מכון ויצמן למדע
WEIZMANN INSTITUTE OF SCIENCE

Thesis for the degree
Doctor of Philosophy

עבודת גמר (תזה) לתואר
דוקטור לפילוסופיה

Submitted to the Scientific Council of the
Weizmann Institute of Science
Rehovot, Israel

מוגשת למועצה המדעית של
מכון ויצמן למדע
רחובות, ישראל

By
Noam Livne

מאת
נעם ליבנה

מסיבוכיות חישובית לקריפטוגרפיה ולתורת המשחקים

From Computational Complexity to Cryptography and to Game Theory

Advisorors:
Prof. Oded Goldreich
Dr. Alon Rosen

מנחים:
פרופ' עודד גולדרייך
ד"ר אלון רוזן

August 10, 2010

ל' אב תש"ע

FROM COMPUTATIONAL COMPLEXITY TO CRYPTOGRAPHY AND TO GAME THEORY

NOAM LIVNE

To my parents

Acknowledgments

First and foremost I would like to thank my primary advisor, Oded Goldreich. Oded encouraged me in any direction I chose to pursue, and supplied me with his bright observations and extremely broad knowledge. He also taught me a great deal of writing clear and concise presentations of mathematical ideas. Oded is a real example of sharp and deep thought, and amazingly enough, this does not come on the expense of swiftness. His ability to process new ideas I present to him, sometimes reacting before I finish presenting them, continues to amaze me. Oded is also a rare example of modesty, simply refusing to join as an author to papers, in cases most others would, claiming that "all the ideas are yours". But above all this I thank Oded for his friendship. Oded has the rare ability to understand not only the research, but also the researcher. Our conversations on research and on life in general are invaluable. I feel extremely lucky having had Oded as my advisor.

Second, I would like to thank Alon Rosen, who joined as a second advisor during the last period of my Ph.D. Collaborating with Alon is a great pleasure. His wide view of the subjects he studies together with his openness to new ideas and directions makes the collaboration with him really stimulating. Alon has the ability to bring the best out of every person, both personally and professionally, and I still haven't figured out how he does it. It is some kind of mixture of slight harshness and generosity. His slightly sarcastic sense of humor only adds fun to the whole process.

I would also like to thank my other collaborators in the works presented in this dissertation, besides Alon Rosen: Amos Beimel, Ronen Gradwohl and Carles Padró.

I also take this opportunity to thank the wonderful people in the Weizmann Institute of Science, and in particular in the Feinberg Graduate School and the Department of Applied Mathematics and Computer Science. I feel really privileged having had the opportunity to study in this wonderful place.

I would like to end by saying that I am hoping for the day when all people between the Mediterranean sea and the Jordan river enjoy equal opportunities to pursue their intellectual curiousness and fulfill their desires.

Summary

The works in this thesis are spanned over the historical development in the approach to cryptography: from information-theoretic cryptography, to computational cryptography, and up to the new approach of rational cryptography.

In the first chapter we prove lower bounds on the size of shares in perfect secret sharing schemes. We do so for matroidial access structures, using non-Shannon inequalities, by this circumventing impossibility results regarding Shannon-inequalities. Our work sheds some light on the problem of proving stronger lower bounds for perfect secret sharing schemes, which is open for a long time.

In the second chapter we contribute to the theory of average-case complexity initiated by Levin in 1984, by proving that all natural $\mathcal{NP}$-complete decision problems have average-case complete versions. While this widens the class of average-case complete distributional problems, the resulted problems do not seem as natural as one might have expected.

In the third chapter we consider the possibility of basing one-way functions on average-case hardness. We consider constructing samplers that yield one-way functions from samplers that output hard-on-average instances. We show limitations on this approach.

Finally, in Chapter 4 we propose a new solution concept for the modeling of cryptographic protocols as extensive games. Our solution concept is the first that is both sequential and computational. As an application of our new definition we revisit the problem of removing the mediator from a correlated equilibrium. We show a new protocol that works for a non-trivial class of correlated equilibria, that achieves sequential rationality.

Contents

Introduction

Since this thesis is in the published papers format, each chapter contains a detailed introduction on the subject it handles. In this introduction we will thus concentrate on framing the different and diverse results in a wider context.

In 1965 Hartmanis and Stearns published their paper On the Computational Complexity of Algorithms [3], which initiated the study of computational complexity. While during the 1940's and 1950's a lot of research was carried on the question of "what can be computed", they showed that even when problems can be solved by computers, they can be arbitrarily hard to compute. While this could look as nothing but "bad news" at first sight, in fact the concept of computational complexity gave birth to a profound revolution in the field of cryptography, starting in the mid 1970's. Until then, cryptography was regarded mainly as establishing means of secretely communicating in the presence of eavesdroppers. A basic paradigm that governed all endeavors towards that end, was that the two communicating parties need to agree on a secret "key", that is, a secret information known only to them. They can then use this key to manipulate the information they exchange, and any adversary attempting to reveal the exchanged information will fail as he lacks the information needed to reverse that manipulation. Thus, this paradigm is based on a *gap in the information* between the two communicating parties and any potential eavesdropper.

However, the aforementioned paradigm bears a few deep problems: the main one is that the parties need at some point in the time a safe channel to communicate (or, alternatively, to physically meet), in order to agree on that key; only then they can start communicating safely. These conditions may not always be met. Another problem this paradigm raises, is that in all encrypting systems based on it, the security of the system degrades as the amount of communication grows. Yet another problem, is that the parties need to keep the secret key safe - once it is revealed, they cannot longer communicate safely.

The new paradigm that initiated the aforementioned revolution, is that there does not necessarily need to be a gap in the information between the communicating parties and the eavesdropper. Even if in fact the manipulation can be reversed, that is, given the manipulated message one can compute the original message, as long as the *computational complexity* of this task is hard enough, the system can be regarded safe (for example, think of a situation where performing this computation is more expensive than any benefit could be gained by breaking the system). Thus, this new paradigm builds on the possibility of arguing about the computational complexity of computational tasks.

This new approach of considering not only "what can be computed", but rather "what can be practically computed", opens a whole new world of possibilities. First of all, it enables two parties to safely communicate without ever needing to agree on a secret key. This naturally solves all three problems mentioned above. But the impact of this paradigm goes way beyond safe communication. In the following years, and until today, the academic research of cryptography has provided cryptographic solutions for a vast variety of tasks, from relatively simple tasks like electronic signatures to extremely complicated tasks as electronic voting systems. At the heart of this research lies not only the construction of these systems, but also a very wide collective endeavor of formulating the right definitions to argue about the security of these systems. Today's cryptosystems do not rely on a vague intuition that "breaking the system is complicated", but rather meet very rigorous definitions of security, definitions that withstand constant examining, and that develop as new ideas are discovered, and new possible attacks on the systems need be considered.

One of the results of this research, however, is that certain tasks are proven to be impossible to achieve under these definitions, and others, while achievable in theory, are not adopted in practice as the known solutions require too much computing power from the participating parties, making them impractical. Starting in the early 2000's, a new approach was considered. Until then, as a result of the desire for best possible security, cryptographic definitions required resistance to any possible malicious behavior of the adversaries. In most cases, this required assuming that some fraction of the players, "the honest players", behave "nicely" unconditionally.

The new approach suggests looking at the parties neither "as malicious as possible" nor as "unconditionally honest", but rather as "rational". Rational parties simply strive to maximize their gain, regardless of the others (put aside altruism). This view naturally relates to the well-established field of Game Theory, which studies interaction between rational entities. Indeed, this line of research combines ideas from both cryptography and game theory, contributing each one by incorporating ideas from the other. This new approach enables providing cryptographic protocols for a few tasks otherwise provably impossible, and in other cases provides more practical solutions. Game theory benefits from this approach too. Traditionally, there are two classic ways to view game theory. One is the descriptive approach, which views game theory as describing the interactive behavior of various entities, and in particular human beings. If one thinks of game theory as modeling interaction between human beings, it is unjustified to assume they have unbounded computational power (even if they use computers, as reflected above). Thus, incorporating computational limitations into the strategic analysis of interaction is unavoidable. The other approach is the prescriptive approach. According to this approach, game theory can be used to prescribe behavior to entities, ensuring rational ones would follow it. Here too, the question of what prescription should and can be rationally followed cannot avoid computational limitations of the parties. Just like in cryptography, the computational limitations of the parties can be exploited to solve classic game-theoretic problems (see Chapter 4 for examples). The fruitfulness of this research direction is yet to be discovered, but considering the amount of research it produced in the last decade, it seems promising.

The results in this dissertation has a representative in each of the approaches overviewed above. Chapter 1 deals with lower bounds for perfect secret sharing schemes. Secret sharing schemes are a cryptographic tool for distributing the access to some information between sets of parties. In a perfect secret sharing scheme there is a finite set of parties, and a collection $\mathcal{A}$ of "authorized" subsets of the parties called the *access structure*. A secret-sharing scheme for $\mathcal{A}$ is a method by which a dealer that holds a secret (which we think of as a random variable) distributes "shares" (pieces of information) to the parties such that (1) any subset in $\mathcal{A}$ can reconstruct the secret from its shares, and (2) any subset not in $\mathcal{A}$ cannot reveal any partial information about the secret in the information-theoretic sense. That is, the collective information of any authorized set determines the secret, while the collective information of any unauthorized set does not reveal any information about the secret. Clearly, the access structure $\mathcal{A}$ must be monotone, that is, all supersets of a set in $\mathcal{A}$ are also in $\mathcal{A}$.

As opposed to all other works in this dissertation, this work does not involve any aspect of computation. In the special case of secret sharing schemes one can achieve "perfect security", that is, security that holds even against computationally "all-powerful" malicious parties. Thus, this cryptographic tool does in fact build on a gap in the information between honest and dishonest (sets of) parties. We mention however, that on the other hand this model also does not require that the reconstruction of the secret be computationally efficient (although in all known constructions it is in fact efficient. Moreover, typically, applications that use secret sharing schemes as a building block do require so). Because of this property of perfect secret sharing schemes, a fundamental tool in their analysis is Information Theory, which was introduced by Shannon in 1948 [6]. Since the secret is a random variable, applying on the secret the scheme by which the dealer distributes the shares to the parties, induces a set of random variables – one for the share of each party. Given a set of random variables that can be dependent in one another, information theory lets us argue about the amount of *entropy* in subsets of random variables, that is, the amount of "uncertainty" in these subsets. Moreover, using the tools of information theory we can argue about the *conditional entropy* of random variables, given other random variables. These arguments take the form of *information inequalities*. The basic inequalities that appeared in Shannon's paper are known as "Shannon inequalities".

One of the major areas of research in secret sharing schemes is the size of the shares (where by "size" we mean the number of bits used to represent them). It is easy to show that the size of the shares must be at least the size of the secret. Thus, we measure the ratio between the size of the shares and the size of the secret. An access structure that admits a secret sharing scheme with the optimal ratio (i.e., where the size of the shares is equal to the size of the secret), is called an *ideal* access structure[1].

Very little is known about the required size for general access structures. In [4] it is shown that every access structure has a perfect secret sharing scheme. However, their construction yields a ratio of exponential size in the number of parties. Other techniques are slightly better than that of [4], but all yield exponential ratio. On the other hand, the best lower bound, due to Csirmaz [1], shows that there exists a family of access structures for which every secret sharing scheme requires a ratio of $n/\log(n)$, where n is the number of parties. Thus, there is an exponential (in the number of parties) gap between the lower and upper bounds for the size of shares.

In his paper [1], Csirmaz showed another very interesting and beautiful result: he showed that Shannon inequalities cannot prove lower bound higher than n for the size of shares. However, in 1998 Zhang and Yeung [7] discovered new inequalities that cannot be deduced from the Shannon inequalities. These are the so-called *non-Shannon inequalities*. Other examples have been found subsequently. These inequalities gave hope that maybe one can circumvent Csirmaz's impossibility result by using these new inequalities.

In the work in chapter 1 we study the power of non-Shannon inequalities. While we still do not know how to circumvent Csirmaz's impossibility result and prove super-linear lower bounds for the aforementioned ratio, using non-Shannon inequalities we manage to circumvent another impossibility result, which we describe in the following.

An interesting class of access structures are those induced by *matroids*. A matroid is an axiomatic abstraction of the notion of linear independence. Matroids induce in a natural way access structures. We call an access structure induced by a matroid a *matroidial* access structure. It is known that all ideal access structures are matroidial. However, the opposite is not true. That is, there are matroidial access structures that are not ideal. Yet, using similar arguments to those in Csirmaz's impossibility result, one can show that for matroidial access structures too non-trivial lower-bounds cannot be proven by Shannon inequalities.

In the main result in our work we prove, using non-Shannon inequalities, non-trivial lower bounds for a matroidial access structure, thus circumventing an impossibility result regarding Shannon inequalities. This work demonstrates the usefulness of non-Shannon inequalities, and raises some hope that maybe in the future non-Shannon inequalities will enable proving stronger lower bounds on the size of shares in secret sharing schemes. Our results also solve

[1] We mention that we disregard some subtleties in this introduction. In particular, there may be access structures for which the ratio is optimal, yet they are not ideal. This is an artifact of the fact the ratio is defined as a supremum over all possible secret sharing schemes. See Chapter 1 for more details.

some combinatorial open questions regarding the size of shares in matroidial access structures.

In Chapter 2 we study *average case complexity*. In computational complexity, algorithms are most often measured by the running time they require, as a function of the input length. Traditional complexity theory studied the *worst-case* performance of algorithms. Worst-case complexity accounts an algorithm for the running time on the worst input of every length. In contrast, average-case complexity considers the *average* running time of the algorithm over all inputs of each length. The reasoning behind this approach is that sometimes we can settle for algorithms that under some distribution on the inputs, fail in rare cases, but otherwise perform well. Thus, this approach studies the complexity of computational problems under different distributions. In 1984 Levin [5] initiated the study of average-case complexity. In this paper he defined, in particular, the following notions:

- *distributional problem*: a distributional problem is a pair consisting of a decision problem and a distribution over the strings. (There are various ways to define distributions on strings, more details can be found in Chapter 2.)
- P-computable distributions: these are distributions on inputs that have strong computational structure: there exists a polynomial-time algorithm that can output for any two strings the probability that a string drawn according to this distribution falls within the interval defined by these two strings (in lexicographical order).
- $\mathrm{dist}\mathcal{NP}$: the class of all distributional problems consisting of an $\mathcal{NP}$-decision problem coupled with a P-computable distribution.
- $\mathrm{avg}\mathcal{P}$: the class of all distributional problems that can be solved "in polynomial-time on average". This is a very important contribution, as it turns out that the "naive" definition is inadequate.

He considered the class $\mathrm{dist}\mathcal{NP}$ to be the "average-case analogue" of $\mathcal{NP}$. (Restricting the class of distributions coupled with $\mathcal{NP}$ problems is essential for a sensible treatment. See Chapter 2 for more details.) He then defined a suitable class of reductions between distributional problems, which we call here AP-reductions, that "preserve easiness on average". That is, if A can be AP-reduced to B, then $B \in \mathrm{avg}\mathcal{P}$ implies that $A \in \mathrm{avg}\mathcal{P}$. He then proved his main result: that there exists a $\mathrm{dist}\mathcal{NP}$-complete distributional problem, that is, a problem in $\mathrm{dist}\mathcal{NP}$ that every problem in $\mathrm{dist}\mathcal{NP}$ can be AP-reduced to it. Thus, this complete problem is in $\mathrm{avg}\mathcal{P}$ if and only if $\mathrm{dist}\mathcal{NP} \subseteq \mathrm{avg}\mathcal{P}$. This result can be viewed as an average-case analogue of the famous Cook-Levin Theorem on the $\mathcal{NP}$-completeness of SAT. However, as not like in the Cook-Levin case, a major deficiency of Levin's theory was the lack of a wide variety of "natural" $\mathrm{dist}\mathcal{NP}$-complete problems. In Chapter 2 we show that every "natural" $\mathcal{NP}$-complete problem can be coupled with some P-computable distribution to form a

$\mathrm{dist}\mathcal{NP}$-complete problem. (The exact meaning of "natural" will be clarified in Chapter 2.) This substantially widens the class of average-case complete distributional problems. However, the resulted distributions on the problems do not seem as natural as one might have expected, which suggests that towards obtaining "natural" complete distributional problems one must relate to the specific structure of each problem.

In Chapter 3 we consider the challenge of utilizing computational hardness for cryptography. A necessary condition for nearly all cryptographic applications studied today (and in a lot of cases a sufficient condition too), is a computational object called *one-way function*. One-way functions are functions that can be efficiently computed (i.e, in polynomial time in their input-length), but cannot be efficiently inverted. However, proving the existence of one-way functions seems unreachable these days. The existence of one-way functions implies that $\mathcal{P} \neq \mathcal{NP}$, but solving the problem of $\mathcal{P}$ versus $\mathcal{NP}$, which is seemingly an easier task, is an open question for about half-century. Thus, a line of research studied the possibility of proving the existence of one-way functions *based* on the assumption that $\mathcal{P} \neq \mathcal{NP}$. More specifically, these works studied the existence of a (security) reduction from inverting a one-way function to solving an $\mathcal{NP}$-complete problem. Such reduction would prove that if $\mathcal{P} \neq \mathcal{NP}$ then the function is a one-way function. These works presented limitations on possible such reductions. Specifically, these works show that, under certain assumptions, certain reductions of this type do not exist. However, the assumption that $\mathcal{P} \neq \mathcal{NP}$ is an assumption about worst-case hardness. It assumes there are problems in $\mathcal{NP}$ that are hard in the worst-case, but not necessarily that there are problems that are hard on average. On the other hand, even the assumption that there are $\mathcal{NP}$ problems that are hard on average is not known to imply that one-way functions exist.

Thus, the starting point of the work presented in Chapter 3 is the observation that the reductions studied by these papers are actually supposed to overcome two gaps at once: (1) the gap between average-case hardness and worst-case hardness; and (2) the gap between one-way functions and average-case hardness. Since the problem of basing average-case hardness on worst-case hardness is hard by itself, this work suggest looking at an even "easier" challenge: the task of basing one-way functions on average-case hardness. One approach to solve this challenge follows the research line sketched above: to study reductions from inverting a one-way function to solving an $\mathcal{NP}$-complete problem *on average* under some distribution. Such reduction would show that if the $\mathcal{NP}$-complete problem is hard on average under this distribution then one-way functions exist. But, there is also another approach. The assumption that there are hard on average problems assumes there exists a search problem R and a sampler S such that the search problem R is hard on average under the distribution induced by S.

It is an easy exercise to show that the existence of one-way functions is equivalent to the following claim: there exists a search problem R and a polynomial-time sampler S^* that outputs instance-solution pairs of R, such that the distribution of S^* restricted to the instances is hard on average

Thus, assuming there is a hard on average problem, one can start from a search problem R and a sampler S where the search problem R is hard on average under the distribution induced by S, and “modify” S to construct a new sampler S^*, that “behaves” similarly to S, but instead of outputting only instances, S^* outputs instance-solution pairs (under R). Using the observation above, this will prove the existence of one-way functions. In fact, one can relax this construction and still get a one-way function. (For more details see Chapter 3.)

The result in this chapter shows a limitation on this approach too. Specifically, it shows that (under standard assumptions) there exists a (universal) sampler, such that for every polynomial λ there is a search problem R such that R is hard on-average under S, but where every sampler S^* as above (for R, S) has randomness complexity at least λ. In other words, the randomness complexity of S^*, who’s distribution on instances should be “similar” to that of S (namely, it should *dominate* that of S, see Chapter 3 for a definition), requires arbitrarily high randomness complexity (which depends on R). This result is somewhat surprising, as one could expect that the randomness complexity of S^* should depend only on S.

The last chapter in this dissertation deals with the intersection of the fields of Cryptography and Game Theory. If one assigns for each outcome of a cryptographic protocol a payoff for each party, it is natural to view the protocol as an *extensive game*. Informally, an extensive game is a game that proceeds in rounds, where in each round one player takes a move. As opposed to the traditional cryptographic security outlined above, where we require that the protocol be resistant to any possible deviation (or at least any such deviation of a “malicious party”), in a game theoretic analysis we require that the parties, or “players” in the game-theoretic terminology, be in *equilibrium*. This means that no player will have an incentive to deviate given the behavior of the others. That is, while it is possible for players to harm the others, it will not be in their interest to do so. Over the years many notions of equilibria, or *solution concepts*, were developed, for different types of games and different assumptions on the players. However, none of these solution concepts suits the cryptographic setting. The reason is that the vast majority of research in game theory considers the players to have no computational limitations. The relatively few studies on computationally bounded players (a subfield called *bounded rationality* in game theory) modeled them in an anachronistic and unrealistic manner, at least for our purposes (such as being represented as finite automata with a bounded number of states).

In the past years a few new solution concepts were suggested, to deal with modern notions of computational bounds. However, none of these considered the *sequential* nature of cryptographic protocols. Thus, we have today on one hand numerous solution concepts, that arise from traditional game theory, that deal with sequential games, where players play in rounds, but where they are computationally unbounded, and on the other hand we have a few solution concepts, that arise from the study of rational cryptography, for games with computationally bounded players, but that suit only *normal-form games*, that is, games that consist of players playing exactly one move, all simultaneously.

When applying solution concepts for normal-form games in extensive-form games, the fundamental problem is the existence of *empty threats*. Consider a traffic light in a junction with two cars coming from perpendicular directions, each wanting to enter the junction first. When we simulate this situation as a normal-form game, the only two equilibria consist of one player entering and one waiting. In this case no player would like to unilaterally deviate, as either he will crash the other, or he will loose his precedence. The other two possibilities are not in equilibrium: if both are waiting each would like to unilaterally enter, and if both are entering, each would like to avoid the crash and wait.

Now consider a slightly different game, where one player decides on his action, and the other, a split of a second later, after viewing the first player's action, decides on his action too. If we model this game as a normal-form game, we get the same set of equilibria. In particular, the first player waiting and the second entering, is an equilibrium. Indeed, if the first player believes the second will enter, he is better-off waiting. However, this equilibrium does not reflect our intuition about this game. If the first player enters the junction, it is fairly rational for him to assume the other will avoid entering too. Once the second player is facing the fact that the first already entered the junction, it is irrational for him to enter too. Thus, if the second player declares that he enters the junction first, it is an *empty threat*. There is no reason for the first player to believe this threat.

The traditional solution for this problem in game theory is the notion of sub-game perfect equilibrium (SPE), known also as sub-game perfect Nash equilibrium (SPNE). This solution concept requires that each player plays and declares to play optimally at every possible point in the game. Thus, for the two drivers from the aforementioned example, the only SPNE is when the first enters and the second waits.

However, when we try to apply this solution concept for computationally bounded players, we face a fundamental difficulty: we cannot require computationally bounded players to play optimally at every point in the game, as this might be computationally intractable. Moreover, as explained above, most cryptographic protocols actually build on the fact players cannot play optimally at every point, as this would mean breaking the protocol. Thus, although an SPNE might exist for protocols when modeled as games, the players typically will not be able to actually follow it.

We consider a few variants of SPNE and demonstrate why they all fail through simple examples. We then propose a new solution concept for 2-players cryptographic protocols. Our first observation is that one does not necessarily have to require optimal behavior at every point in the game in order to eliminate empty threats, as not every non-optimal behavior carries a threat. Moreover, as mentioned above, typically players are actually building on the others not playing optimally. We then proceed to define formally an empty threat, a notion that turns out to be quite elusive. The idea behind our definition is demonstrated informally as following. When am I threatened? I am threatened if I believe that if I will deviate from my strategy, and then the other will play rationally after facing my deviation, I will improve. But what does it mean for the other player to play rationally? He too will not believe to any empty threat of mine. Thus, the rational assumption for me is that after I deviate, the game will proceed with no empty threats. This leads to a regressive definition. Our new definition, called TFNE, for Threat-Free Nash Equilibrium, requires that the players be in equilibrium, and in addition that they will not impose empty threats on one another.

We then turn to define the computational variant of TFNE, called CTFNE, to deal with computationally bounded players. This raises a few new difficulties, and in particular calls for a set of new definitions to model the computational setting in a game-theoretic manner. We formalize in a game-theoretic terminology, in a very rigorous manner, the behavior of *interactive Turing machines* (ITMs), the objects representing players in cryptographic protocols. Since traditionally computational hardness is stated asymptotically, and correspondingly cryptographic protocols are parameterized by a security parameter, we define sequences of games with restricted strategies to model the computational inabilities of the players. Towards this end we define the notion of *strategy filters*, which state which ITMs are allowed to play in which games. We also modify the definition of TFNE to deal with a *slackness parameter*, reflecting the fact that in cryptographic protocols players might succeed in breaking the security with some small probability.

While we end up with quite an involved definition, we attempt to justify our choices, and disqualify a few alternatives. We also believe that the fact that no other definition was found until now, although this was an open question for quite a while, might suggest that a simpler definition might not exist.

To demonstrate the applicability of our definition, we revisit the problem of removing the mediator from a correlated equilibrium. In a correlated equilibrium (CE) a mediator privately recommends to each player how to play. The guarantee is that no player can gain by deviating given the recommendations to the other players. The advantage of correlated equilibrium is that the recommendations can depend on one another, that is, the resulted equilibrium is not necessarily a product-distribution. This often results in better payoffs for some or all players, and in other cases achieves payoffs that are more “fair” (for example, in the traffic light game above, both equilibria are “unfair”, as in each one of them, one waits and one enters. In a correlated equilibrium, each can enter with probability $1/2$, without the two ever entering together).

While there exist game-theoretic solutions for games with more than two players, the case of two players is provably unsolvable in the game-theoretic setting (where players are computationally unbounded). In 2000, Dodis, Halevy and Rabin [2] showed a very nice cryptographic solution for the case of two players. However, the solution concept they used is

Nash equilibrium, a solution concept that does not consider the sequential form of their solution, resulting in a solution that cannot be rationally justified. We show a modification of their protocol, for a non-trivial subset of CE, and prove it is in CTFNE.

Bibliography

[1] L. Csirmaz. The size of a share must be large. *J. of Cryptology*, 10(4):223–231, 1997.

[2] Y. Dodis, S. Halevi, and T. Rabin. A cryptographic solution to a game theoretic problem. In *In Advances in Cryptology Crypto*, pages 11–15, 2000.

[3] J. Hartmanis and R.E. Stearns. On the computational complexity of algorithms. *Transactions of the AMS*, 117:285–306, 1965.

[4] M. Ito, A. Saito, and T. Nishizeki. Secret sharing schemes realizing general access structure. In *Proc. of the IEEE Global Telecommunication Conf., Globecom 87*, pages 99–102, 1987. Journal version: Multiple assignment scheme for sharing secret. *J. of Cryptology*, 6(1):15-20, 1993.

[5] Leonid A Levin. Average case complete problems. *SIAM J. Comput.*, 15(1):285–286, 1986.

[6] C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423,623–656, July, October 1948.

[7] Z. Zhang and R. W. Yeung. On characterization of entropy function via information inequalities. *IEEE Trans. on Information Theory*, 44(4):1440–1452, 1998.

Chapter 1

Matroids Can be Far From Ideal Secret Sharing[1]

Amos Beimel and Noam Livne and Carles Padró

Abstract

In a secret-sharing scheme, a secret value is distributed among a set of parties by giving each party a share. The requirement is that only predefined subsets of parties can recover the secret from their shares. The family of the predefined authorized subsets is called the access structure. An access structure is ideal if there exists a secret-sharing scheme realizing it in which the shares have optimal length, that is, in which the shares are taken from the same domain as the secrets. Brickell and Davenport (J. of Cryptology, 1991) proved that ideal access structures are induced by matroids. Subsequently, ideal access structures and access structures induced by matroids have received a lot of attention. Seymour (J. of Combinatorial Theory, 1992) gave the first example of an access structure induced by a matroid, namely the Vamos matroid, that is non-ideal. Beimel and Livne (TCC 2006) presented the first non-trivial lower bounds on the size of the domain of the shares for secret-sharing schemes realizing an access structure induced by the Vamos matroid.

In this work, we substantially improve those bounds by proving that the size of the domain of the shares in every secret-sharing scheme for those access structures is at least $k^{1.1}$, where k is the size of the domain of the secrets (compared to $k+\Omega(\sqrt{k})$ in previous works). Our bounds are obtained by using non-Shannon inequalities for the entropy function. The importance of our results are: (1) we present the first proof that there exists an access structure induced by a matroid which is not nearly ideal, and (2) we present the first proof that there is an access

[1] Appeared in *Theory of Cryptography Conference (TCC) 2008.*

structure whose information rate is strictly between $2/3$ and 1. In addition, we present a better lower bound that applies only to *linear* secret-sharing schemes realizing the access structures induced by the Vamos matroid.

1.1 Introduction

1.1.1 Ideal Secret-Sharing Schemes and Matroids

Secret-sharing schemes, which were introduced by Shamir [31] and Blakley [5] in 1979, are nowadays used in many cryptographic protocols. In these schemes there is a finite set of parties, and a collection $\mathcal{A}$ of subsets of the parties (called the access structure). A secret-sharing scheme for $\mathcal{A}$ is a method by which a dealer distributes shares of a secret value to the parties such that (1) any subset in $\mathcal{A}$ can reconstruct the secret from its shares, and (2) any subset not in $\mathcal{A}$ cannot reveal any partial information about the secret in the information-theoretic sense. Clearly, the access structure $\mathcal{A}$ must be monotone, that is, all supersets of a set in $\mathcal{A}$ are also in $\mathcal{A}$.

Ito, Saito, and Nishizeki [18] proved that there exists a secret-sharing scheme for every monotone access structure. Their proof is constructive, but the obtained schemes are very inefficient: the ratio between the length in bits of the shares and that of the secret is exponential in the number of parties. Nevertheless, some access structures admit secret-sharing schemes with much shorter shares. A secret-sharing scheme is called *ideal* if the shares of every participant are taken from the same domain as the secret. As proved in [20], this is the optimal size for the domain of the shares. The access structures which can be realized by ideal secret-sharing schemes are called *ideal access structures*.

The exact characterization of ideal access structures is a longstanding open problem, which has interesting connections to combinatorics and information theory. The most important result towards giving such characterization is by Brickell and Davenport [8], who proved that every ideal access structure is induced by a matroid, providing a necessary condition for an access structure to be ideal. A sufficient condition is obtained as a consequence of the linear construction of ideal secret-sharing schemes due to Brickell [7]. Namely, an access structure is ideal if it is induced by a matroid that is representable over some finite field. However, there is a gap between the necessary condition and the sufficient condition. Seymour [30] proved that the access structures induced by the Vamos matroid are not ideal. Other examples of non-ideal access structures induced by matroids have been presented by Matúš [26]. Hence, the necessary condition above is not sufficient. Moreover, Simonis and Ashikmin [33] constructed ideal secret-sharing schemes for the access structures induced by the non-Pappus matroid, which is not representable over any field. This means that the sufficient condition is not necessary. Therefore, the study of the access structures that are induced by matroids is useful in the search of new results about the characterization of ideal access structures.

Another motivation in studying access structures induced by matroids arises from the separation result of Martí-Farré and Padró [24]. Namely, by using an old result by Seymour [29], they generalized the result by Brickell and Davenport [8], proving that in every secret-sharing scheme whose access structure is *not* induced by a matroid there is at least one participant whose domain of shares has size at least $k^{1.5}$, where k is the size of the domain of secrets. In other words, by proving that an access structure is not induced by a matroid, we prove a lower bound of $k^{1.5}$ for the size of the shares' domain. Therefore, the access structures that are not induced by matroids are clearly far from being ideal.

We rephrase the above result using the notion of information rate of [9]. The *information rate* of a secret-sharing scheme is $\log k / \log s$, where k is the size of the domain of the secrets and s is the maximum size of the domains of shares. That is, the information rate is the relation between the length in bits of the secret and the maximum length of the shares. Ideal secret-sharing schemes are those having information rate equal to 1. The information rate of an access structure $\mathcal{A}$ is the supermum of the information rates of all secret-sharing schemes realizing the access structure with a finite domain of shares. Stating the aforementioned result in the new notation, if $\mathcal{A}$ is not induced by a matroid, the information rate of every secret-sharing scheme for $\mathcal{A}$ is at most $2/3$, hence the information rate of $\mathcal{A}$ is at most $2/3$. This is not the case for the non-ideal access structures induced by matroids, which can be very close to ideal. An access structure $\mathcal{A}$ is *nearly ideal* if its information rate is 1. A non-ideal but nearly-ideal access structure is presented in [22, 27].

At this point, two natural open questions arise. First, which matroids induce ideal access structures? And second, what can be said about the optimal size of the shares' domain for access structures induced by matroids?

Even though several interesting results have been given in [33, 26, 27], the first question is far from being solved. Since an ideal secret-sharing scheme can be seen as a representation of the corresponding matroid, this question can be thought of as a representability problem. Very little is known about the second question. For instance, the only known non-trivial lower bound on the optimal size of the shares' domain for access structures induced by matroids has been presented by Beimel and Livne [2]. Specifically, for an access structure induced by the Vamos matroid, they prove a lower bound of $k + \Omega(\sqrt{k})$, where k is the size of the domain of the secrets.

The best constructions of secret-sharing realizing access structures induced by matroids are the constructions for general access structures, e.g., in [4, 32, 7, 19]; in these constructions most access structures induced by matroids require shares of exponential length. However, prior to this work, even the following question was open.

Question 1.1. *Does there exist a matroid such that its induced access structures are not nearly ideal?*

Observe that the lower bound given in [2] for an access structure induced by the Vamos matroid does not imply that it is not nearly ideal. For comparison, for general access structures the best known lower bound is given by Csirmaz [13] who proves that for every n there is an access structure $\mathcal{A}_n$ with n participants such that for every secret-sharing scheme realizing $\mathcal{A}_n$ there is at least one participant whose share has length at least $(n/\log n)\log k$.

Moreover, the following open problem, which was posed by Martí-Farré and Padró [23], was unsolved.

Question 1.2. *Does there exist an access structure whose optimal share size is* $\Theta(k^{\alpha})$ *for some constant* $1 < \alpha < 3/2$*?*

That is, Martí-Farré and Padró ask if there is an access structure whose information rate is strictly between $2/3$ and 1. As a consequence of the result of [24], if such an access structure exists, it must be induced by a matroid.

1.1.2 Our Results

In this paper we answer the above two questions about access structures induced by matroids. Specifically, we prove new lower bounds on the size of the domains of shares in secret-sharing schemes for the access structures induced by the Vamos matroid, substantially improving the bound given in [2]. The Vamos matroid induces two non-isomorphic access structures. We prove for them lower bounds on the size of the domains of shares of, respectively, $k^{10/9}$ and $k^{11/10}$, where k is the size of the domain of the secrets (compared to $k + \Omega(\sqrt{k})$ in [2]).

Therefore, we present here the first examples of access structures induced by matroids that are not nearly ideal, resolving Question 1.1. Moreover, we solve Question 1.2 in the affirmative: As a consequence of our lower bound and the upper bound of $k^{4/3}$ that was proved in [25], the access structures induced by the Vamos matroid are the required examples.

The interest of our result is increased by the use of the so called non-Shannon inequalities in our proof. By using the basic properties of the entropy function, namely, the so-called Shannon inequalities, Csirmaz [13] proved the best known lower bounds for secret-sharing schemes mentioned above. On the negative side, Csirmaz proved that using only Shannon inequalities one cannot improve his lower bounds by a factor larger than $\log n$. More relevant to this work, several bounds on the joint entropy of the shares of subsets of parties for access structures induced by matroids were proved in [2] using Shannon inequalities (see Theorem 1.4 and Theorem 1.5 in Section 1.2 below). However, these bounds are only on the joint entropy of the shares and the authors of [2] could not use them to prove lower bounds for access structures induced by matroids. This is not a coincidence as in [24] it is proved that it is not possible to obtain bounds for access structures induced by matroids by using only this technique (since the rank function of the matroid satisfies the Shannon inequalities).

Nevertheless, there exist several inequalities for the entropies of a set of random variables that cannot be deduced from the Shannon inequalities. These are the so-called non-Shannon inequalities. The first examples of such inequalities were given by Zhang and Yeung [36], and other examples have been found subsequently [15]. In this paper, we combine the entropy inequalities of [2] and the non-Shannon inequality of Zhang and Yeung [36] to obtain a simple and elegant proof of our result. The inequality of [36] was previously used related to the Vamos matroid in [16] for proving lower bounds for network coding and in [27] for proving that this matroid is not asymptotically entropic (the latter result gives an alternative proof that the access structures induced by the Vamos matroid are not ideal). We believe that non-Shannon inequalities will be used for proving new lower bounds for secret-sharing schemes, possibly improving the best known lower bound given by Csirmaz [13].

