U=L\setminus L_S$. Similarly, let R_S be the subset of R consisting of nodes identified by a literal which is true under σ and define $R_U=R\setminus R_S$. Clearly, $|L_S|=3|L|/4$, $|L_U|=|L|/4$, and $|R_S|=|R_U|=|R|/2$. Finally, let R_2^{SU} be the subset of R_2 consisting of pairs in $R_S\times R_U$, i.e., having a node from R_S and a node from R_U. Clearly, $|R_2^{SU}|=n^2$ and $|R_2\setminus R_2^{SU}|=n^2-n$.

Define $P=\sum_{(v,v')\in R_2^{SU}}|B_{v,v'}|$ and $Q=\sum_{(v,v')\in R_2\setminus R_2^{SU}}|B_{v,v'}|$. We will bound the number $\kappa(\widetilde{C})$ using

$$\kappa(\widetilde{C})=\sum_{(v,v')\in R_2}\binom{|B_{v,v'}|}{2}$$

$$
\begin{aligned}
&= \frac{1}{2} \sum_{(v,v') \in R_2^{SU}} |B_{v,v'}|^2 - \frac{1}{2} \sum_{(v,v') \in R_2^{SU}} |B_{v,v'}| \\
&\quad + \frac{1}{2} \sum_{(v,v') \in R_2 \setminus R_2^{SU}} |B_{v,v'}|^2 - \frac{1}{2} \sum_{(v,v') \in R_2 \setminus R_2^{SU}} |B_{v,v'}| \\
&\geq \frac{1}{2|R_2^{SU}|} \left(\sum_{(v,v') \in R_2^{SU}} |B_{v,v'}| \right)^2 - \frac{1}{2} \sum_{(v,v') \in R_2^{SU}} |B_{v,v'}| \\
&\quad + \frac{1}{2|R \setminus R_2^{SU}|} \left(\sum_{(v,v') \in R_2 \setminus R_2^{SU}} |B_{v,v'}| \right)^2 - \frac{1}{2} \sum_{(v,v') \in R_2 \setminus R_2^{SU}} |B_{v,v'}| \\
&\geq \frac{1}{2n^2}(P^2 + Q^2) - \frac{1}{2}(P+Q) \\
&= \frac{1}{4n^2}(P+Q)(P+Q-2n^2) + \frac{1}{4n^2}(Q-P)^2. \qquad (6.14)
\end{aligned}
$$

The first inequality follows by Claim 6.5.4 and the second one by the fact $|R_2 \setminus R_2^{SU}| \leq |R_2^{SU}| = n^2$.

For a node $u \in L_S$, denote by α_u and β_u the number of nodes in R_S and R_U that are adjacent to node u in $G(\widetilde{C})$. For a node $u \in L_U$, denote by γ_u the number of nodes in R_S that are adjacent to node u in $G(\widetilde{C})$; notice that u cannot be adjacent to any node of R_U as such an edge would correspond to an unsatisfied clause under σ. These definitions imply that

$$\sum_{u \in L_S} (\alpha_u + \beta_u) + \sum_{u \in L_U} \gamma_u = 3\widetilde{m}. \qquad (6.15)$$

Observe that for each node $u \in L_S$, there are $\alpha_u \beta_u$ node pairs in R_2^{SU} and $\binom{\alpha_u}{2} + \binom{\beta_u}{2}$ node pairs in $R_2 \setminus R_2^{SU}$ to which node u is adjacent. Also, for each node $u \in L_U$, there are $\binom{\gamma_u}{2}$ node pairs in $R_2 \setminus R_2^{SU}$ to which node u is adjacent. Thus, using the definition of $B_{v,v'}$, we get

$$P = \sum_{(v,v') \in R_2^{SU}} |B_{v,v'}| = \sum_{u \in L_S} \alpha_u \beta_u \qquad (6.16)$$

$$Q = \sum_{(v,v') \in R_2 \setminus R_2^{SU}} |B_{v,v'}| = \sum_{u \in L_S} \left(\binom{\alpha_u}{2} + \binom{\beta_u}{2} \right) + \sum_{u \in L_U} \binom{\gamma_u}{2}. \qquad (6.17)$$

To complete the proof, we will bound the RHS of Equation (6.14) with P and Q defined as in Equations (6.16) and (6.17), under the constraint imposed by Equation (6.15).

Observe that

$$P + Q = \sum_{u \in L_S} \left(\binom{\alpha_u}{2} + \binom{\beta_u}{2} + \alpha_u \beta_u \right) + \sum_{u \in L_U} \binom{\gamma_u}{2}$$

$$= \frac{1}{2}\sum_{u\in L_S}(\alpha_u+\beta_u)^2 + \frac{1}{2}\sum_{u\in L_U}\gamma_u^2 - \frac{1}{2}\sum_{u\in L_S}(\alpha_u+\beta_u) - \frac{1}{2}\sum_{u\in L_U}\gamma_u$$

$$\geq \frac{1}{2|L_S|}\left(\sum_{u\in L_S}(\alpha_u+\beta_u)\right)^2 + \frac{1}{2|L_U|}\left(\sum_{u\in L_U}\gamma_u\right)^2 - \frac{1}{2}\sum_{u\in L_S}(\alpha_u+\beta_u) - \frac{1}{2}\sum_{u\in L_U}\gamma_u$$

$$= \frac{1}{2|L_S|}\left(3\widetilde{m} - \sum_{u\in L_U}\gamma_u\right)^2 + \frac{1}{2|L_U|}\left(\sum_{u\in L_U}\gamma_u\right)^2 - \frac{3\widetilde{m}}{2}$$

$$\geq \frac{1}{3n^2}\left(3\widetilde{m} - \sum_{u\in L_U}\gamma_u\right)^2 + \frac{1}{n^2}\left(\sum_{u\in L_U}\gamma_u\right)^2 - \frac{3\widetilde{m}}{2}$$

$$= \frac{4}{3n^2}\left(\sum_{u\in L_U}\gamma_u\right)^2 - \frac{2\widetilde{m}}{n^2}\sum_{u\in L_U}\gamma_u + \frac{3\widetilde{m}^2}{n^2} - \frac{3\widetilde{m}}{2}. \tag{6.18}$$

The first inequality is due to Claim 6.5.4, the third equation is due to Equation (6.15), and the second inequality is due to the facts $|L_S| = \frac{3}{4}|L|$, $|L_U| = \frac{1}{4}|L|$, and $|L| = 4\binom{n}{2} < 2n^2$. I.e., for the quadratic function $f(x) \stackrel{\text{def}}{=} \frac{4}{3n^2}x^2 - \frac{2\widetilde{m}}{n^2}x + \frac{3\widetilde{m}^2}{n^2} - \frac{3\widetilde{m}}{2}$ the equation (6.18) reads as $P+Q \geq f(x_0)$ for $x_0 = \sum_{u\in L_U}\gamma_u$. This function will also be useful later. Similarly, we have

$$Q-P = \sum_{u\in L_S}\left(\binom{\alpha_u}{2} + \binom{\beta_u}{2} - \alpha_u\beta_u\right) + \sum_{u\in L_U}\binom{\gamma_u}{2}$$

$$= \frac{1}{2}\sum_{u\in L_S}(\alpha_u-\beta_u)^2 + \frac{1}{2}\sum_{u\in L_U}\gamma_u^2 - \frac{1}{2}\sum_{u\in L_S}(\alpha_u+\beta_u) - \frac{1}{2}\sum_{u\in L_U}\gamma_u$$

$$\geq \frac{1}{2}\sum_{u\in L_U}\gamma_u^2 - \frac{1}{2}\sum_{u\in L_S}(\alpha_u+\beta_u) - \frac{1}{2}\sum_{u\in L_U}\gamma_u$$

$$\geq \frac{1}{2|L_U|}\left(\sum_{u\in L_U}\gamma_u\right)^2 - \frac{1}{2}\sum_{u\in L_U}(\alpha_u+\beta_u) - \frac{1}{2}\sum_{u\in L_U}\gamma_u$$

$$= \frac{1}{2|L_U|}\left(\sum_{u\in L_U}\gamma_u\right)^2 - \frac{3\widetilde{m}}{2} \geq \frac{1}{n^2}\left(\sum_{u\in L_U}\gamma_u\right)^2 - \frac{3\widetilde{m}}{2}. \tag{6.19}$$

We conclude the proof by considering two cases for $x_0 = \sum_{u\in L_U}\gamma_u$.

Case 1: $x_0 = \sum_{u\in L_U}\gamma_u \geq \frac{\widetilde{m}}{2}$**.** Notice that $f(x)$ is minimized for $x = \frac{3\widetilde{m}}{4}$ at the value $\frac{9\widetilde{m}^2}{4n^2} - \frac{3\widetilde{m}}{2}$, which is greater than n^2 due to the fact $\widetilde{m} > 6n^2$. This observation together with Equation (6.18) yields $P+Q \geq f(x_0) > n^2$. Now notice that the quadratic function $x(x-2n^2)$ is increasing for $x > n^2$, and using the bound $P+Q \geq f(x_0) \geq \frac{9\widetilde{m}^2}{4n^2} - \frac{3\widetilde{m}}{2}$, we get

$$(P+Q)(P+Q-2n^2) \geq \left(\frac{9\widetilde{m}^2}{4n^2} - \frac{3\widetilde{m}}{2}\right)\left(\frac{9\widetilde{m}^2}{4n^2} - \frac{3\widetilde{m}}{2} - 2n^2\right)$$

$$\geq \frac{81\widetilde{m}^4}{16n^4} - \frac{27\widetilde{m}^3}{4n^2} - \frac{9\widetilde{m}^2}{4}. \tag{6.20}$$

As $x_0 \geq \widetilde{m}/2$, (6.19) yields $Q - P \geq \frac{\widetilde{m}^2}{4n^2} - \frac{3\widetilde{m}}{2}$, which is positive because $\widetilde{m} > 6n^2$. Thus,

$$(Q-P)^2 \geq \left(\frac{\widetilde{m}^2}{4n^2} - \frac{3\widetilde{m}}{2}\right)^2 = \frac{\widetilde{m}^4}{16n^4} - \frac{3\widetilde{m}^3}{4\widetilde{m}^2} + \frac{9\widetilde{m}^2}{4}. \tag{6.21}$$

By equations (6.14), (6.20), and (6.21), we get

$$\kappa(\widetilde{C}) \geq \frac{82\widetilde{m}^4}{64n^6} - \frac{15\widetilde{m}^3}{8n^4} \geq \frac{82\widetilde{m}^4}{64n^6} - \frac{123\widetilde{m}^3}{16n^4}$$

as desired.

Case 2: $x_0 = \sum_{u \in L_U} \gamma_u \leq \frac{\widetilde{m}}{2}$**.** Notice that the quadratic function $f(x)$ is decreasing in $[0, \frac{3\widetilde{m}}{4})$. Thus, $f(x_0) \geq \frac{7\widetilde{m}^2}{3n^2} - \frac{3\widetilde{m}}{2}$, which is higher than n^2, as $\widetilde{m} > 6n^2$. Hence, $P + Q \geq f(x_0) > n^2$. As the quadratic function $x(x - 2n^2)$ is increasing for $x > n^2$ and $P + Q \geq \frac{7\widetilde{m}^2}{3n^2} - \frac{3\widetilde{m}}{2} > n^2$, we get

$$\begin{aligned}(P+Q)(P+Q-2n^2) &\geq \left(\frac{7\widetilde{m}^2}{3n^2} - \frac{3\widetilde{m}}{2}\right)\left(\frac{7\widetilde{m}^2}{3n^2} - \frac{3\widetilde{m}}{2} - 2n^2\right) \\ &\geq \frac{49\widetilde{m}^4}{9n^4} - \frac{7\widetilde{m}^3}{n^2} - \frac{29\widetilde{m}^2}{12} \geq \frac{82\widetilde{m}^4}{16n^4} - \frac{123\widetilde{m}^3}{4n^2}. \end{aligned} \tag{6.22}$$

The last inequality holds as $\widetilde{m} > 6n^2$. The lemma now follows by Equations (6.14) and (6.22). □

The next lemma relates $\widetilde{m}$ with m and n.

Lemma 6.5.7. $\mathbf{E}_{C \sim \mathscr{D}}[\widetilde{m}^4] \geq m^4 - \frac{125}{6} \cdot m^3 \cdot n^2$.

Proof. For two clauses c and c' of C, let $X(c, c')$ be the binary random variable indicating whether $c = c'$ (then $X(c, c') = 1$) or not (then, $X(c, c') = 0$). We first observe that

$$m \leq \widetilde{m} + \sum_{\{c,c'\} \in \binom{C}{2}} X(c, c'). \tag{6.23}$$

Indeed, let $k = k(\theta)$ be the multiplicity of each clause type θ in C, then the left hand side $m = \sum_\theta k(\theta)$. For any value of k, the type θ contributes at least $k(\theta)$ to the right hand side of (6.23): (i) for $k = 1$ we count one θ in $\widetilde{m}$; (ii) for $k = 2$ we count θ once in $\widetilde{m}$ and another time in $X(c, c')$ for $c = \theta = c'$; (iii) for $k \geq 3$ the terms $X(c, c')$ with $c = \theta$ and $c' = \theta$ contribute $\binom{k}{2} \geq k$ to the right hand side of (6.23). After taking expectation over $C \sim \mathscr{D}$, we get

$$\mathop{\mathbf{E}}_{C \sim \mathscr{D}}[\widetilde{m}] \geq m - \sum_{\{c,c'\}} \mathop{\mathbf{Pr}}_{C \sim \mathscr{D}}[c = c'] = m - \binom{m}{2} \cdot \frac{1}{8\binom{n}{3}}$$

$$\geq m - \frac{3m^2}{8n(n-1)(n-2)} \geq m - \frac{25m^2}{48n^3} = m\left(1 - \frac{25m}{48n^3}\right).$$

Thus, using Jensen's inequality, the fact $(1-x)^4 \geq 1-4x$, and the assumption $m \leq \sqrt{10} \cdot n^{5/2}$, we have

$$\mathop{\mathbf{E}}_{C\sim\mathscr{D}}\left[\widetilde{m}^4\right] \geq \mathop{\mathbf{E}}_{C\sim\mathscr{D}}\left[\widetilde{m}\right]^4 \geq m^4\left(1 - \frac{25m}{48n^3}\right)^4 \geq m^4 - \frac{25m^5}{48n^3} \geq m^4 - \frac{125}{6} \cdot m^3 n^2,$$

as desired. $\square$

We prove a lower bound on $\mathbf{E}[\kappa(C)]$ using the last three lemmas.

Lemma 6.5.8. *For any $\mathscr{D} \in \mathscr{F}_4(n,m)$, we have* $\mathbf{E}_{C\sim\mathscr{D}}[\kappa(C)] \geq \frac{81m^4}{64n^6} - \frac{1215m^3}{32n^4} + \frac{m^4}{64n^6} \cdot \mathbf{Pr}_{C\sim\mathscr{D}}[SAT(C)]$.

Proof. We prove the lemma with the following derivation:

$$\begin{aligned}
\mathop{\mathbf{E}}_{C\sim\mathscr{D}}[\kappa(C)] &\geq \mathop{\mathbf{E}}_{C\sim\mathscr{D}}\left[\kappa(\widetilde{C})\right] \\
&= \mathop{\mathbf{E}}_{C\sim\mathscr{D}}\left[\kappa(\widetilde{C})|SAT(C)\right] \cdot \mathop{\mathbf{Pr}}_{C\sim\mathscr{D}}[SAT(C)] \\
&\quad + \mathop{\mathbf{E}}_{C\sim\mathscr{D}}\left[\kappa(\widetilde{C})|\neg SAT(C))\right] \cdot \mathop{\mathbf{Pr}}_{C\sim\mathscr{D}}[\neg SAT(C)] \\
&\geq \mathop{\mathbf{E}}_{C\sim\mathscr{D}}\left[\frac{82\widetilde{m}^4}{64n^6} - \frac{123\widetilde{m}^3}{16n^4}\middle| SAT(C)\right] \cdot \mathop{\mathbf{Pr}}_{C\sim\mathscr{D}}[SAT(C)] \\
&\quad + \mathop{\mathbf{E}}_{C\sim\mathscr{D}}\left[\frac{81\widetilde{m}^4}{64n^6} - \frac{27\widetilde{m}^3}{8n^4}\middle| \neg SAT(C)\right] \cdot \mathop{\mathbf{Pr}}_{C\sim\mathscr{D}}[\neg SAT(C)] \\
&\geq \mathop{\mathbf{E}}_{C\sim\mathscr{D}}\left[\frac{82\widetilde{m}^4}{64n^6} - \frac{123\widetilde{m}^3}{16n^4}\right] - \mathop{\mathbf{E}}_{C\sim\mathscr{D}}\left[\frac{\widetilde{m}^4}{64n^6}\middle| \neg SAT(C)\right] \cdot \mathop{\mathbf{Pr}}_{C\sim\mathscr{D}}[\neg SAT(C)] \\
&\geq \frac{82}{64n^6} \mathop{\mathbf{E}}_{C\sim\mathscr{D}}\left[\widetilde{m}^4\right] - \frac{123m^3}{16n^4} - \frac{m^4}{64n^6} \cdot \mathop{\mathbf{Pr}}_{C\sim\mathscr{D}}[\neg SAT(C)] \\
&\geq \frac{82m^4}{64n^6} - \frac{861m^3}{32n^4} - \frac{123m^3}{16n^4} - \frac{m^4}{64n^6} \cdot \mathop{\mathbf{Pr}}_{C\sim\mathscr{D}}[\neg SAT(C)] \\
&= \frac{81m^4}{64n^6} - \frac{1215m^3}{32n^4} + \frac{m^4}{64n^6} \cdot \mathop{\mathbf{Pr}}_{C\sim\mathscr{D}}[SAT(C)],
\end{aligned}$$

as desired. The second inequality follows by Lemmas 6.5.6 and 6.5.5, the fourth one by the fact $\widetilde{m} \leq m$, and the fifth one by Lemma 6.5.7. $\square$

Putting everything together. Lemmas 6.5.3 and 6.5.8 yield

$$\frac{81m^4}{64n^6} - \frac{1215m^3}{32n^4} + \frac{m^4}{64n^6} \cdot \mathop{\mathbf{Pr}}_{C\sim\mathscr{D}}[SAT(C)] \leq \mathop{\mathbf{E}}_{C\sim\mathscr{D}}[\kappa(C)] \leq \frac{81m^4}{64n^6} + \frac{729m^3}{32n^4},$$

which implies the desired bound $\mathbf{Pr}_{C\sim\mathscr{D}}[SAT(C)] \leq \frac{4288 \cdot n^2}{m}$, completing the proof of Theorem 6.5.1. $\square$

6.6 Bounds on the lower satisfiability threshold

In this section, we present our upper bound on the lower satisfiability threshold and explain why we believe that it is the tight bound for every degree of independence.

