Cryptogenography

Joshua Brody
Swarthmore College
Swarthmore, PA
joshua.e.brody@gmail.com

Sune K. Jakobsen
Queen Mary, University of London
London, United Kingdom
s.k.jakobsen@qmul.ac.uk

Dominik Scheder
Simons Institute for the Theory of Computing, UC Berkeley
Berkeley, CA
dominik.scheder@gmail.com

Peter Winkler
Dartmouth College
Hanover, NH
peter.winkler@dartmouth.edu

ABSTRACT

We consider the following cryptographic secret leaking problem. A group of players communicate with the goal of learning (and perhaps revealing) a secret held initially by one of them. Their conversation is monitored by a computationally unlimited eavesdropper, who wants to learn the identity of the secret-holder. Despite the unavailability of key, some protection can be provided to the identity of the secret-holder. We call the study of such communication problems, either from the group's or the eavesdropper's point of view, *cryptogenography.*

We introduce a basic cryptogenography problem and show that two players can force the eavesdropper to missguess the origin of a secret bit with probability 1/3; we complement this with a hardness result showing that they cannot do better than than 3/8. We prove that larger numbers of players can do better than 0.5644, but no group of any size can achieve 0.75.

Categories and Subject Descriptors

E.4 [**Coding and Information Theory**]; F.m [**Theory of Computation**]: Miscellaneous

Keywords

cryptogenography, cryptography, information theory, communication complexity

1. INTRODUCTION

We consider a cryptographic secret leaking problem where the goal is hiding the identity of the person leaking a secret, rather than protecting the secret itself. Indeed, assuming the communicators share no prior key, the secret is unprotectable unless eavesdroppers are computationally limited *and* some complexity assumptions are made. What is perhaps surprising is that under these circumstances, it is nonetheless possible to protect the identity of the secret-owner.

In the simplest scenario the secret is a bit X held by one of a group of $k > 1$ cooperating players; initially, neither the secret bit nor (except in the $k = 2$ case) the identity of its holder is known to the rest of the group. Ultimately the secret bit is revealed, but the identity of the initial bit-holder must not be revealed to an eavesdropper, even though the eavesdropper hears everything and is not computationally limited.

A protocol's success is measured by the "success probability" p that the secret is correctly revealed and the eavesdropper misguesses the identity of the secret-owner. The group's goal of maximizing p, and the eavesdropper's goal of minimizing it, comprise the two sides of what we call *cryptogenography* ("hidden source writing"). We give a formal definition of a cryptogenography problem in Section 2.

Motivation.

The standard cryptographic setting, in which players want to share a secret but keep it hidden from an adversary, is a well-studied and well-motivated problem. However, there are several applications where hiding the identity of the secret-owner is as important as, or more important than, hiding the secret itself.

The most obvious such situation arises when someone wants to publicize some secret information but fears retribution for making the information public, e.g., when dissidents protest against a government that responds by cracking down on protesters. In this case, protesters might want to broadcast the time and place for anticipated crackdowns, but fear arrest for disseminating this information.

Alternatively, the secret-owner might be publishing information obtained (legally or otherwise) from an organization that wants to keep this information from being leaked, as in the recent mass surveillance disclosures of Edward Snowden [12] and the Wikileaks scandal [13].

Groups who are already in possession of private keys, or capable of setting up a public-key system (and willing to make the necessary assumptions about computational complexity and adversarial limitations), can protect the identity of a source using standard cryptologic methods. Protection against traffic analysis can be achieved by having all parties send messages of sufficient length to include the secret.

ITCS'14, January 12–14, 2014, Princeton, New Jersey, USA.
Copyright 2014 ACM 978-1-4503-2698-8/14/01 ...$15.00.
http://dx.doi.org/10.1145/2554797.2554800.

That cryptogenography can protect the source identity without key or computational assumptions is of potential interest, we think, as are upper bounds on the degree to which this protection can be achieved. The toy problem studied below (in which the secret is just one bit) may not be directly applicable to real-life situations, but will serve, we hope, to get the subject of cryptogenography started.

1.1 Our Results

We introduce the cryptogenography problem[1] and provide both upper and lower bounds on the best possible success probability achievable by the players. Our first results are protocols, both in the two-player and general k-player cases.

THEOREM 1.1. *In the two-player problem, players can achieve success probability* $1/3$.

THEOREM 1.2. *For all large k, there exists a cryptogenography protocol achieving success probablity* 0.5644.

The proofs of the above theorems are constructive. For Theorem 1.2, we provide a series of protocols. First, we give a "majority vote" protocol which achieves success probability $1/2 + \Theta(1/\sqrt{k})$. For large k, this is essentially $1/2$, but for smaller k, we get something better. The success of the majority-votes protocol is maximized at roughly 54% for $k = 23$ players. We then show how a larger number of players can emulate the majority-votes protocol by first communicating to decide on 23 players to participate in the majority-votes protocol. The success probability decreases as a result, but the decrease can be made arbitrarily small. Surprisingly, it turns out that *reversing these operations*—having all k players vote, then adaptively deciding which players' votes will count—can boost the success probability up to ≈ 0.5644. We formalize and analyze these protocols in Section 3.

On the other side, we provide hardness results in Section 4, both in the 2-player case and in the general k-player case.

THEOREM 1.3. *No two-player cryptogenography protocol can achieve success probability greater than* 0.375.

Given that with many players, one can achieve success greater than $1/2$, one might suspect that as $k \to \infty$, players' success probability approaches 1. We show this is actually not the case.

THEOREM 1.4. *No k-player cryptogenography protocol can achieve success probability greater than* 0.75.

To show these upper bounds we generalize the problem, so that instead of starting with secret and secret-holder being uniformly distributed and independent, they can have an arbitrary joint distribution. We show that if a function from the set of such distributions to $[0, 1]$ is an upper bound on the probability of winning if communication is not allowed, and the function satisfy some concavity requirements, then the function is an upper bound on the probability of winning. We further show that the function that sends a probability distribution over secret and secret-holder to the probability of winning starting from this distribution, will satisfy these requirements. Thus, our method can find tight upper bounds.