In addition, by applying a similar technique to the Ingleton's inequality [17, 28], which applies only to linear random variables, we obtain a lower bound of $k^{5/4}$ for the size of the shares' domains for *linear* secret-sharing schemes whose access structures are induced by the Vamos matroid.

1.2 Preliminaries

In this section we define secret-sharing schemes, review some background on matroids, and discuss the connection between secret-sharing schemes and matroids. The definition of secret-sharing presented in this paper uses the entropy function; in the appendix we review the relevant definitions from information theory.

1.2.1 Secret Sharing

Definition 1.2.1 (Access Structure)**.** *Let P be a finite set of parties. A collection $\mathcal{A} \subseteq 2^P$ is* monotone *if $B \in \mathcal{A}$ and $B \subseteq C$ imply that $C \in \mathcal{A}$. An* access structure *is a monotone collection $\mathcal{A} \subseteq 2^P$ of non-empty subsets of P. Sets in $\mathcal{A}$ are called* authorized*, and sets not in $\mathcal{A}$ are called* unauthorized*.*

Definition 1.2.2 (Distribution Scheme)**.** *Let $P = \{p_1, \ldots, p_n\}$ be a set of parties, and $p_0 \notin P$ be a special party called* the dealer*. An n-party* distribution scheme $\Sigma = \langle \Pi, \mu \rangle$ *with domain of secrets K is a pair where μ is a probability distribution on some finite set R (the set of random strings) and Π is a mapping from $K \times R$ to a set of n-tuples $K_1 \times K_2 \times \cdots \times K_n$, where K_i*

is called the share-domain *of p_i. A dealer distributes a secret $s \in K$ according to Σ by first sampling a string $r \in R$ according to μ, computing a vector of* shares $\Pi(s, r) = (s_1, \dots, s_n)$*, and then privately communicating each share s_i to the party p_i.*

We next give a definition of secret-sharing scheme using the entropy function. This definition is the same as that of [20, 10] and is equivalent to the definition of [11, 1, 3]. Before stating the definition, we present some notations. Let $\mathcal{A}$ be an access structure on the set of parties P. We defined a distribution scheme Σ as a probabilistic mapping that given a secret s generates a vector of shares. It will be convenient to view the secret as the share of the dealer, and for every $T \subseteq P \cup \{p_0\}$ to consider the vector of shares of T. Any probability distribution on the domain of secrets, together with the distribution scheme Σ, induces, for any $T \subseteq P \cup \{p_0\}$, a probability distribution on the vector of shares of the parties in T. We denote the random variable taking values according to this probability distribution on the vector of shares of T by S_T, and by S the random variable denoting the secret (i.e., $S = S_{\{p_0\}}$). Note that for disjoint subsets T_1, T_2, the random variable denoting the vector of shares of $T_1 \cup T_2$ can be written either as $S_{T_1 \cup T_2}$ or as $S_{T_1} S_{T_2}$. For a singleton $\{b\}$, we will write S_b instead of $S_{\{b\}}$.

Definition 1.2.3 (Secret-Sharing Scheme)**.** *We say that a distribution scheme is a secret-sharing scheme realizing an access structure $\mathcal{A}$ with respect to a given probability distribution on the secrets, denoted by a random variable S, if the following conditions hold.*

CORRECTNESS. *For every authorized set $T \in \mathcal{A}$, the shares of the parties in T determine the secret, that is,*

$$H(S|S_T) = 0. \tag{1.1}$$

PRIVACY. *For every unauthorized set $T \notin \mathcal{A}$, the shares of the parties in T do not disclose any information on the secret, that is,*

$$H(S|S_T) = H(S). \tag{1.2}$$

Remark 1.2.1. *Although the above definition considers a specific distribution on the secrets, Blundo et al. [6] proved that its correctness and privacy are actually independent of this distribution: If a scheme realizes an access structure with respect to one distribution on the secrets, then it realizes the access structure with respect to any distribution with the same support.*

Karnin et al. [20] have showed that the size of the domain of shares of each non-redundant party (that is, a party that appears in at least one minimal authorized set) is at least the size of the domain of secrets. This motivates the definition of ideal secret sharing.

Definition 1.2.4 (Ideal Secret-Sharing Scheme and Ideal Access Structure)**.** *A secret-sharing scheme with domain of secrets K is* ideal *if the domain of shares of each party is K. An access structure $\mathcal{A}$ is* ideal *if there exists an ideal secret-sharing scheme realizing it over some finite domain of secrets.*

1.2.2 Matroids

A matroid is an axiomatic abstraction of linear independence. There are several equivalent axiomatic systems to describe matroids: by independent sets, by bases, by the rank function, or, as done here, by circuits. For more background on matroid theory the reader is referred to [35, 28].

Definition 1.2.5 (Matroid)**.** *A matroid $\mathcal{M} = \langle V, \mathcal{C} \rangle$ is a finite set V and a collection $\mathcal{C}$ of subsets of V that satisfy the following three axioms:*

(C0) $\emptyset \notin \mathcal{C}$.

(C1) *If $X \neq Y$ and $X, Y \in \mathcal{C}$, then $X \nsubseteq Y$.*

(C2) *If C_1, C_2 are distinct members of $\mathcal{C}$ and $x \in C_1 \cap C_2$, then there exists $C_3 \in \mathcal{C}$ such that $C_3 \subseteq (C_1 \cup C_2) \setminus \{x\}$.*

The elements of V are called points*, or simply* elements*, and the subsets in $\mathcal{C}$ are called* circuits.

For example, let $G = (V, E)$ be an undirected simple graph and $\mathcal{C}$ be the collection of simple cycles in G. Then, $(E, \mathcal{C})$ is a matroid.

Definition 1.2.6 (Rank, Independent and Dependent Sets)**.** *A subset of V is* dependent *in a matroid $\mathcal{M}$ if it contains a circuit. If a subset is not dependent, it is* independent*. The* rank *of a subset $T \subseteq V$, denoted $\text{rank}(T)$, is the size of the largest independent subset of T.*

Definition 1.2.7 (Connected Matroid)**.** *A matroid is* connected *if for every pair of distinct elements x and y there is a circuit containing x and y.*

1.2.3 Matroids and Secret Sharing

In this section we describe the results relating ideal secret-sharing schemes and matroids. We first define access structures induced by matroids.

Definition 1.2.8. *Let $\mathcal{M} = \langle V, \mathcal{C} \rangle$ be a connected matroid and $p_0 \in V$. The* induced access structure of $\mathcal{M}$ with respect to p_0 *is the access structure $\mathcal{A}$ on $P = V \setminus \{p_0\}$ defined by*

$$\mathcal{A} \overset{\text{def}}{=} \{T : \text{ there exists } C_0 \in \mathcal{C} \text{ such that } p_0 \in C_0 \text{ and } C_0 \setminus \{p_0\} \subseteq T\}.$$

That is, a set T is a minimal authorized set of $\mathcal{A}$ if by adding p_0 to it, it becomes a circuit of $\mathcal{M}$. We think of p_0 as the dealer. We say that an access structure is induced *by $\mathcal{M}$, if it is obtained by setting some arbitrary element of $\mathcal{M}$ as the dealer. In this case, we say that $\mathcal{M}$ is the* appropriate matroid *of $\mathcal{A}$, and that $\mathcal{A}$ is* induced *by $\mathcal{M}$ with respect to p_0.*

Remark 1.2.2. *The term* the appropriate matroid *is justified, as if some access structure is induced by a matroid, this matroid is unique.*

The following fundamental result, proved by Brickell and Davenport [8], gives a necessary condition for an access structure to have an ideal secret-sharing scheme.

Theorem 1.3 ([8])**.** *If an access structure is ideal, then it has an appropriate matroid.*

The following result of [21] shows a connection between the rank function of the appropriate matroid and the joint entropy of the collections of shares.

Lemma 1.3.1 ([21])**.** *Assume that the access structure $\mathcal{A} \subseteq 2^P$ is ideal, and let $\langle P \cup \{p_0\}, \mathcal{C} \rangle$ be its appropriate matroid where $p_0 \notin P$. Let Σ be an ideal secret-sharing scheme realizing $\mathcal{A}$ where S is the random variable denoting the secret. Then $H(S_T) = \operatorname{rank}(T) \cdot H(S)$ for any $T \subseteq P \cup \{p_0\}$, where $\operatorname{rank}(T)$ is the rank of T in the matroid.*

Example 1.3.1. Consider the threshold access structure $\mathcal{A}_t$, which consists of all subsets of participants of size at least t, and Shamir's scheme [31] which is an ideal secret-sharing scheme realizing it. In this scheme, to share a secret s, the dealer randomly chooses a random polynomial $p(x)$ of degree $t-1$ such that $p(0) = s$, and the the share of the ith participant is $p(i)$. The appropriate matroid of $\mathcal{A}_t$ is the uniform matroid with $n+1$ points, whose circuits are the sets of size $t+1$ and $\operatorname{rank}(T) = \min\{|T|, t\}$. Since every t points determine a unique polynomial of degree $t-1$, in Shamir's scheme $H(S_T) = \min\{|T|, t\} H(S)$, as implied by Lemma 1.3.1.

We next quote results from [2] proving lower and upper bounds on the size of shares' domains of subsets of parties in matroid-induced access structures. These results generalize the results of [21] on ideal secret-sharing schemes to non-ideal secret-sharing schemes for matroid-induced access structures.

Theorem 1.4 ([2])**.** *Let $\mathcal{M} = \langle V, \mathcal{C} \rangle$ be a connected matroid where $|V| = n+1$, and $p_0 \in V$. Furthermore, let $\mathcal{A}$ be the induced access structure of $\mathcal{M}$ with respect to p_0, and let Σ be* any *secret-sharing scheme realizing $\mathcal{A}$. For every $T \subseteq V$,*

$$H(S_T) \geq \mathrm{rank}(T) \cdot H(S).$$

Theorem 1.5 ([2])**.** *Let $\mathcal{M} = \langle V, \mathcal{C} \rangle$ be a connected matroid where $|V| = n+1$, $p_0 \in V$ and let $\mathcal{A}$ be the induced access structure of $\mathcal{M}$ with respect to p_0. Furthermore, let Σ be* any *secret-sharing scheme realizing $\mathcal{A}$, and let $\lambda \geq 0$ be such that $H(S_v) \leq (1+\lambda)H(S)$ for every $v \in V \setminus \{p_0\}$. Then, for every $T \subseteq V$*

$$H(S_T) \leq \mathrm{rank}(T)(1+\lambda)H(S) + (|T| - \mathrm{rank}(T))\lambda n H(S). \tag{1.3}$$

1.2.4 The Vamos Matroid

In this paper we prove lower bounds on the size of shares in secret-sharing schemes realizing the access structures induced by the Vamos matroid. The Vamos matroid [34] is the smallest known matroid that is non-representable over any field, and is also non-algebraic (for more details on these notions see [35, 28]; we will not need these notions in this paper).

Definition 1.5.1 (The Vamos Matroid)**.** *The Vamos matroid $\mathcal{V}$ is defined on the set $V = \{v_1, v_2, \ldots, v_8\}$. Its independent sets are all the sets of cardinality ≤ 4 except for five: $\{v_1, v_2, v_3, v_4\}$, $\{v_1, v_2, v_5, v_6\}$, $\{v_3, v_4, v_5, v_6\}$, $\{v_3, v_4, v_7, v_8\}$, and $\{v_5, v_6, v_7, v_8\}$.*

Note that these 5 sets are all the unions of two pairs from $\{v_1, v_2\}$, $\{v_3, v_4\}$, $\{v_5, v_6\}$, and $\{v_7, v_8\}$, excluding $\{v_1, v_2, v_7, v_8\}$. The five sets listed in Definition 1.5.1 are circuits in $\mathcal{V}$ while the set $\{v_1, v_2, v_7, v_8\}$ is independent; these facts will be used later.

There are two non-isomorphic access structures induced by the Vamos matroid. First, the access structures obtained by setting v_1, v_2, v_7, or v_8 as the dealer are isomorphic. The other access structure is obtained by setting v_3, v_4, v_5, or v_6 as the dealer.

Definition 1.5.2 (The Access Structures $\mathcal{V}_6$ and $\mathcal{V}_8$)**.** *The access structure $\mathcal{V}_8$ is the access structure induced by the Vamos matroid with respect to v_8. That is, in this access structure the parties are $\{v_1, \ldots, v_7\}$ and a set of parties is a minimal authorized set if this set together with v_8 is a circuit in $\mathcal{V}$. The access structure $\mathcal{V}_6$ is the access structure induced by the Vamos matroid with respect to v_6. That is, in this access structure the parties are $\{v_1, \ldots, v_5, v_7, v_8\}$ and a set of parties is a minimal authorized set if this set together with v_6 is a circuit in $\mathcal{V}$.*

Example 1.5.1. We next give examples of authorized and non-authorized sets in $\mathcal{V}_6$.

1. The set $\{v_5, v_7, v_8\}$ is authorized, since $\{v_5, v_6, v_7, v_8\}$ is a circuit.

2. The circuit $\{v_1, v_2, v_3, v_4\}$ is unauthorized, since the set $\{v_1, v_2, v_3, v_4, v_6\}$ does not contain a circuit that contains v_6. To check this, we first note that this 5-set itself cannot be a circuit, since it contains the circuit $\{v_1, v_2, v_3, v_4\}$. Second, the only circuit it contains is $\{v_1, v_2, v_3, v_4\}$, which does not contain v_6.

3. The set $\{v_1, v_2, v_7, v_8\}$ is a minimal authorized set, since $\{v_1, v_2, v_6, v_7, v_8\}$ is a circuit (as it is dependent, and no circuit of size 4 is contained in it).

1.3 Lower Bounds for the Vamos Access Structure

In this section we prove our main result, stating that the access structures induced by the Vamos matroid cannot be close to ideal. That is, their information rate is bounded away from 1.

We will use a non-Shannon information inequality proved by Zhang and Yeung [36]. This inequality was used related to the Vamos matroid in [16] for proving lower bounds for network coding and in [27] for proving that a function is not asymptotically entropic.

Theorem 1.6 ([36, Theorem 3]). *For every four discrete random variables A, B, C, and D the following inequality holds:*

$$\begin{aligned} &3[H(CD) + H(BD) + H(BC)] + H(AC) + H(AB) \\ &\quad \geq \; H(D) + 2[H(C) + H(B)] + H(AD) + 4H(BCD) + H(ABC). \end{aligned} \tag{1.4}$$

Seymour [30] proved that $\mathcal{V}_6$ and $\mathcal{V}_8$ are not ideal. Inequality 1.4 was used in [27] to give an alternative proof of this fact. We next present the proof of [27]. Assume there is an ideal secret-sharing scheme realizing the Vamos access structure $\mathcal{V}_6$. Define the following random variables

$$\begin{aligned} A &\overset{\text{def}}{=} S_{\{v_1,v_2\}}, \\ B &\overset{\text{def}}{=} S_{\{v_3,v_4\}}, \\ C &\overset{\text{def}}{=} S_{\{v_5,v_6\}}, \\ D &\overset{\text{def}}{=} S_{\{v_7,v_8\}}. \end{aligned} \tag{1.5}$$

By Lemma 1.3.1 $H(S_T) = \text{rank}(T)H(S)$ for every set $T \subseteq \{v_1, \ldots, v_8\}$. Since all sets of size 2 are independent in the Vamos matroid, $H(A) = H(B) = H(C) = H(D) = 2H(S)$. Furthermore, by the definition of the circuits of size 4 in the Vamos matroid $H(AB) = H(AC) = H(BC) = H(BD) = H(CD) = 3H(S)$ while $H(AD) = 4H(S)$. Finally,

$H(BCD) = H(ABC) = 4H(S)$. Under the above definition of A, B, C, and D we notice that the l.h.s. of 1.4 is $33H(S)$ while the r.h.s. of 1.4 is $34H(S)$, a contradiction. Note that this proof strongly exploits the fact that the random variable AD, which corresponds to the shares of the independent set $\{v_1, v_2, v_7, v_8\}$, appears in the r.h.s. of 1.4, while the random variables appearing in the l.h.s. of 1.4 correspond to the shares of circuits in the matroid.

Applying Theorem 1.4 and Theorem 1.5, we can generalize the above proof and prove that $\mathcal{V}_6$ cannot be close to ideal. That is, we can prove that in every secret-sharing scheme realizing $\mathcal{V}_6$, the size of the entropy of the share of at least one party is at least $(1 + 1/110)H(S)$. Using direct arguments, we prove that the size of the entropy of the share of at least one party is at least $(1 + 1/9)H(S)$. Before we formally state our result, we prove two lemmas. First, to aid us in proving the better lower bound, we rearrange Inequality 1.4:

Lemma 1.6.1. *For every four discrete random variables A, B, C, and D the following inequality holds:*

$$\begin{aligned} &3H(C|D) + 2H(C|B) + H(B|C) + H(A|C) \\ &\quad \geq \; H(A|D) + 3H(C|BD) + H(BC|D) + H(C|AB). \end{aligned} \tag{1.6}$$

Proof. The claim is proved by a simple manipulation of 1.4. By 1.28, $3H(BCD) = 3H(C|BD) + 3H(BD)$ and $H(ABC) = H(C|AB) + H(AB)$. Substituting these expressions in 1.4 and rearranging the terms, we get

$$\begin{aligned} &3H(CD) + 3H(BC) + H(AC) \\ &\quad \geq \; H(D) + 2[H(C) + H(B)] \\ &\qquad + H(AD) + 3H(C|BD) + H(BCD) + H(C|AB). \end{aligned} \tag{1.7}$$

By 1.28, $2H(BC) = 2H(B) + 2H(C|B)$, $H(BC) = H(C) + H(B|C)$, and $H(AC) = H(C) + H(A|C)$. Substituting these expressions in 1.7 and rearranging the terms, we get

$$\begin{aligned} &3H(CD) + 2H(C|B) + H(B|C) + H(A|C) \\ &\quad \geq \; H(D) + H(AD) + 3H(C|BD) + H(BCD) + H(C|AB). \end{aligned} \tag{1.8}$$

By 1.28, $3H(CD) = 3H(D) + 3H(C|D)$, $H(AD) = H(D) + H(A|D)$, and $H(BCD) = H(D) + H(BC|D)$. Substituting these expressions in 1.8 and rearranging the terms, we get 1.6. □ □

To prove our lower bounds, we need the following simple lemma whose proof can be found in [2]. For completeness we present its proof here. Informally, this lemma states that if a set T is unauthorized and $T \cup \{b\}$ is authorized for some participant b, then guessing b's share given the shares of T is at least as hard as guessing the secret. Otherwise, the unauthorized set T can guess the share of b, and via the share compute the secret. Since, by the privacy requirement, the unauthorized set T cannot have any information on the secret, the entropy of the share must be at least $H(S)$.

Lemma 1.6.2. *Let* $T \subseteq V \setminus \{p_0\}$ *and* $b \notin T$ *such that* $T \cup \{b\} \in \mathcal{A}$ *and* $T \notin \mathcal{A}$. *Then,* $H(S_b|S_T) \geq H(S)$.

Proof. By applying 1.33 twice,

$$H(S, S_b|S_T) = H(S_b|S_T) + H(S|S_b, S_T) = H(S|S_T) + H(S_b|S, S_T).$$

The proof is straightforward from the second equality by taking into account that $H(S|S_T) = H(S)$, $H(S|S_b, S_T) = 0$, and that the conditional entropy function is nonnegative. □ □

1.3.1 Proving the Lower Bound for $\mathcal{V}_6$

We next state and prove our main result.

Theorem 1.7. *In any secret-sharing scheme realizing* $\mathcal{V}_6$ *with respect to a distribution on the secrets denoted by a random variable* S*, the entropy of the shares of at least one party is at least* $(1 + 1/9)H(S)$.

Proof. We fix any scheme realizing $\mathcal{V}_6$ and define λ as

$$\lambda \overset{\text{def}}{=} \frac{\max_{1 \leq i \leq 8}(H(S_{v_i}))}{H(S)} - 1.$$

In particular, for $1 \leq i \leq 8$:

$$H(S_{v_i}) \quad \leq \quad (1 + \lambda)H(S). \tag{1.9}$$

Recall that $H(S_{v_6}) = H(S)$ as v_6 is the dealer. We use the same random variables A, B, C, and D as defined in 1.5. We will show that Lemma 1.6.1 implies that $\lambda \geq 1/9$.

We start with giving upper-bounds on the terms on the left hand side of 1.6. Recall that v_6 is the dealer, $C = S_{\{v_5,v_6\}}$, and $D = S_{\{v_7,v_8\}}$. Thus, since $\{v_5, v_7, v_8\}$ is authorized,

$$\begin{aligned} H(C|D) &= H(S_{v_5}|S_{v_7}, S_{v_8}) + H(S_{v_6}|S_{v_5}, S_{v_7}, S_{v_8}) \quad \text{(from 1.33)} \\ &\leq H(S_{v_5}) \leq (1 + \lambda)H(S). \end{aligned} \tag{1.10}$$

Similarly,

$$H(C|B) \leq (1 + \lambda)H(S). \tag{1.11}$$

Next, recall that $B = S_{\{v_3,v_4\}}$. By applying 1.29 and 1.33,

$$\begin{array}{rcl} H(B|C) & = & H(S_{v_4}|C) + H(S_{v_3}|S_{v_4}, S_{v_5}, S_{v_6}) \\ & \leq & H(S_{v_4}) + H(S_{v_3}, S_{v_6}|S_{v_4}, S_{v_5}) - H(S_{v_6}|S_{v_4}, S_{v_5}) \\ & = & H(S_{v_4}) + H(S_{v_3}|S_{v_4}, S_{v_5}) + H(S_{v_6}|S_{v_3}, S_{v_4}, S_{v_5}) - H(S_{v_6}|S_{v_4}, S_{v_5}). \end{array}$$

Therefore, since $\{v_3, v_4, v_5\}$ is authorized and $\{v_4, v_5\}$ is unauthorized,

$$H(B|C) \leq H(S_{v_4}) + H(S_{v_3}) - H(S) \leq (1 + 2\lambda)H(S).$$

Similarly,

$$H(A|C) \leq (1 + 2\lambda)H(S). \tag{1.12}$$

So, the l.h.s. of 1.6 is at most $(7 + 9\lambda)H(S)$.

We continue by giving lower-bounds on the terms in the right hand side of 1.6. First, by using 1.32 and 1.33,

$$\begin{array}{rcl} H(A|D) & = & H(S_{v_1}|D) + H(S_{v_2}|D, S_{v_1}) \\ & \geq & H(S_{v_1}|D, S_{v_2}) + H(S_{v_2}|D, S_{v_1}) \\ & \geq & 2H(S), \end{array} \tag{1.13}$$

where the last inequality is obtained from Lemma 1.6.2 as $\{v_1, v_2, v_7, v_8\}$ is a minimal authorized set. Second, from 1.33 and 1.2 as BD is unauthorized

$$H(C|BD) \geq H(S_{v_6}|BD) \geq H(S). \tag{1.14}$$

Third, by 1.33, 1.32, and Lemma 1.6.2,

$$\begin{array}{rcl} H(BC|D) & = & H(B|D) + H(C|BD) \\ & \geq & H(S_{v_3}|D) + H(S) \\ & \geq & H(S_{v_3}|D, S_{v_1}) + H(S). \end{array}$$

From Lemma 1.6.2 and the fact that $\{v_1, v_3, v_7, v_8\}$ is a minimal authorized set,

$$H(BC|D) \geq 2H(S).$$

Fourth, from 1.33 and 1.2 as AB is unauthorized,

$$H(C|AB) \geq H(S_{v_6}|AB) \geq H(S). \tag{1.15}$$

So, the r.h.s. of 1.6 is at least $8H(S)$.

To conclude, we have proved that the l.h.s. of 1.6 is at most $(7 + 9\lambda)H(S)$ and the r.h.s. of 1.6 is at least $8H(S)$. As the l.h.s. of 1.6 should be at least the r.h.s. of 1.6, we deduce that $(7 + 9\lambda)H(S) \geq 8H(S)$, which implies that $\lambda \geq 1/9$. □ □

By Remark 1.2.1, we can assume without loss of generality that the distribution on the secrets is uniform, that is, if the domain of secrets is K, then $H(S) = \log|K|$. Furthermore, by 1.27, if the domain of shares of v_i is K_i, then $H(S_{v_i}) \leq \log|K_i|$. Thus, we can reformulate Theorem 1.7 as follows.

Corollary 1.7.1. *In any secret-sharing scheme realizing $\mathcal{V}_6$ with respect to a distribution on the secrets with support K, the size of the domain of shares of at least one party is at least $|K|^{1+1/9}$.*

1.3.2 Proving the Lower Bound for $\mathcal{V}_8$

In a similar manner to the proof of the lower bound for $\mathcal{V}_6$, we prove a slightly weaker lower-bound for $\mathcal{V}_8$. As before, we begin by rearranging Inequality 1.4. The next lemma is proved similarly to Lemma 1.6.1.

Lemma 1.7.1. *For every four discrete random variables A, B, C, and D the following inequality holds:*

$$\begin{aligned} &3H(D|C) + 2H(D|B) + H(BD) + H(B|A) \\ &\geq \quad H(D) + H(D|A) + H(B|C) + 4H(D|BC) + H(B|AC). \qquad (1.16) \end{aligned}$$

Theorem 1.8. *In any secret-sharing scheme realizing $\mathcal{V}_8$ with respect to a distribution on the secrets denoted by a random variable S, the entropy of the shares of at least one party is at least $(1 + 1/10)H(S)$.*

Proof. We fix any scheme realizing $\mathcal{V}_8$ and we define λ as in the proof of Theorem 1.7. Then $H(S_{v_i}) \leq (1+\lambda)H(S)$ for every $i = 1, \ldots, 8$. Recall that $H(S_{v_8}) = H(S)$ as v_8 is the dealer. We use the same random variables A, B, C, and D as defined in 1.5. In a similar way as in Theorem 1.7, we find bounds on the terms of 1.16 to obtain a bound on λ.

Claim 1.8.1. $H(B|A) \leq (1 + 3\lambda)H(S)$.

To prove this claim, we first observe that

$$\begin{aligned} H(B|A) &= H(S_{v_3}, S_{v_4}|S_{v_1}, S_{v_2}) \\ &\leq H(S_{v_3}) + H(S_{v_4}|S_{v_1}, S_{v_2}, S_{v_3}) \\ &\leq (1 + \lambda)H(S) + H(S_{v_4}|S_{v_1}, S_{v_2}, S_{v_3}). \qquad (1.17) \end{aligned}$$

We now bound $H(S_{v_4}|S_{\{v_1,v_2,v_3\}})$. By applying 1.33 twice,

$$\begin{aligned} H(S_{v_4}, S_{v_5}|S_{\{v_1,v_2,v_3\}}) &= H(S_{v_4}|S_{\{v_1,v_2,v_3\}}, S_{v_5}) + H(S_{v_5}|S_{\{v_1,v_2,v_3\}}) \\ &= H(S_{v_5}|S_{\{v_1,v_2,v_3\}}, S_{v_4}) + H(S_{v_4}|S_{\{v_1,v_2,v_3\}}). \qquad (1.18) \end{aligned}$$

Thus, by 1.18

$$\begin{aligned} H(S_{v_4}|S_{\{v_1,v_2,v_3\}}) &= H(S_{v_4}|S_{\{v_1,v_2,v_3,v_5\}}) + H(S_{v_5}|S_{\{v_1.v_2,v_3\}}) \\ &\quad -H(S_{v_5}|S_{\{v_1.v_2,v_3,v_4\}}). \end{aligned} \tag{1.19}$$

We next bound each of the elements of the above sum, and get the desired result. First,

$$H(S_{v_5}|S_{\{v_1,v_2,v_3,v_4\}}) \leq H(S_{v_5}) \leq (1+\lambda)H(S).$$

Second, from Lemma 1.6.2 we have

$$H(S_{v_5}|S_{\{v_1,v_2,v_3,v_4\}}) \geq H(S).$$

Next observe that $\{v_1, v_2, v_3, v_5\}$ is authorized in $\mathcal{V}_8$, and hence $H(S_{v_8}|S_{\{v_1,v_2,v_3,v_5\}}) = 0$, thus,

$$\begin{aligned} H(S_{v_4}|S_{\{v_1,v_2,v_3,v_5\}}) &= H(S_{\{v_1,v_2,v_3,v_5\}}, S_{v_4}) - H(S_{\{v_1,v_2,v_3,v_5\}}) \\ &= H(S_{\{v_1,v_2,v_3,v_4,v_5\}}) \\ &\quad -[H(S_{v_8}|S_{\{v_1,v_2,v_3,v_5\}}) + H(S_{\{v_1,v_2,v_3,v_5\})}] \\ &= H(S_{\{v_1,v_2,v_3,v_4,v_5\}}) - H(S_{\{v_1,v_2,v_3,v_5,v_8\}}) \\ &\leq H(S_{\{v_1,v_2,v_3,v_4,v_5,v_8\}}) - H(S_{\{v_1,v_2,v_3,v_5,v_8\}}) \\ &= H(S_{v_4}|S_{\{v_1,v_2,v_3,v_5,v_8\}}) \\ &\leq H(S_{v_4}|S_{\{v_1,v_2,v_5,v_8\}}) \\ &= H(S_{v_4}S_{v_8}|S_{\{v_1,v_2,v_5\}}) - H(S_{v_8}|S_{\{v_1,v_2,v_5\}}) \\ &= [H(S_{v_4}|S_{\{v_1,v_2,v_5\}}) + H(S_{v_8}|S_{\{v_1,v_2,v_4,v_5\}})] \\ &\quad -H(S_{v_8}|S_{\{v_1,v_2,v_5\}}) \\ &\leq H(S_{v_4}) + 0 - H(S) \\ &\leq \lambda H(S). \end{aligned} \tag{1.20}$$

In the last steps we used that $\{v_1, v_2, v_4, v_5\}$ is a minimal authorized subset. Now, by summing up the bounds,

$$H(S_{v_4}|S_{\{v_1,v_2,v_3\}}) \;\leq\; \lambda H(S) + (1+\lambda)H(S) - H(S) \;=\; 2\lambda H(S). \tag{1.21}$$

Thus, by 1.17 and 1.21, $H(B|A) \leq (1+3\lambda)H(S)$, which concludes the proof of our claim.

Since $\{v_5, v_6, v_7\}$ is an authorized set,

$$\begin{aligned} H(D) - H(D|C) &= (H(S_{v_7}) + H(S_{v_8}|S_{v_7})) \\ &\quad -(H(S_{v_7}|S_{\{v_5,v_6\}}) + H(S_{v_8}|S_{\{v_5,v_6,v_7\}})) \\ &= H(S_{v_7}) + H(S) - H(S_{v_7}|S_{\{v_5,v_6\}}) - 0 \\ &\geq H(S). \end{aligned} \tag{1.22}$$

Thus, by 1.16 and 1.22,

$$\begin{aligned} & 2H(D|C) + 2H(D|B) + H(BD) + H(B|A) \\ \geq \; & H(D|A) + H(B|C) + 4H(D|BC) + H(B|AC) + H(S). \end{aligned} \tag{1.23}$$

We next give upper bounds for the terms in the l.h.s. of 1.23. We proved before that $H(B|A) \leq (1+3\lambda)H(S)$. For the rest of the terms in the l.h.s. we use straightforward bounds. First,

$$H(D|C) = H(S_{v_7}S_{v_8}|C) \leq H(S_{v_8}|S_{v_7}C) + H(S_{v_7}) \leq (1+\lambda)H(S)$$

because $\{v_5, v_6, v_7\}$ is authorized, and similarly $H(D|B) \leq (1+\lambda)H(S)$. Second, $H(BD) = H(S_{\{v_3,v_4,v_7,v_8\}}) = H(S_{v_8}|S_{\{v_3,v_4,v_7\}}) + H(S_{\{v_3,v_4,v_7\}}) \leq 3(1+\lambda)H(S)$, since $\{v_3, v_4, v_7\}$ is authorized. Thus, the l.h.s. of 1.23 is less than $(8+10\lambda)H(S)$.

We continue by giving lower bounds for the terms in the r.h.s. of 1.23. First, by Lemma 1.6.2,

$$H(D|A) = H(S_{v_8}|S_{\{v_1,v_2,v_7\}}) + H(S_{v_7}|S_{\{v_1,v_2\}}) \geq 2H(S),$$

since $\{v_1, v_2, v_7\}$ is unauthorized and $\{v_1, v_2, v_5, v_7\}$ is authorized. Second,

$$H(B|C) \;\; \geq \;\; H(B|AC) \geq H(S) \tag{1.24}$$

since $\{v_1, v_2, v_5, v_6\}$ is unauthorized and $\{v_1, v_2, v_3, v_4, v_5, v_6\}$ is authorized. Next, $H(D|BC) \geq H(S)$ since the set $\{v_3, v_4, v_5, v_6\}$ is unauthorized, while $\{v_7, v_8\}$ contains the dealer v_8. Finally, $H(B|AC) \geq H(S)$ by 1.24. Thus, we conclude that the r.h.s. of 1.23 is at least $9H(S)$.

Finally, the bounds we obtained for both sides of Inequality 1.23 imply that $\lambda \geq 1/10$. □ □

Corollary 1.8.1. *In any secret-sharing scheme realizing $\mathcal{V}_8$ with respect to a distribution on the secret with support K, the size of the domain of shares of at least one party is at least $|K|^{1+1/10}$.*

1.3.3 Lower Bounds for Linear Secret-Sharing Schemes

In the following, we present a lower bound for the size of the shares' domain that applies only to *linear* secret-sharing schemes with access structure $\mathcal{V}_6$ or $\mathcal{V}_8$. Nearly all known secret-sharing schemes are linear. A secret-sharing scheme is linear if the distribution scheme is such that the domain of secrets K, the domain of random strings R, and the domains of shares of the i-th party K_i, for every i, are vector spaces over some finite field, Π is a linear mapping, and the distribution on random strings μ is uniform. This bound is obtained in a very similar way as the previous ones by using an inequality due to Ingleton [17], which applies only to linear random variables, that is, random variables defined by linear mappings.

Theorem 1.9 ([17, 28])**.** *For every four* linear *discrete random variables A, B, C, and D the following inequality holds:*

$$\begin{aligned}&H(CD) + H(BD) + H(BC) + H(AC) + H(AB)\\ \geq\ & H(C) + H(B) + H(AD) + H(BCD) + H(ABC). \qquad (1.25)\end{aligned}$$

The proof of the next lemma is very similar to the one of Lemma 1.6.1.

Lemma 1.9.1. *For every four linear discrete random variables A, B, C, and D the following inequality holds:*

$$\begin{aligned}&H(C|D) + H(C|B) + H(A|C)\\ \geq\ & H(A|D) + H(C|BD) + H(C|AB). \qquad (1.26)\end{aligned}$$

The following result is proved in a similar way to the proof of Theorem 1.7.

Theorem 1.10. *In any* linear *secret-sharing scheme realizing $\mathcal{V}_6$ with respect to a distribution on the secrets denoted by a random variable S, the entropy of the shares of at least one party is at least* $(1 + 1/4)H(S)$.

Proof. We fix any *linear* scheme realizing $\mathcal{V}_6$ and define

$$\lambda \stackrel{\text{def}}{=} \max_{1\leq i\leq 8}(H(S_{v_i}))/H(S) - 1.$$

We use the same random variables A, B, C, and D as defined in 1.5. Note that all bounds proved in Section 1.3.1 apply, in particular, to linear secret-sharing realizing $\mathcal{V}_6$. Thus, by 1.10, 1.11, and 1.12, the l.h.s. of 1.26 is at most $(3 + 4\lambda)H(S)$. By 1.13, 1.14, and 1.15, the r.h.s. of 1.26 is at least $4H(S)$. This implies that $(3+4\lambda)H(S) \geq 4H(S)$, which implies that $\lambda \geq 1/4$. □

Corollary 1.10.1. *In any linear secret-sharing scheme realizing $\mathcal{V}_6$ with respect to a distribution on the secrets with support K, the size of the domain of shares of at least one party is at least* $|K|^{1+1/4}$.

Finally, the same bound applies to the linear secret-sharing schemes with access structure $\mathcal{V}_8$ by duality. The *dual* of an access structure $\mathcal{A}$ is the access structure

$$\mathcal{A}^* \stackrel{\text{def}}{=} \{T \subseteq P \,:\, P \setminus T \notin \mathcal{A}\}.$$

It is well known that, for every linear secret-sharing scheme Σ with access structure $\mathcal{A}$, there exists a linear secret sharing scheme Σ^* for $\mathcal{A}^*$ such that the domain of the shares of every participant is the same for Σ and for Σ^* (see [14], for instance). Therefore, since $\mathcal{V}_8^*$ is isomorphic to $\mathcal{V}_6$, the bounds in Theorem 1.10 and Corollary 1.10.1 apply also to the access structure $\mathcal{V}_8$.