Theorem 6.6.1. *For every integer $k \geq 2$, $LST_k(n) = \Omega(n^{1-1/k})$.*

Proof. We prove the theorem by showing that for every integers $n \geq 3$ and $m \leq \frac{1}{12} \cdot n^{1-1/k}$, any k-clause independent distribution $\mathscr{D} \in \mathscr{F}_k(n,m)$ satisfies $\mathbf{Pr}_{C\sim\mathscr{D}}[\neg SAT(C)] \leq 1/3$.

Let $G(C) = (V,E)$ be a 3-uniform hypergraph defined by an instance C drawn from a k-clause independent distribution $\mathscr{D}$, $V = X(n)$ and $E = \{\mathrm{Var}(c)\}_{c\in C}$, i.e., every node in G represents a variable and every hyperegde in G represents the variable triplet of a clause. We consider simple *Berge-cycles* (or, simply, cycles) of length $\ell \geq 2$. A cycle of length $\ell > 2$ is defined as a set of ℓ distinct hyperedges $e_1, e_2, \ldots, e_\ell \in E$ such that pairs of consecutive edges share exactly one vertex ($|e_i \cap e_{i+1}| = 1$ and $|e_\ell \cap e_1| = 1$) and all other pairs of hyperedges are disjoint. In a cycle of length $\ell = 2$, the two hyperedges e_1 and e_2 have at least two vertices in common. We similarly define a path of length $\ell \geq 2$, where pairs of consecutive edges share exactly one vertex ($|e_i \cap e_{i+1}| = 1$ for $i \in [\ell - 1]$) and all other pairs of hyperedges are disjoint. We observe that any unsatisfiable instance must contain a subgraph H in G such that every variable in H appears in at least two hyperedges of H, which in turn implies that G has a cycle. That translates into the following upper bound on the probability that instance $C \sim \mathscr{D}$ is not satisfiable:

$$\begin{aligned}
\mathop{\mathbf{Pr}}_{C\sim\mathscr{D}}[\neg SAT(C)] &\leq \mathop{\mathbf{Pr}}_{C\sim\mathscr{D}}[\exists \text{ cycle in } G(C)] \leq \sum_{\ell\geq 2} \mathop{\mathbf{E}}_{C\sim\mathscr{D}}\left[\mathsf{Cycle}_\ell(G(C))\right] \\
&\leq \mathop{\mathbf{E}}_{C\sim\mathscr{D}}[\mathsf{Path}_k(G(C))] + \sum_{\ell\geq 2}^{k-1} \mathop{\mathbf{E}}_{C\sim\mathscr{D}}\left[\mathsf{Cycle}_\ell(G(C))\right] \\
&= \mathop{\mathbf{E}}_{C\sim\mathscr{D}_{\text{Ind.}}}[\mathsf{Path}_k(G(C))] + \sum_{\ell\geq 2}^{k-1} \mathop{\mathbf{E}}_{C\sim\mathscr{D}_{\text{Ind.}}}\left[\mathsf{Cycle}_\ell(G(C))\right]. \qquad (6.24)
\end{aligned}$$

In the above derivation, $\mathsf{Cycle}_\ell(G(C))$ counts the number of distinct cycles of length ℓ in G, and $\mathsf{Path}_k(G(C))$ counts the number of distinct paths of lengths k in G. For each ordered tuple of k hyperedges in G (there are $k! \cdot \binom{m}{k}$ such orderings), we can easily calculate the probability that they form a path. Thus,

$$\mathop{\mathbf{E}}_{C\sim\mathscr{D}_{\text{Ind.}}}[\mathsf{Path}_k(G(C))] = k! \cdot \binom{m}{k} \cdot \frac{3 \cdot \binom{n-3}{2}}{\binom{n}{3}} \cdot \prod_{t=3}^{k} \frac{2 \cdot \binom{n-2t+1}{2}}{\binom{n}{3}} \leq m^k \cdot \frac{9}{n} \cdot \left(\frac{6}{n}\right)^{k-2}. \qquad (6.25)$$

For each ordering of k edges, the first edge e_1 can be chosen arbitrarily; the second edge has exactly one vertex in common with e_1 and the remaining two vertices are chosen from $[n] \setminus e_1$ vertices; each of the remaining edges e_t for $t \geq 3$ has exactly one of the two vertices of e_{t-1} different from $e_{t-1} \cap e_{t-2}$ and has two other vertices chosen from $[n] \setminus (e_1 \cup \ldots \cup e_{t-1})$. The upper bound follows after simplifying each of the terms. We similarly derive an upper bound on $\mathbf{E}[\mathsf{Cycle}_\ell(G(C))]$ for $\ell \geq 3$ as follows:

$$\mathop{\mathbf{E}}_{C\sim\mathscr{D}_{\text{Ind.}}}\left[\mathsf{Cycle}_\ell(G(C))\right] \leq \frac{1}{6}\cdot \ell! \cdot \binom{m}{\ell} \cdot \left[\frac{3\cdot\binom{n-3}{2}}{\binom{n}{3}} \cdot \prod_{t=3}^{\ell-1} \frac{2\cdot\binom{n-2t+1}{2}}{\binom{n}{3}}\right] \cdot \frac{4\cdot(n-2\ell+1)}{\binom{n}{3}},$$

where for each ordering of ℓ edges (now $2\ell \geq 6$ different orderings correspond to the same cycle), the probabilities to choose the first $\ell-1$ edges can be computed in the same way as for Path_k; for the last hyper-edge e_ℓ, there are $4 = 2\cdot 2$ ways to select a vertex of $e_{\ell-1}$ together with a vertex of e_1, and $n-2\ell+1$ choices for the new vertex in $[n] \setminus (e_1 \cup \ldots \cup e_{\ell-1})$. Hence,

$$\mathop{\mathbf{E}}_{C\sim\mathscr{D}_{\text{Ind.}}}\left[\mathsf{Cycle}_\ell(G(C))\right] \leq m^\ell \cdot \left[\frac{9}{n}\cdot\left(\frac{6}{n}\right)^{\ell-3}\right] \cdot \frac{4}{n^2} = \frac{1}{6}\left(\frac{6m}{n}\right)^\ell. \tag{6.26}$$

For $\ell = 2$ we have

$$\begin{aligned}\mathop{\mathbf{E}}_{C\sim\mathscr{D}_{\text{Ind.}}}\left[\mathsf{Cycle}_2(G(C))\right] &= \binom{m}{2}\cdot\frac{3\cdot(n-3)+1}{\binom{n}{3}} \\ &= \frac{3m\cdot(m-1)\cdot(3n-8)}{n\cdot(n-1)\cdot(n-2)} \leq \left(\frac{3m}{n}\right)^2. \end{aligned} \tag{6.27}$$

We conclude the proof by plugging estimates (6.25),(6.26), and (6.27) into (6.24).

$$\begin{aligned}\mathbf{Pr}\left[\neg SAT(C)\right] &\leq \frac{2}{3}\frac{(6m)^k}{n^{k-1}} + \left(\frac{3m}{n}\right)^2 + \frac{1}{6}\sum_{\ell=3}^{k-1}\left(\frac{6m}{n}\right)^\ell \\ &\leq \frac{2}{3\cdot 2^k} + \frac{1}{16} + \frac{1}{6}\cdot\left(\frac{1}{2^3}+\frac{1}{2^4}+\ldots+\frac{1}{2^{k-1}}\right) < \frac{1}{3}.\end{aligned}$$

The second inequality follows since $m \leq \frac{1}{12}\cdot n^{1-1/k}$. □

On the tightness of the $O(n^{1-1/k})$ bound for $\mathtt{LST}_k(n)$

Theorem 6.6.1 says that k-wise independence of $\mathscr{D} \in \mathscr{F}_k(n,m)$ is enough to guarantee satisfiability of a random formula $C \sim \mathscr{D}$ for the number of clauses m of order $n^{1-1/k}$. On the other hand, for the mutually independent distribution $\mathscr{D}_{\text{Ind.}}$ a random formula is unsatisfiable with high probability for $m = O(n)$. Furthermore, our analysis in Theorem 6.6.1 is essentially tight and it seems unlikely that there is a better bound than $m = O(n^{1-1/k})$. We discuss below why this is the case, by outlining a plausible way for constructing k-clause independent distribution $\mathscr{D} \in \mathscr{F}_k(n,m)$ with $\mathbf{Pr}_{C\sim\mathscr{D}}[\neg SAT(C)] \geq \frac{2}{3}$ and $m = \Theta(n^{1-1/k})$.

Informal outline of the construction. In our construction of the k-clause independent distribution $\mathscr{D}$ we are only concerned with the distribution of clauses over variables, as all literals will be assigned uniformly at random and independently. Then, a sufficient condition for a random formula $C \sim \mathscr{D}$ to be unsatisfiable with large probability is that the 3-uniform hypergraph $G(C)$ constructed in Theorem 6.6.1 has a dense subgraph, i.e., a subgraph on $|V(H)|$ vertices (corresponding to variables in C) with at least $|E(H)| \geq 100 \cdot |V(H)|$ hyperedges. Indeed, one can simply count the expected number of satisfying assignments of a random formula ϕ_H on $V(H)$ variables and $|E(H)|$ fixed clauses with randomly assigned literals: initially, all $2^{|V(H)|}$ assignment are satisfying, but then, as we add $|E(H)|$ clauses one-by-one, the expected number of satisfying assignments reduces each time exactly by a factor $7/8$ (each satisfying assignment disappears with probability $1/8$). At the end, we get that the expected number of satisfying assignments is $\left(\frac{7}{8}\right)^{100|V(H)|} \cdot 2^{|V(H)|} < 0.01$, which means that ϕ is unsatisfiable most of the times.

Now, we want to make sure that our hypergraph $G(C)$ often has such a "dense" subgraph H, to get an unsatisfiable random formula. Notice that H needs only have a constant number of vertices. Also, note that according to the analysis in Theorem 6.6.1, we need to avoid any cycles of length smaller than or equal to k, as the probability of having such a cycle is of order $o(1)$ in $G(C)$ for $m = c \cdot n^{1-1/k}$ and any k-clause independent distribution $C \sim \mathscr{F}_k(n,m)$. Luckily, there are many construction of such 3-uniform hypergraphs H on $|V(H)| = O(1)$ vertices with large number of hyperedges $|E(H)| \geq 100|V(H)|$, and also of large girth $g(H) \geq k+1$ (e.g., see [61]). We shall "plant" H (essentially insert H as a connected component into G) with large probability in our construction. However, we need to make sure that by inserting H into our graph G, we still have enough room to match $\mathbf{Pr}[c_i = t_i, \forall i \in S] = \prod_{i \in S} \mathbf{Pr}[c_i = t_i]$ for all $S : |S| \leq k$. Next, we give a high level idea how one could achieve this.

As usual for the construction of distributions with identical marginals, we symmetrize $\mathscr{D}$ over all possible permutations of variables in $C \sim \mathscr{D}$. Then, our goal is to match the expected numbers for each isomorphism class of configurations of k hyperedges in $G(C)$ for $C \sim \mathscr{D}$ with $C \sim \mathscr{D}_{\text{Ind.}}$. It is useful to take note of the structure of typical hypergraph $G(C)$ for $C \sim \mathscr{D}_{\text{Ind.}}$ when $m = c \cdot n^{1-1/k}$: it consists of connected components, each of which is a tree of size at most k; the number of connected components of size k is a constant that grows slightly faster than linearly in c and, more generally, the number of connected components of size $k-j$ is $\Theta(n^{j/k})$. There is also a $o(1)$ probability event of having a Berge-cycle or a connected component of size at least $k+1$ in $G(C)$ for $C \sim \mathscr{D}_{\text{Ind.}}$. We can ignore in our construction of $\mathscr{D}$ those events, by adding small probability mass to $\mathscr{D}$ that consists of $\mathscr{D}_{\text{Ind.}}$ conditional on any of these rare events (can be easily achieved via rejection sampling from $\mathscr{D}_{\text{Ind.}}$). In this way, we only need to worry about matching expected numbers of forest configurations of size k in $G(C)$ between $C \sim \mathscr{D}$ and $C \sim \mathscr{D}_{\text{Ind.}}$. We can pick the constant in $m = c \cdot n^{1-1/k}$ sufficiently large, so that the constant size subgraph H contributes fewer trees of each type than their respective expected numbers in $G(C)$ for $C \sim \mathscr{D}_{\text{Ind.}}$. Then, we can add a few more connected components that are trees of size k to $C \sim \mathscr{D}$, so that we match k-wise statistics on all tree configurations with $C \sim \mathscr{D}_{\text{Ind.}}$. By having H and a few trees of size k in the graph $G(C)$ for $C \sim \mathscr{D}$, we only utilize a constant number of variables. Then, we would like to keep adding smaller connected components that consist of trees with strictly less than k hyperedges and eventually match k-wise statistics on all forest-like configurations of size k.

It seems plausible that the approach outlined above should work and yield a k-clause independent distribution $\mathscr{D}$. Apart from heavy notation that would be needed to formalize all steps in the above outline, the main technical hurdle is to ensure that statistics for all "forest-like" configurations of k hyperedges with more than one connected component are perfectly matched with $G(C)$ for $C \sim \mathscr{D}_{\text{Ind.}}$. Note, however, that even if our approach fails and there is a stronger version of Theorem 6.6.1 with $m = \omega(n^{1-1/k})$ and $\mathbf{Pr}_{C\sim\mathscr{D}}[SAT(C)] \geq \frac{2}{3}$ for any k-clause independent distribution $\mathscr{D}$, such a theorem would require a rather nontrivial argument that relies on subtle dependencies between multiple k-wise statistics.

Chapter 7

The revenue gap of linear pricing for selling a divisible item

Selling a perfectly divisible item to potential buyers is a fundamental task with apparent applications to pricing communication bandwidth and cloud computing services. Surprisingly, despite the rich literature on single-item auctions, revenue maximization when selling a divisible item is a much less understood objective. We introduce a Bayesian setting, in which the potential buyers have concave valuation functions (defined for each possible item fraction) that are randomly chosen according to known probability distributions. Extending the sequential posted pricing paradigm, we focus on mechanisms that use linear pricing, charging a fixed price for the whole item and proportional prices for fractions of it. Our goal is to understand the power of such mechanisms by bounding the gap between the expected revenue that can be achieved by the best among these mechanisms and the maximum expected revenue that can be achieved by any mechanism assuming mild restrictions on the behavior of the buyers. Under regularity assumptions for the probability distributions, we show that this revenue gap depends only logarithmically on a natural parameter characterizing the valuation functions and the number of agents. Our results follow by bounding the objective value of a mathematical program that maximizes the ex-ante relaxation of optimal revenue under linear pricing revenue constraints.

7.1 Introduction

Selling a single item to potential buyers is probably the most fundamental problem in microeconomics, with amazing related discoveries during the last sixty years. In the most standard model, there are several potential buyers (the *agents*), with *private valuations* for the item. The celebrated *Vickrey auction* [138] allocates the item to the *highest bidder* and charges her the second-highest bid as price. At least in theory, it achieves two important goals. First, it motivates the agents to act *truthfully* and report their true valuations as bids. Secondly, it maximizes *social welfare*, in the sense that it allocates the item to the agent who values it the most.

In contrast, for the equally important objective of *revenue maximization*, no solution is possible unless some information is available for the agent valuations. In the classical *Bayesian setting*, which is the most appealing for studying revenue maximization, each agent is assumed to have a random private valuation drawn from a probability distribution. Information about the probability distributions is known to the seller, who can use it to adjust the auction parameters and maximize her expected revenue. Under a regularity assumption about the agents' distributions, the revenue-optimal auction allocates the item to the agent with the highest non-negative *virtual bid* (if any) and charges her the minimum "critical" bid she could still use to win the item. Among other important results, Myerson's seminal paper [119] presents the related theory for revenue-optimal mechanism design, which can be applied to any *single-parameter* environment where the private valuation of each agent is just a scalar.

Inevitably, revenue-optimal auctions can be *complicated* and *counter-intuitive*. For example, as the virtual bid of an agent depends on the distribution from which she draws her valuations, they may discriminate among agents, setting different reserve prices for each of them. Also, the winning bid is not necessarily the highest one. So, the auctions that are used in practice are much simpler. A standard practice is to run a second-price auction with an *anonymous reserve* (i.e., common for all agents). Another even simpler class of mechanisms, known as *sequential posted pricing* [40], approaches the agents in a predefined order and makes a take-it-or-leave-it offer to each of them until the item is bought. When approached, a potential buyer can either accept to buy the item at the proposed price or refuse to buy if the proposed price exceeds her valuation for the item. In general, sequential posted pricing can become notoriously complex (pricing in the airline industry is an annoying example from practice) as the best possible price for an agent can depend on statistical information about the agents' valuations and on the decisions of agents that were approached previously.