1.2 Previous Work

The problem of secret sharing in the presence of an adversarial eavesdropper is a classic and fundamental problem in cryptography [11, 4, 9, 10, 5, 1]. However, to our knowledge our work is the first to consider how to hide the *identity* of the secret-owner rather than the secret itself.

While we are not aware of other work considering other cryptogenography research, some of our techniques are similar to techniques used in recent works in information complexity and information theory. The most relevant is the recent paper of Braverman et al. [3], who give *exact* bounds for the information complexity of zero-error protocols computing the AND of two bits and further use this bound to achieve nearly exact bounds on the randomized communication complexity of computing DISJOINTNESS, perhaps the most studied problem in communication complexity. Specifically, their optimal "protocol" for AND is similar to our "continuous protocol" for cryptogenography (see Section 3.2). Furthermore, their characterization of zero-error information complexity in terms of local concavity constraints is very similar to our convexity arguments in Section 4. Similar arguments appeared first in work of Ma and Ishwar [8, 7].

Cryptography in the absence of private key or computational assumptions is the setting for [2], in which it is shown that information can be passed in secret over an open channel when the parties have shared knowledge, even though that knowledge may not include shared secrets. In our cryptogenography setting, the players do indeed have some shared knowledge: each knows whether he/she is or is not the original secret-owner. This amounts to a shared secret only in the $k = 2$ case, and is otherwise insufficient to protect even a one-bit secret. In our development below, we neutralize even this small scrap of shared information by asking that the secret be revealed—in other words, that a hypothetical extra party, who has zero shared knowledge, be able to deduce the secret from the conversation.

1.3 Paper Outline

We formalize the cryptogenography problem and provide some initial insights in Section 2. In Section 3, we develop our cryptogenography protocols. We give hardness results in Section 4. Finally, Section 5 concludes the paper and lists some open problems for followup work.

2. PRELIMINARIES

Let $[k]$ denote the set $\{1, \ldots, k\}$. We use calligraphic letters, lower case letters, and capital letters to refer respectively to sets, elements of a set, and random variables. Random variables are uniformly distributed unless specified otherwise. We use π to denote a communication protocol

[1] There are many cryptogenography questions that can be asked. The amount of information to be leaked can be varied, as well as the number of players who originally have the information; likewise, what is known about who knows the information (e.g., one player knows the information, another only knows which player was given the information). Conceivably some of the players may try to prevent the information from being leaked by sending misleading messages. The objective itelf can be altered; e.g., instead of measuring success by the probability that Eve guess wrong, it might only be required that Eve cannot be more than 95% sure of who the secret-holder is. Here we only consider one of these questions, calling it "the" cryptogenography problem.

and assume by default that each player has a source of private randomness they can use to decide what messages to send. We further assume that messages are broadcast[2] and abuse notation somewhat by also using π to denote the *protocol transcript*; i.e., the concatenation of all messages sent during a protocol.

The (k-player) cryptogenography problem is formally defined as follows. There are k players, denoted $\textsc{plr}_1, \ldots, \textsc{plr}_k$. Inputs consist of $(X, J) \sim \mu$, where μ is uniform over $\{0,1\} \times [k]$. We refer to X as the *secret* and say that $\textsc{plr}_J$ is the *secret-owner*, or that $\textsc{plr}_J$ owns the secret. Both X and J are given to $\textsc{plr}_J$; other players receive no input. Players communicate using a protocol π, after which they compute a guess $\mathrm{Out} : \{0,1\}^* \to \{0,1\}$ for the secret. Let $\mathrm{Eve} : \{0,1\}^* \to [k]$ be the function that maximizes $\Pr[\mathrm{Eve}(\pi) = J \mid \mathrm{Out}(\pi) = X]$ for each possible value of the protocol transcript. This function represents the best possible guess of an adversary (whom we call Eve), who sees all communication between the players and wants to determine the identity of the secret-owner. Note that $\mathrm{Out}(\pi)$ and $\mathrm{Eve}(\pi)$ are functions of the messages sent in π. We define the success of a protocol as

$$\mathrm{succ}(\pi) := \Pr[\mathrm{Out}(\pi) = X \text{ and } \mathrm{Eve}(\pi) \neq J] .$$

The *communication cost* of π, denoted $\mathrm{CC}(\pi)$, is the maximum amount of communication sent during π, taken over all possible inputs (x, j) and all choices of randomness. In this work, we focus on understanding the maximum possible $\mathrm{succ}(\pi)$ of a protocol, not the communication cost.

The following lemma shows that one can assume without loss of generality that players learn the secret with certainty.

Lemma 2.1. *For all protocols π there exists a cryptogenography protocol π' with $\mathrm{succ}(\pi') = \mathrm{succ}(\pi)$, $\mathrm{CC}(\pi') = \mathrm{CC}(\pi) + k$, and such that $\Pr[\mathrm{Out}(\pi') = X] = 1$.*

Proof. Players first execute π, after which each player sends an additional bit, which equals 1 if (i) the player is the secret-owner and (ii) $\mathrm{Out}(\pi) \neq X$, and is 0 otherwise. Define $\mathrm{Out}(\pi')$ to equal $\mathrm{Out}(\pi)$ if all players communicate a 0 in the extra round of communication; otherwise, set $\mathrm{Out}(\pi') = 1 - \mathrm{Out}(\pi)$.