Bibliography

[1] A. Beimel and B. Chor. Universally ideal secret sharing schemes. *IEEE Trans. on Information Theory*, 40(3):786–794, 1994.

[2] A. Beimel and N. Livne. On matroids and non-ideal secret sharing. In S. Halevi and T. Rabin, editors, *Proc. of the Third Theory of Cryptography Conference – TCC 2006*, volume 3876 of *Lecture Notes in Computer Science*, pages 482–501, 2006.

[3] M. Bellare and P. Rogaway. Robust computational secret sharing and a unified account of classical secret-sharing goals. In *Proc. of the 14th conference on Computer and communications security*, pages 172–184, 2007.

[4] J. Benaloh and J. Leichter. Generalized secret sharing and monotone functions. In S. Goldwasser, editor, *Advances in Cryptology – CRYPTO '88*, volume 403 of *Lecture Notes in Computer Science*, pages 27–35. Springer-Verlag, 1990.

[5] G. R. Blakley. Safeguarding cryptographic keys. In R. E. Merwin, J. T. Zanca, and M. Smith, editors, *Proc. of the 1979 AFIPS National Computer Conference*, volume 48 of *AFIPS Conference proceedings*, pages 313–317. AFIPS Press, 1979.

[6] C. Blundo, A. De Santis, and U. Vaccaro. On secret sharing schemes. *Inform. Process. Lett.*, 65(1):25–32, 1998.

[7] E. F. Brickell. Some ideal secret sharing schemes. *Journal of Combin. Math. and Combin. Comput.*, 6:105–113, 1989.

[8] E. F. Brickell and D. M. Davenport. On the classification of ideal secret sharing schemes. *J. of Cryptology*, 4(73):123–134, 1991.

[9] E. F. Brickell and D. R. Stinson. Some improved bounds on the information rate of perfect secret sharing schemes. *J. of Cryptology*, 5(3):153–166, 1992.

[10] R. M. Capocelli, A. De Santis, L. Gargano, and U. Vaccaro. On the size of shares for secret sharing schemes. *J. of Cryptology*, 6(3):157–168, 1993.

[11] B. Chor and E. Kushilevitz. Secret sharing over infinite domains. *J. of Cryptology*, 6(2):87–96, 1993.

[12] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley & Sons, 1991.

[13] L. Csirmaz. The size of a share must be large. *J. of Cryptology*, 10(4):223–231, 1997.

[14] M. van Dijk, W.A. Jackson, and K. M. Martin. A note on duality in linear secret sharing schemes. *Bull. of the Institute of Combinatorics and its Applications*, 19:98–101, 1997.

[15] R. Dougherty, C. Freiling, and K. Zeger. Six new non-Shannon information inequalities. In *IEEE International Symposium on Information Theory (ISIT)*, pages 233–236, 2006.

[16] R. Dougherty, C. Freiling, and K. Zeger. Networks, matroids, and non-Shannon information inequalities. *IEEE Trans. on Information Theory*, 53(6):1949–1969, 2007.

[17] A.W. Ingleton. Conditions for representability and transversability of matroids. In *Proc. Fr. Br. Conf 1970*, pages 62–67. Springer-Verlag, 1971.

[18] M. Ito, A. Saito, and T. Nishizeki. Secret sharing schemes realizing general access structure. In *Proc. of the IEEE Global Telecommunication Conf., Globecom 87*, pages 99–102, 1987. Journal version: Multiple assignment scheme for sharing secret. *J. of Cryptology*, 6(1):15-20, 1993.

[19] M. Karchmer and A. Wigderson. On span programs. In *Proc. of the 8th IEEE Structure in Complexity Theory*, pages 102–111, 1993.

[20] E. D. Karnin, J. W. Greene, and M. E. Hellman. On secret sharing systems. *IEEE Trans. on Information Theory*, 29(1):35–41, 1983.

[21] K. Kurosawa, K. Okada, K. Sakano, W. Ogata, and S. Tsujii. Nonperfect secret sharing schemes and matroids. In *Advances in Cryptology – EUROCRYPT '93*, volume 765 of *Lecture Notes in Computer Science*, pages 126–141. Springer-Verlag, 1994.

[22] N. Livne. On matroids and non-ideal secret sharing. Master's thesis, Ben-Gurion University, Beer-Sheva, 2005.

[23] J. Martí-Farré and C. Padró. Secret sharing schemes with three or four minimal qualified subsets. *Designs, Codes and Cryptography*, 34(1):17–34, 2005.

[24] J. Martí-Farré and C. Padró. On secret sharing schemes, matroids and polymatroids. In S. Vadhan, editor, *Proc. of the Fourth Theory of Cryptography Conference – TCC 2007*, volume 4392 of *Lecture Notes in Computer Science*, pages 253–272. Springer-Verlag, 2007.

[25] J. Martí-Farré and C. Padró. On secret sharing schemes, matroids and polymatroids. Journal version of [24], in preperation, 2007.

[26] F. Matúš. Matroid representations by partitions. *Discrete Mathematics*, 203:169–194, 1999.

[27] F. Matúš. Two constructions on limits of entropy functions. *IEEE Trans. on Information Theory*, 53(1):320–330, 2007.

[28] J. G. Oxley. *Matroid Theory*. Oxford University Press, 1992.

[29] P. D. Seymour. A forbiden minor characterization of matroid ports. *Quart. J. Math. Oxford Ser.*, 27:407–413, 1976.

[30] P. D. Seymour. On secret-sharing matroids. *J. of Combinatorial Theory, Series B*, 56:69–73, 1992.

[31] A. Shamir. How to share a secret. *Communications of the ACM*, 22:612–613, 1979.

[32] G. J. Simmons, W. Jackson, and K. M. Martin. The geometry of shared secret schemes. *Bulletin of the ICA*, 1:71–88, 1991.

[33] J. Simonis and A. Ashikhmin. Almost affine codes. *Designs, Codes and Cryptography*, 14(2):179–197, 1998.

[34] P. Vamos. On the representation of independence structures. Unpublished manuscript, 1968.

[35] D. J. A. Welsh. *Matroid Theory*. Academic press, London, 1976.

[36] Z. Zhang and R. W. Yeung. On characterization of entropy function via information inequalities. *IEEE Trans. on Information Theory*, 44(4):1440–1452, 1998.

1.4 Appendix: Basic Definitions from Information Theory

In this appendix, we review the basic concepts of information theory used in this paper. For a complete treatment of this subject see, e.g., [12]. All the logarithms here are of base 2.

Given a finite random variable X, we define the *entropy* of X, denoted $H(X)$, as

$$H(X) \overset{\text{def}}{=} - \sum_{x, \Pr[X=x]>0} \Pr[X = x] \log \Pr[X = x].$$

It can be proved that

$$0 \leq H(X) \leq \log |\operatorname{support}(X)|, \tag{1.27}$$

where $|\operatorname{support}(X)|$ is the size of the support of X (the number of values with probability greater than zero). The upper bound is obtained if and only if the distribution of X is uniform.

Given two finite random variables X and Y (possibly dependent), we define the *conditioned entropy of X given Y* as

$$H(X|Y) \overset{\text{def}}{=} H(XY) - H(Y). \tag{1.28}$$

For convenience, when dealing with the entropy function, XY will denote $X \cup Y$. From the definition of the conditional entropy, the following properties can be proved:

$$0 \leq H(X|Y) \leq H(X), \tag{1.29}$$

$$H(Y) \leq H(XY), \tag{1.30}$$

and

$$H(XY) \leq H(X) + H(Y). \tag{1.31}$$

Given three finite random variable X, Y and Z (possibly dependent), the following properties hold:

$$H(X|Y) \geq H(X|YZ), \tag{1.32}$$

$$H(XY|Z) = H(X|YZ) + H(Y|Z) \geq H(Y|Z), \tag{1.33}$$

and

$$H(XY|Z) \leq H(X|Z) + H(Y|Z). \tag{1.34}$$

Chapter 2

All Natural NPC Problems Have Average-Case Complete Versions[1]

Noam Livne

Abstract

The theory of *average case complexity* studies the expected complexity of computational tasks under various specific distributions on the instances, rather than their worst case complexity. Thus, this theory deals with *distributional problems*, defined as pairs each consisting of a decision problem and a probability distribution over the instances. While for applications *utilizing* hardness, such as cryptography, one seeks an efficient algorithm that outputs random instances of some problem that are hard for any algorithm with high probability, the resulting hard distributions in these cases are typically highly artificial, and do not establish the hardness of the problem under ”interesting” or ”natural” distributions. This paper studies the possibility of proving generic hardness results (i.e., for a wide class of $\mathcal{NP}$-complete problems), under ”natural” distributions. Since it is not clear how to define a class of ”natural” distributions for general $\mathcal{NP}$-complete problems, one possibility is to impose some strong computational constraint on the distributions, with the intention of this constraint being to force the distributions to ”look natural”. Levin, in his seminal paper on average case complexity from 1984, defined such a class of distributions, which he called *P-computable distributions*. He then showed that the $\mathcal{NP}$-complete Tiling problem, under some P-computable distribution, is hard for the complexity class of distributional $\mathcal{NP}$ problems (i.e. $\mathcal{NP}$ with P-computable distributions). However, since then very few $\mathcal{NP}$-complete problems (coupled with P-computable distributions), and in particular ”natural” problems, were shown to be hard in this sense. In

[1] To appear in *Journal of Computational Complexity*.

this paper we show that all natural $\mathcal{NP}$-complete problems can be coupled with P-computable distributions such that the resulting distributional problem is hard for distributional $\mathcal{NP}$.

2.1 Introduction

While most of the research in complexity theory concentrates on worst-case complexity, in practice one would often settle for an algorithm that fails only with a negligible probability over the relevant distribution of the instances. To address this observation, the theory of *average case complexity*, initiated by Levin [8], studies *distributional problems*, defined as pairs consisting of some decision (or search[2]) problem and a probability distribution over all strings. Feasibly solving such a problem means providing an algorithm that solves all instances and, loosely speaking, runs in expected polynomial time (or, alternatively, that runs in polynomial time and decides the problem with high probability over the related distribution of the inputs).

For applications *utilizing* hardness (such as cryptography), we are typically content with the mere ability to *generate* hard instances of some problem, and thus seek for an efficient probabilistic algorithm that outputs random instances that are hard for any algorithm with high probability. Such distributions are called *samplable distributions*, as for any such distribution there exists an algorithm that samples it. It is well-known ([1]) that any $\mathcal{NP}$-complete decision problem has a distributional version that is complete for $\mathcal{NP}$ with samplable distributions. In other words, any $\mathcal{NP}$-complete decision problem can be coupled with a samplable distribution such as to form a distributional problem that is hard on the average if and only if there exists some $\mathcal{NP}$ problem that is hard on the average under some samplable distribution. However, typically these distributions are very artificial, and these results do not imply hardness of these $\mathcal{NP}$-complete problems under ”natural”, or ”interesting” distributions. On the other hand, one can show that certain decision problems are complete in this sense under ”natural” distributions, such as the uniform distribution, but in this case, the problems are somewhat unnatural[3] (e.g. Bounded Halting and Tiling [8]). These results, however, do not show (conditional) hardness of ”natural” $\mathcal{NP}$-complete decision problems under ”natural” distributions. Loosely speaking, all known results do not eliminate the possibility that even if $\mathcal{P} \neq \mathcal{NP}$, we can solve most $\mathcal{NP}$-complete problems for most practical applications.

This paper studies the possibility of showing *generic*, ”natural” hardness-on-average results for $\mathcal{NP}$-complete decision problems, that is, hardness results that apply for a large family of $\mathcal{NP}$-complete problems under ”simple” distributions. When dealing with the class $\mathcal{NP}$ as a whole, it is not clear how to define “natural” distributions, because these may depend on the

[2] By a result of Ben-David et al. [1], the discussion here, although focuses on decision problems, holds also for search problems.

[3] The notion of a *natural decision problem* is of course subjective. See further discussion in Section 2.1.4.

specific combinatorial structure of each problem. One approach to overcome this difficulty is to define some ”natural” constraint on the probability distributions over the strings (with the intention of this constraint forcing the distributions to ”look natural”), and then show that a large family of $\mathcal{NP}$-complete problems can be coupled with these distributions as to make them hard for $\mathcal{NP}$ problems under some (as wide as possible) class of distributions. In his seminal paper [8], Levin defined such a family of probability distributions, which he called *P-computable distributions*. He then showed that some rather artificial decision problem (namely, the Tiling Problem), when coupled with some P-computable distribution, is hard for $\mathcal{NP}$ with P-computable distributions. (Later, Impagliazzo and Levin [6] showed that any distributional problem that is hard for $\mathcal{NP}$ with P-computable distributions is in fact hard for $\mathcal{NP}$ with samplable distributions.) However, since then very few $\mathcal{NP}$-complete problems, and in particular ”natural” ones (for example, the 21 problems in Karp's seminal paper on $\mathcal{NP}$-completeness [7]), were shown to be complete in this sense. This was raised as an open problem, for example, in [2].

2.1.1 Our Contribution

In this paper, we show that a very wide family of $\mathcal{NP}$-complete problems (apparently all known ones, see Section 2.4.1), have P-computable distributions under which they are hard for $\mathcal{NP}$ with samplable distributions. That is, assuming some $\mathcal{NP}$ problem is hard on the average under some samplable distribution (which in particular follows from the assumption that cryptography is possible), each of these $\mathcal{NP}$-complete decision problems is hard on the average under some P-computable distribution. We call such a problem *average case $\mathcal{NP}$-complete* (or dist*$\mathcal{NP}$-complete*, see Section 2.1.2 for a more detailed definition, or Section 2.2.2 for the formal definition). In Section 2.4.1 we analyze our constructed distributions, and discuss the interpretation of our result.

In order to further present our result, we review informally the basic notions of Levin's theory [8].

2.1.2 Levin's Theory of Average-Case Complexity

The theory of average case complexity was initiated by Levin [8]. It refers to the complexity of solving problems with respect to certain probability distributions over their instances. Levin set the foundations to an average case complexity theory analogous to the theory of $\mathcal{NP}$-completeness. He first had to define which probability distributions will be of interest. Letting these probability distributions range over all possible probability distributions would have collapsed the new theory to classic worst-case complexity[4]. On the other hand, considering only the uniform distribution seems quiet arbitrary and limiting. Levin therefore defined a class of probability distributions, which he called *P-computable distributions*. These are probability distributions over all strings, such that the accumulative probability can be computed in polynomial time (that is, there exists a polynomial time algorithm that given x outputs the probability that a string smaller or equal lexicographically to x is drawn). Focusing on these probability distributions, Levin defined:

- The class $\mathrm{avg}\mathcal{P}$, which is analogous to $\mathcal{P}$, and consists of the distributional problems that can be solved "efficiently on the average".
- The class $\mathrm{dist}\mathcal{NP}$, which is analogous to $\mathcal{NP}$, and consists of decision problems in $\mathcal{NP}$ paired with P-computable probability distributions.
- A class of reductions, which we call here *AP-reductions* (for Average-case Preserving reductions), analogous to polynomial-time reductions (such as Karp or Cook reductions). Such reductions preserve "easiness on the average", that is, if a distributional problem can be AP-reduced to a problem in $\mathrm{avg}\mathcal{P}$, then the reduced problem is also in $\mathrm{avg}\mathcal{P}$. Although we did not specify yet what it means to solve a problem on the average, the crucial point is that these AP-reductions preserve "easiness on the average" with respect to various different definitions, including the original ones of Levin. The crucial aspect in these reductions is that instances that occur with some probability are not mapped to instances that occur with much smaller probability.

Next, Levin showed that there exists a $\mathrm{dist}\mathcal{NP}$-complete distributional problem, that is, a problem in $\mathrm{dist}\mathcal{NP}$ that every problem in $\mathrm{dist}\mathcal{NP}$ can be AP-reduced to it. Thus, this complete problem is in $\mathrm{avg}\mathcal{P}$ if and only if $\mathrm{dist}\mathcal{NP} \subseteq \mathrm{avg}\mathcal{P}$. As mentioned above, a major deficiency of this theory was the lack of a wide variety of "natural" $\mathcal{NP}$ decision problems that have a distributional version that is $\mathrm{dist}\mathcal{NP}$-complete.

2.1.3 An Overview of Our Proof

We show a simple sufficient condition for an $\mathcal{NP}$-complete decision problem to have a distributional version that is $\mathrm{dist}\mathcal{NP}$-complete. This condition refers to some natural paddability property. Our technique is based on the construction of a restricted type of Karp-reductions

[4] For a decision problem L to be hard for any problem in $\mathcal{NP}$ under *any* distribution, one would be forced to couple L with a distribution that is easy-on-average if and only if $\mathcal{P} = \mathcal{NP}$. Thus, for $L \in \mathcal{NPC}$, the average-case complexity of the resulting distributional problem would be equivalent to the worst-case complexity of L.

that "preserve order" in some (natural) sense. If such an order preserving reduction exists from some (decisional part of a) $\mathrm{dist}\mathcal{NP}$-complete problem to some problem in $\mathcal{NP}$, then the latter has a probability distribution that when coupled with it, forms a $\mathrm{dist}\mathcal{NP}$-complete decision problem. The aforementioned order preserving reduction is related to the paddability property mentioned above.

Let us demonstrate, informally, the high-level ideas of our technique on SAT. Assume some standard encoding for SAT (we will freely identify a formula and its representation). Let (C, μ) be some $\mathrm{dist}\mathcal{NP}$-complete distributional problem (so, in particular, $C \in \mathcal{NP}$ and μ is P-computable), and let h be a Karp reduction (i.e. polynomial-time many-to-one reduction) from C to SAT such that $|x| \geq |y|$ if and only if $|h(x)| \geq |h(y)|$ (we will show in Section 2.3 how to obtain such reductions). We define a new Karp-reduction f such that $f(w)$ "encodes", in some explicit form, w itself into the formula $h(w)$. For example, let $e_0 = (x_0 \vee \neg x_0)$ and $e_1 = (x_1 \vee \neg x_1)$, and assume that the encoding of e_1 is lexicographically larger than that of e_0 and that the lengths of e_0 and e_1 are equal. Now define

$$f(w) = e_{w_1} \wedge e_{w_2} \wedge \ldots \wedge e_{w_{|w|}} \wedge h(w) \tag{2.1}$$

where w_i is the i-th bit of w. Note that the "encoding" of w in the left part of the formula ensures that f has the following properties:

- Invertibility: given $f(w)$ one can compute w.
- Monotonicity: if w is lexicographically larger than w' then $f(w)$ is lexicographically larger than $f(w')$.
- Preserving satisfiability: $f(w)$ preserves the truth value of $h(w)$ (since e_0 and e_1 are tautologies).

Thus, f is an "order preserving" reduction of C to SAT.

Let us see how such order preserving reductions (which are defined between *standard* decision problems) are related to AP-reductions (which are defined between *distributional* problems). We couple SAT with the following probability distribution η:

$$\eta(x) = \begin{cases} \mu(f^{-1}(x)) & \text{if } x \in \mathrm{image}(f) \\ 0 & \text{otherwise} \end{cases} \tag{2.2}$$

Then the reduction f is a AP-reduction from (C, μ) to (SAT, η), because it maps each instance of C to an instance of SAT that occurs with exactly the same probability. Since (C, μ) is $\mathrm{dist}\mathcal{NP}$-complete, and AP-reductions are transitive, it follows that (SAT, η) is $\mathrm{dist}\mathcal{NP}$-hard (under AP-reductions). Furthermore, because of the special properties of f, and since μ is

P-computable, then so is η. Loosely speaking, since f is monotone, in order to compute the accumulative probability of w under η, it suffices to compute the accumulative probability of its inverse under μ; and since f is invertible, we can compute this inverse. It follows that (SAT, η) is in $\mathrm{dist}\mathcal{NP}$, and therefore (SAT, η) is $\mathrm{dist}\mathcal{NP}$-complete. For more details and a complete proof see Section 2.3.

We have just demonstrated that since such an "order preserving" Karp-reduction exists between (the decisional part of) some $\mathrm{dist}\mathcal{NP}$-complete problem and SAT, the later has a distributional version that is $\mathrm{dist}\mathcal{NP}$-complete. Moreover, we note that the construction of the Karp-reduction exploited only the properties of the target problem, SAT. More specifically, the construction used a technique called "padding", introduced by Berman and Hartmanis [4], in order to encode w into $h(w)$. This "paddability" property is a property of decision problems, rather than of reductions. Using this paddability property one can prove similar results for other $\mathcal{NP}$-complete problems.

Hence, essentially we "reduced" the problem of showing that an $\mathcal{NP}$-complete decision problem has a $\mathrm{dist}\mathcal{NP}$-complete version to the problem of proving some paddability properties for this decision problem. Although we do not know whether these paddability properties hold for every decision problem in $\mathcal{NP}$ (and showing that they do is at least as hard as proving $\mathcal{P} \neq \mathcal{NP}$), they are very easy to verify for individual problems (as examples see sections 2.3.2, 2.3.3 and 2.3.4). In particular, we have verified that these properties hold for the famous twenty-one problems treated in Karp's seminal paper [7]. See further discussion in Section 2.4.1

2.1.4 Reflections on the Notion of Natural Decision Problems

In the context of this paper, a discussion on the notion of *natural decision problems* can not be avoided. Most researchers have some intuition on what makes a decision problem "natural", and furthermore, it seems that these intuitions tend to be similar. This suggests the following (empirically oriented) definition: *A decision problem is natural if most researchers would say it is.*

In an attempt to offer a less fluid definition, we suggest the following definition: *The extent to which a decision problem is natural, with respect to some result, is proportional to the amount of references to that problem that are prior to that result and occur in a different context.* Thus, for example, the Satisfiability is very natural with respect to the Cook-Levin Theorem, because this problem was defined and studied in many other studies in different contexts (such as logic) prior to the Cook-Levin Theorem. To the contrary, the sequence of decision problems constructed in the Hierarchy Theorem of Hartmanis and Stearns [5] is completely artificial, because these decision problems were first defined in this context (let alone that they are never mentioned outside the context of the Hierarchy Theorem).

Organization In section 2.2 we give some definitions that will be used throughout this paper. In Section 2.3 we provide a rigorous presentation of our results, by first showing a sufficient condition for an $\mathcal{NP}$-complete decision problem to have a distributional version that is $\mathrm{dist}\mathcal{NP}$-complete, and then, using this sufficient condition to show that some well-known $\mathcal{NP}$-complete decision problems have distributional versions that are $\mathrm{dist}\mathcal{NP}$-complete. In Section 2.4 we discuss some related issues concerning our results.

2.2 Preliminaries

2.2.1 Strings and Functions Over Strings

For a string x, we denote by $|x|$ the length of x, and by $x_1, x_2, \ldots, x_{|x|}$ the bits of x. Throughout this paper, the symbol "<", when applied between strings, will denote the standard lexicographical order over all strings (i.e., $|y| = |y'| \Rightarrow x0y < x1y'$, and $|x| < |x'| \Rightarrow x < x'$). Given a string x, the strings $x - 1$ and $x + 1$ denote, respectively, the strings preceding and succeeding x.

Given an instance of a decision problem, its *characteristic* refers to the value of the characteristic function for this instance (i.e., it equals 1 if the problem contains this instance and 0 otherwise).

Definition 2.0.1 (P-invertible function)**.** *A function f is* P-invertible *if it is 1-1, and there is a polynomial-time algorithm that given x returns $f^{-1}(x)$ if it is defined, and a failure symbol $\perp$ otherwise.*

Definition 2.0.2 (length-regular function)**.** *A function f is* length-regular *if for every $x, y \in \{0,1\}^*$, it holds that $|x| \leq |y|$ if and only if $|f(x)| \leq |f(y)|$.*

Note that a function f is length-regular if and only if it satisfies the following two conditions: (1) $|x| = |y|$ if and only if $|f(x)| = |f(y)|$ and (2) $|x| > |y|$ if and only if $|f(x)| > |f(y)|$.

Definition 2.0.3 (monotonous and semi-monotonous functions)**.** *A function over the strings is* monotonous *if it is strictly increasing WRT lexicographical order (i.e., if for every $x, y \in \{0,1\}^*$ it holds that $x < y$ if and only if $f(x) < f(y)$), and is* semi-monotonous *if for every n, the function, when restricted to the set of strings of length n, is monotonous. (i.e., if for every $x, y \in \{0,1\}^*$ such that $|x| = |y|$ it holds that $x < y$ if and only if $f(x) < f(y)$).*

While a semi-monotonous function is only monotonous within lengths, a function that is semi-monotonous and length-regular is monotonous (over *all* strings) because, in particular, for a length-regular function f, it holds that $|x| > |y|$ implies $|f(x)| > |f(y)|$.

2.2.2 Notions from Average-Case Complexity Theory

We state here the basic definitions from average case complexity theory that will be used throughout this paper. These are the original definitions used by Levin [8]. For a comprehensive survey on average case complexity, see [3].

Definition 2.0.4 (probability distribution function)**.** *A function $\mu : \{0,1\}^* \to [0,1]$ is a* probability distribution function *if $\mu(x) \geq 0$ for every x and $\sum_{x \in \{0,1\}^*} \mu(x) = 1$. The* accumulative probability function *associated with μ is denoted $\overline{\mu}$ and defined by $\overline{\mu}(x) = \sum_{x' \leq x} \mu(x')$.*

Definition 2.0.5 (P-computable probability distribution)**.** *A* probability distribution function *μ is* P-computable *if there exists a polynomial time algorithm that given x outputs the binary expansion of $\overline{\mu}(x) = \sum_{x' \leq x} \mu(x')$.*

Definition 2.0.6 (distributional problem)**.** *A* distributional problem *is a pair consisting of a decision problem and a probability distribution function. That is, (L, μ) is the distributional problem of deciding membership in the set L with respect to the probability distribution μ.*

Definition 2.0.7 (dist$\mathcal{NP}$)**.** *The class* dist$\mathcal{NP}$ *consists of all distributional problems (L, μ) such that $L \in \mathcal{NP}$ and μ is P-computable.*

Definition 2.0.8 (average-case preserving reduction)**.** *A function f is an* average-case preserving reduction *(abbreviated AP-reduction) of the distributional problem (S, μ_S) to the distributional problem (T, μ_T) if f is a Karp-reduction (i.e. many-to-one polynomial-time reduction) from S to T, and in addition there exists a polynomial q such that for every $y \in \{0,1\}^*$,*

$$\mu_T(y) \geq \frac{1}{q(|y|)} \cdot \sum_{x \in f^{-1}(y)} \mu_S(x).$$

In the special case that f is 1-1, which is the case will be used throughout this paper, the last expression simplifies to the following: For every x it holds that

$$\mu_T(f(x)) \geq \frac{\mu_S(x)}{q(|x|)}.$$

Note that we use the fact that $|f(x)|$ is polynomially related to $|x|$.

AP-reductions preserve “easiness on the average” with respect to various definitions. The reason is that the sum of the probabilities of the preimages of every instance (in the range of the reduction), is not much larger than the probability of the instance itself. Thus, an AP-reduction cannot map “typical” instances of the original problem to “rare” instances of the target problem, on which an “average-case algorithm” can perform exceptionally bad. We note that AP-reductions are also transitive.

Definition 2.0.9 (dist$\mathcal{NP}$-complete distributional problem)**.** *A distributional problem is* dist$\mathcal{NP}$*-complete if it is in* dist$\mathcal{NP}$ *and every problem in* dist$\mathcal{NP}$ *is AP-reducible to it.*

For sake of completeness, we state here the definition of avg$\mathcal{P}$. However, our results only refer to AP-reductions, and not to their particular effect on avg$\mathcal{P}$.

Definition 2.0.10 (avg$\mathcal{P}$)**.** *The class* avg$\mathcal{P}$ *consists of all distributional problems* (L, μ) *such that there exists an algorithm A that decides L and a constant $\lambda > 0$ such that*

$$\sum_{x \in \{0,1\}^*} \mu(x) \cdot \frac{t_A(x)^\lambda}{|x|} < \infty$$

where $t_A(x)$ denotes the running time of A on input x.

For a discussion on the motivation for this somewhat non-intuitive definition see [2].

As mentioned above, it can be shown that if (T, μ_T) is AP-reducible to (S, μ_S) and $(S, \mu_S) \in$ avg$\mathcal{P}$ then $(T, \mu_T) \in$ avg$\mathcal{P}$ too. Thus, a dist$\mathcal{NP}$-complete problem is in avg$\mathcal{P}$ if and only if dist$\mathcal{NP} \subseteq$ avg$\mathcal{P}$. But, as mentioned above, AP-reductions preserve other definitions of "easiness on the average" too.

2.3 Main Results

We show here a sufficient condition for the existence of a dist$\mathcal{NP}$-complete version for an $\mathcal{NP}$-complete decision problem. This condition is very easy to verify for known $\mathcal{NP}$-complete decision problems. To justify this claim, we then show that some famous $\mathcal{NP}$-complete decision problems meet this condition. By doing so we wish to give evidence to our belief that *all* known $\mathcal{NP}$-complete decision problems meet this condition (or, at least have some reasonable encoding such that they do). For a discussion on the generality of our results, see Section 2.4.1.

We mention that our results can be easily modified to hold with respect to slightly different definitional variants, like those of [3] (which deal with probability ensembles rather than one probability distribution over all strings).

2.3.1 The General Technique

Our first technical tool is the following notion of paddability. Its purpose is to transform general reductions into length-regular ones, by "padding-up" instances to specified lengths.

Definition 2.0.11 (regular-padding)**.** *A decision problem L is* regular-paddable *if there exists some strictly increasing function q and a padding function $S : 1^* \times \Sigma^* \mapsto \Sigma^*$ such that:*

- *S is polynomial-time computable.*
- ***Preserving characteristic:** For every x and every n it holds that $S(1^n, x) \in L$ if and only if $x \in L$.*
- ***Length-regular**[5]**:** For every x and every n such that $n \geq |x|$, it holds that $|S(1^n, x)| = q(n)$.*

We call q the *stretch measure* of S. The first parameter of S determines the length to which the string is to be padded. The following holds:

Lemma 2.0.1. *If some decision problem is regular-paddable, then every Karp-reduction to it can be made length-regular.*

Proof. We show this by "pumping up" the lengths of all mapped strings. Let L be regular-paddable via S. Given a Karp-reduction f to L, we choose a strictly increasing polynomial r such that $r(|x|) \geq |f(x)|$ for every x, and define $f'(x) = S(1^{r(|x|)}, f(x))$. One can easily verify that f' is length-regular. □

Our main technical tool is the following notion of paddability.

Definition 2.0.12 (monotonous padding)**.** *A decision problem L is* monotonously-paddable *if there exists a padding function $E : \Sigma^* \times \Sigma^* \mapsto \Sigma^*$ and a decoding function $D : \mathbb{N} \times \Sigma^* \mapsto \Sigma^*$ such that:*

- *E and D are polynomial-time computable.*
- ***Preserving characteristic:** For every $p, x \in \{0,1\}^*$ it holds that $E(p, x) \in L$ if and only if $x \in L$.*
- ***Semi-monotonous:** If $p < p'$, $|p| = |p'|$ and $|x| = |x'|$ then $E(p, x) < E(p', x')$.*
- ***Length-regular:** If $|x| = |x'|$ and $|p| = |p'|$ then $|E(p, x)| = |E(p', x')|$, and if $|x| < |x'|$ and $|p| \leq |p'|$ then $|E(p, x)| < |E(p', x')|$.*
- ***Decoding:** For every $x, p \in \{0,1\}^*$ it holds that $D(|p|, E(p, x)) = p$, and $D(k, w) = \perp$ if there is no x and p such that $|p| = k$ and $E(p, x) = w$.*

[5] Since this condition indeed resembles Definition 2.0.2, we allowed ourself this abuse of the term here and in the following definition.

Loosely speaking, the first parameter for D defines the part of the string to be regarded as the "padding". Note that D is well-defined, that is, if $D(k, w) \neq \perp$ then there exists a unique p such that $D(k, w) = p$ (i.e. a unique $p \in \{0,1\}^k$ such that there exists $x \in \{0,1\}^*$ such that $E(p, x) = w$). Although Definition 2.0.12 may seem somewhat cumbersome, the following holds:

Fact 2.1. *If the function E is defined such that $E(p, x) = e_{p_1} e_{p_2} \ldots e_{p_{|p|}} g(x)$, where:*

- $|e_0| = |e_1|$ *and* $e_0 < e_1$.
- *The function $g(x)$ is length-regular.*
- $E(p, x) \in L$ *if and only if* $x \in L$.

then E is a monotonous padding function for L.

In the example of SAT (used in the introduction), the function g is the identity function, but generally, the encoding does not necessarily only add some prefix to the string, it can also change the string in some simple way (for example, in the example of SAT the function g could also change the indexes of the variables in the original formula).

It is easy to see that famous $\mathcal{NP}$-complete decision problems are both regular-paddable and monotonously-paddable. For details see Sections 2.3.2, 2.3.3 and 2.3.4.

Theorem 2.2. *If L is $\mathcal{NP}$-complete, regular-paddable and monotonously-paddable then there is a distribution that when coupled with L forms a $\mathrm{dist}\mathcal{NP}$-complete problem.*

Proof. We use the following result of Levin [8]:

Theorem 2.3. *There exists a $\mathrm{dist}\mathcal{NP}$-complete distributional problem.*

Let (C, μ) be a $\mathrm{dist}\mathcal{NP}$-complete distributional problem (where $C \in \mathcal{NP}$ and μ is P-computable), let h be a Karp-reduction from C to L, and let E, D be as in Definitions 2.0.11 and 2.0.12. In order for E to "work properly" (that is, to yield a length-regular, semi-monotonous reduction), it has to be composed (in the appropriate manner), with a length-regular Karp-reduction. Thus, using Lemma 2.0.1, we transform h to a length-regular Karp-reduction h' of C to L. We then define $f(x) = E(x, h'(x))$. We notice that f enjoys the following properties:

- f is a Karp-reduction from C to L (since E preserves characteristic).
- f is length-regular (since h' and E are both length-regular).
- f is semi-monotonous (since h' is length-regular and E is semi-monotonous).

- f is P-invertible (see next).

P-invertibility is evidenced by the following algorithm: Given y it first tries to find a number k such that $|f(0^k)| = |y|$ (this can be done, e.g., by computing $|f(0)|, |f(0^2)|, \ldots, |f(0^{|y|})|$, capitalizing on $|f(x)| \geq |x|$, which follows from length-regularity). If no such k exists it returns $\perp$. Else, it computes $x = D(k, y)$. If $x = \perp$, the algorithm also returns $\perp$. Otherwise, it computes $f(x)$. If the result equals y it returns x, else it returns $\perp$.

Note that since f is length-regular and semi-monotonous, f is monotonous over all strings. Next, we couple the decision problem L with the following probability distribution η:

$$\eta(y) = \begin{cases} \mu(f^{-1}(y)) & \text{if } y \in \mathrm{image}(f) \\ 0 & \text{otherwise} \end{cases}$$

It is straightforward that indeed η is a probability distribution function. We claim that the reduction f is a AP-reduction from (C, μ) to the distributional problem (L, η), and that η is P-computable. Since AP-reductions are transitive, the theorem follows. The first claim is straightforward, since every instance of C is mapped to an instance of L of exactly the same probability (i.e., $\mu(x) = \eta(f(x))$). To see the second claim, recall that μ is P-computable, and note that the accumulative probability function induced by η, denoted $\overline{\eta}$, satisfies:

$$\overline{\eta}(y) = \overline{\mu}(x) \text{ where } x \text{ is the largest string such that } f(x) \leq y \tag{2.3}$$

where $\overline{\mu}$ is the accumulative probability function induced by μ. We elaborate. Suppose, as an intermediate step, that we wish to compute $\eta(x)$ rather then $\overline{\eta}(x)$. Then we can simply compute $y = f^{-1}(x)$ (which can be done since f is P-invertible), then if $y = \perp$ we output 0, otherwise we output $\mu(y)$. Hence, the mere fact that f is P-invertible is sufficient to compute $\eta(x)$. Turning to the task of computing $\overline{\eta}$, we notice that since f is also monotonous (over all strings), for any string x in $\mathrm{image}(f)$, it holds that $\overline{\eta}(x) = \overline{\mu}(f^{-1}(x))$. For any other string, its accumulative probability is equal to that of the largest string in $\mathrm{image}(f)$ that is smaller than it (since all strings between them occur with probability 0). Equation 2.3 follows.