However, sequential posted pricing is simple and well-understood when the same *anonymous* price (computed using statistical information about the valuations) is proposed to all agents. Constant approximations to optimal revenue (or a constant *revenue gap*) by sequential posted pricing with an anonymous price is possible under a regularity assumption for the valuations. Such statements (e.g., in [98]) usually read as follows: for any set of buyers who draw their valuations from independent and regular probability distributions, there is a price depending only on these distributions that yields an expected revenue that is a constant fraction of the expected revenue returned by Myerson's optimal auction.

In this work, we deviate from the above setting and assume that the item for sale is *perfectly divisible*. Different agents can get fractions of the item, while some fraction of the item can stay unsold. Agent preferences and behavior are much richer now. Each agent is still interested in obtaining the whole item but gets value from fractions of it as well. In particular, each agent has a private *valuation function* that indicates her value for fractions of the item between 0 and 1. Typically, these functions are non-decreasing and concave, indicating non-increasing non-negative marginal value.

Pricing of a divisible item can be very complicated in this setting. For example, refining sequential posted pricing, we can imagine a seller proposing a *menu of prices* to the agent, with a different price for different fractions of the item. In its extreme, the scheme could include different *pricing functions*, discriminating among agents. Then, taking into account her valuation function when facing a proposed pricing function, the agent can demand a desired fraction of the item that maximizes her utility.

We are interested in simple sequential posted pricing mechanisms that specifically use *linear pricing*. In particular, we restrict pricing to a fixed price for the whole item and proportional prices for fractions. Do such mechanisms have nearly-optimal revenue? Can we have statements like the one for anonymous pricing above? We will address these very challenging questions, making several conceptual and technical contributions.

Our contribution and techniques

Our first conceptual contribution is an extension of the standard Bayesian setting. We assume that each agent has a random non-decreasing concave valuation function. An important assumption we make is that the derivative (or *marginal*) of the valuation function of each agent at each fraction point follows a regular probability distribution. All the information about the probability distributions of each agent is known to the seller.

As our setting is very far from the single-parameter environment of an indivisible single item and agents with single-valued valuations for it, Myerson's characterization of the revenue-maximizing allocation does not carry over. Actually, similar characterizations seem out of reach in our case, but we would still like to have an estimate of the optimal expected revenue that can be extracted by the agents or a reasonable benchmark for it. Hence, we assume a *shortsighted behavior* for agents. Each agent responds to a pricing function with a demand for as large a fraction of the item as possible so that no smaller fraction yields her negative marginal utility. This allows us to reason about the revenue extracted by quite general mechanisms.

Our main goal is to provide bounds on the *revenue gap* between the highest expected revenue that can be achieved among given shortsighted agents and the highest expected revenue that can be achieved by a linear posted pricing mechanism. We do so by extending the concept of the *ex-ante relaxation* previously considered in [6, 7, 39]. Instead of comparing the revenue of the best linear pricing to the optimal expected revenue that can be extracted by shortsighted agents, we compare it to the best possible revenue of an ex-ante relaxation of the problem, in which an *expected* fraction of at most 1 is sold to them. We formulate the question regarding the revenue gap as a mathematical program that extends a mathematical program considered earlier for bounding the revenue gap of anonymous pricing in the indivisible item setting [7]. Essentially, assuming n shortsighted agents, our mathematical program has as variables the *selling probabilities* of the "optimal" mechanism and the probability distributions of the valuation marginals. The objective is to find an instance which yields the highest possible expected revenue under the constraints that the average total fraction of the item that is sold to the agents and the expected revenue of any linear pricing do not exceed 1. Then, the maximum objective value of the mathematical program is an upper bound of the revenue gap.

Our technical contribution consists of bounds on the objective value of this mathematical program in three scenarios. Interestingly, these bounds depend on a natural parameter of the valuation functions, which we call *initial marginal over total value* (or IMOTV, for short), defined as the ratio of the marginal of the valuation function at point 0 over its value at point 1.

First, as a warm up, we consider the case of a single agent and bound the revenue gap by $\mathcal{O}(\ln \kappa)$, where κ denotes the maximum IMOTV of the valuation functions. This result holds even for non-regular distributions of the marginals of the valuations functions. A simple instance shows that it is tight, even assuming regular marginal distributions.

Second, we bound the objective value of the mathematical program for the revenue gap by exploiting its relation to the mathematical program of [7] via a black-box reduction. In this way, we obtain an $\mathcal{O}(\kappa^2)$ upper bound. This result is interesting for two reasons. First, it showcases that the maximum IMOTV of the valuation functions is the parameter of importance in the multi-agent case as well. Second, the proof applies to the more general setting of regular initial marginal distributions only; all valuation marginals at points different than 0 can be non-regular.

Finally, our main result is an $\mathcal{O}(\ln \kappa + \ln n)$ upper bound, where n is the number of agents. The main idea of the proof is to identify an almost optimal feasible solution to the mathematical program, which has nice properties (e.g., for every agent, the marginal contribution of each item fraction to the expected revenue is upper bounded by a polynomial of n and κ) that allow us to bound the maximum objective value, assisted by the constraints and additional technical statements for random variables. We remark that previous work on proving bounds on revenue gaps by bounding the objective value of mathematical programs [7, 98–100] relied heavily on extreme regular probability distributions for valuations that have a *triangle-shaped revenue-quantile curve*. Unfortunately, such tools cannot be used in our case since the distributions of the valuation marginals should correspond to a probability distribution over concave valuation functions. We bypass this technical obstacle by a simple geometric lemma in the proof of our third bound.

Related work

Our paper falls within the "simple vs. optimal" line of research in Bayesian mechanism design, initiated with the work of Harline and Roughgarden [90], who studied the revenue gap of the second price auctions with an anonymous reserve. In the same thread, sequential posted pricing with an anonymous price has received much attention recently. Alaei et al. [7] prove that this mechanism achieves a constant approximation of 2.72 of the optimal expected revenue. They use a general strategy that we discuss in more detail later in Section 7.2. The tight bound of 2.62 follows by two papers by Jin et al. [98, 99]. We remark that all results for anonymous pricing carry over to our setting if the valuation functions are restricted to be linear. Indeed, the behavior of an agent with a linear valuation function against a linear pricing with a price of p per unit is either to buy the whole item if her value for the whole item is higher than p or refuse to buy otherwise. We mostly focus on non-linear valuation functions where divisibility differentiates our problem a lot.

A recent extension of single-item mechanisms is considered by Jin et al. [100], who study the setting with k item copies available to be sold to potential buyers. However, a unit-demand assumption restricts the agent valuations to a single scalar value for the single item copy they aim to buy. Their main result is a tight revenue gap of $\Theta(\ln k)$ under the unit-demand case. Instead, our modeling can be used to accommodate *multi-demand* agents with concave piecewise-linear valuation functions (with k segments and, hence, of IMOTV at most k). Our third result implies a revenue gap of $O(\ln k + \ln n)$ in this more general case.

A recent line of research that also deviates from the standard agent modeling assumptions focuses on agents with non-linear utility [68, 69] that, e.g., captures budget constraints or risk aversion, and includes bounds on the revenue gap of anonymous pricing for such agents [68]. However, these works usually assume single-parameter agents and a non-linear definition of their utility, while we focus on a standard quasi-linear definition of utility in a multi-parameter environment of a specific structure for the definition of valuations.

In the economics literature, divisible items have received attention, with the focus being mostly on whether bundling can be beneficial or not [125] and on how to structure the sale as many auctions of shares [140]. Perfect divisibility is considered in [108, 137]. In another more related direction, the operations research community has considered resource allocation mechanisms to divide an item based on signals received by the agents, that are further used to impose payments to the agents. Among them, the proportional mechanism, first defined by Kelly [105] and analyzed by Johari and Tsitsiklis [101] is the most popular one. Even though its social welfare has been analyzed extensively in stochastic settings that are very similar to ours [28, 45, 101] (see also [29] and the references therein), no revenue guarantees are known.

Roadmap

The rest of the paper is structured as follows. We present our setup, introducing several notions and making connections to the relevant literature in Section 7.2. Our mathematical program for the revenue gap is also introduced there. In Section 7.3, we present our tight revenue gap for a single agent. Section 7.4 is devoted to the multi-agent case; our black-box reduction to the result of Alaei et al. [7] is presented there. Our strongest logarithmic revenue gap is presented in Section 7.5. We conclude with a short discussion on open problems in Section 7.6.

7.2 Concepts, techniques, and notation

We denote by n the number of *agents* and use the integers in set $[n] = \{1,2,...,n\}$ to identify them. A *valuation function* for agent i is a monotone non-decreasing concave function $v_i : [0,1] \to \mathsf{R}_{\geq 0}$. We use v' to refer to the (left) derivative of v.

Our main focus is on *linear posted pricing* mechanisms. Such a mechanism uses a scalar price p and considers the agents in a predefined order. Whenever an agent is considered, she responds with her *demand*, i.e., the fraction of the item she is willing to buy at a price of p per unit. The mechanism then allocates the agent a fraction of the item equal to either her demand or the fraction of the item that has not been previously allocated to other agents, whichever is smaller, and charges the agent an amount that is equal to p multiplied with the fraction of the item the agent got. The demand $D(v,p)$ of an agent with valuation function v at a price-per-unit p is defined as the maximum fraction that maximizes the utility of the agent $D(v,p) \in \text{argmax}_{x\in[0,1]} \{v(x) - p\cdot x\}$. Due to the concavity of the valuation functions, the demand is the maximum fraction $x \in [0,1]$ so that $v'(x) \geq p$. Hence, when the n agents have a vector of valuation functions $\mathbf{v} = \langle v_1, v_2, ..., v_n\rangle$, the revenue of the linear posted pricing mechanism that uses price-per-unit p is $p \cdot \min\left\{1, \sum_{i\in[n]} D(v_i, p)\right\}$.

We will assume that all valuation functions v have an *initial marginal over total value* (or IMOTV for short) ratio $v'(0)/v(1)$ bounded by a parameter $\kappa \geq 1$. We denote by $\mathscr{V}_\kappa$ the set of all such valuation functions.

We consider a Bayesian setting in which each agent $i \in [n]$ draws her valuation function v_i, independently from the other agents, from a publicly known probability distribution $\mathbf{F}_i$ over the valuation functions in $\mathscr{V}_\kappa$. We write $v_i \sim \mathbf{F}_i$ and $\mathbf{v} \sim \mathbf{F}$ to denote that the valuation function of agent i and the vector of valuation functions $\mathbf{v} = \langle v_1, v_2, ..., v_n\rangle$ are drawn randomly according to the probability distribution $\mathbf{F}_i$ and the joint probability distribution $\mathbf{F}$, respectively. For every agent $i \in [n]$, the probability distribution $\mathbf{F}_i$ is such that the value of the derivative $v'(x)$ at any point $x \in [0,1]$ follows a regular probability distribution with cumulative density function $F_{i,x}$.

Our modeling assumption includes, as a special case, the model of [137], where the valuation function $v_i(z) = t_i \cdot h_i(z)$ has a scalar part t_i and a function part $h_i(z)$. The function part is known to the seller and can be used by the mechanism. The scalar part is drawn from a probability distribution, in which regularity can be imposed. This is essentially a *single-parameter* environment, where it is a simple exercise to apply Myerson's approach [119] of maximizing revenue by maximizing virtual welfare. This means that revenue-optimal mechanisms have a well-known structure in this setting. In contrast, our setting is *multi-parameter*, and our modeling allows for richer distributions over valuation functions. Unfortunately, Myerson's machinery cannot be applied anymore, so we do not have any clean form of the revenue-maximizing mechanism. This makes even the definition of the revenue gap in our case, as well as the proofs of bounds on its value, more challenging.

Selling an indivisible item with an anonymous price

We will adapt the approach of [7] for selling an indivisible item using sequential posted pricing and an *anonymous price*. In that classic setting, each agent $i \in [n]$ has a scalar valuation v_i drawn from a probability distribution with regular cumulative density function H_i.

Alaei et al. [7] use the mathematical program (7.1) below to bound the gap between the optimal expected revenue and the maximum expected revenue that can be achieved with an anonymous price. Actually, instead of comparing directly to the optimal revenue (in fact, this is done in follow-up work by Jin et al. [98]), they compare to the best possible revenue of the *ex-ante relaxation*, i.e., the maximum expected revenue that can be achieved by a mechanism that sells the item to at most one agent *in expectation*. Their mathematical program uses as variables (1) the regular cumulative density function H_i of the valuation of agent i for the item and (2) the probability r_i that the item is given to agent i in the revenue-maximizing allocation for the ex-ante relaxation.

$$\begin{aligned} \text{maximize} \sum_{i\in[n]} r_i \cdot H_i^{-1}(1-r_i) & \qquad (7.1)\\ \text{subject to} \sum_{i\in[n]} r_i \leq 1 & \\ p\cdot\left(1-\prod_{i\in[n]} H_i(p)\right) \leq R, \forall p > R & \\ r_i \in [0,1], \forall i \in [n] & \\ H_i \text{ is a regular cdf}, \forall i \in [n] & \end{aligned}$$

Viewed together with the first constraint, the objective of the mathematical program (7.1) is to maximize the revenue of the ex-ante relaxation. The second set of constraints implements the restriction that no anonymous pricing p yields a revenue higher than R. The ratio between the objective value of (7.1) and R is an upper bound to the revenue gap. The main result of [7] is as follows.

Theorem 7.2.1 (Alaei et al. [7])**.** *For every $R \geq 1$, the objective value of the mathematical program (7.1) is at most $e \cdot R$.*

Our second bound on the revenue gap when selling a divisible item with linear pricing will use the result of [7] as a black box. We remark that the original result in [7] uses specifically $R = 1$. The extension we consider here is without loss of generality and is used to simplify our exposition in Section 7.4.

Shortsighted agents

Let us now return to our setting. As a benchmark for the comparison and assessment of the revenue of linear pricing, we consider general pricing schemes that may discriminate between different agents using different *pricing functions* of the form

$p:[0,1]\to \mathsf{R}_{\geq 0}$, where $p(x)$ denotes the price an agent is asked to pay for getting an item fraction of x. We extend the notion of the demand by slightly abusing notation as follows. We say that, given a pricing function p with left derivative p', the demand $D(v,p)$ of an agent with a valuation function v is the maximum item fraction x so that $v'(t)\geq p'(t)$ for every $t\in[0,x]$. Thus, we assume that agents are *shortsighted* and aim to only locally maximize their utility.

Let us explain the notion of shortsightness with an example. Consider an agent with the valuation function

$$v(x)=\begin{cases}12x & x\in[0,1/3]\\ 1+9x & x\in[1/3,2/3]\\ 5+3x & x\in[2/3,1]\end{cases}$$

and the pricing function

$$p(x)=\begin{cases}10x & x\in[0,1/2]\\ 3+4x & x\in[1/2,1]\end{cases}.$$

The utility of the agent as a function of the item fraction x is now

$$\text{utility}(x)=\begin{cases}2x & x\in[0,1/3]\\ 1-x & x\in[1/3,1/2]\\ -2+5x & x\in[1/2,2/3]\\ 2-x & x\in[2/3,1]\end{cases}.$$

Hence, it increases as x ranges from point 0 to point $1/3$ up to the value of $2/3$, then decreases and gets the value of $1/2$ at point $1/2$, then increases again getting the highest value of $4/3$ at point $2/3$, and decreases afterwards. According to our assumption of a shortsighted agent, her demand is $D(v,p)=1/3$ (i.e., the first local maximum of the utility) even though the utility is globally maximized later at point $2/3$.

Our objective is to prove bounds on the *revenue gap* of linear posted pricing, i.e., on the maximum ratio, over all possible instances that follow our setting, between the expected revenue that can be achieved with shortsighted agents and the maximum revenue that can be achieved with linear posted pricing. We remark that, due to the concavity of the valuation functions, shortsighted agents actually respond as utility-maximizers to linear pricing.

A mathematical program for the revenue gap

Extending the approach of Alaei et al. [7], we also use an ex-ante relaxation to upper-bound the maximum revenue that can be achieved with shortsighted agents. We consider general pricing schemes consisting of a pricing function $p_i:[0,1]\to\mathsf{R}^+$ for each agent $i\in[n]$. For every agent $i\in[n]$ and every $x\in[0,1]$, let $q_i(x)$ be the probability that a fraction of at least x is bought by agent i with pricing p_i. For $i \in [n]$ and $x \in [0,1]$, observe that the shortsighted agent i will have a demand $D(v_i, p_i)$ that is at least x when $v_i'(x) \geq p_i'(x)$. Hence, $q_i(x) = \Pr[v_i'(x) \geq p_i'(x)] = 1 - F_{i,x}(p_i'(x))$ and, equivalently, $p_i'(x) = F_{i,x}^{-1}(1-q_i(x))$. Using the quantities $q_i(x)$ and $p_i'(x)$, we can express the expected payment by agent i as $\int_0^1 q_i(x) \cdot p_i'(x)\,\mathrm{d}x = \int_0^1 q_i(x) \cdot F_{i,x}^{-1}(1-q_i(x))\,\mathrm{d}x$. Intuitively, $q_i(x)$ denotes the probability that agent i buys her x-th point of the item, $p_i'(x)$ denotes the payment increase due to this point, and the quantity $q_i(x) \cdot F_{i,x}^{-1}(1-q_i(x))$ represents the marginal contribution of agent i and point x to the expected revenue. Hence, the maximum expected revenue that can be achieved by shortsighted agents is upper-bounded by the quantity

$$\sum_{i\in[n]} \int_0^1 q_i(x) F_{i,x}^{-1}(1-q_i(x))\,\mathrm{d}x, \tag{7.2}$$

under the constraint that *on average*, no more than the whole item is available for purchase by the agents. Thus, the RHS of equation (7.2) is indeed the expected revenue of the ex-ante relaxation.