It is easy to see that $\mathrm{Out}(\pi') = X$ with certainty—either π correctly computes X already, or the secret-owner announces that $\mathrm{Out}(\pi) \neq X$. It is also trivial to verify that $\mathrm{CC}(\pi') = \mathrm{CC}(\pi) + k$. Thus, it remains to show that $\mathrm{succ}(\pi') = \mathrm{succ}(\pi)$. This can be seen through the following chain of equalities.

$$\begin{aligned}
\mathrm{succ}(\pi') &= \Pr[\mathrm{Out}(\pi') = X \wedge \mathrm{Eve}(\pi') \neq J] \\
&= \Pr[\mathrm{Eve}(\pi') \neq J] \\
&= \Pr[\mathrm{Eve}(\pi') \neq J \mid \mathrm{Out}(\pi) = X] \cdot \Pr[\mathrm{Out}(\pi) = X] \\
&\quad + \Pr[\mathrm{Eve}(\pi') \neq J \mid \mathrm{Out}(\pi) \neq X] \cdot \Pr[\mathrm{Out}(\pi) \neq X] \\
&= \Pr[\mathrm{Eve}(\pi') \neq J \mid \mathrm{Out}(\pi) = X] \cdot \Pr[\mathrm{Out}(\pi) = X] \\
&= \Pr[\mathrm{Eve}(\pi) \neq J \mid \mathrm{Out}(\pi) = X] \cdot \Pr[\mathrm{Out}(\pi) = X] \\
&= \mathrm{succ}(\pi) ,
\end{aligned}$$

where the second equality holds because players always learn X in π', the third equality holds by conditioning on $\mathrm{Out}(\pi)$, and the penultimate equality holds because, conditioned on π correctly computing X, the eavesdropper in π' learns nothing new about J. □

3. CRYPTOGENOGRAPHY PROTOCOLS

In this section, we present a series of protocols that demonstrate what is possible for the players to achieve.

3.1 Two Player Cryptogenography

When $k = 2$, we refer to players as Alice and Bob instead of $\textsc{plr}_1$ and $\textsc{plr}_2$.

Theorem 3.1 (Restatement of Theorem 1.1). *There is a two-player cryptogenography protocol π with $\mathrm{succ}(\pi) = 1/3$ and $\mathrm{CC}(\pi) = 2$.*

Proof. This protocol proceeds in two rounds. In the first round of communication, Alice decides whether to "pass" or "speak". If she passes, then Bob speaks in the second round; otherwise she speaks. In the second round of communication, whoever speaks will (i) send the secret if she owns it and (ii) send a random bit otherwise. Both players output the second bit of communication as their guess for the secret.

All that remains is to complete the protocol is to specify how Alice chooses to pass or speak in the first round. If Alice owns the secret, she passes with probabilty $2/3$ and speaks with probability $1/3$; otherwise, she passes with probability $1/3$ and speaks with probability $2/3$.

Note that Alice is more likely to speak in round 2 when she *doesn't* own the secret. This is perhaps counterintuitive—the players output the second bit of communication, so intuitively Alice should speak more often when she actually owns the bit. Is there an a priori reason why Alice shouldn't just announce the secret if she owns it and pass otherwise? Unfortunately in this case, Eve will learn with certainty who owns the bit. Alice's probablities of passing are chosen to give Eve no information about the secret-owner *conditioned on players successfully outputting the secret.*

Claim 3.2. $\Pr[\mathrm{Out}(\pi) = X] = 2/3$.

Proof. The secret-owner speaks in the second round with probablity $1/3$. In this case, players output correctly with certainty. Otherwise, players output a random bit and are correct with probability $1/2$. Overall, they output the correct bit with probability $1/3 + (2/3) \cdot (1/2) = 2/3$. □

Claim 3.3. $\Pr[\mathrm{Eve}(\pi) = J \mid \mathrm{Out}(\pi) = X] = 1/2$.

Proof. Without loss of generality, assume Alice speaks in the second round. From Eve's point of view, there are three cases: (i) Alice is the secret-owner and therefore outputs the correct bit in round 2, (ii) Alice is not the secret-owner but outputs the correct bit anyway, and (iii) Alice is not the secret-owner and outputs incorrectly in round 2. Simple calculation shows that all three events are equally likely; however, in the third case, the players have already failed. Thus, *conditioned on players correctly outputting the secret*, Alice and Bob are equally likely to be the secret-owner, and Eve can only guess at random. □

In this protocol, players output the secret with probability $2/3$ and given this, Eve guesses the secret-owner with probablity $1/2$. Thus, the overall success probability is $1/3$.

[2] We make this assumption for concreteness only. Our focus in this work is on the success probability and not the communication complexity, and in this case, the type of communication (e.g., broadcast vs. point-to-point) is equivalent.

3.2 General Cryptogenography Protocols

Next, we present a series of protocols for the general case.

The Majority-Votes Protocol.

In this protocol π_{MAJ}, there is a single round of communication, with each player sending a single bit. PLR_j sends X if she owns the secret; otherwise, PLR_j sends a random bit. At the end of the protocol, players output the MAJORITY of the bits communicated. Let b_j denote the bit communicated by PLR_j, and define $\text{Out}(\pi_{\text{MAJ}}) := \text{MAJ}(b_1, \ldots, b_k)$. When k is odd, the distribution of $\text{MAJ}(b_1, \ldots, b_k)$ is biased slightly towards X. This enables players to achieve success probablity somewhat larger than $1/2$.