The string $\max(\{x|f(x) \leq y\})$ can be computed in polynomial time, since the reduction f is monotonous. An algorithm for computing this string can first compute $x = f^{-1}(y)$. If $x \neq \perp$ it outputs x, else it performs a binary search to find the string x' such that $f(x') < y$ and $f(x' + 1) > y$, and outputs x'. We mention that in fact, P-invertibility is implied by monotonicity (via a binary search), and is mentioned explicitly here only for clarity. □

Using Theorem 2.2 we now turn to prove that some $\mathcal{NP}$-complete decision problems have $\mathrm{dist}\mathcal{NP}$-complete distributional versions. We have verified that all twenty-one $\mathcal{NP}$-complete decision problems that are treated in Karp's paper [7] do meet the sufficient condition of Theorem 2.2. In the rest of this section we describe three of them. The first one is SAT, which we chose since it is the most canonical $\mathcal{NP}$-complete decision problem. We then show that CLIQUE meets this condition, as an example of a typical graph problem. Finally we provide a proof that the same is true for HAM, the problem of Hamiltonian cycle, since this proof is a little less straightforward than the other problems in Karp's paper. We believe that these three examples in particular, and the fact that same results hold for all $\mathcal{NP}$-complete decision problems in Karp's paper, give strong evidence that the results hold for all known $\mathcal{NP}$-complete decision problems (see further discussion in Section 2.4.1, as well as further evidence for our claim).

2.3.2 SAT, Revisited

The following theorem can be proved using Theorem 2.2.

Theorem 2.4. SAT *has a distributional version that is* $\mathrm{dist}\mathcal{NP}$*-complete.*

Proof. To show that SAT meets the hypotheses of Theorem 2.2 one can use similar ideas to those presented in the introduction. We just have to assume some assumptions on the standard encoding of SAT (e.g. that the encoding acts on each clause, and each variable in the clause, in a context-free manner).

We choose two strings e_0, e_1 such that both are encodings of CNF clauses such that:

1. Both clauses are tautologies.
2. $e_0 < e_1$.
3. $|e_0| = |e_1|$.

We first sketch the ideas used to show SAT is regular-paddable. In order to ”stretch” some formula ϕ we ”pad-up” ϕ by prefixing it with a series of e_0's. Since the added clauses are tautologies, the padding function does not affect the characteristic of ϕ.[6]

Following the ideas in the introduction, by using e_0 and e_1 to encode 0's and 1's, one can show that SAT is also monotonously-paddable. If it is required that instances of SAT do not have multiple occurrences of the same clause, then this requirement can be met by allocating sufficient amount of variables for the padding (i.e., using for the padding variables that are disjoint to the ones used in the original formula), and using different variables for each clause in the padding.

[6] There are some small technicalities to be concerned, like assuring that $|e_0|$ divides the difference between the desired length and the length of the initial formula. However, there are various ways of coping with such difficulties, e.g., by using various e_0's with different lengths, and by ”normalizing” the lengths of the variables in ϕ (see next).

We did not define here rigorously the encoding of SAT (e.g., how is a variable encoded, how is a clause encoded, etc). Different encodings will yield different padding functions. However, Theorem 2.4 can be proved under any reasonable encoding of SAT.

□

2.3.3 Clique

We consider the CLIQUE decision problem, consisting of all pairs of an undirected graph G and a natural number k such that there exists a complete induced subgraph of G of size k. We assume the graph is given as an incidence matrix (which can either be symmetric, or upper-triangular), that the first row of the matrix is encoded by the leftmost bits, and that k is represented as a $\lceil \log(n) \rceil$-bit number to the right of the matrix.

Theorem 2.5. CLIQUE *has a distributional version that is* dist$\mathcal{NP}$*-complete.*

Proof. It is straightforward to see that CLIQUE is regular-paddable. We simply add ”dummy” nodes with degree 0 and leave k as is. Thus we can transform any input of size $n^2 + \lceil \log(n) \rceil$ to an input of size $m^2 + \lceil \log(m) \rceil$ for any $m \geq n$, and thus we can achieve a regular-padding function with stretch measure $q(n) = n^2 + \lceil \log n \rceil$.

We now show that CLIQUE is monotonously-paddable. The idea is as follows. Given a graph $G = (V, E)$ where $V = \left\{ v_1, v_2, \ldots, v_{|V|} \right\}$ to be padded with p, we first ”shift” all vertices by raising their index by $|p|$, the number of bits we wish to encode (and of course change the edges accordingly). This ”frees” the vertices indexed lower or equal to $|p|$. We then encode the bits of p by edges connected to v_1, such that each bit is encoded by the edge indexed as the bit's position, and such that the edge will appear if and only if the bit value is 1 (that is, the edge (v_1, v_i) will appear if and only if $p_i = 1$). Thus, these edges will result in 0's and 1's in the first row of the incidence matrix of the graph. This will add a star-shaped subgraph (rooted at v_1) to the original graph. We will ensure that this will not change the characteristic of the instance.

Formally, for M, an 0-1-matrix of size $n \times n$ we define $E(p, (M, k)) = (M', k)$ where M' is an 0-1-matrix of size $(n + |p|) \times (n + |p|)$, such that $M'_{i+|p|, j+|p|} = M_{i,j}$ for $1 \leq i, j \leq n$, and $M'_{1,j} = p_j$ for $1 \leq j \leq |p|$ (where p_j is the j-th bit of p). This, of course, may add to the graph cliques of size 2 (but does not add larger cliques). If $k = 2$ and the graph did not have a clique of size 2 (i.e., the graph was edgeless), this could be a problem. To fix this, the padding function can check (in polynomial time) if indeed $k = 2$ and the graph is edgeless. If this is the case, it simply changes k to 3 and we are done.

This transformation preserves the characteristic of the instance. Moreover, we have p encoded in the most trivial manner, i.e. bit-by-bit, as a prefix of the string $E(p, x)$. It is straightforward to see that E meets all the conditions of a monotonous padding function.[7] $\square$

2.3.4 Hamiltonian Cycle

We consider the Hamiltonian Cycle decision problem, denoted HAM. The Hamiltonian Cycle decision problem consists of all undirected graphs that have a simple cycle that contains all nodes of the graph. We assume the graph is given as an incidence matrix (which can either be symmetric, or upper-triangular), and that the first row of the matrix is encoded at the beginning of the string.

Theorem 2.6. HAM *has a distributional version that is* $\mathrm{dist}\mathcal{NP}$*-complete.*

Proof. We first show that HAM is regular-paddable. We do so by showing that any graph over n nodes can be transformed in polynomial time into a graph over $n+k$ nodes for any $k \geq 2$ such that preserves Hamiltonianicity (thus achieving regular-padding with stretch measure $(n+k)^2$). Given a graph $G = (V, E)$ where $V = \{v_1, v_2, \ldots, v_n\}$, and $k \geq 2$ we define $G' = (V', E')$ where $|V'| = n + k$. Intuitively, we are going to "split" v_n (an arbitrary choice) into $k+1$ nodes, denoted $v_n, v_{n+1}, \ldots, v_{n+k}$, and "force" any Hamiltonian cycle to regard them as one node, that is, any Hamiltonian cycle in the new graph will have to contain the sub-path[8] $v_n, v_{n+1}, \ldots, v_{n+k}$ (see Figure 2.1). The idea is as follows: after adding the mentioned nodes to the graph, we connect them such as to form the path mentioned above. We then connect the last node, v_{n+k}, to all the nodes connected to v_n. Formally:

$$E' = E \cup \{(v_{n+i}, v_{n+i+1}) | 0 \leq i \leq k-1\} \cup \{(u, v_{n+k}) | (u, v_n) \in E\} .$$

We show that indeed this transformation preserves Hamiltonianicity. For every Hamiltonian cycle x, v_n, y in G (where x and y are sub-paths), the path $x, v_n, v_{n+1}, \ldots, v_{n+k}, y$ is a Hamiltonian cycle in G' (where (v_{n+k}, y) is a sub-path in G' because (v_n, y) is a sub-path in G, and due to the construction). On the other hand, for any Hamiltonian cycle in G', in order to reach the nodes $v_{n+1}, v_{n+2}, \ldots, v_{n+k-1}$, it has to be of the form $x, v_n, v_{n+1}, \ldots, v_{n+k-1}, v_{n+k}, y$, which implies that x, v_n, y is a Hamiltonian cycle in G.

[7] We note that any other reasonable encoding can be shown do yield the same result. For example, if k was encoded to the left of the matrix, the constructed reduction here would not be monotonous, since the "encoding row" would not be added at the beginning of the string. To fix this, the function E could fix k for every length of x, thus disabling its effect on the lexicographical order: Given an input (M, k) where M is an $n \times n$ matrix, the reduction would transform it to a matrix M' of size $2n \times 2n$, then add a clique of size $n - k$ to the original graph, using the added nodes, and connect all nodes of the added clique to all nodes of the original graph. The reduction would then generate the instance (M', n), which has the same characteristic as (M, k).

[8] Or its reverse. Since the graph is undirected, we regard the cycles as undirected too.

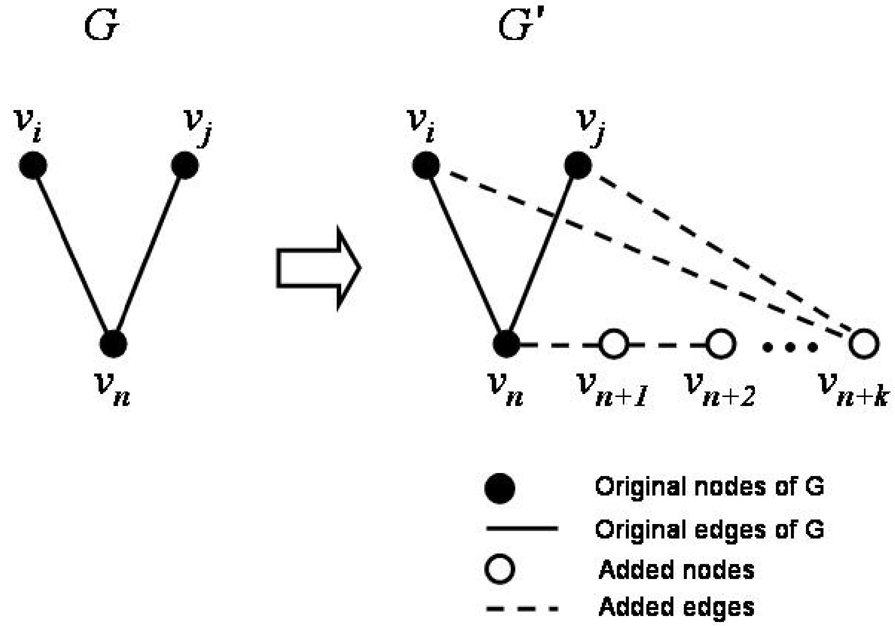

Figure 2.1: The construction of regular padding.

We turn to show that HAM is monotonously-paddable. The idea is similar to the regular-padding described above. In order to pad some (incidence matrix of a) graph with the string p, we first add a path of length roughly $|p|$ to the graph, similarly to the regular-padding described above (see Figure 2.2). We then encode the bits of p by adding edges within the path. We do so such that the path will still have to be taken in the natural order of the nodes, and such that the added edges will be encoded in the prefix, i.e. first row, of the incidence matrix. In order to achieve the later goal, we ”shift” the nodes of the graph by raising their indexes, thus enabling the added nodes to posses the smallest indexes, and then replace v_1 by the path $v_1, v_2, \ldots, v_{|p|+2}$, similarly to the construction of the regular-padding. We then encode the bits of the padding such that the edge (v_1, v_3) encodes the first bit (i.e., existence of the edge encodes a ’1’, and non-existence encodes a ’0’), the edge (v_1, v_4) encodes the second bit and so on. We skip (v_1, v_2) since this edge is necessary in order for the aforementioned path to exist (i.e., this edge will anyway exist). This construction ensures that the added path will still have to be taken in its natural order in any Hamiltonian cycle. We describe the construction formally. For the $n \times n$ incidence matrix M of the graph $G = (V, E)$, we define $E(p, M) = M'$ where M' is the incidence matrix of size $(n + |p| + 1) \times (n + |p| + 1)$ of the graph $G' = (V', E')$ where $|V'| = |V| + |p| + 1$ and

$$E' = \Big\{(v_{i+|p|+1}, v_{j+|p|+1})|(v_i, v_j) \in E\Big\} \cup \Big\{(v_1, v_{i+|p|+1})|(v_1, v_i) \in E\Big\} \cup$$

$$\{(v_i, v_{i+1})|1 \leq i \leq |p| + 1\} \cup \{(v_1, v_{i+2})|p_i = 1\}$$

(where p_i is the i-th bit of p). The first set in the union above is the original graph, with its nodes "shifted" by raising their indexes by $|p| + 2$. The second set connects v_1 to all the nodes $v_{|p|+3}$ is connected to (which are the nodes v_1 of the original graph was connected to, with their indexes "shifted"). The third set forms the path $v_1, v_2, \ldots, v_{|p|+3}$. Finally, the last set encodes p by edges connected to v_1 (thus the bits representing them will be encoded in the first row of the matrix).

We observe that every node that was connected in the original graph to v_1 is now connected to both v_1 and $v_{|p|+3}$, and that $v_1, v_2, \ldots, v_{|p|+3}$ is a path in the new graph. Setting the bits in the diagonal all to zeros[9], the resulting encoding of M' starts with '01', followed by the bits of p. Using similar argument to the one used to prove that the regular-padding preserves Hamiltonianicity, one can verify that indeed such transformation preserves Hamiltonianicity too (in particular, note that any Hamiltonian cycle in G', in order to meet $v_{|p|+2}$, has to contain the sequence $v_1, v_2, \ldots, v_{|p|+3}$ or its reverse). Again, it is straightforward to see that E meets all the conditions of a monotonous padding function. □

We note that the similar decision problem of Parameterized Hamiltonian Cycle, which consists of all couples of a graph and a natural number k such that there is a simple cycle over k nodes in the graph, is easier to be shown to have a $\mathrm{dist}\mathcal{NP}$-complete version. The same is true for the similar decision problem over *directed* graphs.

2.4 Conclusions

We discuss some issues related to the results presented in this work. In Section 2.4.1 we discuss the possibility of generalizing our results to all decision problems in $\mathcal{NP}$, and show that it seems that this cannot be achieved using our techniques. On the other hand, we provide further evidence to our claim that all *known* $\mathcal{NP}$-complete problems have distributional versions which are $\mathrm{dist}\mathcal{NP}$-complete. In Section 2.4.2 we show that our results hold also with respect

[9] The bits in the diagonal are meaningless.

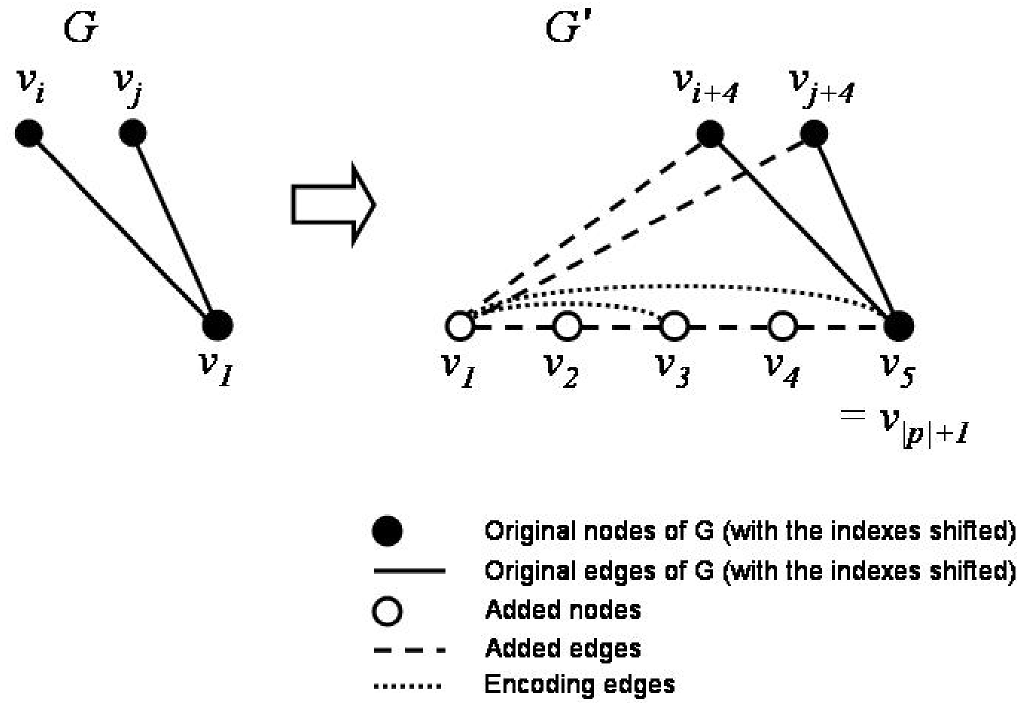

Figure 2.2: The construction of monotonous padding with $p =' 101'$.

to different notions of ”easiness on average”. In Section 2.4.3 we suggest a possible interpretation of our results, and finally, in Section 2.4.4 we discuss some directions for improving our results.

2.4.1 On the Generality of Our Results

A natural question arises regarding our results. Since apparently, for all known $\mathcal{NP}$-complete decision problems we can prove that our result hold, can we expect to prove (using our techniques) that our result holds for *all* $\mathcal{NP}$-complete decision problems? Apparently the answer is negative. Recall that in order to prove that some $\mathcal{NP}$-complete decision problem has a distributional version that is $\mathrm{dist}\mathcal{NP}$-complete, our technique involves proving that this problem is paddable in some natural sense. However, proving that all $\mathcal{NP}$-complete problems are paddabale involves, in particular, proving that all $\mathcal{NP}$-complete problems are infinite. But such a proof would imply $\mathcal{P} \neq \mathcal{NP}$ (because if $\mathcal{P} = \mathcal{NP}$ then any non-empty finite set is $\mathcal{NP}$-complete). For this reason we cannot hope to do better than prove these results for all *known* $\mathcal{NP}$-complete decision problems.

The same phenomena occurs with respect to the *isomorphism conjecture* of Berman and Hartmanis [4]. This conjecture states that every two $\mathcal{NP}$-complete decision problems are related via a 1-1, onto, polynomial-time, and polynomial-time invertible Karp-reduction. Berman and Hartmanis showed that every two $\mathcal{NP}$-complete decision problems that are paddable in some simple manner, are related via such a reduction. They observed that the paddability condition holds for numerous $\mathcal{NP}$-complete decision problems and concluded that these problems are pairwise isomorphic. They conjectured that the same is true for *all* $\mathcal{NP}$-complete decision problems. Thus, both their result and ours build on some paddability properties that are very easy to verify for given $\mathcal{NP}$-complete decision problems, but that are very hard to prove for *all* $\mathcal{NP}$-complete decision problems.

The paddability condition required for Berman and Hartmanis's result is slightly weaker than ours. Loosely speaking, they do not require monotonicity. However, typically their padding functions, as ours, encode bit by bit. Thus, ensuring that the padding is added at the beginning of the string, and that the encoding of 1 is larger than the encoding of 0, results in a monotonous padding function. To date, no counter-example to Berman and Hartmanis's isomorphism conjecture was found. Furthermore, one can show that their paddability condition is not only sufficient, but is also necessary. That is: If a decision problem is isomorphic to SAT, then it is paddable in the sense they define[10]. Thus, the fact that no counter-example to Berman and Hartmanis's isomorphism conjecture was found, implies that no non-paddable decision problem was found. We believe that, although our condition is slightly stronger than Berman and Hartmanis's notion of paddability, this gives a strong evidence that all known $\mathcal{NP}$-complete problems meet our condition too.

2.4.2 The Extension of Our Results to Different Definitions

A different notion of solvability can be considered. While in our definition of $\mathrm{avg}\mathcal{P}$ we require that the algorithm solves the decision problem for all instances, one can consider a relaxation such that the algorithm runs in polynomial time, but solves "almost all instances", that is, for

[10] Let f be a 1-1, onto, polynomial-time, and polynomial-time invertible Karp-reduction of $L \in \mathcal{NPC}$ to SAT. Then in order to pad the instance x of the problem L with p, we first pad $f(x)$ with p using the padding function of SAT. Denote the result by w. We then compute $f^{-1}(w)$ to obtain the padded instance. In order to retrieve the padding from some string y, the decoding function first computes $f(y)$, and then uses the decoding function of the padding function of SAT to retrieve the padding from $f(y)$. Finally, one has to show that this padding function is also length-increasing. This can be achieved, e.g., by modifying the padding function of SAT to pad, instead of p, a longer string, such as p concatenated to itself a polynomial number of times.

every polynomial p and for every n the probability that given an input of length n the algorithm errs on it is upper bounded by $\frac{1}{p(n)}$. (Note that in the former definition, the algorithm, although is permitted to run in super-polynomial time for a negligible fraction of the instances, has to be correct on all instances.) Since AP-reductions preserve "easiness on the average" also under this definition of solvability, our results, which refer to AP-reductions, hold also under this definition.

As mentioned above, our results can be easily modified as to hold for definitional variants that relate to *probability ensembles* rather than one probability distribution over all instances (see [3] for the definitions).

2.4.3 A Possible Interpretation of Our Results

As stated in the introduction, we believe that average-case hardness results, when viewed as "negative" results (i.e., as results evidencing our inability to solve problems on the average), are of more interest when shown with respect to more "simple", or "natural" probability distributions. Arguably, such distributions are more likely to occur in "real life". Furthermore, from the theoretical point of view, hardness under such distributions indicate that the hardness of the complete problem is not a property of some "esoteric" instances, but rather is "inherent in the decision problem at large". The last claim becomes stronger when "simple" is regarded also as "close to uniform".

Let us analyze the structure of the probability distribution of the complete distributional version of SAT implied by the construction presented in the Introduction. (Typically, applying our technique to other $\mathcal{NP}$-complete decision problems yields similarly structured distributions.) The complete problem guaranteed by [8] has a probability distribution that is "close to uniform" in a very strong sense. For simplicity, let us assume we take a complete problem with uniform probability distribution. Combining Equations 2.1 and 2.2, the left side of η is uniform over encodings of all strings (under some simple encoding). The right side is determined by the left, and can be computed in polynomial time from it. Thus, the structure of the resulting probability distribution is a simple structure from the computational point of view. From the combinatorial point of view, however, the distribution does not seem to have a simple interpretation.

A possible interpretation of our result is the following. While in the past it was believed that the constraint of P-computability was too strong, and that for this reason so few average case $\mathcal{NP}$-complete problems were found, our work, in some sense, shows the contrary: we show that in fact a very wide family of known $\mathcal{NP}$-complete decision problems have average case complete versions, but yet, our constructions yield distributions that does not look as "natural" as one could expect, and in this sense, the family of P-computable distributions is in fact not restrictive enough.

2.4.4 Further Directions

Two natural suggestions for improvement arise. The first is to show (conditional) hardness of natural $\mathcal{NP}$-complete problems under distributions that are natural *with respect to the specific combinatorial structure of these problems* (for example, random graphs where the edges are drawn independently with equal probability). Such distributions relate to the underlying combinatorial structure of the decision problem, rather than to the strings representing the objects. However, by nature, such results are not generic, and have to be proven independently for each problem.

The second, is to show generic results (i.e., that relate to a very wide family of decision problems), for families of distributions that are defined by a more restrictive condition than Levin's P-computability, and hope that such families will yield more "simple" distributions.

Acknowledgements

Deep thanks to my advisor, Oded Goldreich, for many helpful discussions, for his advice in presenting the ideas in this paper, and for his clear surveys and lectures that presented me the topics treated in this paper.

Bibliography

[1] S. Ben-David, B. Chor, and O. Goldreich. On the theory of average case complexity. In *STOC '89: Proceedings of the twenty-first annual ACM symposium on Theory of computing*, pages 204–216, New York, NY, USA, 1989. ACM Press.

[2] O. Goldreich. Notes on levin's theory of average-case complexity. Technical Report TR97-058, ECCC, 1997.

[3] O. Goldreich. Computational complexity: A conceptual perspective, draft of a book, 2006. Unpublished manuscript, available from `http://www.wisdom.weizmann.ac.il/∼oded/cc-book.html`.

[4] J. Hartmanis and L. Berman. On isomorphisms and density of np and other complete sets. In *STOC '76: Proceedings of the eighth annual ACM symposium on Theory of computing*, pages 30–40, New York, NY, USA, 1976. ACM Press.

[5] J. Hartmanis and R.E. Stearns. On the computational complexity of algorithms. *Transactions of the AMS*, 117:285–306, 1965.

[6] R. Impagliazzo and L.A. Levin. No better ways to generate hard np instances than picking uniformly at random. In *Proc. of the 31st IEEE Symp. on Foundation of Computer Science*, pages 812–821, 1990.

[7] Richard M. Karp. Reducibility among combinatorial problems. In Raymond E. Miller and James W. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972.

[8] Leonid A Levin. Average case complete problems. *SIAM J. Comput.*, 15(1):285–286, 1986.

Chapter 3

On the Construction of One-Way Functions from Average Case Hardness[1]

Noam Livne

Abstract

In this paper we study the possibility of proving the existence of one-way functions based on average case hardness. It is well-known that if there exists a polynomial-time sampler that outputs instance-solution pairs such that the distribution on the instances is hard on average, then one-way functions exist. We study the possibility of constructing such a sampler based on the assumption that there exists a sampler that outputs only instances, where the distribution on the instances is hard on the average. Namely, we study the possibility of ”modifying” an ordinary sampler S that outputs (only) hard instances of some search problem R, to a sampler that outputs instance-solution pairs of the same search problem R. We show that under some restriction, not every sampler can be so modified. That is, we show that if some hard problem with certain properties exists (which, in particular is implied by the existence of one-way permutations), then there exists a polynomial-time sampler S such that for every polynomial λ, there exists a search problem R such that (1) R is hard under the distribution induced by S, and (2) there is no sampler S^* with randomness complexity bounded by λ, that outputs instance-solution pairs of R, where the distribution on the instances of S^* is closely related to that of S (i.e., dominates it). A possible interpretation of our result is that a generic approach for transforming samplers of instances to samplers of instance-solution pairs cannot succeed.

[1] Appeared in *Innovations in Computer Science (ICS) 2010.*

3.1 Introduction

The existence of one-way functions is a necessary condition for nearly all cryptographic applications, and is sufficient for a large portion of them. Since establishing the existence of one-way functions does not seem reachable these days, as it implies that $\mathcal{P} \neq \mathcal{NP}$, a line of papers studied the possibility of proving the existence of one-way functions based on the assumption that $\mathcal{P} \neq \mathcal{NP}$ ([3],[4],[2],[1]). These works presented limitations on possible reductions of the security of one-way functions to $\mathcal{NP}$-hardness. Specifically, these works show that, under certain assumptions, certain reductions of this type do not exist.

The starting point of this paper is the observation that the transformations studied by these papers are actually supposed to overcome two gaps at once: (1) the gap between average-case hardness and worst-case hardness; and (2) the gap between one-way functions and average-case hardness, where ”average case hardness” refers to a search problem R and a sampler S such that the search problem R is hard on average under the distribution induced by S. Since the problem of basing average-case hardness on worst-case hardness is hard by itself, it is inviting to study the seemingly more modest task of basing one-way functions on average-case hardness.

The assumption that average case hardness exists (as defined above), implies the existence of objects (namely, R and S as above) one can hope to base a construction of a one-way function upon (while to the best of our understanding, the assumption that $\mathcal{P} \neq \mathcal{NP}$ does not seem to imply such objects). Indeed, this paper concentrates on the question of *constructing* a one-way function from average case hardness, rather than on *proving the security* of one-way functions based on average case hardness. Informally, our result gives some restriction on a certain approach to achieve this construction. In the following we describe our result more formally.

3.1.1 Main Result

Before describing our result, let us describe the approach we refer to above. How can one base the construction of a one-way function on the existence of average case hardness? As aforesaid, the assumption implies a search problem R and a polynomial-time sampler S where the search problem R is hard under the distribution induced by S. A natural way to go is to use the following known fact, stated informally (for more details see [5], Section 7.1.1.):

Observation 3.1. *If there exists a search problem R and a polynomial-time sampler S^* that outputs instance-solution pairs of R, such that the distribution of S^* restricted to the instances is hard on average, then one-way functions exist.*

Using this fact, one can try to construct a new sampler S^*, that "behaves" similarly to S, but instead of outputting only instances, S^* outputs instance-solution pairs (under R). That is, the distribution induced by S^*, when restricted to the instances only, is similar to the distribution of S. Thus, the sampler S^* "inherits" its hardness from S. In fact, in order for S^* to inherit the hardness of S, it is sufficient that the (restricted) distribution of S^* dominates that of S (see Section 3.2 for definitions). Upon constructing such S^*, one can use Observation 3.1 to construct the one-way function. Thus, this approach of basing one-way functions on average case hardness calls for the following challenge:

Desired Construction:
Given a search problem R and a polynomial-time sampler S, where R is hard under the distribution induced by S, construct a polynomial-time sampler S^* that outputs instance-solution pairs of R, such that the distribution of S^* restricted to the instances, dominates the distribution of S.

Informally, our main result is that for any fixed polynomial bounding the randomness complexity of S^*, the Desired Construction cannot always exist. That is, under plausible assumptions (which in particular are implied by the existence of one-way permutations) there exists a polynomial time sampler S such that for any polynomial λ there exists a search problem R such that R is hard under S, but where a sampler S^* as in the Desired Construction with randomness complexity bounded by λ, does not exist.

We note, that the search problem R we construct is polynomially bounded (i.e., the lengths of the solutions are polynomially bounded in the lengths of the instances), but not polynomial time verifiable. Rather, it can be verified in arbitrarily small super-polynomial time. This still leaves open the possibility that for any R and S, where R is polynomial time verifiable, the Desired Construction exists (even for some universal polynomial randomness complexity of S^*). However, we note that in order for the Desired Construction to yield a one-way function, there is no need that R be polynomial time verifiable. Thus, one might hope that the construction exists also when R is not polynomial time verifiable. While we cannot eliminate this possibility, we can give arbitrarily strong polynomial lower bounds on the randomness complexity of S^* that achieves this construction. Interestingly, these lower bounds depend only on R. That is, the randomness required by S^* is not a result of the randomness used by S. (For a discussion on this issue see Section 3.4.2.)

We also note that placing no restrictions on R and S, the result is trivial (even without fixing the randomness complexity of S^*):

- If the lengths of the solutions in R are not polynomially bounded in the lengths of the instances, then trivially no such S^* exists simply because it cannot write the solutions in polynomial time.

- If the lengths of the solutions in R are polynomially bounded in the lengths of the instances, but R is super-exponentially (worst-case) hard, then again no such S^* exists: take S to be an ”onto sampler” (i.e., a sampler that outputs any instance with positive probability). Then the assumption that a sampler S^* as above exists implies that R can be solved in the worst case in exponential time, in contradiction: Given some instance x, one can find an instance-solution pair where the instance is x, by performing an exhaustive search on the randomness for S^* until it outputs an appropriate pair (notice that S^* outputs for every x a pair where the instance is x, since the distribution on the instances in S^* should dominate S).

While for the first case above such R and S can be constructed (simply by padding) from the basic assumption that hardness on average exists (as defined above), the second case seems to require a stronger assumption regarding the hardness on average.

3.1.2 Related Work

We mention an observation by Salil Vadhan [6] that is related to this work. Suppose there exists a unary set in $(\mathcal{NP} \cap \mathrm{co}\mathcal{NP}) \setminus \mathcal{P}$ (where by ”unary” we mean that all strings are in 1^*). Then there is a search problem R for which it is infeasible to generate instance-solution pairs although all instances have solutions.

Specifically, let L be a unary set in $(\mathcal{NP} \cap \mathrm{co}\mathcal{NP}) \setminus \mathcal{P}$ and let R_0 (respectively R_1) be an $\mathcal{NP}$-relation for L (respectively $\overline{L}$). Consider the $\mathcal{NP}$-relation $R = \{(x, (b, w)) : (1^{|x|}, w) \in R_b\}$. Then, all instances have solutions under R (as for any $x \in \{0,1\}^*$ it holds that either $1^{|x|} \in L$ or $1^{|x|} \in \overline{L}$), but any instance-solution sampler for R yields an algorithm for deciding L because any sample generated on input 1^n determines whether or not $1^n \in L$.

Thus, for such R the desired construction does not exist. We note the following corollary: if we strengthen the assumption and assume that L is hard on average (here there is no need to specify any distribution, because deciding L depends only on the length of the instances), we obtain a similar result to our result here. Specifically, take S to be any efficient sampler, then R is hard under S, but there is no S^* as above.

It is easy to see that the hypothesis used for this corollary (that there exists a unary set in $(\mathcal{NP} \cap \mathrm{co}\mathcal{NP}) \setminus \mathcal{P}$ that is hard on average) implies our hypothesis (see Theorem 3.4), while the opposite does not seem to be known. This makes our result stronger. Moreover, the hypothesis of the aforementioned corollary is rather non-standard. In contrast, our hypothesis is weaker than the existence of onto one-way functions, which is a standard assumption (and, in particular is implied by the existence of one-way permutations).

3.1.3 Technique

In the following we explain informally the idea of our proof. Suppose, as a mental experiment, that for some R and S there *does exist* S^* as in the Desired Construction. Then, although R is hard under S, using S^* one can sample *random* instance-solution pairs of R. Intuitively, given a specific instance one cannot find a solution, but one can always easily find a *random* instance together with a solution. Moreover, one can obtain instance-solution pairs that relate to arbitrary coin-vectors for S^* at his choice. In light of these observations, the basic idea in our proof is the following. Given some R', S', we construct R and S such that:

- If R' is hard under S' then R is hard under S (that is, the pair (R, S) "inherit" the hardness of the pair (R', S')). This is achieved by "embedding" R' in R and S' in S. That is, any instance of R' is embedded in some instance of R, and any solution for an instance x of R "embeds" a solution for the instance of R' that is embedded in x. Moreover, roughly speaking, S imposes the distribution of S' on the embedded instances of R'.

- If an S^* exists for (R, S) then, if $(x, y) \in R$, the solution y helps finding pairs $(\hat{x}, \hat{y})$ for $\hat{x}$'s that are "close" to x. That is, for any $\hat{x}$ that is a neighbour of x under some (Hamming-like) metric, the solution y contains a coin-vector for S^* that yield an instance-solution pair of the form $(\hat{x}, \hat{y})$. Thus, on input a specific instance x for which one wants to find a solution, through repeated invocations of S^*, one can start from an arbitrary instance-solution pair, and go through a sequence of pairs, where the instance in each pair is closer to x than the preceding one by one unit under the aforementioned metric, until reaching x. By choosing an appropriate coin-vector for S^* from the solution in each pair, one can indeed have the instance in the next pair closer to x.

Thus, the assumption that S^* exists for (R, S), yields that it is easy to find an instance-solution pair with respect to R, for any *specific* instance. It follows that R is easy in the worst case. Now, since R "embeds" R', it follows that R' is also easy in the worst case, in contradiction.

It should be noted, however, that in order for this idea to work, some technicalities need to be dealt with, and these yield some restrictions on R' and S' (but, as aforementioned, in particular, the assumption that one-way permutations exist yields R' and S' as required for our proof).

3.2 Preliminaries

Standard Notation and Conventions For $1 \leq i \leq \ell$ we define $e_i^\ell \triangleq 0^{i-1}10^{\ell-i}$. Given a TM M we denote by $\langle M \rangle$ the code of M. We say that a function f is *noticeable* if there exists a positive polynomial q such that for large enough n's, $f(n) \geq 1/q(n)$. Given a string x we denote by $|x|$ the length of x. We use interchangeably the terms *search problem* and *relation*. Given a search problem R we say that x *is a YES-instance of* R if x has a solution under R, that is, if there exists y such that $(x, y) \in R$. We say that a search problem is *total* if all strings are YES-instances of that problem. We say that a search problem is *polynomially bounded* if there exists some polynomial q such that for every y which is a solution for x, it holds that $|y| \leq q(|x|)$. When defining strings in the form $(\cdot)$, $(\cdot, \cdot)$ etc., we implicitly assume some 1-1, onto, efficient, efficiently invertible encoding[2] from $\bigcup_{n \in \mathbb{N}} (\Sigma^*)^n$ to Σ^*. We will use the term *support* in relation to an ensemble of random variables to denote the sequence of supports of the variables in the ensemble.

In our proof we would like to enumerate all samplers. However, since any reasonable definition of a sampler assumes some structural properties (for example, to the least the requirement that it always halt), it is not clear how one can enumerate all samplers. Instead, we enumerate a superset of the set of samplers, which we call *potential samplers*, which are basically just probabilistic TM's.

Definition 3.1.1 (Potential Sampler). *A potential sampler is a TM with 2 input tapes. The input to the first tape is simply called "input", and the input to the second tape is called "randomness".*

We implicitly assume some enumeration on all such machines. We explain how we interpret a potential sampler. The run of a potential sampler on some input, when not stating explicitly the randomness, is defined as a random variable (ranging over all strings plus a special "not halting" symbol), which is the result of running the machine on a random infinite vector of coins (equivalently, one can think of the machine as flipping coins on the fly). We define the output of a potential sampler given some input and some *explicit* randomness as the output on that input and randomness, and for sake of completeness, we define the output in the case the sampler requires more coins than the explicit randomness, as the failure symbol $\perp$.