Next, we extend the approach of [7] to relate the ex-ante revenue relaxation with shortsighted agents to the maximum revenue that can be achieved with linear pricing. To determine the revenue gap, we can further require that any linear pricing with price per unit p has revenue at most 1 and ask: "How large can the expected revenue of the ex-ante relaxation be?" The answer to this question is given by the mathematical program (7.3) below. The objective is to maximize expression (7.2), which upper-bounds the ex-ante relaxation of the revenue that can be achieved with shortsighted agents. Constraint (7.4) ensures that the average fraction allocated to the agents does not exceed the available item. The RHS of (7.5) is just the expected revenue of linear pricing with a price p per unit. The mathematical program has as variables the probability distributions over the valuation functions of $\mathscr{V}_\kappa$, which are further described by cdf's $F_{i,x}$ for $i \in [n]$ and $x \in [0,1]$, and the probabilities $q_i(x)$ for $i \in [n]$ and $x \in [0,1]$; by its intending meaning as the probability that agent i gets a fraction of at least x, q_i (called *selling probability* from now on) is required to be monotone non-increasing in x.

$$\text{maximize} \sum_{i\in[n]} \int_0^1 q_i(x) \cdot F_{i,x}^{-1}(1-q_i(x))\,\mathrm{d}x \tag{7.3}$$

$$\text{subject to} \sum_{i\in[n]} \int_0^1 q_i(x)\,\mathrm{d}x \leq 1 \tag{7.4}$$

$$p \cdot \mathsf{E}_{\mathbf{v}\sim\mathbf{F}}\left[\min\left\{1, \sum_{i\in[n]} D(v_i, p)\right\}\right] \leq 1, \forall p > 1 \tag{7.5}$$

$$\text{monotone non-increasing } q_i : [0,1] \to [0,1], \forall i \in [n], x \in [0,1]$$

$$\text{supp}(\mathbf{F}_i) \subseteq \mathscr{V}_\kappa, F_{i,x} \text{ is a regular cdf}, \forall i \in [n], x \in [0,1]$$

In the following, we frequently abbreviate the quantity $q_i(x) \cdot F_{i,x}^{-1}(1-q_i(x))$ as $R_{q_i,\mathbf{F}_i}(x)$ for $i \in [n]$ and $x \in [0,1]$.

7.3 Warming up: the single-agent case

We begin by presenting a tight bound on the revenue gap for the case of a single agent. For simplicity of exposition, we drop the index i from notation. Actually, we use a simplified version of the mathematical program (7.3) where we require neither the selling probability function $q(x)$ to be monotone non-increasing nor the probability distributions F_x to be regular. Since we have just a single agent, the constraint (7.4) is now redundant. In particular, we will upper-bound the objective value of the following simplified mathematical program:

$$\begin{aligned}
\text{maximize} \ & \int_0^1 q(x) \cdot F_x^{-1}(1-q(x))\,\mathrm{d}x \qquad (7.6)\\
\text{subject to} \ & p \cdot \mathsf{E}_{\mathbf{v}\sim\mathbf{F}}[D(v,p)] \le 1, \forall p > 1\\
& q(x) \in [0,1], \forall x \in [0,1]\\
& \mathrm{supp}(\mathbf{F}) \subseteq \mathscr{V}_{\mathbf{K}}, F_x : [0,1] \to [0,1] \text{ is a cdf}, \forall x \in [0,1]
\end{aligned}$$

To do so, we will need three technical lemmas. The first one follows easily by the concavity of the valuation functions and the optimality of selling probabilities.

Lemma 7.3.1. *Let $(\mathbf{q},\mathbf{F})$ be an optimal solution of the mathematical program (7.6). Then, $R_{q,\mathbf{F}}(x)$ is non-increasing in x.*

Proof. Let $0 \le x_1 < x_2 \le 1$. Due to the concavity of the valuation functions, $F_x^{-1}(t)$ is monotone non-increasing in x for every $t \in [0,1]$. Thus,

$$\begin{aligned}
R_{q,\mathbf{F}}(x_2) &= q(x_2) \cdot F_{x_2}^{-1}(1-q(x_2)) \le q(x_2) \cdot F_{x_1}^{-1}(1-q(x_2))\\
&\le q(x_1) \cdot F_{x_1}^{-1}(1-q(x_1)) = R_{q,\mathbf{F}}(x_1)
\end{aligned}$$

as desired. The second inequality follows since $q(x_1)$ is the optimal selling probability at point x_1 (under no constraints). □

The next lemma is due to the first set of constraints of the mathematical program (7.6).

Lemma 7.3.2. *Let $(\mathbf{q},\mathbf{F})$ be a solution of the mathematical program (7.6). Then, $R_{q,\mathbf{F}}(x) \le 1/x$ for $x \in (0,1]$.*

Proof. Notice that, for every $x \in [0,1]$, by setting the linear price $p = F_x^{-1}(1-q(x))$, the derivative of the valuation v drawn from distribution $\mathbf{F}$ satisfies $v'(x) \ge p$ with probability at least $q(x)$. This yields an expected demand of $\mathsf{E}_{v\sim\mathbf{F}}\left[D(v,F_x^{-1}(1-q(x)))\right] \ge x \cdot q(x)$. The lemma follows since, by the constraint of the mathematical program (7.6) with $p = F_x^{-1}(1-q(x))$, we get $\mathsf{E}_{v\sim\mathbf{F}}\left[D(v,F_x^{-1}(1-q(x)))\right] \le 1/F_x^{-1}(1-q(x))$. □

The next lemma indicates that the function $\int_0^t R_{q,\mathbf{F}}(x)\,\mathrm{d}x$, which was essentially proved to be concave in Lemma 7.3.1, has an IMOTV bounded by a polynomial in κ.

Lemma 7.3.3. *Let* $(\mathbf{q},\mathbf{F})$ *be a solution of the mathematical program (7.6). Then, it holds*

$$R_{q,\mathbf{F}}(0) \leq 4\kappa^2 \int_0^1 R_{q,\mathbf{F}}(x)\,dx. \tag{7.7}$$

Proof. Denote by $V \subseteq \mathscr{V}_\kappa$ the set of valuation functions that are drawn with non-zero probability and let $S = \left\{v \in V : v'(0) \geq F_0^{-1}(1-q(0))\right\}$ be the set of valuation functions with derivative at least $F_0^{-1}(1-q(0))$ at point 0. Since $q(x)$ is the optimal selling probability at point $x \in [0,1]$, we get

$$R_{q,\mathbf{F}}(x) = q(x)\cdot F_x^{-1}(1-q(x)) \geq q(0)\cdot F_x^{-1}(1-q(0)). \tag{7.8}$$

By the definition of set S, we have $\Pr_{v\sim\mathbf{F}}[v \in S] = 1-q(0)$. Hence,

$$F_x^{-1}(1-q(0)) \geq \min_{v\in S} v'(x). \tag{7.9}$$

By Equations (7.8) and (7.9), we get

$$\int_0^1 \min_{v\in S} v'(x)\,\mathrm{d}x \leq \frac{1}{q(0)}\cdot\int_0^1 R_{q,\mathbf{F}}(x)\,\mathrm{d}x \tag{7.10}$$

Now, let $t \in [0,1]$ and $\widehat{v}$ a valuation function of S satisfying $\widehat{v}'(t) = \min_{v\in S} v'(t)$. We have

$$\begin{aligned}
\int_0^1 \widehat{v}'(x)\,\mathrm{d}x &= \int_0^t \widehat{v}'(x)\,\mathrm{d}x + \int_t^1 \widehat{v}'(x)\,\mathrm{d}x \\
&\leq t\cdot\widehat{v}'(0) + (1-t)\cdot\widehat{v}'(t) \\
&\leq t\cdot\widehat{v}'(0) + \frac{1-t}{t}\cdot\int_0^t \min_{v\in S} v'(x)\,\mathrm{d}x \\
&\leq t\cdot\widehat{v}'(0) + \frac{1-t}{t\cdot q(0)}\cdot\int_0^1 R_{q,\mathbf{F}}(x)\,\mathrm{d}x.
\end{aligned} \tag{7.11}$$

The first two inequalities follow due to the concavity of the valuation function $\widehat{v}$ and its definition, while the third one is due to Equation (7.10). Now, we divide both sides of Equation (7.11) by $\widehat{v}'(0)$ and use the facts that the valuation function $\widehat{v}$ has IMOTV at most κ and belongs to set S (implying that $\widehat{v}'(0) \geq F_0^{-1}(1-q(0))$). We obtain

$$\frac{1}{\kappa} \leq \frac{\int_0^1 \widehat{v}'(x)\,\mathrm{d}x}{\widehat{v}'(0)} \leq t + \frac{1-t}{t}\cdot\frac{\int_0^1 R_{q,\mathbf{F}}(x)\,\mathrm{d}x}{q(0)\cdot\widehat{v}'(0)} \leq t + \frac{1-t}{t}\cdot\frac{\int_0^1 R_{q,\mathbf{F}}(x)\,\mathrm{d}x}{R_{q,\mathbf{F}}(0)}. \tag{7.12}$$

Rearranging Equation (7.12), we get

$$\frac{R_{q,\mathbf{F}}(0)}{\int_0^1 R_{q,\mathbf{F}}(x)\,\mathrm{d}x} \leq \frac{1-t}{t\cdot\left(\frac{1}{\kappa}-t\right)}, \tag{7.13}$$

and setting $t = 1-\sqrt{1-\frac{1}{\kappa}}$, Equation (7.13) yields

$$\frac{R_{q,\mathbf{F}}(0)}{\int_0^1 R_{q,\mathbf{F}}(x)\,\mathrm{d}x} \le \left(1-\sqrt{1-\frac{1}{\kappa}}\right)^{-2} \le 4\kappa^2,$$

as desired. The last inequality follows since the function $1-\sqrt{1-z}$ is convex in $[0,1]$ and has a derivative equal to $1/2$ at $z=0$. Hence, $1-\sqrt{1-z} \ge z/2$ and the inequality follows using $z = 1/\kappa$. $\square$

We are ready to prove our first result using Lemmas 7.3.1, 7.3.2, and 7.3.3; notice that the derivatives of the valuation functions may follow non-regular probability distributions.

Theorem 7.3.4. *The revenue gap of linear posted pricing for a single agent with random concave valuations of IMOTV at most κ is at most $\mathcal{O}(\ln\kappa)$.*

Proof. Let $t \in [0,1]$. The objective of the mathematical program (7.6) is

$$\begin{aligned}
\int_0^1 R_{q,\mathbf{F}}(x)\,\mathrm{d}x &= \int_0^t R_{q,\mathbf{F}}(x)\,\mathrm{d}x + \int_t^1 R_{q,\mathbf{F}}(x)\,\mathrm{d}x \\
&\le t\cdot R_{q,\mathbf{F}}(0) + \int_t^1 \frac{\mathrm{d}x}{x} \\
&\le 4t\kappa^2 \cdot \int_0^1 R_{q,\mathbf{F}}(x)\,\mathrm{d}x + \ln\frac{1}{t}.
\end{aligned}$$

The first inequality follows by Lemmas 7.3.1 and 7.3.2 and the second one by Lemma 7.3.3. Equivalently, we have

$$\int_0^1 R_{q,\mathbf{F}}(x)\,\mathrm{d}x \le \frac{\ln\frac{1}{t}}{1-4t\kappa^2}.$$

Using $t = \frac{1}{8\kappa^2}$, we get that the objective of the mathematical program (7.6) is upper-bounded by $2\ln(8\kappa^2) \in \mathcal{O}(\ln\kappa)$. $\square$

It is not hard to extend the result to the case of n agents (by just considering the agent who contributes the most to the optimal revenue); again, the derivatives of the valuation functions may follow non-regular probability distributions.

Corollary 7.3.5. *The revenue gap of linear posted pricing for n agents with random concave valuations with IMOTV at most κ is at most $\mathcal{O}(n\ln\kappa)$.*

We remark that the linear dependency on n is necessary, as indicated by a lower bound of [7] on the revenue gap of anonymous pricing for agents with non-regular valuations.

A lower bound

We now show that our bound in Theorem 7.3.4 is tight, even for valuation function with derivative values following regular probability distributions.

Theorem 7.3.6. *For every $\kappa > 1$, there exists an instance with a single agent with random concave valuation functions of IMOTV at most κ and with all valuation derivative values following a regular probability distribution, so that the revenue gap of linear pricing is $\Omega(\ln \kappa)$.*

Proof. Let ρ be such that $1+\ln(\kappa\cdot\rho) = \rho$, i.e., $\rho \geq 1+\ln\kappa$. For a parameter $t \geq 0$, let v_t be the valuation function defined as

$$v_t(x) = \begin{cases} (\kappa+t)\cdot x, & x \in \left[0, \frac{1}{\kappa\cdot\rho}\right] \\ \frac{1}{\rho}\cdot(1+\ln(\kappa\cdot\rho\cdot x)) + t\cdot x, & x \in \left[\frac{1}{\kappa\cdot\rho}, 1\right] \end{cases}$$

The function $v_t(x)$ has derivative (with respect to x) equal to $\kappa+t$ for $x \in \left[0, \frac{1}{\kappa\cdot\rho}\right]$ and to $\frac{1}{\rho\cdot x}+t$ for $x \in \left[\frac{1}{\kappa\cdot\rho}, 1\right]$. Thus, it is concave. Since $t \geq 0$, its IMOTV is $\frac{v_t'(0)}{v_t(1)} = \frac{\kappa+t}{\frac{1}{\rho}\cdot(1+\ln(\kappa\cdot\rho)+t} = \frac{\kappa+t}{1+t} \leq \kappa$.

Consider a single agent who draws a random valuation function v_t by selecting t uniformly at random from the interval $[0, 1/\rho]$. Thus, the derivative $v_t'(x)$ is uniformly random in $[\kappa, \kappa+1/\rho]$ if $x \in \left[0, \frac{1}{\kappa\cdot\rho}\right]$ and uniformly random in $\left[\frac{1}{\rho\cdot x}, \frac{1}{\rho\cdot x}+\frac{1}{\rho}\right]$ if $x \in \left[\frac{1}{\kappa\cdot\rho}, 1\right]$, i.e., drawn from regular probability distributions.

Using the pricing function $p(x) = v_0(x)$, the agent will buy the whole item, yielding a revenue of $v_0(1) = 1$. Now, consider the linear pricing p and first observe that $D(v_t, p) \leq D(v_{1/\rho}, p)$ for every $t \in [0, 1/\rho]$. Next, notice that $D(v_{1/\rho}, p) = 0$ if $p > \kappa+1/\rho$, $D(v_{1/\rho}, p) = \frac{1}{\rho\cdot(p-1/\rho)}$ if $2/\rho \leq p \leq \kappa+1/\rho$, and $D(v_{1/\rho}, p) = 1$ if $p < 2/\rho$. Thus, the upper bound of $p\cdot D(v_{1/\rho}, p)$ on the expected revenue with the linear pricing p is 0, at most $\frac{p}{p\cdot(p-1/\rho)} \leq \frac{2}{\rho}$, and less than $2/\rho$, respectively. In this way, we obtain a revenue gap of at least $\rho/2 \in \Omega(\ln\kappa)$. □

7.4 The multi-agent case: a black-box reduction

We now prove an upper bound on the objective value of the mathematical program (7.3) by relating it to the mathematical program (7.1), and exploiting the revenue gap of [7] for anonymous item pricing (Theorem 7.2.1). In particular, we prove the following statement for multiple agents.

Theorem 7.4.1. *The revenue gap when selling a divisible item using linear pricing to multiple agents with random concave valuations of IMOTV at most κ and with derivatives following regular probability distributions is at most $\mathcal{O}(\kappa^2)$.*

Proof. Our main tool is the following transformation. We use the notation $(\mathbf{q},\mathbf{F})$ as abbreviation for the probabilities $q_i(x)$ for every agent $i \in [n]$ and $x \in [0,1]$ and the cumulative density functions $F_{i,x}(t)$ for every agent $i \in [n]$, $x \in [0,1]$, and $t \geq 0$. Given the pair $(\mathbf{q},\mathbf{F})$, we define $r_i = \int_0^1 q_i(x)\,\mathrm{d}x$ and $H_i(t) = F_{i,0}(2\kappa t)$ for every agent $i \in [n]$ and $t \geq 0$, and use the pair $(\mathbf{r},\mathbf{H})$ as their abbreviation.

Bounding the objective value of mathematical program (7.3) has two steps, implemented in Lemmas 7.4.2 and 7.4.6, respectively.

Lemma 7.4.2. *Given a feasible solution $(\mathbf{q},\mathbf{F})$ of the mathematical program (7.3), the solution $(\mathbf{r},\mathbf{H})$ is a feasible solution of the mathematical program (7.1) with $R = 2\kappa - 1$.*

Proof. Clearly, $r_i \in [0,1]$ and H_i is a regular cumulative density function (since $F_{0,i}$ is regular as well). Furthermore, the solution $(\mathbf{r},\mathbf{H})$ satisfies the first constraint of the mathematical program (7.1) since $(\mathbf{q},\mathbf{F})$ satisfies constraint (7.4) of the mathematical program (7.3). In the following, we show that $(\mathbf{r},\mathbf{H})$ satisfies the second constraint of the mathematical program (7.1) as well. We will need three technical lemmas.