LEMMA 3.4. *If k is odd, then π_{MAJ} succeeds with probability $1/2 + \Theta(1/\sqrt{k})$.*

PROOF. The communication $b_1, \ldots, b_k$ consists of $k-1$ random bits, along with the secret X. It will be helpful to be more explicit about the success probability. Let z_j be an indicator variable for the event $b_j = X$. Note that since $\{b_j\}$ are uniform and independent, so are $\{z_j\}$. In π_{MAJ}, players output $\text{MAJ}(b_1, \ldots, b_k)$; therefore, $\text{Out}(\pi_{\text{MAJ}}) = X$ iff $\sum_{j \neq j} z_j \geq \frac{k-1}{2}$. Thus, we have

$$\begin{aligned}\Pr[\text{Out}(\pi_{\text{MAJ}}) = X] &= \Pr\left[\sum_{j \neq J} z_j \geq \frac{k-1}{2}\right] \\ &= \frac{1}{2} + 2^{-k}\binom{k-1}{(k-1)/2} \\ &= \frac{1}{2} + \Theta\left(\frac{1}{\sqrt{k}}\right) .\end{aligned}$$

It is easy to see that the best choice for Eve is to guess a random player j whose bit agrees with the majority. There are at least $k/2$ such bits; therefore, $\Pr[\text{Eve}(\pi_{\text{MAJ}}) = J \mid \text{Out}(\pi_{\text{MAJ}}) = X] = 1/2 + \Theta(1/\sqrt{k}) - O(1/k) = 1/2 + \Theta(1/\sqrt{k})$. □

We can achieve a more precise analysis by conditioning on $\sum_j z_j$. We have

$$\begin{aligned}&\text{succ}(\pi_{\text{MAJ}}) = \\ &\sum_{\ell=(k-1)/2}^{k-1} \Pr\left[\sum_j z_j = \ell\right] \Pr\left[\text{Eve}(\pi_{\text{MAJ}}) \neq J \mid \sum z_j = \ell\right] \\ &= \sum_{\ell=(k-1)/2}^{k-1} 2^{-(k-1)} \cdot \binom{k-1}{\ell} \cdot \left(1 - \frac{1}{\ell+1}\right) .\end{aligned}$$

A straightforward calculation shows that the success probablity of π_{MAJ} is maximized at $\text{succ}(\pi_{\text{MAJ}}) \approx 0.5406$ when $k = 23$. However, for large k, the success probablity decreases and approaches $1/2$. Furthermore, when k is even, π_{MAJ} has success probability less than one half.[3] Our next protocol handles both cases by emulating a protocol for a smaller number of players.

A Continuous Protocol for large k.

Let $k > k'$ be given, and fix a k'-player protocol π'. We construct a protocol π for k players in the following manner. This protocol assumes the existence of a real-valued "clock" that all players see; i.e., the protocol assumes that all players have access to some $\eta \in \mathbb{R}_{\geq 0}$. When the protocol begins, $\eta = 0$, and η increases as the protocol progresses.

Each player generates a real number $t_j \in [0, 1]$. The secret-owner PLR_J sets $t_J := 1$; for $j \neq J$, PLR_j sets t_j uniformly in $[0, 1]$. As η increases, each player announces when $\eta = t_j$. When all but k' players have spoken, the remaining players run π'. We call the communication before emulating π' the *continuous* phase of communication. It is easy to see that at the end of this continuous phase, J is uniformly distributed over the k' remaining players. Thus π has precisely the same success probability as π'.

LEMMA 3.5. *Given any k' player protocol π' and any $k > k'$, there exists a k-player continuous protocol achieving $\text{succ}(\pi) = \text{succ}(\pi')$.*

Together with Lemma 3.4, we get an efficient protocol for all large k.

COROLLARY 3.6. *For all $k \geq 23$, there is a continuous protocol π achieving $\text{succ}(\pi) \geq 0.5406$.*

The assumption that all players have shared access to a continuous clock is perhaps unnatural, and it is unclear how players can emulate such a protocol without access to this clock. Nevertheless, it is a useful abstraction, and while it is hard to see how such protocols can be emulated, it is easy to construct a protocol that *approximates* them. Our next protocol is just such a construction.

LEMMA 3.7. *Fix k, k' with $k > k'$, and let $\epsilon > 0$. For any k'-player protocol π', there exists a k-player protocol π with $\text{succ}(\pi) \geq \text{succ}(\pi') - \epsilon$ and $\text{CC}(\pi) = \text{CC}(\pi') + O(k^3/\epsilon)$.*

PROOF. Given ϵ, let $m = \Theta(k^2/\epsilon)$ be determined later. Similar to the continuous protocol, each PLR_j ($j \neq J$) generates $t_j \in [m]$ uniformly. The secret-owner then sets $t_J := m + 1$. In the first phase of communication, players proceed in rounds $i = 1, 2$, etc. In the ith round, each PLR_j announces whether $t_j < i$. Call PLR_j *alive* if $t_j > i$. Communication in the first phase continues until $i = m$ or until at most k' players remain alive. In the second phase, the remaining alive players execute π' if exactly k' players remain; otherwise, they output something arbitrary.

There are $O(k^2/\epsilon)$ rounds of communication in the first phase of π, and each player sends a single bit in each of these rounds. Thus, π uses $O(k^3/\epsilon)$ additional communication over π'.

It is easy to see that conditioned on the communication in the first phase of π, J is uniformly distributed over the remaining alive players. Simple balls-and-bins analysis shows that the set $\{t_j \leq m\}$ are distinct with probability at least $1-\epsilon$. Thus, the probability that players do *not* execute π' is at most ϵ. □

Taking the majority-votes protocol and fixing ϵ to be a suitably small constant yields the following corollary.

COROLLARY 3.8. *For all $k \geq 23$, there exists a protocol π with $\text{succ}(\pi) > 0.5406$ and $\text{CC}(\pi) = O(k^3)$.*

[3] If k is even then $\Pr[\text{Out}(\pi_{\text{MAJ}}) = X] = 1/2$, and the overall success is only $1/2 - O(1/k)$.

Beating Majority-Votes.

For our final protocol we show that, perhaps surprisingly, one can boost success by reversing the above operations. Specifically, we consider a k-player protocol with two phases of communication. In the first phase, each player votes, as in π_{MAJ}. In the second phase of communication, players communicate to decide one-by-one who will *not* participate in the vote. Call a player "dead" if he has been chosen to no longer participate. Eventually, players decide to end the second phase of communicate and compute the majority of the remaining votes. By voting first, and elminating players from the vote one-by-one, the protocol can *adaptively* decide when to stop the protocol. At a high level, the protocol ends when the votes of the remaining players form a *super-majority.* Say that $b_1, \dots b_k$ form a t-super-majority if t of the k bits agree.