Definition 3.1.2 (Sampler). *A sampler is a potential sampler that on input 1^n, with probability 1 halts and outputs a string of length n.*

We explain how we interpret a sampler. Typically we will run a sampler on inputs of the form 1^n (which will be referred to as "input"). In such a case, for a sampler S, the term $S(1^n)$ will be a random variable with values from Σ^n. Thus, any sampler S defines an ensemble of

[2] For simplicity we assume $|(x, y)| = |x| + |y|$. While this is not the case, it will make the presentation clearer. Clearly, this technicality can be handled, without any essential change in the statements and proofs.

random variables $\{S(1^n)\}_{n\in\mathbb{N}}$. When needed, we will relate explicitly to the randomness as (a second) input to the sampler.

We note, that our rigorous definition of a sampler is for sake of formality of the proof, and is arbitrary. To the best of our knowledge, if there exists a sampler under any "reasonable" definition that generates hard instances of R, then there exists a sampler under our definition that generates hard instances of R (see definition of hardness below).

Definition 3.1.3 (Onto Sampler)**.** *A sampler is an onto sampler if it outputs every string with positive probability. Formally, a sampler is an onto sampler if its support is* $\{\{0,1\}^n\}_{n\in\mathbb{N}}$.

Definition 3.1.4. (Randomness Complexity of a Potential Sampler) *We say that the function f is the randomness complexity of some potential sampler, if on input 1^n, the maximal number of coins it uses is $f(n)$.*

We also define the following transformation, that transforms any machine M that output pairs of strings, to a machine $\tilde{M}$ such that on any input and randomness for which M outputs a pair, $\tilde{M}$ outputs only the first element in the pair:
Unpairing Transformation:
On input $\langle M\rangle$ (a code of a potential sampler), the transformation outputs the code of the following potential sampler $\tilde{M}$:
On input x and randomness r, the machine $\tilde{M}$ runs M on x with randomness r, if it halts it checks if the output is a pair, and if it is, it deletes the second element in the pair.

It is easy to see that the transformation is efficiently computable and that the running time of $\tilde{M}$ is linearly related to that of M. From now on, given a potential sampler M, we denote by $\tilde{M}$ the potential sampler who's code is the result of applying the transformation on $\langle M\rangle$.

Definition 3.1.5 (Pairs-Sampler)**.** *A pairs-sampler is a potential sampler M that on every input and randomness outputs a pair, and where $\tilde{M}$ is a sampler according to Definition 3.1.2 (that is, on input 1^n, with probability 1 M halts and outputs a pair where the first element is of length n).*

Throughout the paper we consider search problems that are not total, along with samplers that may not output only YES-instances (for these search problems). The following definition relates to this setting. It states that in the aforementioned setting, a potential solver A fails on a search problem R under the sampler S if when running A on random instances output by S, the event that a YES-instance is output and A does not respond with a correct solution, is noticeable. (Trivially, it implies that S must output YES-instances with noticeable probability.) We then say that R is hard under S if every potential probabilistic polynomial time solver A fails on R under S.

Definition 3.1.6. (Failure of a Solver, Hardness of a Search Problem Under a Sampler) *For an algorithm A, a search problem R and a sampler S we say that A* fails on R under S *if:*

$$\Pr_{x \leftarrow S(1^n)}[(x, A(x)) \notin R \wedge \exists y (x, y) \in R]$$

is noticeable (where the probability is taken both over the randomness of S, and the randomness of A).

We say that R is hard under S *if every probabilistic polynomial time algorithm fails on R under S.*

We note that one can alternatively define the failure of a solver, instead of failing with noticeable probability (on the aforementioned distribution), as not failing with negligible probability. The difference is that now we only require that the solver fail infinitely often, and not on every n. Our main result is true also under that definition, and the changes in the proof are minor.

Definition 3.1.7 (Domination)**.** *Given two ensembles of random variables $\{X_n\}_{n \in \mathbb{N}}$ and $\{X'_n\}_{n \in \mathbb{N}}$, we say that $\{X_n\}$* dominates *$\{X'_n\}$ if there exists a positive polynomial q such that for any n and any $x \in X'_n$, $\Pr[X_n = x] \geq (1/q(n)) \Pr[X'_n = x]$.*

As aforementioned, every sampler induces an ensemble of random variables. For brevity, sometimes we will say that a sampler S dominates a sampler S' to mean that the ensemble induced by S dominates that of S'.

The following facts, which will be used through the paper, are straightforward:

Fact 3.2. *If a sampler S dominates a sampler S' and S' is an onto sampler then so is S.*

Fact 3.3. *Let $\{X_n\}_{n \in \mathbb{N}}$ be an ensemble that dominates the ensemble $\{X'_n\}_{n \in \mathbb{N}}$ and let $\{S_n\}_{n \in \mathbb{N}}$ be a sequence of sets. Suppose $\Pr[X'_n \in S_n]$ is noticeable. Then $\Pr[X_n \in S_n]$ is noticeable.*

3.3 Main Result

Theorem 3.4. *Suppose that there exists a polynomially bounded total search problem R' that is hard under some polynomial time onto sampler. Then, there exists a polynomial time sampler S such that for any super-polynomial, time-constructible function τ, and every polynomial $\lambda(n) \geq n$, there exists a search problem R such that:*

- *R is hard under S.*

- *There is no polynomial time pairs-sampler S^* with randomness complexity bounded by λ, such that the first elements in the pairs output by S^* are distributed in a distribution that dominates $\{S(1^n)\}_{n\in\mathbb{N}}$, and the second element in every pair is a solution for the first element with respect to R if such a solution exists, and any string otherwise.*
- *The relation R is polynomially bounded, and can be verified in time $\tau(n)$ using one oracle call to a verifier for R'.*

We note that the resulted search problem R is not total, and that the sampler S does not output only YES-instances. Nevertheless, the existence of a pairs-sampler S^* as (asserted not to exist) in the theorem for such R and S implies the existence of one-way functions.

Proof. Let R' be a search problem that is hard under the polynomial time sampler S', from the hypothesis. Let τ be some super polynomial time-constructible function (for sake of concreteness one can think of $\tau(n) = n^{log^*(n)}$). Let τ' be a (smaller) super polynomial time-constructible function to be defined later. Let λ be a polynomial. We define S and R as follows:

Definition of R:
$((\langle M\rangle, x), (w, r_1, \ldots, r_{|x|})) \in R$ if and only if the following conditions hold:

- $(x, w) \in R'$
- For all $1 \leq i \leq |x|$ it holds that $|r_i| \leq \lambda(|\langle M\rangle| + |x|)$.
- For all $1 \leq i \leq |x|$, on input $1^{|\langle M\rangle|+|x|}$ and randomness r_i, the potential sampler $\tilde{M}$ outputs $(\langle M\rangle, x \oplus e_i^{|x|})$ in at most $\tau'(|\langle M\rangle| + |x|)$ steps.

In the following we describe the main ideas behind the definitions (disregarding issues of running time and randomness complexity, which will be dealt later). Suppose that M is some potential sampler (we will eventually consider $M = S^*$). Then, if $(w, r_1, \ldots, r_{|x|})$ is a solution for $(\langle M\rangle, x)$, then the r_i's, when used as randomness for M (with the appropriate input length), output an instance-solution pair where in the instance, the first element (the machine code) is again the code of the same machine M, and the second element in the instance is x with the i-th bit flipped. Thus, using the appropriate r_i from the solution as randomness for M, one can basically "flip any bit of x at his choice" without changing the (code of) the machine in the instance. Now, suppose the machine M is guaranteed to always output legal pairs with respect to R, when the instance in the pair is a YES-instance. Then, if the new instance (with the "flipped" bit) is a YES-instance, the new pair output by M is again a legal pair. It follows, that if all involved instances are YES-instances, then by repeated invocations of M (each time using the appropriate r_i from the last solution), one can sequentially flip bit after bit, and arrive at any desired x'. We then note that if $(w, r_1, \ldots, r_{|x|})$ is a solution for $(\langle M\rangle, x)$ under R, then w is a solution for x under R'. Thus, once arriving at such x' (i.e., once having M output a pair $((\langle M\rangle, x'), (w, r_1, \ldots, r_{|x|}))$, indeed w is a solution for x' under R'. It follows that R' can be efficiently solved (in the worst case), in contradiction to the hardness of R' under S'.

The following definition of S, together with the hypotheses of the theorem, guarantee that if a sampler S^* exists, the idea outlined above works. It also guarantees that R is hard under S, as required. We elaborate. Since we would like to diagonalize against all possible S^*'s, roughly speaking, we let S output (the code of) any machine (in the left part of the instance) with noticeable probability (see Claim 3.5). This implies (by the hypothesis of the theorem) that any potential S^* must also output (the code of) any machine with noticeable probability (as it is required to dominate S). If follows that one can efficiently (i.e., in expected polynomial time), find randomness for S^* that yield any desired machine. In particular, one can efficiently find randomness for S^* that will make it output *its own code* (and again, this is true for any possible S^*). Since S^* is assumed to output only legal pairs with respect to R when these exist, the process described above can then be performed, provided that legal solutions always exist. We will show that Since $\tilde{S}^*$ must be an onto-sampler (since S is an onto sampler and $\tilde{S}^*$ dominates it), and since R' is total, throughout the process described above solutions indeed will always exist. Moreover, since the right side of the instances output by S are distributed similar to S', it follows that R together with S "inherit" the hardness of R' under S'.

Definition of S:
On input 1^n:

1. Choose i uniformly from $[0, n]$.
2. Choose a potential sampler $\langle M\rangle$ uniformly from $\{0,1\}^i$.
3. Run S' on 1^{n-i} (supplying it with random coins from the randomness tape of S itself) and denote the output by x.
4. Output $(\langle M\rangle, x)$.

We now formalize the ideas described above. We will use the following simple fact, that states that every potential sampler is output by S with noticeable probability:

Fact 3.5. *For every fixed potential sampler M_0, the probability that $S(1^n) = (\langle M_0\rangle, \cdot)$ is noticeable.*

Proof. By our definition, on input $1^{\ell+|\langle M_0\rangle|}$ the sampler S outputs a tuple $(\langle M\rangle, \cdot)$ where $|\langle M\rangle| = |\langle M_0\rangle|$ with probability $1/(\ell + |\langle M_0\rangle| + 1)$. Given that this event occurs, the probability that $\langle M\rangle = \langle M_0\rangle$ is fixed (namely, it is $2^{-|\langle M_0\rangle|}$). Thus, S outputs a tuple of the form $(\langle M_0\rangle, \cdot)$ with noticeable probability. □

In the following we show that R and S are as required. Since R' is polynomially bounded, and λ is a fixed polynomial, it is straightforward that R too is polynomially bounded. It is easy to see that for some polynomial q (i.e., which depends on the model of computation) the relation R can be verified in time $q(\tau'(n))$ using one oracle call to a verifier for R'. Setting $\tau' = q^{-1}(\tau)$ we have that the relation R can be verified in time $\tau(n)$ using one oracle call to a verifier for R'. Since τ is super polynomial and time-constructible, it follows that $q^{-1}(\tau)$ is also super polynomial and time-constructible.

We show that R is hard under S. We first explain informally the idea of the proof. Let S_0 be some polynomial time onto sampler with randomness complexity bounded by λ. By Fact 3.5 any such S_0 is chosen by S with noticeable probability. As we will show, since R' is total, and since τ' is super-polynomial, for long enough x's, any instance of the form $(\langle S_0 \rangle, x)$ is a YES-instance of R (and thus a solver should succeed on it). The second element in the pair, x, is chosen by S (independently from the first) according to the distribution of S'. It follows that conditioned on some noticeable event:

- The distribution on the x's is identical to the distribution of S'.

- The instances are always YES-instances.

Now, if a solver B does not fail on R under S (according to Definition 3.1.6), it does not fail on R under S conditioned on this event. This follows from (1) the fact that when this event occurs the instance is a YES-instance (2) the event is noticeable. Thus, failure on R under the distribution of S conditioned on this event implies failure on R under S (with no conditioning). Moreover, according to the definition of R, a solution for the instance $(\langle S_0 \rangle, x)$ under R contains a solution for the instance x under R'. From all the above it follows that if B solves R under S, it implicitly solves in particular R' under S'. Thus the solver B can be used to solve R' under S', in contradiction.

We elaborate. Let S_0 be some polynomial time onto sampler with randomness complexity bounded by λ (for example, the sampler that on input 1^n outputs the first n coins from the randomness tape, which has randomness complexity bounded by λ since $\lambda(n) \geq n$). Let $p_{\tilde{S}_0}$ be a polynomial bounding the running time of $\tilde{S}_0$. By Fact 3.5, S outputs a tuple of the form $(\langle S_0 \rangle, x)$ with noticeable probability. Let n_0 be such that for any $n \geq n_0$ it holds that $\tau'(n + |\langle S_0 \rangle|) \geq p_{\tilde{S}_0}(n + |\langle S_0 \rangle|)$ (clearly such number exists as τ' is super-polynomial).

Claim 3.5.1. *Any instance of the form $(\langle S_0 \rangle, x)$ where $|x| \geq n_0$, is a YES-instance of R.*

Proof. The claim follows from the following facts:

- Every x has a w such that $(x, w) \in R'$, as R' is total.

- S_0 is an onto sampler, thus there are $r_1, \ldots, r_{|x|}$ such that for all $1 \leq i \leq |x|$, on input $1^{|\langle S_0 \rangle| + |x|}$ and randomness r_i, the sampler $\tilde{S}_0$ outputs $(\langle S_0 \rangle, x \oplus e_i^{|x|})$.
- Since the randomness complexity of S_0 is bounded by λ, it follows that for all $1 \leq i \leq |x|$, the randomness r_i can be taken such that $|r_i| \leq \lambda(|\langle S_0 \rangle| + |x|)$.
- Since $|x| \geq n_0$ it follows that $p_{\tilde{S}_0}(|x| + |\langle S_0 \rangle|) \leq \tau'(|x| + |\langle S_0 \rangle|)$, thus S_0 halts on the r_i's in at most $\tau'(|\langle S_0 \rangle| + |x|)$ steps as required.

□

Now, suppose to the contrary there exists an algorithm B that succeeds on R under S. That is,

$$\Pr_{(\langle M \rangle, x) \leftarrow S(1^n)}[((\langle M \rangle, x), B((\langle M \rangle, x))) \notin R \wedge \exists y((\langle M \rangle, x), y) \in R]$$

is not noticeable. Then (for $n \geq |\langle S_0 \rangle|$):

$$\Pr_{(\langle M \rangle, x) \leftarrow S(1^n)}[((\langle M \rangle, x), B((\langle M \rangle, x))) \notin R \wedge \exists y((\langle M \rangle, x), y) \in R | \langle M \rangle = \langle S_0 \rangle]$$

is also not noticeable, as we condition on a noticeable event (by Fact 3.5). Moreover, for large enough n's

$$\begin{aligned} & \Pr_{(\langle M \rangle, x) \leftarrow S(1^n)}[((\langle M \rangle, x), B((\langle M \rangle, x))) \notin R \wedge \exists y((\langle M \rangle, x), y) \in R | \langle M \rangle = \langle S_0 \rangle] \\ = & \Pr_{(\langle M \rangle, x) \leftarrow S(1^n)}[((\langle M \rangle, x), B((\langle M \rangle, x))) \notin R | \langle M \rangle = \langle S_0 \rangle], \end{aligned}$$

as for large enough n's, when $\langle M \rangle = \langle S_0 \rangle$ there is always a solution (by Claim 3.5.1). Since S chooses the second element (independent of the first) according to the distribution of S', it follows that for large enough n's, the last term equals

$$\Pr_{x \leftarrow S'(1^{n - |\langle S_0 \rangle|})}[((\langle S_0 \rangle, x), B((\langle S_0 \rangle, x))) \notin R].$$

Let B_1 denote the algorithm that behaves similar to B but outputs only the first element (out of the pair) output by B. Then the last term, which by the above is not noticeable, upper bounds

$$\Pr_{x \leftarrow S'(1^{n - |\langle S_0 \rangle|})}[(x, B_1((\langle S_0 \rangle, x))) \notin R'],$$

since if B succeeds on the instance $(\langle S_0 \rangle, x)$ then the first element output by B is a solution for x under R'. Finally, trivially

$$\begin{aligned} & \Pr_{x \leftarrow S'(1^{n - |\langle S_0 \rangle|})}[(x, B_1((\langle S_0 \rangle, x))) \notin R'] \\ \geq & \Pr_{x \leftarrow S'(1^{n - |\langle S_0 \rangle|})}[(x, B_1((\langle S_0 \rangle, x))) \notin R' \wedge \exists w (x, w) \in R'] \end{aligned}$$

(in fact equality holds here as R' is total), and we conclude that the algorithm $B_1((\langle S_0\rangle, \cdot))$ does not fail with noticeable probability on R' under S', in contradiction to the hardness of R' under S'. We conclude that R is hard under S.

We now prove the main claim, i.e., that there does not exist a sampler S^* as above. Assume towards contradiction that a sampler S^* as above does exist. Then, we use (the code of) S^* to construct an algorithm A that succeeds (i.e., does not fail with noticeable probability) on R' under S', in contradiction to the assumed hardness of R' under S'. In fact, A will solve R' in the worst case, in expected polynomial time. We note, that this part of the proof only uses the fact that S^* is an onto sampler (which follows from the fact that the sampler S we construct is an onto sampler). Following is the definition of A. (See elucidating remarks below.)

Definition of A:
Let $p_{\tilde{S}^*}$ be a polynomial bounding the running time of $\tilde{S}^*$. Let n_0 be such that for any $n \geq n_0$ it holds that $\tau'(n) \geq p_{\tilde{S}^*}(n)$ (clearly such number exists as τ' is super-polynomial).

On input x of length n (an instance of R'):

1. If $n + |\langle S^*\rangle| < n_0$ output the answer out of a pre-computed (fixed) table.

2. Invoke $S^*(1^{n+|\langle S^*\rangle|})$ (supplying it with random coins). Denote the output by $((\langle M\rangle, x^{(0)}), y)$.

3. If $\langle M\rangle \neq \langle S^*\rangle$ go back to 2. Else, If $\langle M\rangle = \langle S^*\rangle$, proceed (note that in that case $|x^{(0)}| = |x| = n$).

4. Parse y to $(w^{(0)}, r_1^{(0)}, \ldots, r_n^{(0)})$ (the fact that y can be so parsed will be proven later).

5. Let $i_1, \ldots, i_h$ be the locations of the bits that are different between x and $x^{(0)}$.
   For $j = 1$ to h:
   Use $r_{i_j}^{(j-1)}$ as randomness for $S^*(1^{n+|\langle S^*\rangle|})$ to obtain output $((\langle M\rangle, x^{(j)}), (w^{(j)}, r_1^{(j)}, \ldots, r_n^{(j)}))$
   (Again, the fact that the second element in the output of S^* can be so parsed will be proven later. Also, we will show that it must hold that $\langle M\rangle = \langle S^*\rangle$ and $x^{(j)} = x^{(j-1)} \oplus e_{i_j}^n$.)

6. Output $w^{(h)}$.

Before proving the correctness and the running time of the algorithm, let us explain informally its idea. In Lines 2 and 3 the algorithm invokes S^* (with appropriate input-length) until it outputs an instance-solution pair where the instance is of the form $(\langle S^*\rangle, x^{(0)})$ (where, due to the choice of the input-length, the length of $x^{(0)}$ is $|x|$). As we will show, this part takes expected polynomial time. The idea is that once such an instance is output, the string y that accompanies it is a solution with respect to R (since the instance is a YES-instance and S^* is assumed to output solutions when they exist). Moreover, this solution relates to the machine S^* *itself*. That is, y is of the form $(w^{(0)}, r_1^{(0)}, \ldots, r_n^{(0)})$, where the $r_i^{(0)}$'s can be used as randomness for S^* to "flip" the i-th bit of $x^{(0)}$, and obtain a new pair $(((\langle S^* \rangle, x^{(1)}), (w^{(1)}, r_1^{(1)}, \ldots, r_n^{(1)}))$, where $x^{(1)}$ is Hamming-closer to x by one bit. Then, the same idea is repeated in the loop in Line 5. At each step j the algorithm chooses the j-th bit that is different between $x^{(0)}$ and x, and using the appropriate randomness from the previous solution (i.e., $r_{i_j}^{(j-1)}$) "flips" it to become Hamming-closer to x by one bit. After reaching x (that is, after having S^* output a pair where the instance is $(\langle S^* \rangle, x)$), in Line 6, the algorithm outputs $w^{(h)}$, which, by the definition of R, and since S^* always outputs legal solutions with respect to R when these exist, is a solution for x under R'.

We proceed to a formal proof.

Claim 3.5.2. *The algorithm A runs in expected polynomial time.*

Proof. We first show that the algorithm reaches Line 4 in expected polynomial time. By Fact 3.5 the probability of the event that on input $1^{n+|\langle S^* \rangle|}$ the sampler S outputs $(\langle S^* \rangle, x^{(0)})$ (for some $x^{(0)}$ with $|x^{(0)}| = n$) is noticeable in n.

Since by assumption $\tilde{S}^*$ dominates S, it follows that the event that on input $1^{n+|\langle S^* \rangle|}$ the sampler S^* outputs $(\langle S^* \rangle, x^{(0)})$ (for some $x^{(0)}$ with $|x^{(0)}| = n = |x|$) is also noticeable in n. It follows that the algorithm reaches Line 4 in expected polynomial time. It is easy to see (considering that the running time of S^* is polynomial) that Lines 5,6 run in (strict) polynomial time. The claim follows. □

We now prove the correctness of the algorithm. That is, we show that the output of the algorithm A on input x is always a solution for x under R'.

Claim 3.5.3. *If the algorithm A reaches Line 4, then y is a valid solution for $(\langle M \rangle, x^{(0)})$ under R.*

Proof. We show that when the algorithm reaches Line 4, $(\langle M \rangle, x^{(0)})$ is a YES-instance, and therefore y is a valid solution for $(\langle M \rangle, x^{(0)})$ (as by assumption the second element in every pair output by S^* is a solution for the first element with respect to R if such exists).

The proof resembles the proof of Claim 3.5.1. Note that if the algorithm reaches Line 4 then $\langle M \rangle = \langle S^* \rangle$. We then note that:

- $x^{(0)}$ has a w such that $(x^{(0)}, w) \in R'$, as R' is total.
- $\tilde{S}^*$ is an onto sampler since S is an onto sampler, and $\tilde{S}^*$ dominates S. Thus, there are $r_1^{(0)}, \ldots, r_{|x|}^{(0)}$ such that for all $1 \leq i \leq |x|$, on input $1^{|\langle S^* \rangle|+n}$ and randomness $r_i^{(0)}$, the sampler $\tilde{S}^*$ outputs $(\langle S^* \rangle, x \oplus e_i^{|x|})$.

- Since the randomness complexity of S^* is assumed to be bounded by λ, it follows that for all $1 \leq i \leq |x|$, the randomness $r_i^{(0)}$ can be taken such that $|r_i^{(0)}| \leq \lambda(|\langle S^* \rangle| + n)$.
- If A reaches Line 4 then $n + |\langle S^* \rangle| \geq n_0$ (because of Line 1), thus $\tau'(n + |\langle S^* \rangle|) \geq p_{\tilde{S}^*}(n + |\langle S^* \rangle|)$. Thus $\tilde{S}^*$ halts on the $r_i^{(0)}$'s in at most $\tau'(|\langle S^* \rangle| + n)$ steps as required.

The claim follows. □

Finally, we show that the algorithm outputs the correct output. First we note:

Claim 3.5.4. *The second element in the output of S^* in each step j in the loop in Line 5 is a valid solution for $(\langle M \rangle, x^{(j)})$ with respect to R.*

The proof is identical to the proof of Claim 3.5.3.

Claim 3.5.5. *For every j, at the end of step j in the loop in Line 5 the Hamming distance between $x^{(j)}$ and x is $h - j$.*

Proof. By Claims 3.5.3 and 3.5.4, and since by assumption the second element in every pair output by S^* is a solution for the first element with respect to R if such exists, it follows that throughout the algorithm, "all $r_i^{(j)}$'s are correct" (formally, for every $0 \leq j \leq h$ and every $1 \leq i \leq n$, any $r_i^{(j)}$ is such that $S^*(1^{n+|\langle S^* \rangle|})$ with randomness $r_i^{(j)}$ outputs a pair of the form $((\langle S^* \rangle, x^{(j)} \oplus e_i^n), \cdot))$. Since in step j the algorithm A chooses to run S^* on $r_{i_j}^{(j-1)}$, where i_j is the j-th bit that is different between $x^{(0)}$ and x, it follows that $x^{(j+1)}$ is Hamming-closer to x than $x^{(j)}$ by one bit. The claim follows. □

It follows that $x^{(h)} = x$. Since $(w^{(h)}, r_1^{(h)}, \ldots, r_n^{(h)})$ is a solution for $(\langle M \rangle, x^{(h)})$ under R (again, by Claim 3.5.4), by the definition of R it follows that $w^{(h)}$ is a solution for $x^{(h)} = x$ under R'. The theorem follows. □

3.4 Conclusion

3.4.1 On Our Assumptions

We have shown that if there exists a polynomially bounded total search problem that is hard under some polynomial time onto sampler, then there exist efficiently samplable distributions that are hard on average, but that cannot be transformed into one-way functions (in the manner we described).

We note that the hypothesis of the theorem is implied by the existence of one-way permutations, is further implied by the weaker assumption of the existence of onto one-way functions (where by ”onto” we mean that the function’s image is Σ^*), and is even weaker than the latter. Thus, our assumptions are plausible. In fact, one does not even have to assume that S' is an onto sampler. If S' is a sampler with the property that the size of its image is noticeable, it can be made onto by standard (straightforward) techniques without harming the hardness of any search problem under this sampler.

It might seem contradictory at first sight that we mention that our hypothesis is implied by an assumption that in fact yields the *existence* of one-way functions. But, our result is essentially about approaches to *construct* one-way functions, and not about their *existence*. Thus, a way to interpret the result, is that if onto one-way functions exist, then a certain way to prove the existence of one-way functions cannot work (although one-way functions in fact do exist under this assumption).

3.4.2 On Restricting the Randomness Complexity of S^*

To the best of our knowledge, it is unknown if restricting the randomness of the pairs-sampler S^* is with loss of generality. It is fairly easy to see that if the answer to the following (open, to the best of our knowledge) question is affirmative, then restricting the randomness of the pairs-sampler S^* is *without* loss of generality.

Open Question 3.6. *There exists a (universal) polynomial q such that for every pairs-sampler S^* such that $\tilde{S}^*$ is an onto sampler, there exists a pairs-sampler S^{**} such that $\tilde{S}^{**}$ is an onto sampler and such that:*

- *For every n, it holds that* $\text{support}(S^{**}(1^n)) \subseteq \text{support}(S^*(1^n))$.
- *The sampler $\tilde{S}^{**}$ dominates $\tilde{S}^*$.*
- *The randomness complexity of S^{**} is bounded by q.*

3.4.3 Possible Interpretations of Our Result

First we note that the conclusion of our theorem can be presented in a different way: *There exists a polynomial time sampler S such that for any polynomials q and λ there exists a polynomial time verifiable search problem R such that R is hard under S, but where any sampler S^* as in the Desired Construction must run in time exceeding q and use randomness exceeding λ.* It is easy to modify the proof accordingly (basically, one just has to change the function τ in the definition of R, to q).

A possible interpretation of our result is as an indication for the generality of the classes for which one can achieve the Desired Construction. Following the discussion in the Introduction, the most generic manner to achieve the Desired Construction is to achieve it for any R and S where R is hard under S, and where R is polynomially bounded and exponential time verifiable. Our result implies that in any such generic construction, the randomness complexity of S^* must depend on R. This implication is somewhat surprising, as one might have expected that once we fixed S, we can fix the randomness complexity of S^* for any R, with the (very informal) justification being that the ”random” requirement from S^* is that its distribution on the instances dominates S, while the requirement that it outputs instance-solution pairs under R is ”computational”.

We note, however, that our result does not imply that any ”generic” approach to achieve the Desired Construction cannot succeed (where by ”generic” we mean a transformation that works for a wide class of pairs (R, S)). The two main reasons for this statement (besides the loss of generality implied by the restriction on the randomness complexity of S^*) are:

- The search problem we construct is not polynomial time verifiable (but is ”nearly polynomial time verifiable”). This still leaves open the possibility that any polynomial time verifiable search problem that is hard on average under some efficient sampler can be transformed to a one-way function (via the Desired Construction we refer to).

- We require that the desired S^* relates strictly to R and S. A slightly different approach could be to first transform R and S to some R_1 and S_1, and then construct S^* for R_1 and S_1. Our result does not eliminate the possibility that this approach can work for a wide class of pairs (R, S).

Acknowledgements

I would like to deeply thank Oded Goldreich for many helpful discussions, for the help in the presentation of this paper, and for encouraging me to work in this area. I would also like to thank Salil Vadhan for an observation regarding this work.

Bibliography

[1] Adi Akavia, Oded Goldreich, Shafi Goldwasser, and Dana Moshkovitz. On basing one-way functions on np-hardness. In Jon M. Kleinberg, editor, *STOC*, pages 701–710. ACM, 2006.

[2] Andrej Bogdanov and Luca Trevisan. On worst-case to average-case reductions for np problems. *SIAM J. Comput.*, 36(4):1119–1159, 2006.

[3] Gilles Brassard. Relativized cryptography. In *FOCS*, pages 383–391. IEEE, 1979.

[4] Joan Feigenbaum and Lance Fortnow. Random-self-reducibility of complete sets. *SIAM J. Comput.*, 22(5):994–1005, 1993.

[5] Oded Goldreich. *Computational Complexity: A Conceptual Perspective*. Cambridge University Press, 2008.

[6] S. Vadhan. Personal communication, 2010.

Chapter 4

Sequential Rationality in Cryptographic Protocols[1]

Ronen Gradwohl and Noam Livne and Alon Rosen

Abstract

Much of the literature on rational cryptography focuses on analyzing the strategic properties of cryptographic protocols. However, due to the presence of computationally-bounded players and the asymptotic nature of cryptographic security, a definition of sequential rationality for this setting has thus far eluded researchers.

We propose a new framework for overcoming these obstacles, and provide the first definitions of computational solution concepts that guarantee sequential rationality. We argue that natural computational variants of subgame perfection are too strong for cryptographic protocols. As an alternative, we introduce a weakening called threat-free Nash equilibrium that is more permissive but still eliminates the undesirable "empty threats" of non-sequential solution concepts.

To demonstrate the applicability of our framework, we revisit the problem of implementing a mediator for correlated equilibria (Dodis-Halevi-Rabin, Crypto'00), and propose a variant of their protocol that is sequentially rational for a non-trivial class of correlated equilibria. Our treatment provides a better understanding of the conditions under which mediators in a correlated equilibrium can be replaced by a stable protocol.

Keywords: rational cryptography, Nash equilibrium, subgame perfect equilibrium, sequential rationality, cryptographic protocols, correlated equilibrium

[1] To appear in *Foundations of Computer Science (FOCS) 2010.*

4.1 Introduction

A recent line of research has considered replacing the traditional cryptographic modeling of adversaries with a game-theoretic one. Rather than assuming arbitrary *malicious* behavior, participants are viewed as being self-interested, *rational* entities that wish to maximize their own profit, and that would deviate from a protocol's prescribed instructions if and only if it is in their best interest to do so.

Such game theoretic modeling is expected to facilitate the task of protocol design, since rational behavior may be easier to handle than malicious behavior. It also has the advantage of being more realistic in that it does not assume that some of the parties honestly follow the protocol's instructions, as is frequently done in cryptography.

The interplay between cryptography and game theory can also be beneficial to the latter. For instance, using tools from secure computation, it has been shown how to transform games in the mediated model into games in the unmediated model.

But regardless of whether one analyzes cryptographic protocols from a game theoretic perspective or whether one uses protocols to enhance game theory, it is clear that the results are meaningful only if one provides an adequate framework for such analyses.

4.1.1 Computational Nash Equilibrium

Applying game-theoretic reasoning in a cryptographic context consists of modeling interaction as a *game*, and designing a protocol that is in *equilibrium*. The game specifies the model of interaction, as well as the utilities of the various players as a function of the game's outcome. The protocol lays out a specific plan of action for each player, with the goal of realizing some pre-specified task. Once a protocol has been shown to be in equilibrium, rational players are expected to follow it, thus reaching the desired outcome.

A key difficulty in applying game-theoretic reasoning to the analysis of cryptographic protocols stems from the latter's use of computational infeasibility. Whereas game theory places no bounds on the computational ability of players, in cryptography it is typically assumed that players are computationally bounded. Thus, in order to retain the meaningfulness of cryptographic protocols, it is imperative to restrict the set of strategies that are available to protocol participants. This gives rise to a natural analog of Nash equilibrium (NE), referred to as *computational Nash equilibrium* (CNE): any polynomial-time computable deviation of a player from the specified protocol can improve her utility by only a negligible amount (assuming other players stick to the prescribed strategy).

Consider, for example, the following (two-stage, zero-sum) game (related to a game studied by Ben-Sasson et al. [4] and Fortnow and Santhanam [7]), which postulates the existence of a one-way permutation $f : \{0,1\}^n \mapsto \{0,1\}^n$.

Example 4.0.1. (One-way permutation game):

1. *P_1 chooses some $x \in \{0,1\}^n$, and sends $f(x)$.*

2. *P_2 sends a message $z \in \{0,1\}^n$.*

3. *P_2 wins (gets payoff 1) if $z = x$ (and gets -1 otherwise).*

In classical game theory, in all NE of this game P_2 wins, since there always exists some z such that $z = x$. However, in the computational setting, the following is a CNE: both players choose their messages uniformly at random (resulting in an expected loss for P_2). This is true because if P_2 chooses z at random, then P_1 can never improve his payoff by not choosing at random. If P_1 chooses x at random, then by the definition of a one-way permutation, any computationally-bounded strategy σ_2 of P_2 will be able to guess the value of x with at most negligible (in n) probability. Thus, the expected utility of P_2 using σ_2 is negligible, and so he loses at most that much by sticking to his CNE strategy (i.e. picking some z at random).

4.1.2 Computational Subgame Perfection

The notion of CNE serves as a first stepping stone towards a game-theoretic treatment of cryptographic protocols. However, protocols are typically *interactive*, and CNE does not take their sequential nature into consideration.

In traditional game theory interaction is modeled via extensive games. The most basic equilibrium notion in this setting is *subgame perfect equilibrium* (SPE), which requires players' strategies to be in NE at any point of the interaction, regardless of the history of prior actions taken by other players. Basically, this ensures that players will not reconsider their actions as a result of reaching certain histories (a.k.a. "empty threats").

As already noted in previous works (cf. [16, 19, 26]), it is not at all clear how to adapt SPE to the computational setting. A natural approach would be to require the strategies to be CNE at every possible history. However, if we condition on the history, then this means that *different* machines can and will do much better than the prescribed equilibrium strategy. For example, in the one-way permutation game of Example 4.0.1, given any message history, a machine M can simply have the correct inverse hardwired.

Although this requirement can be relaxed to ask that the prescribed strategy should be better than any other fixed machine on all inputs, this again may be too strong, since a fixed machine can always do better on some histories. Therefore, it seems that we must accept the following: for any machine M, with *high probability* over possible message histories, the prescribed strategy does at least as well as M. However, it turns out that this approach also fails to capture our intuitive understanding of a computational SPE (CSPE). Consider the following (two-stage) variant of the one-way permutation game from Example 4.0.1:

Example 4.0.2. (Modified one-way permutation game):

1. *P_1 chooses some $x \in \{0,1\}^n$, and sends $f(x)$.*
2. *P_2 sends a message $z \in \{0,1\}^n$.*
3. *If exactly one of P_1 and P_2 send message 0, both players get payoff -2. If both players send message 0, both players get payoff $+2$. Otherwise, P_2 wins (with payoff $+1$) if and only if $z = x$, and the non-winning player loses (with payoff -1).*

Using a similar argument to the one applied in Section 4.1.1, it can be shown that the strategies in which both players choose a message uniformly at random from $\{0,1\}^n \setminus \{0\}$ satisfy the above "probabilistic" variant of CSPE. However, this equilibrium does not match our intuitive understanding of SPE: P_1 will prefer to send message 0 regardless of P_2's strategy, knowing that P_2 will then respond with 0 as well. The threat of playing uniformly from all other messages is empty, and hence should not be admitted by the definition.[2]

The examples above are rather simple, so it is reasonable to expect that issues arising in their analyses are inherent in many other cryptographic protocols. This raises the question of whether a computational variant of SPE is at all attainable in a cryptographic setting.