Lemma 7.4.3. *Let X_1, X_2, ..., X_k be independent random variables with $X_i \in [0,1]$ for $i \in [k]$ and let $X = \sum_{i\in[k]} X_i$. Then,* $\mathsf{E}[\min\{1,X\}] \geq 1 - \prod_{i\in[k]}(1-\mathsf{E}[X_i])$.

Proof. We claim that $\mathsf{E}[\min\{1,X\}]$ is minimized when X_1, X_2, ..., X_k are Bernoulli random variables. Then, it will be

$$\mathsf{E}[\min\{1,X\}] = 1 - \prod_{i\in[k]} \Pr[X_i = 0] = 1 - \prod_{i\in[k]} (1-\mathsf{E}[X_i]),$$

and the lemma will follow.

To prove this claim, we show that by replacing the random variable X_k with the Bernoulli random variable Y_k with $\Pr[Y_k = 1] = \mathsf{E}[X_k]$ and $\Pr[Y_k = 0] = 1-\mathsf{E}[X_k]$ the expectation $\mathsf{E}[\min\{1,X\}]$ can only become smaller. The claim will follow by repeating this argument and replacing each random variable X_i with a Bernoulli one that has the same expectation.

Formally, let $Y = X_1 + \ldots + X_{k-1} + Y_k = X' + Y_k$; we will show that $\mathsf{E}[\min\{1,X\}] \geq \mathsf{E}[\min\{1,Y\}]$. Denoting by G the cdf of the random variable Y_k, we have that, conditioned on $X' = w$, the expected contribution of X_k to $\min\{1,X\}$ is

$$\begin{aligned}
\mathsf{E}[\min\{1,X\} - X' | X' = w] &= \int_0^{1-w} (1-G(z))\,\mathrm{d}z \geq (1-w)\int_0^1 (1-G(z))\,\mathrm{d}z \\
&= (1-w)\mathsf{E}[X_k] = (1-w)\Pr[Y_k = 1] \\
&= \mathsf{E}[\min\{1,Y\} - X' | X' = w].
\end{aligned}$$

The inequality follows since $1-G(z)$ is non-increasing in z and, subsequently, $\int_0^t (1-G(z))\,\mathrm{d}z$ is concave in t and has no point below the line $t\int_0^t (1-G(z))\,\mathrm{d}z$ for $t \in [0,1]$. Denoting the pdf of the random variable X' by f, we have

$$\mathsf{E}[\min\{1,X\}] = \int_0^1 f(w)(w + \mathsf{E}[\min\{1,X\} - X' | X' = w])\,\mathrm{d}w$$

$$
\begin{aligned}
&\geq \int_0^1 f(w)(w + \mathsf{E}[\min\{1,Y\} - X' | X' = w])\,\mathrm{d}w \\
&= \mathsf{E}[\min\{1,Y\}],
\end{aligned}
$$

as desired. □

Lemma 7.4.4. *For every agent $i \in [n]$ and any price p, it holds that $\mathsf{E}_{v_i \sim \mathbf{F}_i}[D(v_i,p)] \geq \frac{1-H_i(p)}{2\kappa-1}$.*

Proof. Assume that $v_i'(0) \geq 2\kappa p$; hence, $v_i(1) \geq 2p$. By the definition of $D(v_i,p)$, we have that $v_i'(z) \leq p$ for $z \geq D(v_i,p)$. Hence,

$$
v_i(D(v_i,p)) \geq v_i(1) - v_i'(D(v_i,p)) \cdot (1 - D(v_i,p)) \geq v_i(1) - p \cdot (1 - D(v_i,p)) \tag{7.14}
$$

Furthermore, by our assumption on the IMOTV bound of the valuation functions, we have $v_i'(z) \leq \kappa \cdot v_i(1)$ for $z \leq D(v_i,p)$. Hence,

$$
v_i(D(v_i,p)) \leq \kappa \cdot v_i(1) \cdot D(v_i,p). \tag{7.15}
$$

Now, (7.14) and (7.15) yield

$$
D(v_i,p) \geq \frac{v_i(1) - p}{\kappa \cdot v_i(1) - p} \geq \frac{1}{2\kappa - 1}.
$$

Thus,

$$
\mathsf{E}_{v_i \sim \mathbf{F}_i}[D(v_i,p)] \geq \mathsf{E}_{v_i \sim \mathbf{F}_i}[D(v_i,p)\mathbb{1}\{v_i'(0) \geq 2\kappa p\}] \geq \frac{1 - F_{i,0}(2\kappa p)}{2\kappa - 1} = \frac{1 - H_i(p)}{2\kappa - 1}. \square
$$

Lemma 7.4.5. *Let k be an integer and $0 \leq z_1, z_2, \ldots, z_k \leq 1$. Then, for every $t \in (0,1)$, it holds that*

$$
1 - \prod_{i \in [k]} (1 - t \cdot z_i) \geq t \left(1 - \prod_{i \in [k]} (1 - z_i)\right)
$$

Proof. It suffices to show that the LHS is concave as a function of t; then it is at least as high as the line that connects points $(0,0)$ and $(1, 1 - \prod_{i \in [k]} (1 - z_i))$, i.e., the RHS of the above inequality. Indeed, the first derivative of the LHS is equal to

$$
\prod_{i \in [k]} (1 - t \cdot z_i) \cdot \sum_{i \in [k]} \frac{z_i}{1 - t \cdot z_i}
$$

and its second derivative is equal to

$$
-\prod_{i \in [k]} (1 - t \cdot z_i) \left(\left(\sum_{i \in [k]} \frac{z_i}{1 - t \cdot z_i} \right)^2 - \sum_{i \in [k]} \left(\frac{z_i}{1 - t \cdot z_i} \right)^2 \right) < 0,
$$

as desired. □

Now, using the constraint (7.5) of the mathematical program (7.3) for $p > 2\kappa - 1$ and applying Lemma 7.4.3 (with $k = n$ and $X_i = D(v_i, p)$ for $i \in [n]$), Lemma 7.4.4 and Lemma 7.4.5, we have

$$\begin{aligned}
1 &\geq p \cdot \mathsf{E}_{\mathbf{v}\sim\mathbf{F}}\left[\min\left\{1, \sum_{i\in[n]} D(v_i, p)\right\}\right] \geq p \cdot \left(1 - \prod_{i\in[n]} (1 - \mathsf{E}_{v_i\sim\mathbf{F}_i}[D(v_i, p)])\right) \\
&\geq p \cdot \left(1 - \prod_{i\in[n]} \left(1 - \frac{1 - H_i(p)}{2\kappa - 1}\right)\right) \geq \frac{p}{2\kappa - 1} \cdot \left(1 - \prod_{i\in[n]} H_i(p)\right),
\end{aligned}$$

i.e., equivalently,

$$p\left(1 - \prod_{i\in[n]} H_i(p)\right) \leq R, \forall p > R,$$

with $R = 2\kappa - 1$, as the second constraint of the mathematical program (7.1) requires. □

The second step of our proof is to relate the objective values of the two mathematical programs (7.3) and (7.1).

Lemma 7.4.6. *The objective value of the mathematical program (7.3) at solution* $(\mathbf{q}, \mathbf{F})$ *is at most* 2κ *times the objective value of the mathematical program (7.1) with* $R = 2\kappa - 1$ *at solution* $(\mathbf{r}, \mathbf{H})$.

Proof. Notice that, by the definition of $\mathbf{H}$, we have $F_{i,0}^{-1}(t) = 2\kappa H_i^{-1}(t)$. Hence,

$$\begin{aligned}
\int_0^1 q_i(x) F_{i,x}^{-1}(1 - q_i(x))\,\mathrm{d}x &\leq \int_0^1 q_i(x) F_{i,0}^{-1}(1 - q_i(x))\,\mathrm{d}x \\
&\leq \int_0^1 q_i(x)\,\mathrm{d}x \cdot F_{i,0}^{-1}\left(1 - \int_0^1 q_i(x)\,\mathrm{d}x\right) \\
&= 2\kappa r_i H_i^{-1}(1 - r_i).
\end{aligned}$$

The first inequality follows by the concavity of valuations which implies $F_{i,x}^{-1}(t)$ is non-increasing in x. The second inequality follows by applying Jensen inequality; recall that, by the regularity of $v_i'(0)$, the function $q \cdot F_{i,0}^{-1}(1 - q)$ is concave in q. □

Theorem 7.4.1 now follows by Theorem 7.2.1 and Lemmas 7.4.2 and 7.4.6. Indeed, these three statements imply a revenue gap of at most $2\kappa(2\kappa - 1)e$. □

7.5 The multi-agent case: a direct approach

We now present a direct approach to bound the objective value of the mathematical program (7.3), which will give us a better (logarithmic) dependency on κ at the expense of a logarithmic dependency on the number of agents. In particular, we prove the following statement for multiple agents.

Theorem 7.5.1. *The revenue gap when selling a divisible item using linear pricing to n agents with random concave valuations of IMOTV at most κ and with derivatives that follow regular probability distributions is at most $\mathcal{O}(\ln\kappa+\ln n)$.*

Proof. The proof of Theorem 7.5.1 is decomposed into two main parts. First, we show that the mathematical program (7.3) has a feasible solution that satisfies certain properties. Then, we use this solution and its properties to bound the maximum objective value.

Lemma 7.5.2. *Any feasible solution $(\mathbf{q},\mathbf{F})$ of the mathematical program (7.3) satisfies*

$$\sum_{i\in[n]}\int_0^1 (1-F_{i,x}(p))\,dx \leq \frac{1}{p-1}, \forall p>1.$$

Proof. We will prove the statement using the constraint (7.5) of the mathematical program (7.3) and the next technical lemma.

Lemma 7.5.3. *Let X_1, X_2, ..., X_k be independent random variables with $X_i\in[0,1]$ for $i\in[k]$ and let $X=\sum_{i\in[k]}X_i$. Then,* $\mathsf{E}[X]\leq\frac{\mathsf{E}[\min\{1,X\}]}{1-\mathsf{E}[\min\{1,X\}]}$.

Proof. Using Lemma 7.4.3 and applying the geometric-arithmetic mean inequality, we get

$$\begin{aligned}\mathsf{E}[\min\{1,X\}] &\geq 1-\prod_{i\in[k]}(1-\mathsf{E}[X_i])\\ &\geq 1-\left(\frac{1}{k}\cdot\sum_{i\in[k]}(1-\mathsf{E}[X_i])\right)^k = 1-\left(1-\frac{1}{k}\cdot\mathsf{E}[X]\right)^k,\end{aligned}$$

which implies that

$$\mathsf{E}[X]\leq k\cdot\left(1-(1-\mathsf{E}[\min\{1,X\}])^{1/k}\right).$$

By the concavity of function $z^{1/k}$, we get $z_1^{1/k}-z_2^{1/k}\leq\frac{1}{k}\cdot(z_1-z_2)\cdot z_1^{1/k-1}$ for $z_2<z_1$. Applying this inequality to the RHS above with $z_1=1$ and $z_2=1-\mathsf{E}[\min\{1,X\}]$, we get

$$\begin{aligned}\mathsf{E}[X] &\leq \mathsf{E}[\min\{1,X\}]\cdot(1-\left(1-\mathsf{E}[\min\{1,X\}]\right)^{1/k-1}\right)\\ &\leq \mathsf{E}[\min\{1,X\}]\cdot(1-\left(1-\mathsf{E}[\min\{1,X\}]\right)^{-1}\right)\\ &= \frac{\mathsf{E}[\min\{1,X\}]}{1-\mathsf{E}[\min\{1,X\}]},\end{aligned}$$

as desired. □

Lemma 7.5.3 clearly implies that $\mathsf{E}[X] \leq \frac{1}{p-1}$ when $\mathsf{E}[\min\{1,X\}] \leq 1/p$. Lemma 7.5.2 now follows by applying Lemma 7.5.3 with $k=n$ and the random variable $X_i = D(v_i,p)$ for $i \in [n]$, where v_i is drawn from the probability distribution $\mathbf{F}_i$. Notice that $\mathsf{E}_{v_i \in \mathbf{F}_i}[D(v_i,p)] = \int_0^1 (1-F_{i,x}(p))\,\mathrm{d}x$ and, hence, $\mathsf{E}[X] = \sum_{i\in[n]} X_i = \sum_{i\in[n]} \int_0^1 (1-F_{i,x}(p))\,\mathrm{d}x$, while $\mathsf{E}[\min\{1,X\}] \leq 1/p$ is due to constraint (7.5) of the mathematical program (7.3). □

Our next lemma shows that the mathematical program (7.3) has an approximately optimal feasible solution in which no selling probability is very close to 0.

Lemma 7.5.4. *The mathematical program (7.3) has a feasible solution $(\widetilde{\mathbf{q}},\mathbf{F})$ which furthermore satisfies $\widetilde{q}_i(x) \geq \frac{1}{2n}$ for every $i \in [n]$ and $x \in [0,1]$ and approximates its maximum objective value within a factor of 2.*

Proof. Consider the optimal feasible solution $(\mathbf{q},\mathbf{F})$ of the mathematical program (7.3) and define $\widetilde{q}_i(x) = \frac{1}{2n} + \frac{q_i(x)}{2}$ for $i \in [n]$ and $x \in [0,1]$. The only constraint of the mathematical program (7.3) that can be affected by the change from $\mathbf{q}$ to $\widetilde{\mathbf{q}}$ is (7.4). Due to the feasibility of solution $(\mathbf{q},\mathbf{F})$, we get

$$\sum_{i\in[n]} \int_0^1 \widetilde{q}_i(x)\,\mathrm{d}x = \sum_{i\in[n]} \int_0^1 \left(\frac{1}{2n} + \frac{q_i(x)}{2}\right)\mathrm{d}x = \frac{1}{2} + \frac{1}{2} \cdot \sum_{i\in[n]} \int_0^1 q_i(x)\,\mathrm{d}x \leq 1,$$

implying that solution $(\widetilde{\mathbf{q}},\mathbf{F})$ is feasible as well.

To prove that the objective value $(\widetilde{\mathbf{q}},\mathbf{F})$ approximates $(\mathbf{q},\mathbf{F})$ within 2, it suffices to prove that $2 \cdot R_{\widetilde{q}_i,\mathbf{F}_i}(x) \geq R_{q_i,\mathbf{F}_i}(x)$ for every $i \in [n]$ and $x \in [0,1]$. Let $i \in [n]$ and $x \in [0,1]$. The claim is obvious if $q_i(x) = 1/n$ (since $\widetilde{q}_i(x) = 1/n$ as well). We distinguish between two further cases.

Case 1: $q_i(x) < 1/n$. Notice that $\widetilde{q}_i(x) > q_i(x)$ in this case. Recall that due to the regularity of probability distribution $F_{i,x}$, the revenue-quantile curve $z \cdot F_{i,x}^{-1}(1-z)$ is concave. Thus, it takes values above the line $q(z) = \frac{q_i(x)}{1-q_i(x)} \cdot F_{i,x}^{-1}(1-q_i(x)) \cdot (1-z)$ connecting points $(q_i(x), q_i(x) \cdot F_{i,x}^{-1}(1-q_i(x)))$ and $(1,0)$. Hence,

$$\begin{aligned}
2 \cdot R_{\widetilde{q}_i,\mathbf{F}_i}(x) &= 2 \cdot \left(\frac{1}{2n} + \frac{q_i(x)}{2}\right) \cdot F_{i,x}^{-1}\left(1 - \frac{1}{2n} - \frac{q_i(x)}{2}\right) \\
&\geq 2 \cdot \frac{q_i(x)}{1-q_i(x)} \cdot F_{i,x}^{-1}(1-q_i(x)) \cdot \left(1 - \frac{1}{2n} - \frac{q_i(x)}{2}\right) \\
&\geq q_i(x) \cdot F_{i,x}^{-1}(1-q_i(x)) = R_{q_i,\mathbf{F}_i}(x),
\end{aligned}$$

as desired.

Case 2: $q_i(x) > 1/n$. Notice that $\widetilde{q}_i(x) < q_i(x)$ in this case. Again, due to the regularity of probability distribution $F_{i,x}$, the revenue-quantile curve $z \cdot F_{i,x}^{-1}(1-z)$ is concave and takes values above the line $g(z) = F_{i,x}^{-1}(1-q_i(x)))\cdot z$ connecting points $(0,0)$ and $(q_i(x), q_i(x)\cdot F_{i,x}^{-1}(1-q_i(x)))$. Hence,

$$\begin{aligned}
2\cdot R_{\widetilde{q}_i,\mathbf{F}_i}(x) &= 2\cdot\left(\frac{1}{2n}+\frac{q_i(x)}{2}\right)\cdot F_{i,x}^{-1}\left(1-\frac{1}{2n}-\frac{q_i(x)}{2}\right)\\
&\geq 2\cdot F_{i,x}^{-1}(1-q_i(x))\cdot\left(\frac{1}{2n}+\frac{q_i(x)}{2}\right)\\
&\geq q_i(x)\cdot F_{i,x}^{-1}(1-q_i(x)) = R_{q_i,\mathbf{F}_i}(x),
\end{aligned}$$

completing the proof. □

The proof of the next lemma follows a similar structure to the proof of Lemma 7.3.3 in Section 7.3. However, due to the constraint (7.4) on the selling probabilities, the simple argument about their optimality that we used in the proof of Lemma 7.3.3 cannot be used here; the dependency on the number of agents in our analysis is mainly introduced due to this subtle issue.