Fix a function $\tau : \mathbb{N} \to \mathbb{N}$. For each τ, we define a protocol π_τ as follows. First, the k players vote. Then, while there is no $\tau(k')$-super-majority among the remaining k' live players, they communicate to decide on a player to bow out of the protocol. The protocol ends when a super-majority of the remaining votes is achieved. In general, determining the optimal τ appears to be nontrivial; however, for small k (we used $k = 1200$), we can compute τ and the resulting $\mathrm{succ}(\pi_\tau)$ easily using Mathematica and dynamic programming. Along with Lemma 3.7, this gives a protocol with success probability greater than 0.5644, thus proving Theorem 1.2.

THEOREM 3.9 (RESTATEMENT OF THEOREM 1.2). *For all $k \geq 1200$, there exists a k-player cryptogenography protocol π with $\mathrm{succ}(\pi) > 0.5644$.*

4. HARDNESS RESULTS

In this section, we show the limits of cryptogenography—in both the two-player and general case, we give upper bounds on the best possible success probability. The proofs in this section are more technical, so we start with a high-level description of our approach.

In Section 3 we gave several protocols achieving high success probability under the uniform distribution on inputs (X, J). In this section, it will be helpful to consider the space of all possible input distributions. Let $\Delta(\{0,1\} \times [k])$ denote the set of all possible distributions on (X, J). Given a (partial) communication transcript $t \in \{0,1\}^m$, define μ_t to be the input distribution μ, conditioned on the first m bits of communication equaling t. Our motivation here is two-fold: first, examining general distributions allows us to appeal to the geometry of $\Delta(\{0,1\} \times [k])$. In particular, we show that the success of a protocol satisfies certain concavity conditions when viewed as a function $s : \Delta(\{0,1\} \times [k]) \to [0,1]$ over the distribution space. Second, our arguments will examine how a protocol π affects μ_t. We show that μ is a convex combination of $\{\mu_t\}$. We are particularly interested in how μ "splits" into distributions μ_0 and μ_1; i.e., we look at convex combinations on conditional distributions one bit at a time. Importantly, we show that for each player PLR_p, the set of all possible distributions obtainable by splitting μ forms a plane in $\Delta(\{0,1\} \times [k])$; we call this the PLR_p-*allowed plane through* μ. Any plane, that is an allowed plane through μ' for some distribution μ' is called an *allowed plane.* Our first lemma characterizes the possible distribution splits made by a cryptogenography protocol.

LEMMA 4.1. *Let π be a protocol where only one message gets sent, this message is in $\{0,1\}$, and this message is sent by PLR_p. If π is used with prior distribution μ, let $\nu(i)$ denote the probability that PLR_p sends message i and let μ_i be the distribution given that PLR_p sent message i. Then*

1. *$\mu = \nu(0)\mu_0 + \nu(1)\mu_1$.*
2. *Each μ_i is proportional to μ on $\{0,1\} \times ([k] \setminus \{p\})$.*

PROOF. 1: Let M denote the message sent in π. Then we have

$$\begin{aligned}\mu(x, j) &= \Pr(X = x, J = j)\\ &= \sum_{i=0}^{1} \Pr(X = x, J = j, M = i)\\ &= \sum_{i=0}^{1} \nu(i)\mu_i(x, j)\end{aligned}$$

2: Let $x' \in \{0,1\}$ and $p' \in [k] \backslash \{p\}$. Then by Bayes' theorem

$$\begin{aligned}\mu_i(x', p') =& \Pr(X = x', J = p' | M = i)\\ =& \frac{\Pr(M = i | X = x', J = p') \Pr(X = x', J = p')}{\Pr(M = i)}\\ =& \frac{\Pr(M = i | X = x', J = p')}{\Pr(M = i)} \mu(x', p')\end{aligned}$$

The probability distribution PLR_p use to chose his message only depends on his information, and thus does not depend on x' and p' (as long as $p' \neq p$). So $\frac{\Pr(M=i|X=x',J=p')}{\Pr(M=i)}$ is a constant, and μ_i is indeed proportional to μ on $\{0,1\} \times ([k] \setminus \{p\})$ □

Our second lemma is the converse of Lemma 4.1. It says that every possible split conforming to the restrictions of Lemma 4.1 are possible in a communication protocol.

LEMMA 4.2. *Let PLR_p be a player, μ, μ_0, μ_1 be distributions over $\{0,1\} \times [k]$, let ν be a distribution with support $\{0,1\}$ such that*

1. *$\mu = \nu(0)\mu_0 + \nu(1)\mu_1$.*
2. *Each μ_i is proportional to μ on $\{0,1\} \times ([k] \setminus \{p\})$.*

Then there is a protocol π where only player PLR_p sends messages, he only sends one message, he sends message $i \in \{0,1\}$ with probability $\nu(i)$, and the posterior probability distribution given that he sends the message i is μ_i.

PROOF. If PLR_p has the information and it is 0, he should send the message $i \in \{0,1\}$ with probability $\frac{\nu(i)\mu_i(0,p)}{\mu(0,p)}$, if he has the information and it is 1 he should send the message $i \in \{0,1\}$ with probability $\frac{\nu(i)\mu_i(1,p)}{\mu(1,p)}$ and if he does not have the information, he should send message i with probability $\frac{\nu(i)\mu_i(\{0,1\}\times([k]\setminus\{k\}))}{\mu(\{0,1\}\times([k]\setminus\{k\}))}$. It is easy to check that this protocol satisfies the claim in the theorem. □

Instead of playing the cryptogenography game starting from the uniform distribution over $\{0,1\} \times [k]$, we could start from any other distribution μ (and let all the players know that we are starting from distribution μ). Let $\mathrm{succ}(\mu, \pi)$ denote the probability of winning, when using protocol π starting from distribution μ. Let $\mathrm{succ}(\mu) = \sup_\pi \mathrm{succ}(\mu, \pi)$ where the supremum is over all protocols π, and let $\mathrm{succ}_n(\mu) = \sup_{CC(\pi)\le n} \mathrm{succ}(\mu,\pi)$.