At the heart of this question is the fact that essentially any cryptographic protocol carries some small (but positive) probability of being broken. This means that, while there may be a polynomial-time TM that can "perform well" on the *average* message history, there is no single TM that will do better than *all* other TMs on every history (as for any history there exists some TM that has the corresponding "secret information" hardwired).

This state of affairs calls for an alternative approach. While such an approach should be meaningful enough to express strategic considerations in an interactive setting, it should also be sufficiently weak to be realizable. As demonstrated above, any approach for tackling this challenge should explicitly address the associated probability of error. It should also take asymptotics into consideration.

4.2 Our Results

We propose a new framework for guaranteeing sequential rationality in a computational setting. Our starting point is a weakening of subgame perfection, called *threat-free Nash equi-*

[2] We note that a simple change to the payoffs yields a game whose empty threat is more "typical": For the case in which both players send message 0, let P_2's payoff be $-3/2$.

librium, that is more permissive, but still eliminates the undesirable empty threats of non-sequential solution concepts.

To cast our new solution concept into the computational setting, we develop a methodology that enables us to "translate" arguments that involve computational infeasibility into a purely game theoretic language. This translation enables us to argue about game theoretic concepts directly, abstracting away complications that are related to computation.

In order to demonstrate the applicability of our framework, we revisit the problem of implementing a mediator for correlated equilibria [6], and propose a protocol that is sequentially rational for a non-trivial class of correlated equilibria (see Section 4.2.3 for details). Our treatment provides a better understanding of the conditions under which mediators in a correlated equilibrium can be replaced by a stable protocol.

4.2.1 Threat-Free Nash Equilibria

We introduce *threat-free Nash equilibria* (TFNE), a weakening of subgame perfection whose objective is to capture strategic considerations in an interactive setting. Loosely speaking, a pair of strategies in an extensive game is a TFNE if it is a NE, and if in addition no player is facing an empty threat at any history.

The problem of empty threats is the following: in a NE of an extensive game, it is possible that a player plays sub-optimally at a history that is reached with probability 0. The other player may strategically choose to deviate from his prescribed strategy and arrive at that history, knowing that this will cause the first player to play an optimal response rather than the prescribed one. In an SPE this problem is eliminated by requiring that no player can play sub-optimally at any history, and so no other player will strategically deviate and take advantage of this.

The main observation leading to the definition of TFNE is that the above requirement may be too strong a condition to eliminate such instability: if an optimal response of a player *decreases* the utility of the other, then this other player would not want to strategically deviate. By explicitly ruling out this possibility, the instability caused by empty threats is eliminated, despite the equilibrium notion being more permissive than subgame perfection.

To make this precise, we give the first formal definition of an empty threat in extensive games. The definition is regressive: Roughly speaking, a player i is facing a threat at a history if there is some deviation at that history, along with a threat-free continuation from that history onwards, so that i increases his overall expected payoff when the players play this new deviation and continuation.

We note that the notion of TFNE is strong enough to eliminate the undesirable strategy of playing randomly in the modified OWP game from Example 4.0.2 – Claim 4.0.2 shows that in any computational TFNE of this game the second player outputs 0 after history 0.

4.2.2 Strategy-Filters and Tractable Strategies

To cast the definition of TFNE into a computational setting, we map the given protocol into a sequence of extensive games using *strategy-filters* that map computable strategies into their "strategic representation" (the strategic representation corresponds to the strategy effectively played by a given interactive Turing machine). We can then apply pure game theoretic solution concepts, and in particular our newly introduced concept of TFNE, to understand the strategic behavior of players.

Similarly to the definition of CNE, the computational treatment departs from the traditional game theoretic treatment in two crucial ways. First of all, our definition is framed *asymptotically* (in order to capture computational infeasibility), whereas traditional game-theory is framed for finitely sized games. Second, it allows for a certain *error probability*. This is an artifact of the (typically negligible) probability with which the security of essentially any cryptographic scheme can be broken.

Given a cryptographic protocol, we consider a corresponding sequence of extensive games. The sequence is indexed by a security parameter k and an error parameter ε. For each game, we "constrain" the strategies available to players to be a subset of those that can be generated by PPT players in the protocol. Intuitively, the game indexed by (k, ε) contains those strategies that run in time polynomial in k and "break crypto" with probability at most ε. We also require that strategy-filters be *PPT-covering*: that for any polynomially-small ε, every PPT is eventually a legal strategy, far enough into the sequence of extensive games.

Using this framework we formalize the notion of a computational threat-free Nash equilibrium (CTFNE). To the best of our knowledge this is the first attempt at analyzing sequential strategic reasoning in the presence of computational infeasibility.

4.2.3 Applications

Our treatment provides a powerful tool for arguing about the strategic behavior of players in a cryptographic protocol. It also enables us to isolate sequential strategic considerations that are suitable for use in cryptographic protocols (so that the solution concept is not too weak and not too strong).

As a warm up, we demonstrate the applicability of our framework and solution concept to the "coin-flipping game" that corresponds to Blum's coin-flipping protocol [5]. One may view this as playing the classic game of match pennies without simultaneity (but with cryptography). We show that it is possible to exploit the specific structure of the game to implement a correlating device resulting in a CTFNE. This is in contrast to the general approach of [6] that only enables one to argue CNE. This result already demonstrates the added strength of our framework and definition.

We then revisit the general problem of implementing a mediator for correlated equilibria [6], and propose a protocol that is sequentially rational for a non-trivial class of correlated equilibria. In particular, our protocol is in a CTFNE for correlated equilibria that are convex combinations of Nash equilibria and that are "undominated": There does not exist any convex combination of Nash equilibria for which both players get a strictly higher expected payoff.

Our treatment explores the conditions under which mediators in a correlated equilibrium can be replaced by a stable protocol, and sheds light on some structural properties of such equilibria.

Finally, we prove a general theorem that identifies sufficient conditions for a TFNE in extensive games. Namely, we show that if an undominated NE has the additional property that no player can harm the other by a unilateral deviation, then that NE must also be threat-free.

4.2.4 Related Work

This paper contributes to the growing literature on rational cryptography. Many of the papers in this line of research, such as [6, 13, 15, 1, 9, 20, 22, 16, 18, 19, 17, 26, 23, 2, 10], explore various solution concepts for cryptographic protocols viewed as games (often in the context of rational secret-sharing). Aside from the works of Lepinski et al. [15, 20], Ong et al. [26], and Gradwohl [10], who work in a different model[3], all prior literature has considered solution concepts that are non-sequential. More specifically, they all use variants of NE such as strict NE, NE with stability to trembles, and everlasting equilibrium.

An additional related work is that of Halpern and Pass [14], in which the authors present a general framework for game theory in a setting with computational cost. While their approach to computational limitations is more general than ours, they only address NE. Finally, Fortnow and Santhanam [7] study a different framework for games with computational limits, but also only in the context of NE.

4.2.5 Future Work

One potential application of our new definition is an analysis of rational secret-sharing protocols. While the design of such a protocol that is in a CTFNE is not within the scope of the current paper, we do provide some intuition about why known gradual release protocols satisfy a slightly weaker solution concept. Consider the following simple setting: each of two players knows a bit, and the XOR of the two bits is the secret. Secret exchange protocols, for

[3] More specifically, [15, 20] make strong physical assumptions, [26] assume the existence of a fraction of honest (non-rational) players, and [26, 10] work in an information-theoretic setting.

example [21], allow the players to exchange their respective bits and thus learn the secret in such a way that even if one of the players cheats, he can reconstruct the secret with probability at most ε more than the other player. Then under the assumptions on players' utilities used by [17], any unilateral deviation from this protocol can get the deviating player an increase of only $O(\varepsilon)$ in utility. However, since the other player can always correctly guess the secret with almost the same probability (up to the additive ε), the potential benefit to a player of deviating, causing the other to deviate, and so on, is also at most $O(\varepsilon)$. Thus, this protocol is in a computational variant of ε-NE and is also ε-threat-free. The reason this is weaker than our current solution concept is that we require the benefit from a threat or a deviation to be negligible, whereas in [21] the ε is polynomially-small (in the number of rounds of the protocol).

There are numerous other compelling problems left for future work. The first problem is to extend our definition to games with simultaneous moves. While we do offer a partial extension tailored to the problem of implementing a mediator, the problem of defining CTFNE for general games with simultaneous moves is open. Such a definition would be particularly useful for a sequential analysis of protocols with a simultaneous channel. Another natural extension of the definition is to multiple players, as opposed to 2. Such an extension comes with its own challenges, particularly with regard to the possibility of collusion. A third extension is to incorporate the threat-freeness property with stronger variants of NE, such as stability with respect to trembles, strict NE, or survival of iterated elimination of dominated strategies. Finally, we would like to find more applications for our definition. One particularly interesting problem is to extend our results on the implementation of mediators to a larger class of correlated equilibria.

4.3 Game Theory Definitions

4.3.1 Extensive Games

Informally, a game in extensive form can be described as a game tree in which each node is owned by some player and edges are labeled by legal actions. The game begins at the root, and at each step follows the edge labeled by the action chosen by the current node's owner. Utilities of players are given at the leaves of the tree. More formally, we have the following standard definition of extensive games (see, for example, Osborne and Rubinstein [27]):

Definition 4.0.1 (Extensive game)**.** A 2-person *extensive game* is a tuple $\Gamma = (H, P, A, u)$ where

- H is a set of (finite) *history* sequences such that the empty word $\epsilon \in H$. A history $h \in H$ is *terminal* if $\{a : (h, a) \in H\} = \emptyset$. The set of terminal histories is denoted Z.

- $P : (H \setminus Z) \to \{1, 2\}$ is a function that assigns a "next" player to every non-terminal history.

- A is a function that, for every non-terminal history $h \in H \setminus Z$, assigns a finite set $A(h) = \{a : (h, a) \in H\}$ of available actions to player $P(h)$.

- $u = (u_1, u_2)$ is a pair of payoff functions $u_i : Z \mapsto \mathbb{R}$.

We will denote the two players by P_1 and P_2 and by P_i and P_{-i}, where $i \in \{1, 2\}$ and $-i$ is shorthand for $2 - i$.

Definition 4.0.2 (Behavioral strategy)**.** *Behavioral strategies* of players in an extensive game are collections $\sigma_i = (\sigma_i(h))_{h:P(h)=i}$ of independent probability measures, where $\sigma_i(h)$ is a probability measure over $A(h)$.

For any extensive game $\Gamma = (H, P, A, u)$, any player i, and any history h satisfying $P(h) = i$, we denote by $\Sigma_i(h)$ the set of all probability measures over $A(h)$. We denote by Σ_i the set of all strategies σ_i of player i in Γ. For each *profile* $\sigma = (\sigma_1, \sigma_2)$ of strategies, define the *outcome* $O(\sigma)$ to be the probability distribution over terminal histories that results when each player i follows strategy σ_i. Note that if both σ_1 and σ_2 are deterministic (i.e. deterministic on every history), then so is the outcome $O(\sigma)$.

4.3.2 Nash Equilibrium

Each profile of strategies yields a distribution over outcomes, and we are interested in profiles that guarantee the players some sort of optimal outcomes. There are many solution concepts that capture various meanings of "optimal," and one of the most basic is the Nash equilibrium (NE).

Definition 4.0.3 (Nash equilibrium (NE))**.** An *ε-Nash equilibrium* of an extensive game $\Gamma = (H, P, A, u)$ is a profile σ^* of strategies such that for each player i,

$$\mathrm{E}\left[u_i\left(O(\sigma^*)\right)\right] \geq \mathrm{E}\left[u_i\left(O(\sigma^*_{-i}, \sigma_i)\right)\right] - \varepsilon$$

for every strategy σ_i of player i. It is a *NE* if the above holds for $\varepsilon \leq 0$ and a *strict NE* if it holds for some $\varepsilon < 0$.

One of the premises behind the stability of profiles that are in an ε-NE is that players will not bother to deviate for a mere gain of ε. For applications in cryptography we will generally have ε be some negligible function, and this corresponds to our understanding that we do not care about negligible gains.

4.3.3 Subgame Perfect Equilibrium

One of the problems with NE in extensive games is the presence of empty threats: a player's equilibrium strategy may specify a sub-optimal strategy at a history that is reached with probability 0. The other player, knowing this, may strategically deviate to reach that history, predicting that the first player will also deviate. For more details and explicit examples see any textbook on game theory, such as [27].

The most basic solution to the problem of empty threats is to refine the NE solution, and require a strategy profile to be in a NE at every history in the game. This results in a profile that is in *subgame perfect equilibrium* (SPE).

Definition 4.0.4 (Subgames of extensive game)**.** *For any 2-person extensive game $\Gamma = (H, P, A, u)$ and any non-terminal history $h \in H$, the subgame $\Gamma|_h$ is the 2-person extensive game $\Gamma|_h = (H|_h, P|_h, A|_h, u|_h)$, where*

- *$h' \in H|_h$ if and only if $h \circ h' \in H$,*
- *$P|_h(h') = P(h \circ h')$,*
- *$A|_h(h') = A(h \circ h')$, and*
- *$u_i|_h(h') = u_i(h \circ h')$.*

For each profile $\sigma = (\sigma_1, \sigma_2)$ of strategies and history $h \in H$, define the *conditional outcome $O(\sigma)|_h$* to be the probability distribution over terminal histories that results when the game starts at a history h, and from that point onwards each player i follows strategy σ_i.

Definition 4.0.5 (Subgame perfect equilibrium (SPE))**.** *An* ε-subgame perfect equilibrium *of an extensive game $\Gamma = (H, P, A, u)$ is a profile σ^* of strategies such that for each player i and each non-terminal history $h \in H$,*

$$\mathrm{E}\left[u_i\left(O(\sigma^*)|_h\right)\right] \geq \mathrm{E}\left[u_i\left(O(\sigma^*_{-i}, \sigma_i)|_h\right)\right] - \varepsilon$$

for every strategy σ_i of player i. It is an SPE *if the above holds for $\varepsilon = 0$ and a* strict SPE *if it holds for some $\varepsilon < 0$.*

4.3.4 Constrained Games

In the standard game theory literature, where there are no computational constraints on the players, the available strategies σ_i of player i are all possible collections $(\sigma_i(h))_{h:P(h)=i}$, where $\sigma_i(h)$ is an arbitrary distribution over $A(h)$. In our setting, however, we will only consider strategies that can be implemented by computationally bounded ITMs. This requires being able to constrain players' strategies to a strict subset of the possible strategies. One natural way to restrict the strategies is to allow only a subset of all distributions over $A(h)$ at each history h. However, this does not enable us to capture more elaborate restrictions, and specifically ones that might result from requiring strategies to be implementable by polynomial time ITMs. (For example, a player might have for every possible history a strategy that plays best response on that history, but no strategy that plays best response on *all* histories.) To capture these more elaborate restrictions, we consider player i strategies that are restricted to an arbitrary subset T_i of all possible (mixed) strategies. Given a pair $T = (T_1, T_2)$ of such sets we can then define a constrained version of a game, in which only strategies that belong to these sets are considered.

Definition 4.0.6 (Constrained game)**.** Let $\Gamma = (H, P, A, u)$ be an extensive game and let $T = (T_1, T_2)$, where $T_i \subseteq \bigotimes_{h:P(h)=i} \Sigma_i(h)$ for each $i \in \{1, 2\}$. The *T-constrained version* of Γ is the game in which the only allowed strategies for player i belong to T_i.

NE of constrained games are defined similarly to regular NE, except that players' strategies and deviations must be from the constraint sets.

Definition 4.0.7 (NE in constrained games)**.** *An* ε-Nash equilibrium *of a (T_1, T_2)-constrained version of an extensive game $\Gamma = (H, P, A, u)$ is a profile $\sigma^* \in (T_1, T_2)$ of strategies such that for each player i,*

$$\mathrm{E}\left[u_i\left(O(\sigma^*)\right)\right] \geq \mathrm{E}\left[u_i\left(O(\sigma^*_{-i}, \sigma_i)\right)\right] - \varepsilon$$

for every strategy $\sigma_i \in T_i$ of player i. It is a NE *if the above holds for $\varepsilon \leq 0$ and a* strict NE *if it holds for some $\varepsilon < 0$.*

4.4 Threat-Free Nash Equilibrium

Our starting point is the inadequacy of subgame perfection in capturing sequential rationality in a computational context. As argued in Section 4.1.2, it is unreasonable to require computationally-bounded players to play optimally at every node of a game. In particular, in cryptographic settings this requires breaking the security of the protocol, which is assumed impossible under the computational constraints.

A possible idea might be to require that players "play optimally at every node of the game, under their computational constraints." However, this idea cannot be interpreted in a sensible way. Computational constraints must be defined "globally," and thus the notion of playing optimally under some computational constraint on a particular history is senseless. In particular, for any history of some cryptographic protocol, there is a small machine that plays optimally on this specific history *unconditionally* (and breaks "cryptographic challenges" appearing in this history, by having the solutions hardwired). This machine is efficient, and so meets essentially any computational constraint. So, while under computational constraints every machine fails on cryptographic challenges in most histories, for every history there is a machine that succeeds. We thus assume that a player chooses his machine before the game starts, and cannot change his machine later.

4.4.1 A New Solution Concept

In light of the above discussion, it seems like the solution concept we are looking for has to reconciliate between the following seemingly conflicting properties:

1. It implies an optimal strategy for the players *under their computational constraints*, which implies *non-optimal* play on certain histories.

2. It does not allow empty threats, thus implying "sequential rationality."

The crucial observation behind our definition is that in order to rule out empty threats, one does not necessarily need to require that players play optimally at *every* node, because not every non-optimal play carries a threat to other players. In fact, in a typical cryptographic protocol, the security of each player is *building* on other players not playing optimally (because playing optimally would mean breaking the security of the protocol). Thus, a player's "declaration" to play non-optimally does not necessarily carry a threat: the other players may even gain from it. More generally, even in non-cryptographic protocols, at least in 2-player perfect information games, we can use the following observation: in any computational challenge, either a player gains from the other not playing optimally, or, if he does not gain, he can avoid introducing that computational challenge to the other player.[4]

Following the above observation, we introduce a new solution concept for extensive games. The new solution concept requires that players be in NE, and moreover, that no player impose an empty threat on the other. At the same time, it does not require players to play optimally at every node. In other words, players may (declare to) play non-optimally on non-equilibrium support, yet this declaration of non-optimal play does not carry an empty threat. We call our new solution concept TFNE, for threat-free Nash equilibrium.

To make the above precise, we introduce a formal definition of an empty threat. An empty threat occurs when a player threatens to play "non-rationally" on some history in order to coerce the other player to avoid this history. Crucially, empty threats are such that, had the

[4] This is indeed an informal statement. In fact, we should add the disclaimer that computational hardness for one player does not necessarily have to stem from the strategy of the other. For example, the utility function may be computationally hard.

threatened *not* believed the threat, had he deviated accordingly, and had the threatening player played “rationally,” the threatened player would have benefitted. To rephrase our intuition: a player faces an empty threat with respect to some strategy profile if by deviating from his prescribed strategy, and having the other player react “rationally,” he improves his payoff (in comparison with sticking to the prescribed strategy and having the other player react “rationally” from then on).

But what does it mean for the other player to react “rationally”? The other player may assume, recursively, that the first player will play a best response, and will not carry out empty threats against him, and so on, leading to a regressive definition.

4.4.2 Vanilla Version

Before giving the general definition of TFNE that we will use, we present a simpler version that has no slackness parameter and that works for games without constrained strategies.

For a player i and a history h, two strategies σ_i and π_i are *equivalent for player i on h* if $P(h) = i$ and $\sigma_i(h) = \pi_i(h)$, or $P(h) \neq i$. Two strategies *differ only on the subgame h* if they are equivalent on every non-terminal history that does not have h as a prefix. Formally, they are equivalent on every history in $H \setminus \{h' \in H : h' = h \circ h''$ for some $h''\}$. For a history $h \in H$, a strategy σ, and a distribution $\tau = \tau(h)$ on $A(h)$, let

$$\mathrm{Cont}(h, \sigma, \tau) \overset{\text{def}}{=} \Big\{\pi : (\pi \text{ differs from } \sigma \text{ only on the subgame } h) \ \&\ (\pi(h) = \tau(h))\Big\}.$$

We now proceed to define a threat. For simplicity, we will do so for generic games, in which each player’s possible payoffs are distinct. For such games, the set $\mathrm{Cont}(h, \sigma, \tau)$ always contains exactly one “threat-free” element (defined below).

Definition 4.0.8 (Threat)**.** Let $\Gamma = (H, P, A, u)$ be an extensive game with distinct payoffs. Let σ be a strategy profile, and let $h \in H$. Player $i = P(h)$ is facing a *threat* at history h with respect to σ if there exists a distribution $\tau = \tau(h)$ over $A(h)$ such that the unique $\pi \in \mathrm{Cont}(h, \sigma, \tau)$ and $\pi' \in \mathrm{Cont}(h, \sigma, \sigma)$ that are threat-free on h satisfy

$$\mathrm{E}\left[u_i\left(O(\pi)\right)\right] > \mathrm{E}\left[u_i\left(O(\pi')\right)\right],$$

where strategy π is threat-free on h if for *all* $h' \neq \epsilon$ satisfying $h \circ h' \in H$ player $P(h \circ h')$ is not facing a threat at $h \circ h'$ with respect to π.

Note that if h is such that for all $a \in A(h)$ it holds that $h \circ a \in Z$, then any profile π is threat free on h.

Definition 4.0.9 (Threat-free Nash equilibrium)**.** Let $\Gamma = (H, P, A, u)$ be an extensive game. A strategy profile σ^* is a *threat-free Nash equilibrium (TFNE)* if:

1. σ^* is a NE of Γ, and

2. for any $h \in H$, player $P(h)$ is not facing a threat at history h with respect to σ^*.

Note that in every profile that is in a TFNE, the effective play matches some SPE profile (more precisely, there is an SPE profile that yields exactly the same distribution on outcomes). This and other properties of threats and TFNE are formalized in the companion paper to this work [11].

In the definition of a threat we used the fact that $\mathrm{Cont}(h, \sigma, \tau)$ and $\mathrm{Cont}(h, \sigma, \sigma)$ each contain exactly one profile that is threat-free on h. To show that this must be the case, we have the following proposition, which is not unlike the fact that generic games have unique subgame perfect equilibria.

Proposition 4.0.1. *For any extensive game $\Gamma = (H, P, A, u)$, strategy profile σ, player i, history $h \in H \setminus Z$ with $P(h) = i$, and distribution τ over $A(h)$, the set $\mathrm{Cont}(h, \sigma, \tau)$ contains exactly one profile that is threat-free on h.*

Proof. For any history $h \in H \setminus Z$, let $\mathrm{height}(h)$ be the maximal distance between h and a descendant of h (i.e. the leaf that is furthest away from h but lies on the subtree rooted by h). The proof of the proposition is by induction on $\mathrm{height}(h)$.

For the base case $\mathrm{height}(h) = 1$, note that there is exactly one element in $\mathrm{Cont}(h, \sigma, \tau)$ and that this profile is threat-free on h (since h is a last move of the game).

Next, suppose the claim of the proposition holds for all histories h with $\mathrm{height}(h) < k$. We will prove that it holds for histories h with $\mathrm{height}(h) = k$. To this end, fix such a history h^0, and suppose the children of h^0 in the game tree are $h^1, \ldots, h^t$. Suppose also that $P(h^0) = i$ and $P(h^1) = \ldots = P(h^t) = -i$, and note that this is without loss of generality.

Consider the profile π^0 that is identical to σ except at history h, and fix $\pi^0(h) = \tau(h)$. We now repeat the following process in succession for each $j \in \{1, \ldots, t\}$: For any such j, let

$$\mathrm{TF}(h^j) \stackrel{\text{def}}{=} \left\{ \pi \in \bigcup_{\tau^j} \mathrm{Cont}(h^j, \pi^{j-1}, \tau^j) : \pi \text{ is threat-free on } h^j \right\}.$$

We then choose a profile $\pi^j \in \mathrm{TF}(h^j)$ that satisfies

$$u_{-i}\left(\pi^j\right) \geq u_{-i}\left(\pi''\right)$$

for all $\pi'' \in \mathrm{TF}(h^j)$. Because payoffs for player $-i$ are distinct, it must be the case that there exists a unique maximal π^j. That is, there can be no π'' that is different from π^j and has the same payoff for player $-i$.

After doing this for all $h^j \in \{h^1, \ldots, h^t\}$ we have a profile π^t that we claim is threat-free on h. To see this, observe that for all $j \in \{1, \ldots, t\}$, π^j is threat-free on h^j because we chose it to be a threat-free profile from $\mathrm{Cont}(h^j, \pi^{j-1}, \tau'')$. However, since for each j we chose a *maximal* τ^j, there are no threats at the histories h^j either. Finally, uniqueness of π^j is guaranteed by the fact that for each j, our choice of a maximal τ^j was unique. □

4.4.3 Round-Parameterized Version

For games induced by cryptographic protocols we will need a more general definition of TFNE. We assume that in these games players alternate moves, and thus there is a natural notion of the "rounds" in the game: Player i makes a move in round 1, then player $-i$ makes a move in round 2, and so on until the end of the game.

For the general definition, we introduce a few modifications to the vanilla version:

- We add a slackness parameter ε. This is necessary for our applications in order to handle the probability of error inherent in almost all cryptographic protocols.

- We allow players to be threatened at rounds, rather than just specific histories. This is needed because when we add the slackness parameter, a player might be threatened at a set of histories, where the weight of each individual threat does not exceed the slackness parameter, but the overall weight does.

- Finally, for a player to be threatened, we require that he improve on *all* threat-free continuations π. The reason we need this is that in the general case, there may be more than one π that is threat-free. If a player deviates from his prescribed behavior, he cannot choose *which* (threat-free) continuation will be played.

The definitions below make use of the notion of a round R strategy of player i: This is simply a function mapping every history h that reaches round R to a distribution over $A(h)$. For a round $R \in \mathbb{N}$ we let $\sigma_i(R)$ represent player i's round R strategy implied by σ. Let $\sigma(R) = (\sigma_1(R), \sigma_2(R))$, and let

$$\mathrm{Cont}(\sigma(1), \ldots, \sigma(R)) \overset{\text{def}}{=} \Big\{\pi \in T : \pi(S) = \sigma(S)\ \forall S \leq R\Big\},$$

where $T = (T_1, T_2)$ consists of constraints for players' strategies.

Definition 4.0.10 (ε-threat)**.** Let $\Gamma = (H, P, A, u)$ be an extensive game with constraints $T = (T_1, T_2)$. Let $\varepsilon \geq 0$, let $\sigma \in T$ be a strategy profile, and let $R \in \mathbb{N}$ be a round of Γ. Player $i = P(R)$ is facing an *ε-threat* at round R with respect to σ if there exists a round R strategy $\tau = \tau(R)$ for player i such that

(i) the set $\mathrm{Cont}(\sigma(1), \ldots, \sigma(R-1), \tau(R))$ is nonempty, and

(ii) for all $\pi \in \mathrm{Cont}(\sigma(1), \ldots, \sigma(R-1), \tau(R))$ and $\pi' \in \mathrm{Cont}(\sigma(1), \ldots, \sigma(R))$ that are ε-threat-free on R

$$\mathrm{E}\left[u_i\left(O(\pi)\right)\right] > \mathrm{E}\left[u_i\left(O(\pi')\right)\right] + \varepsilon,$$

where strategy π is *ε-threat-free on R* if for *all* rounds $S > R$ it holds that player $P(S)$ is not facing an ε-threat at round S with respect to π.

Note that if R is the last round of the game, then any profile $\pi \in T$ is ε-threat-free on R. Using Definition 4.0.10, we can now define an ε-TFNE.

Definition 4.0.11 (ε-threat-free Nash equilibrium)**.** Let $\Gamma = (H, P, A, u)$ be an extensive game with constraints $T = (T_1, T_2)$. A strategy profile $\sigma^* \in T$ is an *ε-threat-free Nash equilibrium (ε-TFNE)* if:

1. σ^* is an ε-NE of Γ, and

2. for any round R of Γ, player $P(R)$ is not facing an ε-threat at round R with respect to σ^*.

As is the case for Definition 4.0.8, Definition 4.0.10 (and hence Definition 4.0.11) would not be (semantically) well-defined if either one of the sets $\mathrm{Cont}(\sigma(1), \ldots, \sigma(R-1), \tau(R))$ or $\mathrm{Cont}(\sigma(1), \ldots, \sigma(R))$ would not contain at least one profile π that is ε-threat-free on R. The following proposition shows that this can never be the case.

Proposition 4.0.2. *Let $\Gamma = (H, P, A, u)$ be an extensive game with constraints $T = (T_1, T_2)$. Let $\varepsilon \geq 0$, let $\sigma \in T$ be a strategy profile, and let R be a round of Γ. For any round R strategy $\tau = \tau(R)$ for player $i = P(R)$, if the set $\mathrm{Cont}(\sigma(1), \ldots, \sigma(R-1), \tau(R))$ is nonempty then it contains at least one profile π that is ε-threat-free on R.*

Proof. For any round R of Γ, let $\mathrm{height}(R)$ be the distance between h and the last round of Γ. The proof of the proposition is by induction on $\mathrm{height}(R)$.

For the base case $\mathrm{height}(R) = 0$, note that, by the hypothesis of the proposition, the set $\mathrm{Cont}(\sigma(1), \ldots, \sigma(R-1), \tau(R))$ is nonempty. Since R is the last round of the game, the set contains exactly one profile, $(\sigma(1), \dots, \sigma(R{-}1), \tau(R))$, and this profile is vacuously ε-threat-free on R.

Next, suppose the claim of the proposition holds for all rounds R with $\mathrm{height}(R) < k$. We will prove that it holds for round R satisfying $\mathrm{height}(R) = k$. Let $i = P(R)$, and assume that there exists some $\pi' \in \mathrm{Cont}(\sigma(1), \dots, \sigma(R-1), \tau(R))$. We would like to show that $\mathrm{Cont}(\sigma(1), \dots, \sigma(R{-}1), \tau(R))$ contains at least one profile π that is ε-threat-free on R.

By the inductive hypothesis we have that, for any round $R+1$ strategy τ' of player $-i$, if the set $\mathrm{Cont}(\sigma(1), \dots, \sigma(R-1), \tau(R), \tau'(R+\ 1))$ is nonempty then it contains at least one profile that is ε-threat-free on $R+1$ (since $\mathrm{height}(R+1) < k$). We will choose a profile that has a *maximal* τ' as follows. Let

$$\mathrm{TF}(R+1) \overset{\text{def}}{=} \left\{ \pi \in \bigcup_{\tau'} \mathrm{Cont}(\sigma(1), \dots, \sigma(R-1), \tau(R), \tau'(R+1)) : \pi \text{ is } \varepsilon\text{-threat-free on } R+1 \right\},$$

and note that $\mathrm{TF}(R+1)$ must be nonempty. This is because there always exists at least one τ' for which $\mathrm{Cont}(\sigma(1), \dots, \sigma(R-1), \tau(R), \tau'(R+1))$ is nonempty: namely, we could have $\tau'(R+1) = \pi'(R+1)$. Since $\mathrm{Cont}(\sigma(1), \dots, \sigma(R-1), \tau(R), \pi'(R+1))$ is nonempty by assumption, it must contain a profile that is ε-threat-free on $R{+}1$ (by the inductive hypothesis).

We now choose a profile $\pi \in \mathrm{TF}(R+1)$ that satisfies

$$u_{-i}(\pi) \geq u_{-i}(\pi'') - \varepsilon$$

for all $\pi'' \in \mathrm{TF}(R+1)$. So now we have a profile $\pi \in \mathrm{Cont}(\sigma(1), \dots, \sigma(R{-}1), \tau(R))$, which we claim is ε-threat-free on round R. To see this, note that π is ε-threat-free on $R+1$ by the way we chose it (i.e. a profile from $\mathrm{Cont}(\sigma(1), \dots, \sigma(R-1), \tau(R), \tau'(R+1))$ that is ε-threat-free on $R+1$). However, since we chose a *maximal* τ' (up to ε), there is no ε-threat at round $R+1$ either. Thus π is ε-threat-free on R. □

4.5 The Computational Setting

In the following we explain how to use the notion of TFNE for cryptographic protocols. In Section 4.5.1 we describe how to view a cryptographic protocol as a sequence of extensive games. In Section 4.5.2 we show how to translate the behavior of an interactive TM to a sequence of strategies. In Section 4.5.3 we show how to express computational hardness in a game-theoretic setting. Finally, in Section 4.5.4 we give our definition of computational TFNE.

4.5.1 Protocols as Sequences of Games

When placing cryptographic protocols in the framework of extensive games, the possible messages of players in a protocol correspond to the available actions in the game tree, and the prescribed instructions correspond to a strategy in the game.

The protocol is parameterized by a security parameter $k \in \mathbb{N}$. The set of possible messages in the protocol, as well as its prescribed instructions, typically depend on this k. Assigning for each k and each party a payoff for every outcome, a protocol naturally induces a sequence $\Gamma^{(k)} = (H^{(k)}, P^{(k)}, A^{(k)}, u^{(k)})$ of extensive games, where:

- $H^{(k)}$ is the set of possible *transcripts* of the protocol (sequences of messages exchanged between the parties). A history $h \in H^{(k)}$ is *terminal* if the prescribed instructions of the protocol instruct the player whose turn it is to play next to halt on input h.
- $P^{(k)} : (H^{(k)} \setminus Z^{(k)}) \to \{1, 2\}$ is a function that assigns a "next" player to every non-terminal history.
- $A^{(k)}$ is a function that assigns to every non-terminal history $h \in H^{(k)} \setminus Z^{(k)}$ a set $A^{(k)}(h) = \{m : (h \circ m) \in H^{(k)}\}$ of possible protocol messages to player $P^{(k)}(h)$.[5]
- $u^{(k)} = (u_1^{(k)}, u_2^{(k)})$ is a vector of payoff functions $u_i^{(k)} : Z^{(k)} \to \mathbb{R}$.

A sequence $\Gamma = \{\Gamma^{(k)}\}_{k \in \mathbb{N}}$ of games defined as above is referred to as a *computational game*.

Remark 4.0.1. *In the following we will consider games played by Turing machines. Thus, actions will be represented by strings. As opposed to traditional game theory, where players are computationally unbounded, in our case the names of the actions will be significant. For example, in the One-way Permutation Game, if we encode player 1's action $f(x)$ by the string x for every $x \in \{0,1\}^k$, then inverting the one-way permutation becomes easy for player 2. However, to avoid too much notation, we will identify actions with their string representation. The reader should keep in mind, however, that actions are always strings, and that changing the string representation of actions might be* with *loss of generality.*

4.5.2 Strategic Representation of Interactive Machines

Protocols are defined in terms of *interactive Turing machines* (ITMs) – see [8] for a formal definition. More specifically, the prescribed behavior for each player is defined via an ITM, and any possible deviation of this player corresponds to choosing a different ITM. In order

[5] We can interpret "disallowed" messages in the protocol as abort, and define "abort" as a possible protocol message. This will imply that every execution of the protocol corresponds to some history in the game.

to argue about the protocol in a game-theoretic manner we formalize, using game-theoretic notions, the strategic behavior implied by ITMs. We believe this formalization is necessary for our treatment or any game-theoretic analysis of ITMs, in particular because, to the best of our knowledge, it has never been done before. However, because this section somewhat departs from the main thrust of the paper, the reader may skip to Section 4.5.3, keeping the following (informally stated) conclusion in mind: The strategic behavior of an ITM for player i in a protocol may be seen as a collection of independent distributions on actions, one for each of player i's histories that are reached with positive probability given the ITM of player i and some strategy profile of the other players. We refer to this collection as the behavioral reduced strategy induced by the ITM.

When considering some computational game $\Gamma^{(k)}$ in a sequence $\Gamma = \{\Gamma^{(k)}\}_{k \in \mathbb{N}}$ and an ITM "playing" this game (with input 1^k), the machine does not, strictly speaking, define a strategy. Informally, the machine specifies how to play *only on histories that are not inconsistent with the specification on earlier histories in the game*. That is, an ITM for player i specifies distributions on actions for all histories on which it is player i's turn, except those it cannot reach based on its own specification on earlier histories. This is the case, because when fixing the other player's moves, the distribution on actions the machine plays on a history that cannot be reached is simply undefined, as we are conditioning on an event with probability 0. In the following, we show that the prescribed behavior of an ITM can be seen as a convex combination of *reduced strategies* (which we call *mixed reduced strategy*), to be defined next. We then define the natural analogue of *behavioral reduced strategy*, and argue that for every mixed reduced strategy there exists a behavioral reduced strategy that is outcome-equivalent. We will eventually use behavioral reduced strategies to describe the behavior induced by ITMs.