Lemma 7.5.5. *Let $(\widetilde{\mathbf{q}},\mathbf{F})$ be any feasible solution of the mathematical program (7.3) that satisfies the additional constraint $\widetilde{q}_i(x) \geq \frac{1}{2n}$ for every $i \in [n]$ and $x \in [0,1]$. Then, for every $i \in [n]$, it holds*

$$R_{\widetilde{q}_i,\mathbf{F}_i}(0) \leq 8n\kappa^2 \int_0^1 R_{\widetilde{q}_i,\mathbf{F}_i}(x)\,dx. \tag{7.16}$$

Proof. Consider an agent $i \in [n]$ and define $S_i = \left\{v_i \in \mathscr{V}_\kappa : v_i'(0) \geq F_{i,0}^{-1}(1-\widetilde{q}_i(0))\right\}$ to be the set of valuation functions drawn from $\mathbf{F}_i$ with derivative at least $F_{i,0}^{-1}(1-\widetilde{q}_i(0))$ at point 0. By the definition of set S_i, we have $\Pr_{v\sim\mathbf{F}_i}[v\in S_i] = 1-\widetilde{q}_i(0)$. Hence,

$$\min_{v\in S_i} v'(x) \leq F_x^{-1}(1-\widetilde{q}_i(0)) \leq F_x^{-1}(1-\widetilde{q}_i(x)) \leq \frac{2n}{\widetilde{q}_i(0)}\cdot\widetilde{q}_i(x)\cdot F_x^{-1}(1-\widetilde{q}_i(x)). \tag{7.17}$$

The second inequality follows due to the constraint for monotone non-increasing selling probability in mathematical program (7.3), and the third one since $\widetilde{q}_i(x) \geq \frac{1}{2n}$ and $\widetilde{q}_i(0) \leq 1$. By Equation (7.17), we get

$$\int_0^1 \min_{v\in S_i} v'(x)\,\mathrm{d}x \leq \frac{2n}{\widetilde{q}(0)}\cdot\int_0^1 R_{\widetilde{q}_i,\mathbf{F}_i}(x)\,\mathrm{d}x. \tag{7.18}$$

Now, let $t\in[0,1]$ and $\widehat{v}$ be a valuation function of S_i satisfying $\widehat{v}'(t) = \min_{v\in S_i} v'(t)$. We have

$$\int_0^1 \widehat{v}'(x)\,\mathrm{d}x = \int_0^t \widehat{v}'(x)\,\mathrm{d}x + \int_t^1 \widehat{v}'(x)\,\mathrm{d}x$$

$$\leq t \cdot \widehat{v}'(0) + (1-t) \cdot \widehat{v}'(t)$$
$$\leq t \cdot \widehat{v}'(0) + \frac{1-t}{t} \cdot \int_0^t \min_{v \in S} v'(x)\, \mathrm{d}x$$
$$\leq t \cdot \widehat{v}'(0) + \frac{2n \cdot (1-t)}{t \cdot \widetilde{q}_i(0)} \cdot \int_0^1 R_{\widetilde{q}_i, \mathbf{F}_i}(x)\, \mathrm{d}x. \tag{7.19}$$

The first two inequalities are due to the concavity of the valuation function $\widehat{v}$ (and its definition), while the third one is due to Equation (7.18). Now, we divide both sides of Equation (7.19) by $\widehat{v}'(0)$ and use the facts that the valuation function $\widehat{v}$ has IMOTV at most κ and belongs to set S_i (and, therefore, $\widehat{v}'(0) \geq F_{i,0}^{-1}(1-\widetilde{q}_i(0))$. We obtain

$$\frac{1}{\kappa} \leq \frac{\int_0^1 \widehat{v}'(x)\, \mathrm{d}x}{\widehat{v}'(0)} \leq t + \frac{1-t}{t} \cdot \frac{\int_0^1 R_{q,\mathbf{F}}(x)\, \mathrm{d}x}{\widetilde{q}(0) \cdot \widehat{v}'(0)} \leq t + \frac{2n \cdot (1-t)}{t} \cdot \frac{\int_0^1 R_{\widetilde{q}_i, \mathbf{F}_i}(x)\, \mathrm{d}x}{R_{\widetilde{q}_i, \mathbf{F}_i}(0)}. \tag{7.20}$$

Rearranging Equation (7.20), we get

$$\frac{R_{\widetilde{q}_i, \mathbf{F}_i}(0)}{\int_0^1 R_{\widetilde{q}_i, \mathbf{F}_i}(x)\, \mathrm{d}x} \leq \frac{2n \cdot (1-t)}{t \cdot \left(\frac{1}{\kappa} - t\right)}. \tag{7.21}$$

Similarly to the last step in the proof of Lemma 7.3.3, setting $t = 1 - \sqrt{1 - \frac{1}{\kappa}}$, Equation (7.21) yields

$$\frac{R_{\widetilde{q}_i, \mathbf{F}_i}(0)}{\int_0^1 R_{\widetilde{q}_i, \mathbf{F}_i}(x)\, \mathrm{d}x} \leq 2n \cdot \left(1 - \sqrt{1 - \frac{1}{\kappa}}\right)^{-2} \leq 8n\kappa^2,$$

as desired. □

The next two lemmas refer to optimal feasible solutions of the mathematical program (7.3), among those satisfying the additional constraint.

Lemma 7.5.6. *Let $(\widetilde{\mathbf{q}}, \mathbf{F})$ be an optimal feasible solution to mathematical program (7.3), under the additional constraint $\widetilde{q}_i(x) \geq \frac{1}{2n}$ for $i \in [n]$ and $x \in [0,1]$. Then, $R_{\widetilde{q}_i, \mathbf{F}_i}(x)$ is monotone non-decreasing in x for every $i \in [n]$.*

Proof. For the sake of contradiction, let $0 \leq x_1 < x_2 \leq 1$ and assume that $R_{\widetilde{q}_i, \mathbf{F}_i}(x) < R_{\widetilde{q}_i, \mathbf{F}_i}(x_2)$ for every $x \in [x_1, x_2)$. Then, notice that due to the concavity of the valuation functions, we have

$$\widetilde{q}_i(x_2) \cdot F_{i,x}^{-1}(1-\widetilde{q}_i(x_2)) \geq \widetilde{q}_i(x_2) \cdot F_{i,x_2}^{-1}(1-\widetilde{q}_i(x_2)) > \widetilde{q}_i(x_1) \cdot F_{i,x_1}^{-1}(1-\widetilde{q}_i(x_1)).$$

This contradicts the optimality of the solution $(\widetilde{\mathbf{q}}, \mathbf{F})$ since decreasing $\widetilde{q}_i(x)$ to $\widetilde{q}_i(x_2)$ for $x \in [x_1, x_2)$ would strictly increase the objective value of the mathematical program (7.3), without violating constraint (7.4), the constraint for monotone non-increasing selling probability, as well as the additional constraint (recall that $\widetilde{q}_i(x_2) \geq \frac{1}{2n}$). □

Lemma 7.5.7. *Let $(\widetilde{\mathbf{q}}, \mathbf{F})$ be an optimal feasible solution to mathematical program (7.3), under the additional constraint $\widetilde{q}_i(x) \geq \frac{1}{2n}$ for $i \in [n]$ and $x \in [0,1]$. Then, $R_{\widetilde{q}_i,\mathbf{F}_i}(x) \leq 256 n^2 \kappa^4$ for every $i \in [n]$ and $x \in [0,1]$.*

Proof. Let $i \in [n]$ and $z \in [0,1]$; for the linear price $p = F_{i,z}^{-1}(1-\widetilde{q}_i(z))$, the constraint (7.5) of the mathematical program (7.3) yields

$$\begin{aligned}\frac{1}{F_{i,z}^{-1}(1-\widetilde{q}_i(z))} &\geq \mathsf{E}_{\mathbf{v}\sim\mathbf{F}}\left[\min\left\{1, \sum_{i\in[n]} D(v_i, F_{i,z}^{-1}(1-\widetilde{q}_i(z)))\right\}\right] \\ &\geq \mathsf{E}_{v_i\sim\mathbf{F}_i}\left[D(v_i, F_{i,z}^{-1}(1-\widetilde{q}_i(z)))\right] \geq z\cdot\widetilde{q}_i(z).\end{aligned}$$

The last inequality is due to the fact that, by definition, the demand $D(v_i, F_{i,z}^{-1}(1-\widetilde{q}_i(z)))$ is at least z with probability $\widetilde{q}_i(z)$. Hence,

$$R_{\widetilde{q}_i,\mathbf{F}_i}(z) = \widetilde{q}_i(z)\cdot F_{i,z}^{-1}(1-\widetilde{q}_i(z)) \leq \frac{1}{z}, \tag{7.22}$$

for every $z \in [0,1]$.

Now, let $t \in [0,1]$. Using Lemma 7.5.5, Equation (7.22), and the inequality $R_{\mathbf{q}_i,\mathbf{F}_i}(0) \geq R_{\widetilde{q}_i,\mathbf{F}_i}(x)$ from Lemma 7.5.6, we have

$$\begin{aligned}\frac{R_{\widetilde{q}_i,\mathbf{F}_i}(0)}{8n\kappa^2} &\leq \int_0^1 R_{\widetilde{q}_i,\mathbf{F}_i}(x)\,\mathrm{d}x = \int_0^t R_{\widetilde{q}_i,\mathbf{F}_i}(x)\,\mathrm{d}x + \int_t^1 R_{\widetilde{q}_i,\mathbf{F}_i}(x)\,\mathrm{d}x \\ &\leq t\cdot R_{\widetilde{q}_i,\mathbf{F}_i}(0) + \int_t^1 \frac{\mathrm{d}x}{x} = t\cdot R_{\widetilde{q}_i,\mathbf{F}_i}(0) + \ln\frac{1}{t}.\end{aligned}$$

Setting $t = \left(R_{\widetilde{q}_i,\mathbf{F}_i}(0)\right)^{-1}$ and using the inequality $1+\ln z \leq 2\sqrt{z}$, we get

$$\frac{R_{\mathbf{q}_i,\mathbf{F}_i}(0)}{8n\kappa^2} \leq 1 + \ln R_{\widetilde{q}_i,\mathbf{F}_i}(0) \leq 2\cdot\sqrt{R_{\widetilde{q}_i,\mathbf{F}_i}(0)}$$

and, equivalently, $R_{\widetilde{q}_i,\mathbf{F}_i}(0) \leq 256 n^2\kappa^4$. The lemma now follows due to Lemma 7.5.6. □

To summarize the proof of Theorem 7.5.1 so far, Lemmas 7.5.2, 7.5.4, and 7.5.7 imply that the mathematical program (7.3) has a feasible solution $(\mathbf{q}, \mathbf{F})$ that approximates its maximum objective value within a factor of 2 and satisfies the next two sets of inequalities:

$$\sum_{i\in[n]} \int_0^1 (1-F_{i,x}(p))\,\mathrm{d}x \leq \frac{1}{p-1}, \quad \forall p > 1 \tag{7.23}$$

$$R_{q_i,\mathbf{F}_i}(x) \leq 256 n^2\kappa^4, \quad \forall i \in [n], x \in 0,1] \tag{7.24}$$

Using this solution, we define the subsets A, B, and C_t for $t \geq 2$ of set $[n] \times [0,1]$:

$$\begin{aligned}
A &= \Big\{(i,x) \in [n] \times [0,1] \text{ s.t. } F_{i,x}^{-1}(1-q_i(x)) < 2\Big\} \\
B &= \Big\{(i,x) \in [n] \times [0,1] \text{ s.t. } F_{i,x}^{-1}(1-q_i(x)) \geq 2 \\
&\qquad \text{and } q_i(x) \cdot F_{i,x}^{-1}(1-q_i(x)) < 2 \cdot (1-q_i(x))\Big\} \\
C_t &= \Big\{(i,x) \in [n] \times [0,1] \text{ s.t. } F_{i,x}^{-1}(1-q_i(x)) \geq t \\
&\qquad \text{and } q_i(x) \cdot F_{i,x}^{-1}(1-q_i(x)) \geq t \cdot (1-q_i(x))\Big\}
\end{aligned}$$

Furthermore, we define

$$\mathbf{I}\left(g_i(x);(i,x) \in X\right) = 2 \cdot \sum_{i \in [n]} \int_0^1 g_i(x)\, \mathbb{1}\{(i,x) \in X\}\, \mathrm{d}x$$

for functions $g_i : [0,1] \to \mathsf{R}_{\geq 0}$ for $i \in [n]$ and $X \subseteq [n] \times [0,1]$. Thus, the maximum objective value of the mathematical program (7.3) is upper-bounded by

$$\begin{aligned}
2 \cdot \sum_{i \in [n]} \int_0^1 q_i(x) \cdot F_{i,x}^{-1}(1-q_i(x))\, \mathrm{d}x &= \mathbf{I}(R_{q_i,\mathbf{F}_i}(x);(i,x) \in [m] \times [0,1]) \\
&= \mathbf{I}(R_{q_i,\mathbf{F}_i}(x);(i,x) \in A) + \mathbf{I}(R_{q_i,\mathbf{F}_i}(x);(i,x) \in B) \\
&\quad + \sum_{j=1}^{\lfloor \log(256n^2\kappa^4) \rfloor} \mathbf{I}(R_{q_i,\mathbf{F}_i}(x);(i,x) \in C_{2^j} \setminus C_{2^{j+1}}).
\end{aligned} \tag{7.25}$$

The last sum runs up to $\lfloor \log(256n^2\kappa^4) \rfloor$ since, due to the inequalities (7.24), the set C_t is empty for $t > 256n^2\kappa^4$. We will complete the proof of Theorem 7.5.1 by upper-bounding each summand in the RHS of Equation (7.25) by a constant. In doing so, we will be assisted by a last technical lemma.

Lemma 7.5.8. *Let F be a regular probability distribution, $z \in [0,1]$, and $\ell > 0$ such that $\ell \leq F^{-1}(1-z)$. Then,*

$$\frac{z \cdot F^{-1}(1-z)}{(1-z) \cdot \ell + z \cdot F^{-1}(1-z)} \leq 1 - F(\ell).$$

Proof. We use lower-left and upper-right points to define axis-parallel rectangles. Notice that the assumption $\ell \leq F^{-1}(1-z)$ implies that $z \leq 1 - F(\ell)$. We distinguish between two cases.

Case 1: $\ell \cdot (1-F(\ell)) \geq z \cdot F^{-1}(1-z)$. Then, the area of the rectangle defined by the points $(0,0)$ and $(1, z \cdot F^{-1}(1-z))$ is upper-bounded by the total area in the two rectangles defined by the points $(0,0)$ and $(1-F(\ell), z \cdot F^{-1}(1-z))$ and by the points $(z,0)$ and $(1, \ell \cdot (1-F(\ell)))$, respectively. Thus,

$$z \cdot F^{-1}(1-z) \leq (1-F(\ell)) \cdot z \cdot F^{-1}(1-z) + (1-z) \cdot \ell \cdot (1-F(\ell)),$$

which is equivalent to the claimed inequality.

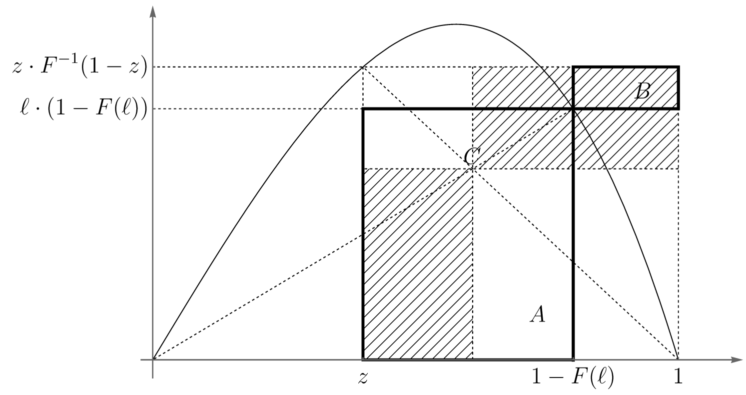

Figure 7.1: An illustration of the argument in the proof of Lemma 7.5.8 (Case 2) using the revenue-quantile curve $q \cdot F^{-1}(1-q)$. To compare the two rectangles with thick sides A and B, we show that due to concavity of the revenue curve $q \cdot F^{-1}(1-q)$, rectangle A contains the lower-left shaded region corresponding to rectangle $\widetilde{A}$ and the upper-right shaded region corresponding to rectangle $\widetilde{B}$ contains rectangle B. To complete the proof, it suffices to observe that the two shaded rectangles have equal areas.

Case 2: $\ell \cdot (1-F(\ell)) < z \cdot F^{-1}(1-z)$. Denote by A the axis-parallel rectangle defined by points $(z,0)$ and $(1-F(\ell), \ell \cdot (1-F(\ell)))$ and observe that its area is equal to $\ell \cdot (1-F(\ell)) \cdot (1-F(\ell)-z)$. Also, denote by B the axis-parallel rectangle defined by points $(1-F(\ell), \ell \cdot (1-F(\ell)))$ and $(1, z \cdot (F^{-1}(1-z))$ and observe that its area is equal to $F(\ell)(z \cdot F^{-1}(1-z) - \ell \cdot (1-F(\ell))$. We claim that rectangle A is at least as large as rectangle B and, hence,

$$
\begin{aligned}
& z \cdot F^{-1}(1-z) \leq z \cdot F^{-1}(1-z) + \text{area}(A) - \text{area}(B) \\
& = z \cdot F^{-1}(1-z) + (1-F(\ell)-z) \cdot \ell \cdot (1-F(\ell)) - F(\ell)(z \cdot F^{-1}(1-z) - \ell \cdot (1-F(\ell))) \\
& = (1-F(\ell) \cdot (z \cdot F^{-1}(1-z) + \ell \cdot (1-z)),
\end{aligned}
$$

which is equivalent to the stated inequality. The argument is illustrated in Figure 7.1.