For a distribution μ we now know that the $\textsc{plr}_p$-allowed plane through μ, as define previously, is the set of all distributions μ' that are proportional to μ on $\{0,1\} \times ([k] \setminus \{p\})$. We see that this is indeed a plane in the set $\Delta(\{0,1\} \times [k])$ of distributions over $\{0,1\} \times [k]$.

Lemma 4.3. *The function* $\mathrm{succ} : \Delta(\{0,1\} \times [k]) \to [0,1]$ *satisfies:*

1. $\mathrm{succ}(\mu) \ge \mathrm{succ}(\mu,\pi_0)$ *where* π_0 *is the protocol where they do not communicate at all.*
2. *For any allowed plane,* succ *restricted to that plane is concave.*

Proof. 1: $\mathrm{succ}(\mu) = \sup_\pi \mathrm{succ}(\mu,\pi) \ge \mathrm{succ}(\mu,\pi_0)$.

2: Whenever μ_0, μ_1 are distributions in the $\textsc{plr}_p$-allowed plane, and $\nu(i)$ is a distribution with support $\{0,1\}$ such that $\mu = \sum_{i=0}^{1} \nu(i)\mu_i$, Lemma 4.2 says that we can find a protocol where $\textsc{plr}_p$ sends one message, sends message i with probability $\nu(i)$, and the distribution given that $\textsc{plr}_p$ sends i is μ_i. For every $\epsilon > 0$ we can now construct a protocol π_ϵ such that $\mathrm{succ}(\mu,\pi_\epsilon) \ge \sum_{i=0}^{1} \nu(i)\,\mathrm{succ}(\mu_i) - \epsilon$. The protocol π_ϵ starts with the one-message protocol we obtain from Lemma 4.2. If the message i is sent, they continue from there, using a protocol π_i with $\mathrm{succ}(\mu_i,\pi_i) \ge \mathrm{succ}(\mu_i) - \epsilon$. The existence of such a protocol follows from the definition of $\mathrm{succ}(\mu_i)$. It is clear that the resulting π_ϵ satisfies the required inequality. As we can do this for all $\epsilon > 0$ we get $\mathrm{succ}(\mu) \ge \sum_{i=0}^{1} \nu(i)\,\mathrm{succ}(\mu_i)$. It follows from the converse of Jensens inequality that succ is concave in the $\textsc{plr}_p$-allowed plane. □

We are now ready for a characterisation of succ.

Theorem 4.4. *The function* $\mathrm{succ} : \Delta(\{0,1\} \times [k]) \to [0,1]$ *is the point-wise smallest function* $s : \Delta(\{0,1\} \times [k])$ *that satisfies*

1. $s(\mu) \ge \mathrm{succ}(\mu,\pi_0)$ *where* π_0 *is the protocol where they do not communicate at all.*
2. *For any allowed plane,* s *restricted to that plane is concave.*

Proof. We know from Lemma 4.3 that succ satisfies the two requirements. It is clear that the point-wise infimum of a family of functions satisfying requirement 1 will itself satisfy requirement 1, and similar for requirement 2. Thus, there is a smallest function s^* satisfying both requirements.

Requirement 1 simply says that $s^*(\mu) \ge \mathrm{succ}_0(\mu)$. Assume for induction that $s^*(\mu) \ge \mathrm{succ}_n(\mu)$, and consider a protocol π with $CC(\pi) \le n+1$. We can view the protocol π as first sending one message $i \in \{0,1\}$ sent by $\textsc{plr}_p$ (if he can send more than two messages in the first round, we simply let him send one bit of the message at a time), and for each possible message i calling some subsequent protocol π_i with $CC(\pi_i) \le n$. If we let $\nu(i)$ denote the probability that $\textsc{plr}_p$ sends i and let μ_i denote probability distribution given the $\textsc{plr}_p$ sends i, we know from Lemma 4.1 that all the μ_is are in the p-allowed plane through μ and that $\mu = \sum_{i=0}^{1} \nu(i)\mu_i$. So

$$\mathrm{succ}(\mu,\pi) \le \sum_{i=0}^{1} \nu(i)\,\mathrm{succ}_n(\mu_i) \le \sum_{i=0}^{1} \nu(i) s^*(\mu_i) \le s^*(\mu)$$

Here the second inequality follows from induction hypothesis, and the third follows from the fact that s is concave in the p-allowed plane. As this holds for all π with $CC(\pi) \le n+1$ we get $\mathrm{succ}_{n+1}(\mu) \le s^*(\mu)$, and by induction we have $\mathrm{succ}_n \le s^*$ for all n.

Now $s^*(\mu) \ge \lim_{n\to\infty} \mathrm{succ}_n(\mu) = \mathrm{succ}(\mu)$ but succ satisfies the two requirement in the theorem, and s^* is the smallest function satisfying the two requirements. Thus $s^* = \mathrm{succ}$. □

This theorem gives us a way to show upper bounds on $\mathrm{succ}(\mu)$: Whenever we have a function s satisfying the two requirements, $s(\mu) \ge \mathrm{succ}(\mu)$. In the rest of this section we will show upper bounds on succ by guessing such functions s. These are the best functions we have, but we do not think that they are optimal.

Theorem 4.5 (Restatement of Theorem 1.3). *Let* μ_2 *denote the uniform distribution on* $\{0,1\} \times [2]$. *Then* $\mathrm{succ}(\mu_2) \le \frac{3}{8}$.