Definition 4.0.12 (Reduced strategy (adapted from [27]))**.** Given a game $\Gamma = (H, P, A, u)$, a (pure) reduced strategy for player i is a function σ_i whose domain is a subset of $\{h \in H | P(h) = i\}$ with the following properties:

- For every h in the domain of σ_i it holds that $\sigma_i(h) \in A(h)$.
- $h = (a_1, \ldots, a_m)$ is in the domain of σ_i if and only if for any $1 \leq \ell \leq m-1$ such that $P(a_1, \ldots, a_\ell) = i$ it holds that $(a_1, \ldots, a_\ell)$ is in the domain of σ_i and $\sigma_i(a_1, \ldots, a_\ell) = a_{\ell+1}$.

Definition 4.0.13 (Mixed reduced strategy)**.** *A* mixed reduced strategy *for player i is a distribution over reduced strategies for player i.*

Given an ITM for $\Gamma^{(k)}$, for every instance of internal randomness for that machine (i.e., a vector of coins), the induced behavior of that ITM is exactly a reduced strategy. This is the case because for every profile of pure strategies (or reduced pure strategies) of the other players, the randomness naturally defines an action for every history that is consistent with its previous actions (the sequence of these actions, together with the profile, defines the outcome of the game), and on the other hand, naturally the randomness does not define an action for histories that are not consistent with that randomness (as with that randomness the machine will never reach these histories). It follows that an ITM defines a distribution over reduced (pure) strategies, i.e., a mixed reduced strategy. We now formalize this claim.

Definition 4.0.14 (Induced mixed reduced strategy of an ITM)**.** *Let M be a probabilistic ITM for player i in the extensive game Γ. Assume that M halts for any infinite vector of coins and any sequence of messages sent by the other players, and let t be a bound on the number of coins it reads. Let r be a (sufficiently long) coin vector for M. Then the induced pure reduced strategy $\sigma_i^{(r)}$ of M with randomness r is defined as follows:*

- *$h = (a_1, \ldots, a_m)$ is in the domain of $\sigma_i^{(r)}$ if and only if:*
  - *$P(a_1, \ldots, a_m) = i$;*
  - *For any $1 \leq \ell \leq m-1$ such that $P(a_1, \ldots, a_\ell) = i$ it holds that $(a_1, \ldots, a_\ell)$ is in the domain of $\sigma_i^{(r)}$ and when M with randomness r participates in an interaction, conditioned on the sequence of sent messages being $(a_1, \ldots, a_\ell)$ (where $a_{\ell+1}$ is a message sent by the ITM representing player $P(a_1, \ldots, a_\ell)$ for any $1 \leq \ell \leq m-1$), the message sent by M is $a_{\ell+1}$.*[6]
- *For any $h = (a_1, \ldots, a_m)$ in the domain of $\sigma_i^{(r)}$, the action $\sigma_i^{(r)}(a_1, \cdots, a_m)$ is the message sent by M with randomness r conditioned on the sequence of sent messages being $(a_1, \ldots, a_m)$.*

The mixed reduced strategy induced by M *is now defined as follows: the probability assigned to any pure reduced strategy σ is the probability that the induced reduced strategy of M with randomness r is σ, where r is uniformly chosen from U_t.*

In [27] it is shown that for perfect-recall extensive games (which are the only games we will consider here), every mixed strategy has a behavioral strategy that is outcome equivalent. (Two strategies are outcome-equivalent if for every profile of pure strategies of the other players the two strategies induce the same distribution on outcomes; A mixed strategy is a distribution on pure strategies). Next, we define the behavioral analogue of a mixed reduced strategy, and argue that the same holds for mixed and behavioral *reduced* strategies: For perfect-recall extensive games, every mixed reduced strategy has a behavioral reduced strategy that is outcome equivalent.

[6] For completeness, we may assume that whenever M outputs on history h an action that is not in $A(h)$, we interpret it as abort, which is denoted in the induced game by $\perp$ and is always a legal action.

Definition 4.0.15 (Behavioral reduced strategy)**.** *Given a game $\Gamma = (H, P, A, u)$, a behavioral reduced strategy for player i is a collection $\sigma_i = (\sigma_i(h))_{h\in\mathcal{H}}$ of independent probability measures, where $\mathcal{H}$ is a subset of $\{h \in H | P(h) = i\}$, with the following properties:*

- *$\sigma_i(h)$ is a probability measure over $A(h)$ for every h in $\mathcal{H}$.*
- *$h = (a_1, \ldots, a_m)$ is in $\mathcal{H}$ if and only if for any $1 \leq \ell \leq m-1$ such that $P(a_1, \ldots, a_\ell) = i$ it holds that $(a_1, \ldots, a_\ell) \in \mathcal{H}$ and $\sigma_i(a_1, \ldots, a_\ell)(a_{\ell+1}) > 0$.*

Claim 4.0.1. *Every mixed reduced strategy has a behavioral reduced strategy that is outcome equivalent.*

Proof Sketch 4.0.1. *Every pure reduced strategy σ_i for player i can be extended to a (full) pure strategy by assigning arbitrary values to all histories in $\{h : P(h) = i\}$ for which σ_i is undefined. The two strategies will be outcome-equivalent, as the outcome is only affected by the consistent histories of σ_i. It follows that every mixed reduced strategy can be extended to a mixed (full) strategy that is outcome-equivalent.*

On the other hand, every behavioral strategy $\sigma_i = (\sigma_i(h))_{h:P(h)=i}$ can be restricted to a behavioral reduced strategy by restricting the collection of probability measures accordingly. Again, the two strategies will be outcome-equivalent, as the distribution on outcomes is only affected by the consistent histories of σ_i.

Finally, as mentioned above, in [27] it is shown that for perfect-recall extensive games, every mixed strategy has a behavioral strategy that is outcome equivalent.

Thus, given some mixed reduced strategy we extended it to a mixed strategy that is outcome-equivalent, then transform it to a behavioral strategy that is outcome-equivalent, and finally we restrict the resulting behavioral strategy to an outcome-equivalent behavioral reduced strategy.

As argued above, ITMs induce mixed reduced strategies, and by Claim 4.0.1, these induce behavioral reduced strategies. Thus, in the following we will model ITMs by behavioral reduced strategies. This is captured by the notion of *strategic representation*.

Definition 4.0.16 (Strategic representation of an ITM)**.** *Let Γ be a game and let $i \in \{1, 2\}$. Let M be an ITM for player i. Assume that M halts for any infinite vector of coins and any sequence of messages sent by the other players. Let σ be the mixed reduced strategy induced by M. Then the* strategic representation *of M is the behavioral reduced strategy that is outcome-equivalent to σ.*[7]

[7] In certain games there may be more than one behavioral reduced strategy that is outcome-equivalent to σ. However, our treatment will always be indifferent to the actual choice.

Similarly, for a sequence of games $\{\Gamma^{(k)}\}_{k\in\mathbb{N}}$ and an ITM M that takes a security parameter 1^k, the strategic representation of M is the sequence of strategic representations of $M(1), M(1^2), M(1^3), \ldots$.

ε-TFNE for Reduced Strategies

In Section 4.4.3 we presented our general definition of TFNE. However, that definition was framed for strategies and, following the conclusion of the previous section, we actually care about reduced strategies. To make Definition 4.0.11 work for reduced strategies we notice that only two small changes need to be made: We need to define the notion of a round R reduced strategy, and we need to allow the constraint sets T_1 and T_2 to include behavioral reduced strategies.

Definition 4.0.17 (Round R reduced strategy)**.** *Let $\Gamma = (H, P, A, u)$ be an extensive game, let R be a round of Γ, and let σ_i be a behavioral reduced strategy of player $i = P(R)$. Then $\tau = \tau(R)$ is a* round R reduced strategy *of player i consistent with σ_i if the following hold:*

- *When $R = 1$, $\tau(1)$ is a distribution over $A(\epsilon)$.*
- *Otherwise, there exists some behavioral reduced strategy π_i of player i for which $\pi_i(j) = \sigma_i(j)$ for all $j \in \{1, \ldots, R-1\}$, and such that $\pi_i(R) = \tau_i(R)$.*

Throughout the paper, the behavioral reduced strategy σ_i with which $\tau(R)$ is consistent will be evident from the context, and so we omit reference to this consistency requirement.

Next, we modify the definition of constraints (Definition 4.0.6) by allowing each constraint set T_i to be a subset of $\bigotimes_{h:P(h)=i}(\Sigma_i(h) \cup \perp)$, where $\sigma_i(h) = \perp$ if the history h is not in the domain of the reduced strategy σ_i.

Finally, we observe that, following the two modifications above, Definitions 4.0.10 and 4.0.11 work for behavioral reduced strategies as well (replacing "strategy" by "behavioral reduced strategy" and "round R strategy" by "round R reduced strategy").

4.5.3 Computational Hardness in the Game-Theoretic Setting

The security of cryptographic protocols stems from the assumption on the limitation of the computational power of the players. In our strategic analysis of games, we also expect to deduce the (sequential) equilibrium from this limitation. However, because protocols are parameterized by a security parameter, a strategic analysis of protocols requires dealing with a *sequence* of games rather than a single game. While relating to the sequence of games is crucial in order to express computational hardness (as this hardness is defined in an asymptotic manner), this raises a new difficulty: How do we extend the definition of TFNE to sequences of games?

An appealing approach might be to try to define empty threats for sequences of games. That is, one might consider the effect of deviations on the expected payoff as k goes to infinity (much like the derivation of CNE from NE). However, to the best of our understanding this approach cannot work. Loosely speaking, this is because in order to relate to empty threats one has to consider deviations in internal nodes of the game tree, and it is not clear how to define such deviations for sequences of games. Typically, the structure of the game tree changes with k, so it is not clear even how to define an "internal node" in a *sequence* of games.

Instead, our approach insists on analyzing empty threats for *individual* games. Thus, our solution concept reflects a hybrid approach that relates to a protocol both as a family of *individual, extensive games* and as a *sequence* of *normal-form games*. To eliminate empty threats one must relate to the *interactive* aspect of each *individual* game (as this is the setting where threats are defined). In order to claim players are playing optimally under their computational constraints, one must think of the protocol as a *sequence* of *one-shot* games (because computational hardness is meaningful only when players are required to choose their machines in advance, and as the traditional notion of hardness is stated asymptotically).

Strategy-filters

When considering computational games $\Gamma = \{\Gamma^{(k)}\}_{k\in\mathbb{N}}$, the computational bounds on the players will be expressed by restricting the space of available strategies for the players. The available sequences of reduced strategies for the players will be exactly those that can be played by the ITMs that meet the computational bound on the players. In our case we will consider PPT ITMs.

While on the one hand every PPT ITM fails on cryptographic challenges for large enough values of the security parameter k (under appropriate assumptions), on the other hand, PPT ITMs can have arbitrarily large size and thus arbitrarily much information hardwired, and so for every k there is a PPT ITM that breaks the cryptographic challenges with security parameter k. In our analysis, we would like to "filter" machines according to their ability to break cryptographic challenges for specific k's, and allow using them only in games that correspond to large enough k's, where these machines fail (and in particular, cannot use hardwiring to solve the cryptographic challenges).

To this end, we define the notion of a *strategy-filter*. For each value k of the security parameter and value ε, a strategy-filter maps the ITM M to either $\perp$ or to its strategic representation, according to whether $M(1^k)$ violates level of security ε or does not (respectively).

Definition 4.0.18 (Strategy-filter)**.** Let $\Gamma = \{\Gamma^{(k)}\}_{k\in\mathbb{N}}$ be a computational game and let i be a player. A *strategy-filter* is a sequence $F_i = \{F_i^{(k)} : \mathcal{M} \times [0,1] \to \Sigma_i^{(k)} \cup \{\perp\}\}_{k \in \mathbb{N}}$ such that for every ITM M, every $k \in \mathbb{N}$ and every $\varepsilon \in [0,1]$, it holds that either $F_i^{(k)}(M, \varepsilon) = \perp$, or $F_i^{(k)}(M, \varepsilon) = \sigma_i^{(k)}$, where $\sigma_i^{(k)}$ is the strategic representation of the machine $M(1^k, \cdot)$.

A strategy-filter is meaningful if it allows us to reason about all reduced strategies that are considered to be feasible, in our case PPT implementable reduced strategies, and in particular does not filter them out. This is captured in the following definition.

Definition 4.0.19 (PPT-covering filter)**.** A strategy-filter F_i is said to be *PPT-covering* if for every PPT ITM M and any positive polynomial $p(\cdot)$ there exists k_0 such that for all $k \geq k_0$, it holds that $F_i^{(k)}(M, 1/p(k)) \neq \perp$.

Typically, protocols have the following security guarantee (under computational assumptions): for every i, every PPT ITM M of P_i and every polynomial $p(\cdot)$, there exists k_0 such that for any $k \geq k_0$, the ITM M does not break level of security $1/p(k)$ in the protocol with security parameter k. Such a protocol will naturally have a PPT-covering filter, where if $F_i^{(k)}(M, \varepsilon) \neq \perp$ then the reduced strategy $F_i^{(k)}(M, \varepsilon)$ "does not break level of security ε in the game $\Gamma^{(k)}$."

Tractable Reduced Strategies

As reflected above, the asymptotic nature of defining security does not determine any level of security for any k. Rather, it dictates that any PPT ITM "eventually fails in violating $1/p(k)$ security" for any $p(\cdot)$ (where "eventually" means for large enough k). Thus, we follow the same approach in our game theoretic analysis: roughly speaking, our solution concept requires that ε-security will imply ε-stability for any k (rather than requiring a particular level of stability for each k). More formally, we require that for any k and any ε, the game induced by the protocol with security parameter k be in ε-TFNE, given that the available strategies for the players are those that do not break level of security ε. Thus, for any pair (k, ε) we will consider the game $\Gamma^{(k)}$ with available reduced strategies restricted to those that guarantee ε-security. The following definition derives from a PPT-covering filter, for each such game, the set of available reduced strategies for each player.

Definition 4.0.20 (Tractable reduced strategies)**.** Let F_i be a PPT-covering filter. For every $k \in \mathbb{N}$ and $\varepsilon \in [0,1]$ we define the set $T_{i,\varepsilon}^{(k)}(F_i)$ of (k, ε)-tractable reduced strategies for player $i \in \{1, 2\}$ as

$$\{F_i^{(k)}(M, \varepsilon) | M \text{ is a PPT ITM and } F_i^{(k)}(M, \varepsilon) \neq \perp\}.$$

Whenever F_i will be understood from the context, we will write $T_{i,\varepsilon}^{(k)}$ to mean $T_{i,\varepsilon}^{(k)}(F_i)$.

4.5.4 Computational TFNE

We can now define our computational variant of TFNE. Roughly, the definition requires that there exist a family of PPT compatible constraints such that for any k and any ε, the strategies played by the machines on input security parameter k are in ε-TFNE in the game indexed by (k, ε).

Definition 4.0.21 (Computational TFNE)**.** Let Γ be a computational game. A pair of PPT machines (M_1, M_2) is said to be in a *computational threat-free Nash equilibrium (CTFNE)* of Γ if there exists a pair of PPT-covering filters (F_1, F_2) such that for every k, ε for which $F_1^{(k)}(M_1, \varepsilon)$ and $F_2^{(k)}(M_2, \varepsilon)$ are tractable the profile $(F_1^{(k)}(M_1, \varepsilon), F_2^{(k)}(M_2, \varepsilon))$ constitutes an ε-TFNE in the $(T_{1,\varepsilon}^{(k)}, T_{2,\varepsilon}^{(k)})$-constrained version of $\Gamma^{(k)}$.

The expressive power of Definition 4.0.21 is illustrated through the following claim, which refers to Example 4.0.2. We omit the proof, and proceed to more interesting applications in sections 4.6 and 4.7.

Claim 4.0.2. *In the modified one-way permutation game,*

(i) the strategy profile in which P_1 plays 0 and P_2 plays 0 after a history of 0 and randomly otherwise is a CTFNE, and

(ii) any profile in which P_2 plays randomly after history 0 is not a CTFNE.

We note that part (ii) of the claim can easily be extended to profiles in which, after history 0, P_2 plays 0 with probability at most $1 - p(k)$ for any polynomial p.

4.6 The Coin-Flipping Game

In the following we describe a classic protocol for coin-flipping, formulated as a sequence of games (parameterized by a security parameter k). We then show that the prescribed behavior according to that protocol constitutes a CTFNE in the sequence of games.

Following is an informal description of the sequence of games. We assume some perfectly binding commitment scheme with the following properties (see Appendix 4.9 for a formal definition):

- For any security parameter k (which is a common input to the sender and receiver), the "commit" phase consists of one message from the sender to the receiver, denoted $\mathsf{com}^{(k)}$, which is of length bounded by $p(k)$ for some polynomial p.

- For any PPT ITM, the advantage in guessing the committed value given the aforementioned message is negligible in k.

The description defines the legal messages in each game. Recall that at any phase where a player is supposed to send a message, the move "abort" is legal (and well-defined). Note also, that any illegal message is interpreted as abort by the other player. The game $\Gamma^{(k)}$ is defined as follows:

1. Player 1 chooses a string c of length at most $p(k)$ and sends it to player 2.
2. Player 2 chooses a bit r_2, and sends r_2 to player 1.
3. Player 1 does one of the following: (1) sends to player 2 decom, where decom is a legal decommitment to c revealing that the committed value was $1 - r_2$ (in that case the payoffs are (1,0)); or (2) aborts (in that case the payoffs are (0,1)).

Any other abort results in the aborting player receiving payoff 0, and the other player receiving 1.

We now describe a pair of interactive ITMs for the game $\Gamma^{(k)}$ that form a CTFNE. We describe them interleaved, in the form of a protocol. We denote the ITMs playing the strategies of P_1, P_2 by M_1, M_2, respectively.

1. Player 1 chooses a random bit r_1, and sends $c = \mathsf{com}^{(k)}(r_1)$ to player 2 (player 1 also obtains decom, which is a legal decommitment to c).
2. Player 2 chooses a random bit r_2, and sends r_2 to player 1.
3. If $r_1 \neq r_2$, player 1 sends decom to player 2. Else, player 1 aborts.

Theorem 4.0.1. *The pair (M_1, M_2) forms a CTFNE for the protocol above.*

Proof. First we define the functions $F_1^{(k)}$ and $F_2^{(k)}$. For any k, the function $F_1^{(k)}$ never maps to $\perp$ (this, roughly speaking, reflects the fact that the protocol is secure against an all-powerful player 1). For F_2 we use the following rule: $F_2^{(k)}(M, \varepsilon) = \perp$ if and only if "for security parameter k, the PPT ITM M guesses the committed value with advantage greater than ε." More formally, $F_2^{(k)}(M, \varepsilon) = \perp$ if and only if when player 1 sends as the first message a random commitment of a random bit (i.e., chooses a random bit and then uses the aforementioned commitment scheme using uniformly random coins), then the message with which M reacts is the committed value of player 1 with probability greater than $1/2 + \varepsilon$.

The fact that F_1 is PPT-covering is straightforward. The fact that F_2 is PPT covering follows directly from the security of the commitment scheme: For any positive polynomial

p, every PPT ITM has advantage smaller than $1/p(k)$ in guessing the committed value with security parameter k, for large enough k's.

Next, we need to show that for every k, ε for which $F_1^{(k)}(M_1, \varepsilon) \neq \perp$ and $F_2^{(k)}(M_2, \varepsilon) \neq \perp$ the profile $(F_1^{(k)}(M_1, \varepsilon), F_2^{(k)}(M_2, \varepsilon))$ constitutes an ε-TFNE in the $T = (T_{1,\varepsilon}^{(k)}, T_{2,\varepsilon}^{(k)})$-constrained version of $\Gamma^{(k)}$. Let k, ε be as above, and let $\sigma = (\sigma_1, \sigma_2) = (F_1^{(k)}(M_1, \varepsilon), F_2^{(k)}(M_2, \varepsilon))$. We first show that σ constitutes an ε-NE in the T-constrained version of $\Gamma^{(k)}$.

The strategy σ_1 chooses a random commitment of a random bit in round 1, and in round 3 decommits whenever it can. It is easy to see that this is optimal, as player 2 always guesses the committed value with probability $1/2$, and so there is no strategy for player 1 for which he can decommit with probability greater than $1/2$ in round 3. It is also easy to see that player 2's strategy is an ε best-response, as any PPT ITM M_2 for player 2 for which $F_2^{(k)}(M_2, \varepsilon) \neq \perp$ does not guess with advantage more than ε. We conclude that σ constitutes an ε-NE in the T-constrained version of the game $\Gamma^{(k)}$.

Next, we show that no player is facing an ε-threat with respect to σ at any round of the T-constrained version of $\Gamma^{(k)}$. Note that for both players, the expected payoff according to σ is $1/2$. Suppose some player is facing an ε-threat with respect to σ. We divide the proof into cases.

Case 1 – P_1 is facing an ε-threat in round 3: In order for P_1 to improve in Step 3 by more than ε, it must play a round 3 strategy $\tau(3)$ in which he sends decom that proves that $r_1 \neq r_2$ with larger probability than in σ. However, since in σ player 1 sends decom whenever $r_1 \neq r_2$ (and otherwise no such decom exists, since the commitment is perfectly binding), we conclude that no such $\tau(3)$ exists.

Case 2 – P_2 is facing an ε-threat in round 2: According to the constraints, P_2 cannot guess r_1 with probability greater than $1/2 + \varepsilon$. So in order for him to improve by *more* than ε, it must be the case that he has some round 2 strategy $\tau(2)$, such that in any ε-threat-free continuation in $\mathrm{Cont}(\sigma(1), \tau(2))$ player 1 aborts with positive probability conditioned on $r_1 \neq r_2$. However, any continuation where P_1 aborts with zero probability conditioned on $r_1 \neq r_2$ (and sends decom) is ε-threat-free, and so there is no deviation for P_2 for which he improves on *all* ε-threat-free continuation.

Case 3 – P_1 is facing an ε-threat in round 1: Since σ is ε-threat-free on round 1, if P_1 is threatened in round 1 then he has a round 1 strategy $\tau(1)$ so that for all ε-threat-free profiles in $\mathrm{Cont}(\tau(1))$ his expected payoff is greater than $1/2 + \varepsilon$. Consider the profile $\sigma' = (\tau(1), \sigma(2), \sigma(3))$. This profile gives both players an expected payoff of $1/2$ (assuming $\tau(1)$ aborts with probability 0, which is clearly optimal), and is ε-threat-free on round 2 (by the same argument as Case 1 above). If σ' is ε-threat-free on round 1 as well, then P_1 does not improve by more than ε using the deviation $\tau(1)$. If σ' is not ε-threat-free on round 1, then in any ε-threat-free profile in $\mathrm{Cont}(\tau(1))$ player 2's payoff must be greater than $1/2 + \varepsilon$. However, this means that P_1's payoff is less than $1/2$, and again he does not improve using the deviation $\tau(1)$. Hence, the postulated $\tau(1)$ does not exist, and so P_1 is not facing an ε-threat in round 1. □

4.7 Correlated Equilibria Without a Mediator

In one of the first papers to consider the intersection of game theory and cryptography, Dodis, Halevi and Rabin proposed an appealing methodology for implementing a correlated equilibrium in a 2-player normal-form game without making use of a mediator [6]. Under standard hardness assumptions, they showed that for any 2-player normal-form game Γ and any correlated equilibrium σ for Γ, there exists a new 2-player extensive "extended game" Γ' and a CNE σ' for Γ', such that σ and σ' achieve the same payoffs for the players. (Strictly speaking Γ' is a sequence of games indexed by a security parameter, and a CNE is defined for a sequence.) However, as already pointed out by Dodis et al., their protocol lacks a satisfactory analysis of its sequential nature – the resulting "extended game" is an extensive game, but the solution concept they use, CNE, is not strong enough for these games.

In the following, we extend the definition of CTFNE to allow handling this setting (that is, we define CTFNE for extensive games with simultaneous moves at the leaves), give some justification for our new definition, and then provide a new protocol for removing the mediator that achieves CTFNE in a wide class of correlated equilibria that are in the convex hull of Nash equilibria (see definition below).

4.7.1 The Dodis-Halevi-Rabin Protocol

The "extended game" Γ' consists of 2 phases. In the first phase ("preamble phase"), the players execute a protocol for sampling a pair under the distribution σ, and in the second phase each player plays the action implied by the sampled pair, in the original normal-form game. The CNE of the extended game is the profile that consists of each player playing the protocol honestly in the first phase, and then in the second phase, if the other player did not abort, choosing the action by the protocol's result, and otherwise "punishing" the other player by choosing a "min-max" action (i.e., choosing an action minimizing the utility resulting from the other player's best response).

This profile is indeed a CNE because an efficient player can achieve only a negligible advantage by trying to break the cryptography in the first phase, cannot achieve any advantage by aborting in the first phase (as this minimizes its best possible move in the second phase), and cannot gain any advantage in the expectation of the payoff by deviating in the second phase, because the players are playing a pair of actions from a correlated equilibrium.

4.7.2 TFNE for Games with Simultaneous Moves at the Leaves

The definition of an extensive game with simultaneous moves is similar to the definition of an ordinary extensive game. The main difference is that now the function P maps to (nonempty) sets of players rather than to single players. The definition of history is then changed to a sequence of sets of actions rather than a sequence of actions, and the definitions of a strategy and a payoff function are both also changed accordingly. For a formal definition see Osborne and Rubinstein [27].

In order to adjust our definition for extensive games with simultaneous moves, we notice that when a player deviates on a history with a simultaneous move, he cannot expect the other to react to this deviation (because they both play at the same time). However, in order to argue that a profile is rational, we still need to require that for every simultaneous move in the equilibrium support, each player is playing a "best response" given the other player's prescribed behavior. This means the prescribed behavior for the players should form some kind of equilibrium for normal-form games. In our case, the prescribed behavior will form a NE. The question of what should a CTFNE profile prescribe in off-equilibrium-support histories is more delicate: Clearly, in order to claim that the profile is "rational," again we need some kind of equilibrium for normal-form games. In our case the only deviation will be prematurely aborting without completing the preamble phase, which leads to the original normal-form game without agreeing on a sampled pair. In this case one can argue that after one player aborted, the other (non-aborting) player cannot assume the aborting player will play his prescribed behavior in the simultaneous move (as he is already not following his prescribed behavior). However, we argue that it is in fact still rational to assume the aborting player will play his prescribed behavior. The justification for this claim is essentially the same as the justification for the rationality of NE. Once there is a prescribed behavior that is a NE, each player knows the other has no incentive to deviate, and so he also has no incentive to deviate. The essential difference between a deviation in an extensive game and a deviation in a simultaneous move, is that in the former, once a player deviated, the other player is facing a fact. He now has to readjust his behavior according to this deviation. However, in the latter, there is no point for a player to deviate from the prescribed NE, because the other player will not know about this deviation prior to choosing his move (if at all). Thus, for terminal leaves that are off-equilibrium-support (i.e., in the original normal-form game that follows an abort of some player), we claim it is sufficient for a CTFNE to prescribe a NE as well.

The bottom line of this discussion is that players cannot assume other players will deviate from any prescribed NE in any terminal leaf. Thus, our new definition of TFNE for extensive games with simultaneous moves at the leaves (abbreviated GSML) is essentially the same as the original definition, except that (i) we require a profile in TFNE to prescribe a NE in any terminal leaf, and (ii) in the definition of a threat we do not allow a player to assume the other will deviate from his strategy in any NE at a terminal leaf. In order to formally modify our definition of TFNE to achieve (ii), essentially we would need to define the only threat-free continuation on a leaf to be the one that assigns to the players the actions in the prescribed NE (which expresses the idea that a player is not allowed to assume the other will deviate from his strategy in any NE).

However, we adopt an equivalent, simpler convention. Given a GSML Γ and a profile σ that assigns a NE at every simultaneous move, we look at a slightly modified game Γ': All simultaneous moves are removed, and instead at each leaf where a simultaneous move was removed each player is assigned his expected payoff in the corresponding NE for that leaf. Note that the modified game is now a regular extensive game with *no* simultaneous moves. We then "prune" the strategy profile to remove all the distributions on actions on all simultaneous leaves and denote the resulting profile σ'. We say that σ is a TFNE in Γ if σ' is a TFNE in Γ'. We call Γ' and σ' the *pruned representation* of Γ and σ.

The definition of CTFNE for GSML is derived from the above definition of TFNE for GSML, similarly to the derivation of CTFNE from TFNE in the non-simultaneous case.

A note on the strength of our definition It seems that for general GSMLs our definition is too strong. The reason is that in certain cases it is computationally intractable for the players to play the prescribed NE in every leaf (it is easy to construct simple sequences of games where one cannot assign tractable Nash equilibria at all leaves). While we do not yet know how to relax our definition to apply to these cases, we believe our definition, when met, is sufficient.

4.7.3 Our Protocol

For a non-trivial class of correlated equilibria, we show how to modify the DHR protocol to achieve CTFNE. Our basic idea is to use Nash equilibria as "punishments" for aborting players. That is, if there is a NE that assigns to a player a payoff at most his expected payoff when not aborting, then assigning this NE in case he aborts serves as a punishment and yields that the player has no incentive to abort. In the following we characterize a family of correlated equilibria for which we can use the aforementioned punishing technique, and prove that for this family we can remove the mediator while achieving CTFNE.

We say that a correlated equilibrium π is a *convex combination of Nash equilibria* if π is induced by a distribution on (possibly mixed) Nash equilibria. (The set of such distributions is sometimes referred to as the *convex hull of Nash equilibria*.) Note that any such distribution is a correlated equilibrium (CE), but the converse is not true.

Let π be a correlated equilibrium for a two-player game Γ that is a convex combination of a set N of NEs. We say that π is *weakly Pareto optimal* if there does not exist a different CE ρ in the convex hull of N for which both $\mathrm{E}[u_1(O(\rho))] > \mathrm{E}[u_1(O(\pi))]$ and $\mathrm{E}[u_2(O(\rho))] > \mathrm{E}[u_2(O(\pi))]$.

We say that a distribution is *samplable* if there exists a probabilistic TM that halts on every infinite randomness vector, and can sample it. This is equivalent to requiring that all probabilities can be expressed in binary (assuming we work over $\{0, 1\}$). Note that every distribution can be approximated arbitrarily accurately by a samplabale distribution.

Theorem 4.0.2. Assume there exists a non-interactive computationally binding commitment scheme. Let π be a weakly Pareto optimal correlated equilibrium for a two-player game Γ that is a samplable convex combination Π of some set of samplable Nash equilibria. Then there exists an extended extensive game and a profile that achieves the same expected payoffs as π and is a CTFNE.

Proof. Since Π is samplable, the common denominator of all probabilities in Π is a power of two. Thus, we can assume Π is a *uniform* distribution on a sequence of Nash equilibria that may contain repetitions, where the length of the sequence is a power of two. Let 2^ℓ be the length of that sequence, and let $(\pi_{0^\ell}, \ldots, \pi_{1^\ell})$ be that sequence. Note that the distribution π can now be generated by first choosing uniformly at random a string r in $\{0, 1\}^\ell$, and then choosing a pair of actions according to π_r.

Let $\widehat{\sigma}^i$ be the NE that assigns the worst payoff for P_i (this value represents the "severest punishment" for player i).

Our protocol embeds a 2-party string sampling protocol, which is a simple generalization of the Blum coin flipping protocol [5]. The protocol consists of simply running the Blum protocol in parallel for a fixed number of times. This protocol, in turn, relies on a perfectly binding commitment scheme as in Section 4.6, whose formal definition can be found in Appendix 4.9.

As in Section 4.6, we describe the two ITMs that form the protocol in an interleaved manner. We denote the ITMs playing the strategies of P_1, P_2 by M_1, M_2, respectively.

- Round 1: Player 1 chooses uniformly at random a string $r = (r_1, \ldots, r_\ell)$ from $\{0, 1\}^\ell$, and sends $c = (c_1 = \mathsf{com}^{(k)}(r_1), \ldots, c_\ell = \mathsf{com}^{(k)}(r_\ell))$ to player 2 (player 1 also obtains $(\mathsf{decom}_1, \ldots, \mathsf{decom}_\ell)$, where decom_i is a legal decommitment with respect to c_i and r_i).

- Round 2: If player 1 aborted, the assigned NE is $\widehat{\sigma}^1$. Else, player 2 chooses a uniformly random string $r' = (r'_1, \ldots, r'_\ell)$ from $\{0, 1\}^\ell$, and sends r' to player 1.

- Round 3: If player 2 aborted, the assigned NE is $\widehat{\sigma}^2$. Else, player 1 sends the message $((r_1, \mathsf{decom}_1), \ldots, (r_\ell, \mathsf{decom}_\ell))$.
- If player 1 aborted, the assigned NE is $\widehat{\sigma}^1$. Else, player 2 verifies that decom_i is a legal decommitment with respect to c_i and r_i for $1 \leq i \leq \ell$. If the verification fails (which is equivalent to an abort of player 1, as it means player 1 sent an illegal message), the assigned NE is $\widehat{\sigma}^1$. Else, the assigned NE is $\pi_{r \oplus r'}$ (where $\oplus$ is bitwise exclusive-or).

Lemma 4.0.1. *The pair (M_1, M_2) forms a CTFNE for the protocol above.*

Proof. Let $\{\tilde{\Gamma}^{(k)}\}_{k \in \mathbb{N}}$ be the sequence of games induced by the protocol. Denote the pruned representation of $\tilde{\Gamma}^{(k)}$ by $\Gamma^{(k)}$. Let $\tilde{\sigma}_1^{(k)}, \tilde{\sigma}_2^{(k)}$ be the strategies of P_1, P_2 in the protocol with security parameter k, and let $\sigma_1^{(k)}, \sigma_2^{(k)}$ be their pruned representations. Let $\sigma^{(k)} = (\sigma_1^{(k)}, \sigma_2^{(k)})$. We prove that $\{\sigma^{(k)}\}$ is a CTFNE in $\{\Gamma^{(k)}\}$, which, by the discussion of Section 4.7.2, implies that $\{\tilde{\sigma}^{(k)}\}$ is a CTFNE in $\{\tilde{\Gamma}^{(k)}\}$.

First we define the functions $F_1^{(k)}$ and $F_2^{(k)}$. For any k, the function $F_1^{(k)}$ never maps to $\perp$ (this, roughly speaking, reflects the fact that the protocol is secure against an all-powerful player 1, which follows from the perfect binding property of the commitment scheme). For F_2 we use the following rule: $F_2^{(k)}(M, \varepsilon) = \perp$ if and only if

$$\mathrm{E}[u_2^{(k)}(O(\sigma_1^{(k)}, \sigma_M^{(k)}))] \geq \mathrm{E}[u_2^{(k)}(O(\sigma^{(k)}))] + \varepsilon, \tag{4.1}$$

where $\sigma_M^{(k)}$ is the strategic representation of machine M and $\sigma_1^{(k)}$ is the strategic representation of machine M_1, both with security parameter k. In other words, P_2 cannot unilaterally ε-improve in the $(T_{1,\varepsilon}^{(k)}, T_{2,\varepsilon}^{(k)})$-constrained version of $\Gamma^{(k)}$.

The fact that F_1 is PPT-covering is straightforward. The fact that F_2 is PPT covering follows from the security of the commitment scheme, as we prove next.

Claim 4.0.3. *The strategy-filter F_2 is PPT-covering.*

Proof. Suppose F_2 is not PPT-covering. Then from (4.1) there is a PPT ITM M and a polynomial p such that

$$\mathrm{E}[u_2^{(k)}(O(\sigma_1^{(k)}, \sigma_M^{(k)}))] \geq \mathrm{E}[u_2^{(k)}(O(\sigma_1^{(k)}, \sigma_2^{(k)}))] + 1/p(k) \tag{4.2}$$

for infinitely many k's, where $\sigma_M^{(k)}$ is the strategic representation of the machine M with security parameter k.