To prove the claim $\text{area}(A) \geq \text{area}(B)$, it suffices to consider the point C at the intersection of the line connecting points $(z, z \cdot F^{-1}(1-z))$ and $(1,0)$ with the line connecting points $(0,0)$ and $(1-F(\ell), \ell \cdot (1-F(\ell)))$. Now, denote by $\widetilde{A}$ and $\widetilde{B}$ the axis-parallel rectangles defined by points $(z,0)$ and C and by points C and $(1, z \cdot F^{-1}(1-z))$, respectively. Due to the concavity of the revenue-quantile curve $q \cdot F^{-1}(1-q)$, the point $(1-F(\ell), \ell \cdot (1-F(\ell)))$ lies inside rectangle $\widetilde{B}$ and point C lies inside rectangle A. Thus, $\text{area}(A) \geq \text{area}(\widetilde{A})$ and $\text{area}(B) \leq \text{area}(\widetilde{B})$. Furthermore, observe that rectangles $\widetilde{A}$ and $\widetilde{B}$ have the same area (this would still hold even if we had picked point C anywhere in the line connecting points $(z, z \cdot F^{-1}(1-z))$ and $(1,0)$). Thus, we have

$$
\text{area}(A) - \text{area}(B) \geq \text{area}(\widetilde{A}) - \text{area}(\widetilde{B}) = 0
$$

as claimed. □

Now, we have

$$\mathbf{I}(R_{q_i,\mathbf{F}_i}(x);(i,x)\in A) \leq 2\cdot \mathbf{I}(q_i(x);(i,x)\in A) \leq 4.$$

The first inequality follows by the definition of set A, and the second one by inequality (7.23). Also,

$$\begin{aligned}
\mathbf{I}(R_{q_i,\mathbf{F}_i}(x);(i,x)\in B) &\leq 4\cdot \mathbf{I}\left(\frac{q_i(x)\cdot F_{i,x}^{-1}(1-q_i(x))}{4\cdot(1-q_i(x))};(i,x)\in B\right)\\
&\leq 4\cdot \mathbf{I}\left(\frac{q_i(x)\cdot F_{i,x}^{-1}(1-q_i(x))}{2\cdot(1-q_i(x))+q_i(x)\cdot F_{i,x}^{-1}(1-q_i(x))};(i,x)\in B\right)\\
&\leq 4\cdot \mathbf{I}(1-F_{i,x}(2);(i,x)\in B) \leq 8.
\end{aligned}$$

The first inequality is obvious, the second one follows by the definition of set B, the third one follows by applying Lemma 7.5.8 with $F=F_{i,x}$, $z=q_i(x)$, and $\ell=2$, and the fourth one follows by applying inequality (7.23) with $p=2$. Finally,

$$\begin{aligned}
&\mathbf{I}(R_{q_i,\mathbf{F}_i}(x);(i,x)\in C_{2^j}\setminus C_{2^{j+1}}) < \mathbf{I}\left(2^{j+1};(i,x)\in C_{2^j}\setminus C_{2^{j+1}}\right)\\
&= 2^{j+2}\cdot \mathbf{I}\left(\frac{1}{2};(i,x)\in C_{2^j}\setminus C_{2^{j+1}}\right)\\
&\leq 2^{j+2}\cdot \mathbf{I}\left(\frac{q_i(x)\cdot F_{i,x}^{-1}(1-q_i(x))}{2^j\cdot(1-q_i(x))+q_i(x)\cdot F_{i,x}^{-1}(1-q_i(x))};(i,x)\in C_{2^j}\setminus C_{2^{j+1}}\right)\\
&\leq 2^{j+2}\cdot \mathbf{I}\left(1-F_{i,x}(2^j);(i,x)\in C_{2^j}\setminus C_{2^{j+1}}\right)\\
&\leq \frac{2^{j+3}}{2^j-1} \leq 16.
\end{aligned}$$

The first inequality follows since $(i,x)\notin C_{2^{j+1}}$ and, hence, $q_i(x)\cdot F_{i,x}^{-1}(1-q_i(x)) < \max\{2^{j+1}\cdot(1-q_i(x),2^{j+1}\cdot q_i(x)\} \leq 2^{j+1}$. The second inequality follows since $(i,x)\in C_{2^j}$ and, hence, $q_i(x)\cdot F_{i,x}^{-1}(1-q_i(x)) \geq 2^j\cdot(1-q_i(x))$. The third inequality follows by applying Lemma 7.5.8 with $F=F_{i,x}$, $z=q_i(x)$, and $\ell=2^j$, and the fourth one by inequality (7.23) for $p=2^j$.

Overall, the RHS of Equation (7.25) and, consequently, the maximum objective value of mathematical program (7.3) and the revenue gap, is upper-bounded by $4+8+16\cdot\lfloor\log\left(256n^2\kappa^4\right)\rfloor \leq 140+32\cdot\log n+64\cdot\log\kappa \in \mathcal{O}(\ln\kappa+\ln n)$. The proof of Theorem 7.5.1 is now complete. □

7.6 Extensions and open problems

Motivated by recent work on “simple vs. optimal” revenue trade-offs in Bayesian mechanism design, we have explored the power of linear posted pricing when selling a perfectly divisible item. Our work leaves several interesting open problems. Of course, we would like to determine the tight bound on the revenue gap for linear pricing, closing the gap between our upper bound of $\mathcal{O}\left(\max\left\{\ln\kappa+\ln n, \kappa^2\right\}\right)$ and our lower bound of $\Omega(\ln\kappa)$. It is tempting to conjecture that the dependency on the number of agents is not necessary and $\Theta(\ln\kappa)$ is the tight bound on the revenue gap. Furthermore, we would like to explore whether the revenue gap increases significantly when agents are utility-maximizing instead of shortsighted. Preliminary attempts to show a separation between the two agent models have not been successful. Finally, the optimal revenue for the ex-ante relaxation of the multi-parameter setting in Section 7.2 could be used to bound the revenue gap of other mechanisms for allocating a divisible item, such as the proportional mechanism [101, 105], which has been extensively studied in terms of social welfare [28, 29, 45]. This could lead to results on the Bayes-Nash price of anarchy for revenue which, with a few exceptions such as [31, 91, 111], are rather sporadic in the literature.

Bibliography

[1] Dimitris Achlioptas. Setting 2 variables at a time yields a new lower bound for random 3-SAT. In *Proceedings of the 32nd Annual ACM Symposium on Theory of Computing (STOC)*, pages 28–37, 2000. 22, 66

[2] Dimitris Achlioptas. Random satisfiabiliy. In *Handbook of Satisfiability*, volume 336 of *Frontiers in Artificial Intelligence and Applications*, pages 437–462. 2nd edition, 2021. 68

[3] Dimitris Achlioptas and Yuval Peres. The threshold for random k-sat is $2klog2 - o(k)$. *Journal of the American Mathematical Society*, 17:947–973, 2004. 23, 66

[4] Dimitris Achlioptas and Gregory B. Sorkin. Optimal myopic algorithms for random 3-sat. In *Proceedings of 41st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 590–600, 2000. 22

[5] Heiner Ackermann, Heiko Röglin, and Berthold Vöcking. On the impact of combinatorial structure on congestion games. *Journal of the ACM*, 55(6): 25:1–25:22, 2008. 8, 43

[6] Saeed Alaei. Bayesian combinatorial auctions: Expanding single buyer mechanisms to many buyers. *SIAM Journal on Computing*, 43(2):930–972, 2014. 28, 95

[7] Saeed Alaei, Jason D. Hartline, Rad Niazadeh, Emmanouil Pountourakis, and Yang Yuan. Optimal auctions vs. anonymous pricing. *Games and Economic Behavior*, 118:494–510, 2019. 5, 28, 30, 31, 35, 36, 95, 96, 97, 99, 100, 101, 104, 105

[8] Noga Alon and Assaf Naor. Approximating the cut-norm via grothendieck's inequality. *SIAM Journal on Computing*, 35(4):787–803, 2006. 44

[9] Noga Alon and Asaf Nussboim. k-wise independent random graphs. In *Proceedings of 49th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 813–822, 2008. 4, 67

[10] Noga Alon, Alexandr Andoni, Tali Kaufman, Kevin Matulef, Ronitt Rubinfeld, and Ning Xie. Testing k-wise and almost k-wise independence. In *Proceedings of the 39th Annual ACM Symposium on Theory of Computing (STOC)*, pages 496–505, 2007. 68

[11] Elliot Anshelevich, Anirban Dasgupta, Jon M. Kleinberg, Éva Tardos, Tom Wexler, and Tim Roughgarden. The price of stability for network design with fair cost allocation. *SIAM Journal on Computing*, 38(4):1602–1623, 2008. 42

[12] Carlos Ansótegui, Maria Luisa Bonet, and Jordi Levy. On the structure of industrial SAT instances. In *Proceedings of 15th International Conference on Principles and Practice of Constraint Programming (CP)*, pages 127–141, 2009. 20

[13] Carlos Ansótegui, Jesús Giráldez-Cru, and Jordi Levy. The community structure of SAT formulas. In *Proceedings of 15th International Conference on Theory and Applications of Satisfiability Testing (SAT)*, pages 410–423, 2012. 20, 66

[14] Baruch Awerbuch, Yossi Azar, Amir Epstein, Vahab S. Mirrokni, and Alexander Skopalik. Fast convergence to nearly optimal solutions in potential games. In *Proceedings of the 9th ACM Conference on Electronic Commerce (EC)*, pages 264–273, 2008. 43

[15] Maria-Florina Balcan, Avrim Blum, and Yishay Mansour. Improved equilibria via public service advertising. In *Proceedings of the 20th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 728–737, 2009. 43

[16] Paul Beame, Richard Karp, Toniann Pitassi, and Michael Saks. The efficiency of resolution and davis–putnam procedures. *SIAM Journal on Computing*, 31 (4):1048–1075, 2002. 26, 69

[17] Eli Ben-Sasson and Avi Wigderson. Short proofs are narrow—resolution made simple. *Journal of the ACM*, 48(2):149–169, 2001. 26, 68, 69

[18] Itai Benjamini, Ori Gurel-Gurevich, and Ron Peled. On k-wise independent distributions and boolean functions. *arXiv:math*, abs/1201.3261, 2012. 4, 67

[19] Anand Bhalgat, Tanmoy Chakraborty, and Sanjeev Khanna. Nash dynamics in congestion games with similar resources. In *Proceedings of the 5th International Conference on Web and Internet Economics (WINE)*, pages 362–373, 2009. 8, 43

[20] Anand Bhalgat, Tanmoy Chakraborty, and Sanjeev Khanna. Approximating pure nash equilibrium in cut, party affiliation, and satisfiability games. In *Proceedings of the 11th ACM Conference on Electronic Commerce (EC)*, pages 73–82, 2010. 3, 8, 40, 42, 63

[21] Armin Biere, Marijn Heule, Hans van Maaren, and Toby Walsh, editors. *Handbook of Satisfiability - Second Edition*. 2021. 19

[22] Vittorio Bilò and Mauro Paladini. On the performance of mildly greedy players in cut games. *Journal of Combinatorial Optimization*, 32(4):1036–1051, 2016. 43

[23] Béla Bollobás, Christian Borgs, Jennifer T. Chayes, Jeong Han Kim, and David Bruce Wilson. The scaling window of the 2-SAT transition. *Random Structures and Algorithms*, 18(3):201–256, 2001. 21, 66

[24] Karl Bringmann. Why walking the dog takes time: Frechet distance has no strongly subquadratic algorithms unless SETH fails. In *Proceedings of 55th IEEE Annual Symposium on Foundations of Computer Science, (FOCS)*, pages 661–670, 2014. 66

[25] Andrei Z. Broder, Alan M. Frieze, and Eli Upfal. On the satisfiability and maximum satisfiability of random 3-CNF formulas. In *Proceedings of the 4th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 322–330, 1993. 22, 66

[26] Ioannis Caragiannis and Angelo Fanelli. On approximate pure nash equilibria in weighted congestion games with polynomial latencies. *Journal of Computer and System Sciences*, 117:40–48, 2021. 8, 13, 43

[27] Ioannis Caragiannis and Zhile Jiang. Computing better approximate pure nash equilibria in cut games via semidefinite programming. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing (STOC)*, pages 710–722, 2023. 3, 5

[28] Ioannis Caragiannis and Alexandros A. Voudouris. Welfare guarantees for proportional allocations. *Theory of Computing Systems*, 59(4):581–599, 2016. 97, 117

[29] Ioannis Caragiannis and Alexandros A. Voudouris. The efficiency of resource allocation mechanisms for budget-constrained users. *Mathematics of Operations Research*, 46(2):503–523, 2021. 97, 117

[30] Ioannis Caragiannis, Angelo Fanelli, Nick Gravin, and Alexander Skopalik. Efficient computation of approximate pure nash equilibria in congestion games. In *Proceedings of the 52nd IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 532–541, 2011. 8, 13, 40, 43

[31] Ioannis Caragiannis, Christos Kaklamanis, Panagiotis Kanellopoulos, and Maria Kyropoulou. Revenue guarantees in the generalized second price auction. *ACM Transactions on Internet Technology*, 14(2-3):17:1–17:19, 2014. 117

[32] Ioannis Caragiannis, Angelo Fanelli, Nick Gravin, and Alexander Skopalik. Approximate pure nash equilibria in weighted congestion games: Existence, efficient computation, and structure. *ACM Transactions on Economics and Computation*, 3(1):2:1–2:32, 2015. 8, 13, 43

[33] Ioannis Caragiannis, Angelo Fanelli, and Nick Gravin. Short sequences of improvement moves lead to approximate equilibria in constraint satisfaction games. *Algorithmica*, 77(4):1143–1158, 2017. 8, 13, 40, 41, 42, 43, 45, 48

[34] Ioannis Caragiannis, Nick Gravin, Pinyan Lu, and Zihe Wang. Relaxing the independence assumption in sequential posted pricing, prophet inequality, and random bipartite matching. In *Proceedings of 17th International Conference on Web and Internet Economics (WINE)*, pages 131–148, 2021. 68

[35] Ioannis Caragiannis, Nick Gravin, and Zhile Jiang. On the satisfiability of random 3-SAT formulas with k-wise independent clauses. *arXiv:math*, abs/2411.03813, 2024. 4, 5, 23, 24

[36] Ioannis Caragiannis, Zhile Jiang, and Apostolis Kerentzis. Bounds on the revenue gap of linear posted pricing for selling a divisible item. *CoRR*, abs/2007.08246, 2024. 4, 5

[37] Moses Charikar and Anthony Wirth. Maximizing quadratic programs: Extending grothendieck's inequality. In *Proceedings of the 45th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 54–60, 2004. 44

[38] Shuchi Chawla and Balasubramanian Sivan. Bayesian algorithmic mechanism design. *SIGecom Exchanges*, 13(1):5–49, 2014. 4

[39] Shuchi Chawla, Jason D. Hartline, and Robert D. Kleinberg. Algorithmic pricing via virtual valuations. In *Proceedings of the 8th ACM Conference on Electronic Commerce (EC)*, pages 243–251, 2007. 36, 95

[40] Shuchi Chawla, Jason D. Hartline, David L. Malec, and Balasubramanian Sivan. Multi-parameter mechanism design and sequential posted pricing. In *Proceedings of the 42nd ACM Symposium on Theory of Computing (STOC)*, pages 311–320, 2010. 94

[41] Peter C. Cheeseman, Bob Kanefsky, and William M. Taylor. Where the really hard problems are. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 331–340, 1991. 66

[42] Xi Chen, Xiaotie Deng, and Shang-Hua Teng. Settling the complexity of computing two-player nash equilibria. *Journal of the ACM*, 56(3):14:1–14:57, 2009. 7

[43] Steve Chien and Alistair Sinclair. Convergence to approximate nash equilibria in congestion games. *Games and Economic Behavior*, 71(2):315–327, 2011. 8, 43

[44] George Christodoulou, Vahab S. Mirrokni, and Anastasios Sidiropoulos. Convergence and approximation in potential games. *Theoretical Computer Science*, 438:13–27, 2012. 43

[45] George Christodoulou, Alkmini Sgouritsa, and Bo Tang. On the efficiency of the proportional allocation mechanism for divisible resources. *Theory of Computing Systems*, 59(4):600–618, 2016. 97, 117

[46] Vasek Chvátal and Bruce A. Reed. Mick gets some (the odds are on his side). In *Proceedings of 33rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 620–627, 1992. 22, 66

[47] Vašek Chvátal and Endre Szemerédi. Many hard examples for resolution. *Journal of the ACM*, 35(4):759–768, 1988. 68

[48] Edmund M. Clarke, Armin Biere, Richard Raimi, and Yunshan Zhu. Bounded model checking using satisfiability solving. *Formal Methods in System Design*, 19(1):7–34, 2001. 19

[49] José Coelho and Mario Vanhoucke. Multi-mode resource-constrained project scheduling using RCPSP and SAT solvers. *European Journal of Operational Research*, 213(1):73–82, 2011. 19

[50] Amin Coja-Oghlan. The asymptotic k-sat threshold. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing (STOC)*, pages 804–813, 2014. 21, 23, 66

[51] Stephen A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the 3rd Annual ACM Symposium on Theory of Computing (STOC)*, pages 151–158, 1971. 19

[52] James M. Crawford and Larry D. Auton. Experimental results on the crossover point in random 3-sat. *Artificial Intelligence*, 81(1-2):31–57, 1996. 21

[53] Marek Cygan, Jesper Nederlof, Marcin Pilipczuk, Michal Pilipczuk, Johan M. M. van Rooij, and Jakub Onufry Wojtaszczyk. Solving connectivity problems parameterized by treewidth in single exponential time. *ACM Transactions on Algorithms*, 18(2):17:1–17:31, 2022. 66