Proof. For brevity, write $x_j := \mu(0,j), y_j := \mu(1,j)$ for $j \in \{1,2\}$ being one of the players. Define

$$f(x_1,x_2,y_1,y_2) := x_1^2 + x_2^2 + y_1^2 + y_2^2 - 6(x_1x_2 + y_1y_2) \quad \text{and}$$

$$s(x_1,x_2,y_1,y_2) := \frac{1 - f(x_1,x_2,y_1,y_2)}{4} \ .$$

Proposition 4.6. *Let* μ_2 *be the uniform distribution on* $\{0,1\} \times [2]$. *Then* $s(\mu_2) = \frac{3}{8}$.

The proof is a simple calculation.

Lemma 4.7. *The function* s *is concave on all allowed planes.*

Proof. We focus on $\textsc{plr}_1$-allowed planes. Let μ be a distribution and let $(\mu_t)_{t\in\mathbb{R}}$ be a line in a $\textsc{plr}_1$-allowed plane through μ (let us say we get μ at $t = 0$). We show that f is convex (and thus s is concave) along this line. Since $(\mu_t)_{t\in\mathbb{R}}$ is an allowed line, the values $(x_2(t), y_2(t))$ will be proportional to (x_2, y_2) throughout.

First we handle the case that $(x_2(t), y_2(t)) = (x_2, y_2)$. That is, $\textsc{plr}_1$'s message does not change the probabilities involving $\textsc{plr}_2$. In words, she talks only about the value of her bit, not about whether she owns it or not. In this case we can assume that

$$\mu_t = (x_1 + t, x_2, y_1 - t, y_2) \ .$$

Now $f(\mu_t)$ is a quadratic polynomial in t with leading monomial $2t^2$, and thus is convex.

From now on, we assume that $(x_2(t), y_2(t)) \ne (x_2, y_2)$ unless $t = 0$. Let $b := x_2 + y_2$ be the probability that $\textsc{plr}_2$ has the bit. Note that $b > 0$, because the case $b = 0$ would mean $(x_2(t) = y_2(t)) = (0,0)$ throughout, and we have handled this case already above. Now μ_t is of the form

	0	1
P_1	$x_1 + ctb$	$y_1 + \bar{c}tb$
P_2	$x_2(1-t)$	$y_2(1-t)$

where c is a parameter that describes, if you will, the "slope" of the line $(\mu_t)_{t\in\mathbb{R}}$, and $\bar{c} := 1 - c$. Again, $t = 0$ recovers the original distribution μ. Again, $f(\mu_t)$ is quadratic in t, and the leading monomial is

$$(c^2 + \bar{c}^2)b^2 + x_2^2 + y_2^2 + 6b(cx_2 + \bar{c}y_2) \ . \tag{1}$$

We want to show that this is non-negative. It is quadratic in c^2 with leading monomial $2b^2c^2$ (note that $\bar{c}^2 = 1-2c+c^2$). Thus (1) is minimized when the derivative with respect to c is 0:

$$\begin{aligned}\frac{\partial(1)}{\partial c} &= 2cb^2 - 2\bar{c}b^2 + 6b(x_2-y_2) = 0 \;\Leftrightarrow \\ cb - (1-c)&b + 3(x_2-y_2) = 0 \Leftrightarrow \\ &cb = 2y_2 - x_2 \ ,\end{aligned}$$

and $\bar{c}b = 2x_2 - y_2$ (recall that $b = x_2 + y_2$ when micro-checking the above calculation). Plugging the values of cb and $\bar{c}b$ into (1), we obtain

$$\begin{aligned}(c^2 + \bar{c}^2)&b^2 + x_2^2 + y_2^2 + 6b(cx_2 + \bar{c}y_2) \\ &= (2y_2 - x_2)^2 + (2x_2 - y_2)^2 + x_2^2 + y_2^2 \\ &\quad + 6x_2(2y_2 - x_2) + 6y_2(2x_2 - y_2) \\ &= 16x_2y_2 \\ &\geq 0 \ .\end{aligned}$$

This shows that f is convex on all allowed planes. □

Lemma 4.8. *Let $\mu \in \Delta(\{0,1\} \times [2])$ be a distribution and let π_0 be the empty protocol, i.e., the one without any communication. Then $s(\mu) \geq \operatorname{succ}(\mu, \pi_0)$.*

Proof. First, let us compute $\operatorname{succ}(\mu, \pi_0)$. Since there is no communication, Out only depends on μ. If Out $= 0$, then Eve guesses the player j that maxizes x_j. If Out $= 1$, she maximizes y_j. It therefore follows that $\operatorname{succ}(\mu, \pi_0) = \max(\min(x_1, x_2), \min(y_1, y_2))$. Next, we claim that $s(\mu) \geq \min(x_1, x_2)$. The proof that $s(\mu) \geq \min(y_1, y_2)$ will be symmetric. We introduce the shorthand $s_x := x_1 + x_2$ and $m_x := \max(x_1, x_2)$, and similarly for y. So $\min(x_1, x_2) = s_x - m_x$.

$$\begin{aligned}s \geq \min(x_1, x_2) &\Leftrightarrow 1 - f - 4\min(x_1, x_2) \geq 0 &(2)\\ &\Leftrightarrow 1 - 4s_x + 4m_x - f \geq 0 \ . &(3)\end{aligned}$$

Let us bound f from above:

$$\begin{aligned}f(x_1, x_2, y_1, y_2) &= x_1^2 + x_2^2 + y_1^2 + y_2^2 - 6(x_1x_2 + y_1y_2) \\ &= 4(x_1^2 + x_2^2) + 4(y_1^2 + y_2^2) \\ &\qquad - 3(x_1 + x_2)^2 - 3(y_1 + y_2)^2 \\ &= 4(x_1^2 + x_2^2) + 4(y_1^2 + y_2^2) - 3s_x^2 - 3s_y^2 \\ &\leq 4s_xm_x + 4s_ym_y - 3s_x^2 - 3s_y^2 \ .\end{aligned}$$