First, we show that we can assume M does not abort in round 2. An abort of P_2 leads to a leaf with $\widehat{\sigma}^2$. But since π is a convex combination of NE's, following the protocol would mean playing a NE. Since by definition $\widehat{\sigma}^2$ is the worst NE for player 2, it follows that the machine M' that behaves the same as M, but whenever M aborts, M' instead follows the protocol, i.e., acts like M_2, does at least as well as M. The machine M' is well-defined, as the reduced strategy $\sigma_2^{(k)}$ is in fact a full strategy, and is defined everywhere.[8]

Since the payoffs in $\{\Gamma^{(k)}\}$ are bounded in k and the number of NEs in π is fixed in k, by (4.2) there exists a polynomial p and (at least one) $s \in \{0,1\}^\ell$ such that infinitely often

$$\Pr[O(\sigma_1^{(k)}, \sigma_M^{(k)}) = \pi_s] - \Pr[O(\sigma_1^{(k)}, \sigma_2^{(k)}) = \pi_s] \geq 1/p(k).$$

It follows that infinitely often

$$\Pr_{(\sigma_1^{(k)},\sigma_M^{(k)})}[r \oplus r' = s] - \Pr_{(\sigma_1^{(k)},\sigma_2^{(k)})}[r \oplus r' = s] \geq 1/p(k). \tag{4.3}$$

Claim 4.0.4. *There exists a polynomial q such that for each k satisfying (4.3) there exists some $i \in \{1, \ldots, \ell\}$ for which*

$$\Pr_{(\sigma_1^{(k)},\sigma_M^{(k)})}[r_i \oplus r'_i = s_i | r_j \oplus r'_j = s_j \; \forall j < i] - 1/2 \geq 1/q(k). \tag{4.4}$$

Proof. We show that the claim holds with $q(k) = 2^\ell \cdot p(k)$. Suppose towards contradiction this is not the case, and let k be such that (4.3) holds. Suppose (4.4) does not hold for any $i \in \{1, \ldots, \ell\}$. Then

$$\begin{aligned}
&\Pr_{(\sigma_1^{(k)},\sigma_M^{(k)})}[r \oplus r' = s] - \Pr_{(\sigma_1^{(k)},\sigma_2^{(k)})}[r \oplus r' = s] \\
&\qquad = \Pr_{(\sigma_1^{(k)},\sigma_M^{(k)})}[r_1 \oplus r'_1 = s_1] \cdot \Pr_{(\sigma_1^{(k)},\sigma_M^{(k)})}[r_2 \oplus r'_2 = s_2 | r_1 \oplus r'_1 = s_1] \cdot \ldots \\
&\qquad\quad \ldots \cdot \Pr_{(\sigma_1^{(k)},\sigma_M^{(k)})}[r_\ell \oplus r'_\ell = s_\ell | r_j \oplus r'_j = s_j \; \forall j < \ell] - \Pr_{(\sigma_1^{(k)},\sigma_2^{(k)})}[r \oplus r' = s] \\
&\qquad < \left(\frac{1}{2} + \frac{1}{q(k)}\right)^\ell - \frac{1}{2^\ell} \\
&\qquad < \frac{2^\ell}{q(k)} = \frac{1}{p(k)}.
\end{aligned}$$

[8] Note that we assume here that there exists such PPT ITM M'. This may not always be the case. One reason is that sometimes detecting with probability 1 whether M aborted cannot be done in polynomial time (or at all). The reason is that any illegal message is regarded as abort, but sometimes a party cannot "know" whether its message is illegal or not. See [3], Section 6.3 for an example. Another reason could be that in order to "emulate" M_2, the machine M' needs to be in some internal state. We note, however, that in our case both problems do not occur.

where the first inequality holds since the distribution on $r \oplus r'$ in $(\sigma_1^{(k)}, \sigma_2^{(k)})$ is uniform on $\{0,1\}^\ell$ and the second inequality follows from the observation that in $(1/2 + 1/q(k))^\ell$ we are summing over 2^ℓ terms, one equal to $1/2^\ell$ and the others strictly smaller than $1/q(k)$. This, of course, yields a contradiction to (4.3). □

Since there are infinitely many k's for which (4.4) holds, and because ℓ is fixed, there must exist some $i \in \{1, \ldots, \ell\}$ for which (4.4) holds infinitely often. This, however, yields a PPT machine A that breaks the hiding property of the commitment scheme, and is thus a contradiction: Given a commitment $c = \mathsf{com}^{(k)}(r)$ for a uniformly chosen random bit r, the machine A chooses uniformly at random a string $(r_1, \ldots, r_{i-1}, \ldots, r_{i+1}, \ldots, r_\ell)$ from $\{0,1\}^{\ell-1}$, and runs M on

$$(c_1 = \mathsf{com}^{(k)}(r_1), \ldots, c_{i-1} = \mathsf{com}^{(k)}(r_{i-1}), c, c_{i+1} = \mathsf{com}^{(k)}(r_{i+1}), \ldots, c_\ell = \mathsf{com}^{(k)}(r_\ell))$$

to get output r'. Then, if $r_j \oplus r'_j = s_j \; \forall j < i$, algorithm A outputs $s_i \oplus r'_i$, and otherwise A outputs a uniformly random bit. Clearly A is a PPT machine. From (4.3) it follows that infinitely often with probability at least $1/2^\ell$ it holds that $r_j \oplus r'_j = s_j \; \forall j < i$. Once r' is such that $r_j \oplus r'_j = s_j \; \forall j < i$, (4.4) implies that $\Pr[s_i \oplus r'_i = r_i | r_j \oplus r'_j = s_j \; \forall j < i] \geq 1/2 + q(k)$. Thus, in total infinitely often it holds that $\Pr[s_i \oplus r'_i = r_i] = (1 - 2^{-\ell}) \cdot \frac{1}{2} + 2^{-\ell} \cdot (\frac{1}{2} + q(k)) = \frac{1}{2} + 2^{-\ell} \cdot q(k)$, which means that A breaks the hiding property of the commitment scheme. □

Next, we show that for all k, ε for which $F_1^{(k)}(M_1, \varepsilon) \neq \perp$ and $F_2^{(k)}(M_2, \varepsilon) \neq \perp$ the profile $(F_1^{(k)}(M_1, \varepsilon), F_2^{(k)}(M_2, \varepsilon))$ constitutes an ε-TFNE in the $T = (T_{1,\varepsilon}^{(k)}, T_{2,\varepsilon}^{(k)})$-constrained version of $\Gamma^{(k)}$. Let k, ε be as above, and let $\sigma = (\sigma_1, \sigma_2) = (F_1^{(k)}(M_1, \varepsilon), F_2^{(k)}(M_2, \varepsilon))$. We first show that σ constitutes an ε-NE in the T-constrained version of $\Gamma^{(k)}$. Suppose P_1 unilaterally ε-improves in the T-constrained version of $\Gamma^{(k)}$. From similar arguments as above we can assume P_1 never aborts. But when P_1 never aborts the outcome is exactly π, as the players are playing $\pi_{r \oplus r'}$, and r' is chosen uniformly at random.

Suppose now that P_2 unilaterally ε-improves in the T-constrained version of $\Gamma^{(k)}$. However, this is a contradiction to the constraints, that state that for any k P_2 cannot unilaterally ε-improve in the $(T_{1,\varepsilon}^{(k)}, T_{2,\varepsilon}^{(k)})$-constrained version of $\Gamma^{(k)}$.

Next, we show that no player is ε-threatened with respect to σ at any round of the T-constrained version of $\Gamma^{(k)}$. To this end, suppose towards a contradiction that some player is ε-threatened with respect to σ. We divide the proof into cases.

Case 1 – P_1 is facing an ε-threat in round 3: In step 3 player 1 has exactly two options: He can (i) play honestly, send $((r_1, \mathsf{decom}_1), \ldots, (r_\ell, \mathsf{decom}_\ell))$ which he generated in round 1, and receive $\mathrm{E}[u_1(O(\sigma))]$, or he can (ii) abort and receive $\mathrm{E}[u_1(O(\widehat{\sigma}^1))]$. The value $\mathrm{E}[u_1(O(\widehat{\sigma}^1))]$ is at most $\mathrm{E}[u_1(O(\sigma))]$, and so P_1 cannot improve over $\mathrm{E}[u_1(O(\sigma))]$. Hence player 1 is not facing an ε-threat at round 3.

Case 2 – P_2 is facing an ε-threat in round 2: We first note that for any round 1 strategy for P_1 and round 2 strategy for P_2, the round strategy of playing honestly in round 3 for P_1 is threat-free, since he cannot improve over that strategy (again, since his only deviation is aborting, which gives him the worst possible NE). Thus, if P_2 is ε-threatened at round 2, he has some round strategy that ε-improves over $\mathrm{E}[u_2(O(\sigma))]$ when P_1 plays in round 3 (and 1) according to the protocol. This means that P_2 unilaterally ε-improves, which contradicts the constraints (as well as the ε-NE).

Case 3 – P_1 is facing an ε-threat in round 1: If P_1 is ε-threatened in round 1, he has some round 1 strategy $\tau(1)$ for which every ε-threat-free continuation ε-improves over every ε-threat-free continuation of $\sigma_1(1)$. We will describe an ε-threat-free continuation of $\tau(1)$ and an ε-threat-free continuation of $\sigma_1(1)$ that contradict this.

The ε-threat-free continuation of $\sigma_1(1)$: We established in Case 2 that when P_1 plays honestly in round 1, if P_2 plays honestly in round 2 he is not ε-threatened. We also established there that P_1 playing honestly in round 3 is always ε-threat-free. If follows that the continuation of both players playing honestly in rounds 2 and 3 is an ε-threat-free continuation of $\sigma_1(1)$. On this profile P_1 receives $\mathrm{E}[u_1(O(\sigma))]$.

The ε-threat-free continuation of $\tau_1(1)$: As we established in Case 2, playing honestly in round 3 is always ε-threat-free for P_1. Now, note that there is no profile in which both players improve simultaneously – because all leaves are Nash equilibria, such a profile would be a distribution on Nash equilibria that contradicts the Pareto-optimality of π. Note also that because P_1 receives the worst possible payoff when he aborts, it follows that he improves also conditioned on not aborting (as this can only help him). Thus, in any threat-free continuation of $\tau(1)$, conditioned on P_1 not aborting in round 1, P_2 again cannot improve over $\mathrm{E}[u_2(O(\sigma))]$, as this again contradicts the Pareto-optimality of π. However, if P_2 plays honestly in round 2 and then P_1 plays honestly in round 3, then P_2 receives exactly $\mathrm{E}[u_2(O(\sigma))]$ conditioned on P_1 not aborting in round 1. It follows that this continuation is the best possible for P_2, and thus P_2 is not ε-threatened in round 2 of this continuation. It follows that this continuation is ε-threat-free. However, in this continuation P_1 receives $\mathrm{E}[u_1(O(\sigma))]$ conditioned on not aborting, and thus receives at most $\mathrm{E}[u_1(O(\sigma))]$ without the conditioning. □

This completes the proof of the theorem. □

4.8 A General Theorem

In this section we prove a general theorem identifying sufficient conditions for a strategy profile to be a TFNE. The first condition is that the profile must be weakly Pareto optimal:

Definition 4.0.22 (Weakly Pareto optimal)**.** A strategy profile $\sigma \in T$ of an extensive game $\Gamma = (H, P, A, u)$ with constraints T is *weakly Pareto optimal* if there does not exist a strategy profile $\pi \in T$ for which both $\mathrm{E}[u_1(O(\pi))] > \mathrm{E}[u_1(O(\sigma))]$ and $\mathrm{E}[u_2(O(\pi))] > \mathrm{E}[u_2(O(\sigma))]$.

Next, we require the profile to be ε-safe. Intuitively, this just means that a player cannot harm the other too much by a unilateral deviation (as opposed to not being able to gain too much, which is the NE condition).

Definition 4.0.23 (ε-safe)**.** A strategy profile $\sigma = (\sigma_1, \sigma_2) \in T$ of an extensive game $\Gamma = (H, P, A, u)$ with constraints $T = (T_1, T_2)$ is ε*-safe* if for each player i,

$$\mathrm{E}\left[u_{-i}\left(O(\sigma)\right)\right] \geq \mathrm{E}\left[u_{-i}\left(O(\sigma_i', \sigma_{-i})\right)\right] - \varepsilon$$

for every strategy $\sigma_i' \in T_i$ of player i.

Finally, we have the following theorem. Note that we are implicitly assuming that the extensive games in the claim are derived from a cryptographic protocol or some other setting in which it is natural to discuss the "rounds" of a game.

Theorem 4.0.3. Let $\Gamma = (H, P, A, u)$ be an extensive game with constraints $T = (T_1, T_2)$, and let $\sigma = (\sigma_1, \sigma_2)$ be a weakly Pareto optimal ε-NE of Γ that is ε-safe. Then σ is an ε-TFNE of Γ.

We also have the following corollary.

Corollary 4.0.1. Let $\Gamma = (H, P, A, u)$ be a *zero-sum* extensive game with constraints $T = (T_1, T_2)$, and let σ be an ε-NE of Γ. Then σ is an ε-TFNE of Γ.

The corollary follows from the observation that any ε-NE of a zero-sum game is both weakly Pareto optimal and ε-safe. Note that the corollary implies the threat-freeness part of Theorem 4.0.1.

We now prove Theorem 4.0.3.

Proof. Suppose towards contradiction that at least one of the players is facing an ε-threat with respect to σ at some round. Let R be the latest such round: that is, player i is facing an ε-threat at round R with respect to σ, and no player is facing an ε-threat at any round R' that follows R.

By Definition 4.0.10 it follows that there exists a round R strategy $\tau = \tau(R)$ for player i such that the set $\mathrm{Cont}(\sigma(1, \dots, R–1), \tau(R))$ is nonempty, and such that for all $\pi \in \mathrm{Cont}(\sigma(1, \dots, R–1), \tau(R))$ and $\pi' \in \mathrm{Cont}(\sigma(1, \dots, R))$ that are ε-threat-free on R it holds that

$$\mathrm{E}\left[u_i\left(O(\pi)\right)\right] > \mathrm{E}\left[u_i\left(O(\pi')\right)\right] + \varepsilon, \tag{4.5}$$

where

$$\sigma(1, \dots, S) \overset{\text{def}}{=} \sigma(1), \dots, \sigma(S)$$

and

$$\mathrm{Cont}(\sigma(1, \dots, R)) \overset{\text{def}}{=} \Big\{\pi \in T : \pi(S) = \sigma(S) \text{ for all } S \leq R\Big\}.$$

Note that $\sigma \in \mathrm{Cont}(\sigma(1, \dots, R))$. Also note that, because R is the latest round on which an ε-threat occurs, the profile σ is ε-threat-free on R.

Using inequality (4.5) we can then infer that for any $\pi \in \mathrm{Cont}(\sigma(1, \dots, R–1), \tau(R))$ that is ε-threat-free on R it holds that

$$\mathrm{E}\left[u_i\left(O(\pi)\right)\right] > \mathrm{E}\left[u_i\left(O(\sigma)\right)\right] + \varepsilon. \tag{4.6}$$

Let $\pi^1 \in \mathrm{Cont}(\sigma(1, \dots, R-1), \tau(R))$ be one such ε-threat-free profile, and let $\sigma^1 = (\pi^1_i, \sigma_{-i})$.

Fix $R^1 = R$ and $\tau^1 = \tau$ for consistent notation. We next ask, is player i facing an ε-threat with respect to σ^1 at any round R' that follows R^1? If yes, let R^2 be the next such round: there is no R' between R^1 and R^2 on which player i is facing an ε-threat with respect to σ^1. By Definition 4.0.10 it follows that there exists a round R^2 strategy τ^2 for player i such that $\mathrm{Cont}(\sigma^1(1, \dots, R^2-1), \tau^2(R^2))$ is nonempty, and such that for all $\pi \in \mathrm{Cont}(\sigma^1(1, \dots, R^2-1), \tau^2(R^2))$ and $\pi' \in \mathrm{Cont}(\sigma^1(1, \dots, R^2))$ that are ε-threat-free on R^2 it holds that

$$\mathrm{E}\left[u_i\left(O(\pi)\right)\right] > \mathrm{E}\left[u_i\left(O(\pi')\right)\right] + \varepsilon.$$

Assume τ^2 is maximal, in the sense that for any $\pi \in \mathrm{Cont}(\sigma^1(1, \dots, R^2-1), \tau^2(R^2))$ that is ε-threat-free on R^2, player i is *not* facing an ε-threat at round R^2 with respect to π. Pick some arbitrary $\pi^2 \in \mathrm{Cont}(\sigma^1(1, \dots, R^2-1), \tau^2(R^2))$, and fix $\sigma^2 = (\pi^2_i, \sigma_{-i})$.

We now repeat the above procedure, finding the next threat to player i and letting him act on that threat, as follows. For $t = 3, 4, \ldots$ we ask, is player i facing an ε-threat with respect to σ^{t-1} at any round R' that follows R^{t-1}? If yes, let R^t be the next such round: there is no R' between R^{t-1} and R^t on which player i is facing an ε-threat with respect to σ^{t-1}.

By Definition 4.0.10 it follows that there exists a round R^t strategy τ^t for player i such that $\mathrm{Cont}(\sigma^{t-1}(1, \ldots, R^t–1), \tau^t(R^t))$ is nonempty, and such that for all $\pi \in \mathrm{Cont}(\sigma^{t-1}(1, \ldots, R^t–1), \tau^t(R^t))$ and $\pi' \in \mathrm{Cont}(\sigma^{t-1}(1, \ldots, R^t))$ that are ε-threat-free on R^t it holds that

$$\mathrm{E}\left[u_i\left(O(\pi)\right)\right] > \mathrm{E}\left[u_i\left(O(\pi')\right)\right] + \varepsilon.$$

Assume τ^t is maximal, in the sense that for any $\pi \in \mathrm{Cont}(\sigma^{t-1}(1, \ldots, R^t - 1), \tau^t(R^t))$ that is ε-threat-free on R^t, player i is *not* facing an ε-threat at round R^t with respect to π. Pick some arbitrary $\pi^t \in \mathrm{Cont}(\sigma^{t-1}(1, \ldots, R^t - 1), \tau^t(R^t))$, and fix $\sigma^t = (\pi_i^t, \sigma_{-i})$.

Finally, after repeating this for all t until there are no more ε-threats to P_i on any round that follows R, we are left with a profile $\sigma^C = (\pi_i^C, \sigma_{-i})$ on which player i is not facing an ε-threat at any round below R.

Fix $\rho = \sigma^C$, and recall that, by construction, $\rho_{-i} = \sigma_{-i}$. Because σ is ε-safe, it must be the case that

$$\mathrm{E}\left[u_{-i}\left(O(\rho)\right)\right] \geq \mathrm{E}\left[u_{-i}\left(O(\sigma)\right)\right] - \varepsilon. \tag{4.7}$$

We next ask, is player $-i$ facing an ε-threat with respect to ρ at any round S that follows R? As the following claim shows, the answer is positive:

Claim 4.0.5. *Player $-i$ is facing an ε-threat with respect to ρ at some round S that follows R.*

Proof. Suppose not. By our construction of ρ, player i is also not facing an ε-threat with respect to ρ at any round that follows R. This means that the profile ρ is ε-threat-free on the subgames R.

Since $\rho \in \mathrm{Cont}(\sigma(1, \ldots, R-1), \tau(R))$ and since $\sigma \in \mathrm{Cont}(\sigma(1, \ldots, R))$ is ε-threat-free on R, we can then use (4.6) to infer that

$$\mathrm{E}\left[u_i\left(O(\rho)\right)\right] > \mathrm{E}\left[u_i\left(O(\sigma)\right)\right] + \varepsilon.$$

However, since $\rho = (\pi_i^C, \sigma_{-i})$ is a *unilateral* deviation of player i, this contradicts the fact that σ constitutes an ε-NE. □

Let S^1 be the latest round on which P_{-i} is facing an ε-threat with respect to ρ. By Definition 4.0.10 it follows that there exists a round S^1 strategy μ^1 for player $-i$ such that

$\mathrm{Cont}(\rho(1,\ldots,S^1-1),\mu^1(S^1))$ is nonempty, and such that for all $\pi \in \mathrm{Cont}(\rho(1,\ldots,S^1-1),\mu^1(S^1))$ and $\pi' \in \mathrm{Cont}(\rho(1,\ldots,S^1))$ that are ε-threat-free on S^1 it holds that

$$\mathrm{E}\left[u_i\left(O(\pi)\right)\right] > \mathrm{E}\left[u_i\left(O(\pi')\right)\right] + \varepsilon.$$

Assume μ^1 is maximal, in the sense that for any $\pi \in \mathrm{Cont}(\rho(1,\ldots,S^1-1),\mu^1(S^1))$ that is ε-threat-free on S^1, player $-i$ is *not* facing an ε-threat at round S^1 with respect to π. Pick some $\rho^1 \in \mathrm{Cont}(\rho(1,\ldots,S^1-1),\mu^1(S^1))$ that is ε-threat-free on S^1 – such a ρ^1 must exist by Proposition 4.0.2.

Now, note that because S^1 was the last round on which P_{-i} is facing an ε-threat, and because P_i is not facing an ε-threat at any round following R with respect to ρ, it must be the case that ρ is ε-threat-free on S^1. Since $\rho \in \mathrm{Cont}(\rho(1,\ldots,S^1))$ we then have that

$$\mathrm{E}\left[u_{-i}\left(O(\rho^1)\right)\right] > \mathrm{E}\left[u_{-i}\left(O(\rho)\right)\right] + \varepsilon \geq \mathrm{E}\left[u_{-i}\left(O(\sigma)\right)\right],$$

where the second inequality follows from (4.7). We now repeat the above procedure, finding the preceding threat to player $-i$ (but that still follows R) and letting him act on that threat, as follows. For $t = 2, 3, \ldots$ we ask, is P_{-i} facing an ε-threat with respect to ρ^{t-1} at any round S that follows R? If yes, let S^t be the latest such round. By Definition 4.0.10 it follows that there exists a round S^t strategy μ^t for player $-i$ such that $\mathrm{Cont}(\rho^{t-1}(1,\ldots,S^t-1),\mu^t(S^t))$ is nonempty, and such that for all $\pi \in \mathrm{Cont}(\rho^{t-1}(1,\ldots,S^t-1),\mu^t(S^t))$ and $\pi' \in \mathrm{Cont}(\rho^{t-1}(1,\ldots,S^t))$ that are ε-threat-free on S^t it holds that

$$\mathrm{E}\left[u_i\left(O(\pi)\right)\right] > \mathrm{E}\left[u_i\left(O(\pi')\right)\right] + \varepsilon.$$

Assume μ^t is maximal, in the sense that for any $\pi \in \mathrm{Cont}(\rho^{t-1}(1,\ldots,S^t-1),\mu^t(S^t))$ that is ε-threat-free on S^t, player $-i$ is *not* facing an ε-threat at round S^t with respect to π. Pick some $\rho^t \in \mathrm{Cont}(\rho^{t-1}(1,\ldots,S^t-1),\mu^t(S^t))$ that is ε-threat-free on S^t – again, such a ρ^t must exist by Proposition 4.0.2.

Now, note that because S^t was the last round on which P_{-i} is facing an ε-threat, P_{-i} is not facing an ε-threat with respect to ρ^{t-1} at any round following S^t. Since ρ^{t-1} was chosen to be ε-threat free on S^{t-1}, player i is not facing an ε-threat with respect to ρ^{t-1} at any round following S^{t-1}. Finally, by construction, P_i is not facing an ε-threat at any round following R with respect to ρ. Since ρ and ρ^{t-1} are equivalent up to round S^{t-1}, it must be the case that P_i is not facing an ε-threat with respect to ρ^{t-1} at any round between S^t and S^{t-1} either. Thus, ρ^{t-1} is ε-threat-free on S^t. Since $\rho^{t-1} \in \mathrm{Cont}(\rho^{t-1}(1,\ldots,S^t))$, we then have that

$$\begin{aligned}\mathrm{E}\left[u_{-i}\left(O(\rho^t)\right)\right] &> \mathrm{E}\left[u_{-i}\left(O(\rho^{t-1})\right)\right] + \varepsilon \\ &> \mathrm{E}\left[u_{-i}\left(O(\rho)\right)\right] + t\cdot\varepsilon \\ &\geq \mathrm{E}\left[u_{-i}\left(O(\sigma)\right)\right] + (t-1)\cdot\varepsilon.\end{aligned}$$

Finally, after repeating this for all t until there are no more ε-threats to P_{-i} at any round that follows R, we are left with a profile $\rho^D \in \mathrm{Cont}(\sigma(1,\ldots,R-1),\tau(R))$ on which both P_i and P_{-i} are not facing an ε-threat at any round that follows R. We can then use (4.6) to infer that

$$\mathrm{E}\left[u_i\left(O(\rho^D)\right)\right] > \mathrm{E}\left[u_i\left(O(\sigma)\right)\right] + \varepsilon.$$

Furthermore, ρ^D satisfies

$$\mathrm{E}\left[u_{-i}\left(O(\rho^D)\right)\right] > \mathrm{E}\left[u_{-i}\left(O(\rho^{D-1})\right)\right] + D\cdot\varepsilon \geq \mathrm{E}\left[u_{-i}\left(O(\sigma)\right)\right],$$

since $D \geq 1$.

We conclude that on the profile ρ^D both players strictly improve over σ, contradicting the weak Pareto optimality of σ. Hence no player is facing an ε-threat with respect to σ at any round R, and this, coupled with the fact that σ is an ε-NE, yields that profile an ε-TFNE. □

Acknowledgments

We thank Eddie Dekel, Oded Goldreich, Ehud Kalai, Eran Omri, and Gil Segev for helpful conversations, and the anonymous referees for careful reading and insightful comments.

Bibliography

[1] I. Abraham, D. Dolev, R. Gonen, , and J. Halpern. Distributed computing meets game theory: robust mechanisms for rational secret sharing and multiparty computation. In *In 25th ACM Symposium Annual on Principles of Distributed Computing*, pages 53–62, 2006.

[2] G. Asharov and Y. Lindell. Utility dependence in correct and fair rational secret sharing. In *Advances in Cryptology Crypto*, pages 559–576, 2009. A full version, containing additional results, is avalable at `http://eprint.iacr.org/2009/373`.

[3] Y. Aumann and Y. Lindell. Security against covert adversaries: Efficient protocols for realistic adversaries. To appear in *Journal of Cryptology*. An extended abstract appeared in TCC 2007. Full version can be found at `http://u.cs.biu.ac.il/ lindell/PAPERS/covert.pdf`.

[4] E. Ben-Sasson, A. Tauman-Kalai, and E. Kalai. An approach to bounded rationality. In *Advances in Neural Information Processing Systems*, 2007.

[5] M. Blum. Coin flipping by telephone. In *CRYPTO*, pages 11–15, 1981.

[6] Y. Dodis, S. Halevi, and T. Rabin. A cryptographic solution to a game theoretic problem. In *In Advances in Cryptology Crypto*, pages 11–15, 2000.

[7] L. Fortnow and R. Santhanam. Bounding rationality by discounting time. In *Proceedings of the First Symposium on Innovations in Computer Science*, 2010.

[8] O. Goldreich. *Foundation of Cryptography – Basic Tools*. Cambridge University Press, 2001.

[9] S. D. Gordon and J. Katz. Rational secret sharing, revisited. In *In 5th Intl. Conf. on Security and Cryptography for Networks (SCN)*, pages 229–241, 2006.

[10] R. Gradwohl. Rationality in the full-information model. In *TCC*, 2010.

[11] R. Gradwohl, N. Livne, and A. Rosen. Incredible threats. In preparation.

[12] I. Haitner and O. Reingold. Statistically-hiding commitment from any one-way function. In *STOC 2007*, pages 1 – 10, 2007.

[13] J. Halpern and V. Teague. Rational secret sharing and multiparty computation: Extended abstract. In *36th Annual ACM Symposium on Theory of Computing (STOC)*, pages 623–632, 2004.

[14] J. Y. Halpern and R. Pass. Game theory with costly computation. In *First Symposium on Innovations in Computer Science*, 2010.

[15] S. Izmalkov, S. Micali, , and M. Lepinski. Rational secure computation and ideal mechanism design. In *FOCS*, 2005.

[16] J. Katz. Bridging game theory and cryptography: Recent results and future directions. In *5th Theory of Cryptography Conference TCC*, pages 251–272, 2008.

[17] J. Katz, G. Fuchsbauer, and D. Naccache. Efficient rational secret sharing in the standard communication model. In *TCC*, 2010.

[18] G. Kol and M. Naor. Cryptography and game theory: Designing protocols for exchanging information. In *5th Theory of Cryptography Conference TCC*, pages 320–339, 2008.

[19] G. Kol and M. Naor. Games for exchanging information. In *40th Annual ACM Symposium on Theory of Computing (STOC)*, pages 423–432, 2008.

[20] M. Lepinski, S. Micali, and A. shelat. Collusion-free protocols. In *STOC*, 2005.

[21] M. Luby, S. Micali, and C. Rackoff. How to simultaneously exchange a secret bit by flipping a symmetrically-biased coin. In *FOCS*, pages 11–21, 1983.

[22] A. Lysyanskaya and N. Triandopoulos. Rationality and adversarial behavior in multi-party computation. In *In Advances in Cryptology Crypto*, pages 180–197, 2006.

[23] S. Micali and A. Shelat. Truly rational secret sharing. In *6th Theory of Cryptography Conference TCC*, pages 54–71, 2009.

[24] M. Naor, R. Ostrovsky, R. Venkatesan, and M. Yung. Perfect zero-knowledge arguments for np using any one-way permutation. *Jour. of Cryptology*, 11:87–108, 1998.

[25] M. Naor and M. Yung. Universal one-way hash functions and their cryptographic applications. In *21st STOC*, pages 33–43, 1989.

[26] S. J. Ong, D. Parkes, A. Rosen, and S. Vadhan. Fairness with an honest minority and a rational majority. In *Theory of Cryptography Conference TCC*, pages 36–53, 2009.

[27] M. J. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, 1994.

[28] I. Damgård, T. Pedersen, and B. Pfitzmann. On the existence of statistically hiding bit commitment schemes and fail-stop signatures. In *Crypto93*, pages 250–265, 1993.

4.9 Appendix: Cryptographic Definitions

4.9.1 One-way Functions and One-way Permutations

A function f is one-way if it is easy to compute but hard to invert given the image of a random input. More formally,

Definition 4.0.24 (One-way functions). *A function $f : \{0,1\}^* \rightarrow \{0,1\}^*$ is said to be* one-way *if the following two conditions hold:*

1. *There exists a polynomial-time algorithm that on input x outputs $f(x)$.*
2. *For every probabilistic polynomial-time algorithm $\mathcal{A}$, every polynomial $p(\cdot)$, and all sufficiently large n's*

$$\Pr\left[\mathcal{A}(1^n, f(U_n)) \in f^{-1}(f(U_n))\right] < \frac{1}{p(n)} \ ,$$

where U_n denotes the uniform distribution over $\{0,1\}^n$.

In this paper we also deal with one-way permutations, and we note that the above definition naturally extends to consider permutations.

4.9.2 Commitment Schemes

A commitment scheme is a two-stage interactive protocol between a sender and a receiver. After the first stage of the protocol, which is referred to as the *commit stage*, the sender is bound to at most one value, not yet revealed to the receiver. In the second stage, which is referred to as the *reveal stage*, the sender reveals its committed value to the receiver. For simplicity of exposition, we will focus on bit-commitment schemes, i.e., commitment schemes in which the committed value is only one bit. A bit-commitment scheme is defined via a triplet of probabilistic polynomial-time Turing-machines $(\mathcal{S},\mathcal{R},\mathcal{V})$ such that:

- $\mathcal{S}$ receives as input the security parameter 1^n and a bit b. Following its interaction, it outputs some information decom (the decommitment).

- $\mathcal{R}$ receives as input the security parameter 1^n. Following its interaction, it outputs a state information com (the commitment).

- $\mathcal{V}$ (acting as the receiver in the reveal stage[9]) receives as input the security parameter 1^n, a commitment com and a decommitment decom. It outputs either a bit b' or $\perp$.

Denote by $(\mathsf{decom}|\mathsf{com}) \leftarrow \langle \mathcal{S}(1^n,b),\mathcal{R}(1^n)\rangle$ the experiment in which $\mathcal{S}$ and $\mathcal{R}$ interact (using the given inputs and uniformly chosen random coins), and then $\mathcal{S}$ outputs decom while $\mathcal{R}$ outputs com. It is required that for all n, every bit b, and every pair $(\mathsf{decom}|\mathsf{com})$ that may be output by $\langle \mathcal{S}(1^n,b),\mathcal{R}(1^n)\rangle$, it holds that $\mathcal{V}(\mathsf{com},\mathsf{decom}) = b$.[10]

The security of a commitment scheme can be defined in two complementary ways, protecting against either an all-powerful sender or an all-powerful receiver. The former are referred to as *statistically-binding* commitment schemes, whereas the latter are referred to as *statistically-hiding* commitment schemes. For simplicity, we assume that the associated “error” is zero, resulting in *perfectly-binding* and *perfectly-hiding* commitments schemes.

In order to define the security properties of such schemes, we first introduce the following notation. Given a commitment scheme $(\mathcal{S},\mathcal{R},\mathcal{V})$ and a Turing machine $\mathcal{R}^*$, we denote by

[9] Note that there is no loss of generality in assuming that the reveal stage is non-interactive. This is since any such interactive stage can be replaced with a non-interactive one as follows: The sender sends its internal state to the receiver, who then simulates the sender in the interactive stage.

[10] Although we assume perfect completeness, it is not essential for our results.

$\mathsf{view}_{\langle \mathcal{S}(b),\mathcal{R}^*\rangle}(1^n)$ the distribution of the view of $\mathcal{R}^*$ when interacting with $\mathcal{S}(1^n, b)$. This view consists of $\mathcal{R}^*$'s random coins and of the sequence of messages it receives from $\mathcal{S}$. The distribution is taken over the random coins of both $\mathcal{S}$ and $\mathcal{R}$. Similarly, given a Turing machine $\mathcal{S}^*$ we denote by $\mathsf{view}_{\langle \mathcal{S}^*(1^n),\mathcal{R}\rangle}(1^n)$ the view of $\mathcal{S}^*$ when interacting with $\mathcal{R}(1^n)$. Note that whenever no computational restrictions are assumed on $\mathcal{S}^*$ or $\mathcal{R}^*$, then without loss of generality they can be assumed to be deterministic.

Definition 4.0.25 (Perfectly-binding commitment)**.** *A bit-commitment scheme* $(\mathcal{S},\mathcal{R},\mathcal{V})$ *is said to be* perfectly-hiding *if it satisfies the following two properties:*

- **Computational hiding:** *for every probabilistic polynomial-time Turing machine* $\mathcal{R}^*$ *the ensembles* $\{\mathsf{view}_{\langle \mathcal{S}(0),\mathcal{R}^*\rangle}(1^n)\}_{n\in\mathbb{N}}$ *and* $\{\mathsf{view}_{\langle \mathcal{S}(1),\mathcal{R}^*\rangle}(1^n)\}_{n\in\mathbb{N}}$ *are computationally indistinguishable.*

- **Perfect binding:** *for every Turing machine* $\mathcal{S}^*$

$$\Pr\left[((\mathsf{decom},\mathsf{decom}')|\mathsf{com}) \leftarrow \langle \mathcal{S}^*(1^n),\mathcal{R}(1^n)\rangle : \begin{array}{c}\mathcal{V}(\mathsf{com},\mathsf{decom})=0\\ \mathcal{V}(\mathsf{com},\mathsf{decom}')=1\end{array}\right] = 0 \ ,$$

  for all sufficiently large n*, where the probability is taken over the random coins of* $\mathcal{R}$*.*

Perfectly-binding commitments can be constructed assuming the existence of any one-way permutation [5]. The construction is "non-interactive", meaning that the commitment phase consists of a single message sent from the sender $\mathcal{S}$ to the receiver $\mathcal{R}$.

Definition 4.0.26 (Perfectly-hiding commitment)**.** *A bit-commitment scheme* $(\mathcal{S},\mathcal{R},\mathcal{V})$ *is said to be* perfectly-hiding *if it satisfies the following two properties:*

- **Perfect hiding:** *for every Turing machine* $\mathcal{R}^*$ *the ensembles* $\{\mathsf{view}_{\langle \mathcal{S}(0),\mathcal{R}^*\rangle}(1^n)\}_{n\in\mathbb{N}}$ *and* $\{\mathsf{view}_{\langle \mathcal{S}(1),\mathcal{R}^*\rangle}(1^n)\}_{n\in\mathbb{N}}$ *are identically distributed.*

- **Computational binding:** *for every probabilistic polynomial-time Turing machine* $\mathcal{S}^*$ *the exists a negligible function* $\mu(n)$ *so that*

$$\Pr\left[((\mathsf{decom},\mathsf{decom}')|\mathsf{com}) \leftarrow \langle \mathcal{S}^*(1^n),\mathcal{R}(1^n)\rangle : \begin{array}{c}\mathcal{V}(\mathsf{com},\mathsf{decom})=0\\ \mathcal{V}(\mathsf{com},\mathsf{decom}')=1\end{array}\right] < \mu(n) \ ,$$

  for all sufficiently large n*, where the probability is taken over the random coins of both* $\mathcal{S}^*$ *and* $\mathcal{R}$*.*

Perfectly-hiding commitments can be constructed assuming the existence of any one-way permutation [24]. This construction is “highly-interactive”, in that the commitment phase requires the exchange of $n - 1$ messages between the sender and the receiver, where n is the security parameter. By relaxing the hiding condition to be only “statistical” it is possible to weaken the underlying assumption to the existence of one-way functions [12]. Assuming the existence of collision resistant hash functions, it is possible to construct two-message statistically-hiding commitments [25, 28].