[54] Constantinos Daskalakis, Paul W. Goldberg, and Christos H. Papadimitriou. The complexity of computing a nash equilibrium. *SIAM Journal on Computing*, 39(1):195–259, 2009. 7

[55] Théo Delemazure, Tom Demeulemeester, Manuel Eberl, Jonas Israel, and Patrick Lederer. Strategyproofness and proportionality in party-approval multiwinner elections. In *Proceedings of 27th AAAI Conference on Artificial Intelligence (AAAI)*, pages 5591–5599, 2023. 19

[56] Josep Díaz, Lefteris M. Kirousis, Dieter Mitsche, and Xavier Pérez-Giménez. On the satisfiability threshold of formulas with three literals per clause. *Theoretical Computer Science*, 410(30-32):2920–2934, 2009. 22, 66

[57] Jian Ding, Allan Sly, and Nike Sun. Proof of the satisfiability conjecture for large k. *Annals of Mathematics*, 196(1):1–388. 21, 23, 66

[58] Olivier Dubois and Yacine Boufkhad. A general upper bound for the satisfiability threshold of random r-SAT formulae. *Journal of Algorithms*, 24(2): 395–420, 1997. 22

[59] Olivier Dubois, Yacine Boufkhad, and Jacques Mandler. Typical random 3-SAT formulae and the satisfiability threshold. In *Proceedings of the 11th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 126–127, 2000. 22, 66

[60] Shaddin Dughmi, Yusuf Hakan Kalayci, and Neel Patel. Limitations of stochastic selection problems with pairwise independent priors. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing (STOC)*, pages 479–490, 2024. 68

[61] David Ellis and Nathan Linial. On regular hypergraphs of high girth. *The Electronic Journal of Combinatorics*, 21(1):1, 2014. 25, 91

[62] Robert Elsässer and Tobias Tscheuschner. Settling the complexity of local max-cut (almost) completely. In *Proceedings of the 38th International Colloquium on Automata, Languages and Programming (ICALP)*, pages 171–182, 2011. 18, 63

[63] Alex Fabrikant, Christos H. Papadimitriou, and Kunal Talwar. The complexity of pure nash equilibria. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing (STOC)*, pages 604–612, 2004. 8, 11, 42

[64] Uriel Feige. Relations between average case complexity and approximation complexity. In *Proceedings of the Thiry-Fourth Annual ACM Symposium on Theory of Computing*, page 534–543, 2002. 70

[65] Uriel Feige and Michel X. Goemans. Aproximating the value of two prover proof systems, with applications to MAX 2sat and MAX DICUT. In *Proceedings of the 3rd Israel Symposium on Theory of Computing and Systems (ISTCS)*, pages 182–189, 1995. 3, 8, 17, 42, 44

[66] Uriel Feige and Eran Ofek. Easily refutable subformulas of large random 3cnf formulas. *Theory of Computing*, 3(1):25–43, 2007. 68, 69, 70, 71

[67] Matthias Feldotto, Martin Gairing, and Alexander Skopalik. Bounding the potential function in congestion games and approximate pure nash equilibria. In *Proceedings of the 10th International Conference on Web and Internet Economics (WINE)*, pages 30–43, 2014. 8, 43

[68] Yiding Feng, Jason D. Hartline, and Yingkai Li. Optimal auctions vs. anonymous pricing: Beyond linear utility. In *Proceedings of the 20th ACM Conference on Economics and Computation (EC)*, pages 885–886, 2019. 97

[69] Yiding Feng, Jason D. Hartline, and Yingkai Li. Simple mechanisms for non-linear agents. In *Proceedings of the 34th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 3802–3816, 2023. 97

[70] Abraham Flaxman. Random planted 3-sat. In *Encyclopedia of Algorithms*, pages 1728–1732. Springer New York, New York, NY, 2016. ISBN 978-1-4939-2864-4. 68

[71] Dimitris Fotakis, Spyros C. Kontogiannis, and Paul G. Spirakis. Selfish unsplittable flows. *Theoretical Computer Science*, 348(2-3):226–239, 2005. 8

[72] John Franco and Marvin C. Paull. Probabilistic analysis of the davis putnam procedure for solving the satisfiability problem. *Discrete Applied Mathematics*, 5(1):77–87, 1983. 22, 66

[73] Joel Friedman, Andreas Goerdt, and Michael Krivelevich. Recognizing more unsatisfiable random k-SAT instances efficiently. *SIAM Journal on Computing*, 35(2):408–430, 2005. 68

[74] Tobias Friedrich and Ralf Rothenberger. Sharpness of the satisfiability threshold for non-uniform random k-SAT. In *Proceedings of 21st International Conference on Theory and Applications of Satisfiability Testing (SAT)*, pages 273–291, 2018. 20, 66

[75] Tobias Friedrich and Ralf Rothenberger. The satisfiability threshold for non-uniform random 2-SAT. In *Proceedings of 46th International Colloquium on Automata, Languages, and Programming, (ICALP)*, pages 61:1–61:14, 2019. 66

[76] Tobias Friedrich, Anton Krohmer, Ralf Rothenberger, Thomas Sauerwald, and Andrew M. Sutton. Bounds on the satisfiability threshold for power law distributed random SAT. In *Proceedings of 25th Annual European Symposium on Algorithms (ESA)*, pages 37:1–37:15, 2017. 20, 66

[77] Alan M. Frieze and Stephen Suen. Analysis of two simple heuristics on a random instance of k-SAT. *Journal of Algorithms*, 20(2):312–355, 1996. 22, 66

[78] Martin Gairing, Kostas Kollias, and Grammateia Kotsialou. Existence and efficiency of equilibria for cost-sharing in generalized weighted congestion games. *ACM Transactions on Economics and Computation*, 8(2):11:1–11:28, 2020. 8, 13, 43

[79] Ian P. Gent and Toby Walsh. The SAT phase transition. In *Proceedings of the 11th European Conference on Artificial Intelligence (ECAI)*, pages 105–109, 1994. 21

[80] Yiannis Giannakopoulos, Georgy Noarov, and Andreas S. Schulz. Computing approximate equilibria in weighted congestion games via best-responses. *Mathematics of Operations Research*, 47(1):643–664, 2022. 13, 43

[81] Julian H Gibbs and Edmund A DiMarzio. Nature of the glass transition and the glassy state. *The Journal of Chemical Physics*, 28(3):373–383, 1958. 20

[82] Michel X. Goemans and David P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM*, 42(6):1115–1145, 1995. 3, 8, 17, 40, 41, 43, 44

[83] Michel X. Goemans, Li (Erran) Li, Vahab S. Mirrokni, and Marina Thottan. Market sharing games applied to content distribution in ad hoc networks. *IEEE Journal of Selected Areas in Communications*, 24(5):1020–1033, 2006. 42

[84] Stuart Alexander Grant. *Phase transition behaviour in constraint satisfaction problems*. PhD thesis, University of Leeds, UK, 1997. 21

[85] Nick Gravin and Zhiqi Wang. On robustness to k-wise independence of optimal bayesian mechanisms. *CoRR*, abs/2409.08547, 2024. 68

[86] Aarti Gupta, Malay K. Ganai, and Chao Wang. Sat-based verification methods and applications in hardware verification. In *Formal Methods for Hardware Verification, Proceeding of the 6th International School on Formal Methods for the Design of Computer, Communication, and Software Systems (SFM)*, 2006. 19

[87] Anupam Gupta, Jinqiao Hu, Gregory Kehne, and Roie Levin. Pairwise-independent contention resolution. In *Proceddings of 25th International Conference on Integer Programming and Combinatorial Optimization (IPCO)*, pages 196–209, 2024. 68

[88] Mohammad Taghi Hajiaghayi and Gregory B Sorkin. The satisfiability threshold of random 3-SAT is at least 3.52. *arXiv:math*, abs/0310193, 2003. 22, 66

[89] Jason D. Hartline and Brendan Lucier. Bayesian algorithmic mechanism design. In *Proceedings of the 42nd ACM Symposium on Theory of Computing (STOC)*, pages 301–310, 2010. 4

[90] Jason D. Hartline and Tim Roughgarden. Simple versus optimal mechanisms. In *Proceedings 10th ACM Conference on Electronic Commerce (EC)*, pages 225–234, 2009. 4, 28, 96

[91] Jason D. Hartline, Darrell Hoy, and Sam Taggart. Price of anarchy for auction revenue. In *Proceedings of the 15th ACM Conference on Economics and Computation (EC)*, pages 693–710, 2014. 117

[92] Pavel Hubácek and Eylon Yogev. Hardness of continuous local search: Query complexity and cryptographic lower bounds. *SIAM Journal on Computing*, 49 (6):1128–1172, 2020. 12

[93] Russell Impagliazzo and Ramamohan Paturi. On the complexity of k-sat. *Journal of Computer and System Sciences*, 62(2):367–375, 2001. 19, 66

[94] Russell Impagliazzo and Avi Wigderson. *P = BPP* if *E* requires exponential circuits: Derandomizing the XOR lemma. In *Proceedings of the 29th Annual ACM Symposium on Theory of Computing (STOC)*, pages 220–229, 1997. 3

[95] Russell Impagliazzo and Avi Wigderson. Randomness vs time: Derandomization under a uniform assumption. *Journal of Computer and System Sciences*, 63(4):672–688, 2001. 3

[96] Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *Journal of Computer and System Sciences*, 63(4):512–530, 2001. 66

[97] Svante Janson, Yannis C. Stamatiou, and Malvina Vamvakari. Bounding the unsatisfiability threshold of random 3-SAT. *Random Structures and Algorithms*, 17(2):103–116, 2000. 22

[98] Yaonan Jin, Pinyan Lu, Qi Qi, Zhihao Gavin Tang, and Tao Xiao. Tight approximation ratio of anonymous pricing. In *Proceedings of the 51st Annual ACM Symposium on Theory of Computing (STOC)*, pages 674–685, 2019. 28, 30, 31, 94, 96, 99

[99] Yaonan Jin, Pinyan Lu, Zhihao Gavin Tang, and Tao Xiao. Tight revenue gaps among simple mechanisms. *SIAM Journal on Computing*, 49(5):927–958, 2020. 30, 96

[100] Yaonan Jin, Shunhua Jiang, Pinyan Lu, and Hengjie Zhang. Tight revenue gaps among multiunit mechanisms. *SIAM Journal on Computing*, 51(5):1535–1579, 2022. 28, 30, 35, 96, 97

[101] Ramesh Johari and John N. Tsitsiklis. Efficiency loss in a network resource allocation game. *Mathematics of Operations Research*, 29(3):407–435, 2004. 97, 117

[102] 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. 11, 42

[103] Anil Kamath, Rajeev Motwani, Krishna V. Palem, and Paul G. Spirakis. Tail bounds for occupancy and the satisfiability threshold conjecture. *Random Structures and Algorithms*, 7(1):59–80, 1995. 22, 66

[104] Alexis C. Kaporis, Lefteris M. Kirousis, and Efthimios G. Lalas. The probabilistic analysis of a greedy satisfiability algorithm. *Random Structures and Algorithms*, 28(4):444–480, 2006. ISSN 1042-9832. 22, 66

[105] Frank Kelly. Charging and rate control for elastic traffic. *European Transactions on Telecommunications*, 8(1):33–37, 1997. 97, 117

[106] Lefteris M. Kirousis, Evangelos Kranakis, Danny Krizanc, and Yannis C. Stamatiou. Approximating the unsatisfiability threshold of random formulas. *Random Structures and Algorithms*, 12(3):253–269, 1998. 21, 22

[107] Elias Koutsoupias and Christos H. Papadimitriou. Worst-case equilibria. *Computer Science Review*, 3(2):65–69, 2009. 42

[108] Omer Levy, Rann Smorodinsky, and Moshe Tennenholtz. Undivide and conquer: On selling a divisible and homogeneous good. *The B.E. Journal of Theoretical Economics*, 15(1):1–23, 2015. 97

[109] Michael Lewin, Dror Livnat, and Uri Zwick. Improved rounding techniques for the MAX 2-sat and MAX DI-CUT problems. In *Proceedings of the 9th International Conference on Integer Programming and Combinatorial Optimization (IPCO)*, pages 67–82, 2002. 3, 18, 42, 44, 63

[110] Richard J. Lipton, Evangelos Markakis, and Aranyak Mehta. Playing large games using simple strategies. In *Proceedings of the 4th ACM Conference on Electronic Commerce (EC)*, pages 36–41, 2003. 7

[111] Brendan Lucier, Renato Paes Leme, and Éva Tardos. On revenue in the generalized second price auction. In *Proceedings of the 21st World Wide Web Conference (WWW)*, pages 361–370, 2012. 117

[112] A. El Maftouhi and Wenceslas Fernandez de la Vega. On random 3-SAT. *Combinatorics, Probability and Computing*, 4:189–195, 1995. 22, 23

[113] Sanjeev Mahajan and H. Ramesh. Derandomizing approximation algorithms based on semidefinite programming. *SIAM Journal on Computing*, 28(5): 1641–1663, 1999. 3

[114] Nimrod Megiddo and Christos H. Papadimitriou. On total functions, existence theorems and computational complexity. *Theoretical Computer Science*, 81(2): 317–324, 1991. 12, 21

[115] M. Mezard, G. Parisi, and R. Zecchina. Analytic and algorithmic solution of random satisfiability problems. *Science*, 297(5582):812–815, 2002. 21, 66

[116] Wil Michiels, Emile HL Aarts, and Jan Korst. *Theoretical aspects of local search*. 2007. 11

[117] David G. Mitchell, Bart Selman, and Hector J. Levesque. Hard and easy distributions of SAT problems. In *Proceedings of the 10th National Conference on Artificial Intelligence (AAAI)*, pages 459–465, 1992. 20, 21, 66

[118] Dov Monderer and Lloyd S Shapley. Potential games. *Games and economic behavior*, 14(1):124–143, 1996. 7, 10, 13, 42

[119] Roger B. Myerson. Optimal auction design. *Mathematics of Operations Research*, 6(1):58–73, 1981. 4, 27, 94, 98

[120] John Nash. Non-cooperative games. *Annals of Mathematics*, 54(2), 1951. 7

[121] Noam Nisan and Amir Ronen. Algorithmic mechanism design. *Games and Economic Behavior*, 35(1-2):166–196, 2001. 42

[122] Noam Nisan and Avi Wigderson. Hardness vs randomness. *Journal of Computer and System Sciences*, 49(2):149–167, 1994. 3

[123] Oleksii Omelchenko and Andrei A. Bulatov. Satisfiability threshold for power law random 2-sat in configuration model. In *Proceedings of 22nd International Conference on Theory and Applications of Satisfiability Testing (SAT)*, pages 53–70, 2019. 20, 66

[124] James B. Orlin, Abraham P. Punnen, and Andreas S. Schulz. Approximate local search in combinatorial optimization. *SIAM Journal on Computing*, 33 (5):1201–1214, 2004. 40

[125] Thomas R. Palfrey. Bundling decisions by a multiproduct monopolist with incomplete information. *Econometrica*, 51(2):463–483, 1983. 97

[126] 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. 12

[127] Christos H. Papadimitriou Paul W. Goldberg. Towards a unified complexity theory of total functions. In *Proceedings of the 9th Innovations in Theoretical Computer Science Conference (ITCS)*, pages 37:1–37:20, 2018. 12

[128] Aleksandar Pekec and Michael H. Rothkopf. Combinatorial auction design. *Management Science*, 49(11):1485–1503, 2003. 27

[129] Dominik Peters. Proportionality and strategyproofness in multiwinner elections. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, pages 1549–1557, 2018. 19

[130] Robert W. Rosenthal. A class of games possessing pure-strategy Nash equilibria. *International Journal of Game Theory*, 2:65–67, 1973. 7, 10, 42

[131] Tim Roughgarden and Éva Tardos. How bad is selfish routing? *Journal of the ACM*, 49(2):236–259, 2002. 42

[132] Ronitt Rubinfeld and Ning Xie. Testing non-uniform k-wise independent distributions over product spaces. In *Proceedings of 37th International Colloquium on Automata, Languages, and Programming (ICALP)*, pages 565–581, 2010. 68

[133] Aviad Rubinstein. Settling the complexity of computing approximate two-player nash equilibria. *SIGecom Exchanges*, 15(2):45–49, 2017. 8

[134] Aviad Rubinstein. Inapproximability of nash equilibrium. *SIAM Journal on Computing*, 47(3):917–959, 2018. 7

[135] Alejandro A. Schäffer and Mihalis Yannakakis. Simple local search problems that are hard to solve. *SIAM Journal on Computing.*, 20(1):56–87, 1991. 8, 13, 40, 42

[136] Alexander Skopalik and Berthold Vöcking. Inapproximability of pure nash equilibria. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing (STOC)*, pages 355–364, 2008. 8

[137] Vladimir Smirnov and Andrew Wait. Bundling in auctions with non-linear valuations. *Australian Economic Papers*, 50(1):17–26, 2011. 97, 98

[138] William Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of Finance*, 16(1):8–37, 1961. 93

[139] Vipin Ravindran Vijayalakshmi and Alexander Skopalik. Improving approximate pure nash equilibria in congestion games. In *Proceedings of the 16th International Conference on Web and Internet Economics (WINE)*, pages 280–294, 2020. 8, 43

[140] Robert Wilson. Auctions of shares. *The Quarterly Journal of Economics*, 93 (4):675–689, 1979. 97

[141] Uri Zwick. Analyzing the MAX 2-SAT and MAX DI-CUT approximation algorithms of Feige and Goemans. Manuscript, 2000. URL `http://www.cs.tau.ac.il/~zwick/online-papers.html`. 44