Let us combine this with (3):

$$\begin{aligned}1-&4s_x + 4m_x - f \\ &\geq 1 - 4s_x + 4m_x - 4m_xs_x - 4s_ym_y + 3s_x^2 + 3s_y^2 \\ &= (1 - s_x)(1 - 3s_x) + 4m_x(1 - s_x) - 4m_ys_y + 3s_y^2 \\ &= s_y(1 - 3s_x + 4m_x - 4m_y + 3s_y) \\ &\qquad\qquad\qquad\qquad \text{(note that } 1 - s_x = s_y) \\ &\geq s_y(1 - 3s_x + 2s_x - 4s_y + 3s_y) \\ &\qquad\qquad\qquad \text{(since } m_x \geq \tfrac{s_x}{2} \text{ and } m_y \leq s_y) \\ &= s_y(1 - s_x - s_y) = 0 \ .\end{aligned}$$

This shows that $s(\mu) \geq \max(\min(x_1, x_2), \min(y_1, y_2)) = \operatorname{succ}(\mu, \pi_0)$ and proves the lemma. □

By Theorem 4.4 this implies that $\operatorname{succ}(\mu_2) \leq s(\mu_2) = \frac{3}{8}$.

A function similar to the above s was suggested by "fedja" on Mathoverflow [6] and this function was then improved by Wadim Zudilin to the above function also on Mathoverflow.

Our final theorem generalizes the above argument to k players.

Theorem 4.9 (Restatement of Theorem 1.4). *Let μ_k denote the uniform distribution on $\{0,1\} \times [k]$. Then $\operatorname{succ}(\mu_k) \leq \frac{3}{4} - \frac{1}{2k}$.*

Proof. For brevity, we denote by x_j the probability that player j has the bit, and it is 0, that is, $x_j := \mu(0, j)$. Similarly, $y_j := \mu(1, j)$. We define

$$f(\vec{x}, \vec{y}) := 2\left\|\vec{x}\right\|_2^2 + 2\left\|\vec{y}\right\|_2^2 - \left\|x\right\|_1^2 - \left\|y\right\|_1^2 \ .$$

where $\left\|\vec{x}\right\|_p := \left(\sum_{i=1}^k x_i^p\right)^{1/p}$ and define

$$s_k(\vec{x}, \vec{y}) := \frac{1 - f(\vec{x}, \vec{y})}{2} \ .$$

We will prove three things. First, $s_k(\mu_k) = \frac{3}{4} - \frac{1}{2k}$. Second, $s_k(\mu) \geq \operatorname{succ}(\mu, \pi_0)$, where π_0 is the "empty" protocol without any communication. Third, and most important, s_k is concave along allowed planes. This will conclude the proof.

Proposition 4.10. *Let μ_k be the uniform distribution on $\{0,1\} \times [k]$. Then $s_k(\mu_k) = \frac{3}{4} - \frac{1}{2k}$.*

Proof. Every (X, J) has probability $\frac{1}{2k}$. Therefore $\left\|\vec{x}\right\|_2^2 = \left\|\vec{y}\right\|_2^2 = k \cdot \left(\frac{1}{2k}\right)^2$ and $\left\|\vec{x}\right\|_1^2 = \left\|\vec{y}\right\|_1^2 = \left(\frac{1}{2}\right)^2 = \frac{1}{4}$. Thus, $f(\mu_k) = 4k \cdot \left(\frac{1}{2k}\right)^2 - 2 \cdot \left(\frac{1}{2}\right)^2 = \frac{1}{k} - \frac{1}{2}$, and $s_k(\mu_k) = \frac{3}{4} - \frac{1}{2k}$. □

Proposition 4.11. *Let $\mu \in \Delta(\{0,1\} \times [k])$ be a distribution and let π_0 be the empty protocol, i.e., the one without any communication. Then $s_k(\mu) \geq \operatorname{succ}(\mu, \pi_0)$.*

Proof. What is $\operatorname{succ}(\mu, \pi_0)$? The transcript of π_0 is empty, thus $\operatorname{Out}(\pi)$ only depends on μ. If Out $= 0$, then Eve optimally guesses the player j that maximizes x_j, and the success probability for the players is $\mu(0, [k]) - \max_j \mu(0, j)$. Similarly, if Out $= 1$, she chooses the j maximizing y_j, and the success probability is $\mu(1, [k]) - \max_j \mu(1, j)$. Thus, the success probability of π_0 is

$$\max\left(\mu(0, [k]) - \max_j \mu(0, j), \quad \mu(1, [k]) - \max_j \mu(1, j)\right) \ .$$

For brevity, we define $m_x := \max_j x_j = \max_j \mu(0, j)$, $m_y := \max_j y_j = \max_j \mu(1, j)$, $s_x := \sum_j x_j = \mu(0, [k])$, and $s_y := \sum_j y_j = \mu(1, [k])$. We want to show that $s_k(\mu) \geq \operatorname{succ}(\mu, \pi_0) = \max(s_x - m_x, s_y - m_y)$. We will show that $s_k(\mu) \geq s_x - m_x$. The inequality $s_m(\mu) \geq s_y - m_y$ will follow analogously.

$$\begin{aligned}s_k(\mu) \geq s_x - m_x \iff& \frac{1 - f(\vec{x}, \vec{y})}{2} \geq s_x - m_x &(4)\\ \iff& 1 - 2s_x + 2m_x - f(\vec{x}, \vec{y}) \geq 0 \ . &(5)\end{aligned}$$

Let us bound $f(\vec{x}, \vec{y})$ from above:

$$\begin{aligned}f(\vec{x}, \vec{y}) &= 2\left\|\vec{x}\right\|_2^2 + 2\left\|\vec{y}\right\|_2^2 - \left\|x\right\|_1^2 - \left\|y\right\|_1^2 \\ &\leq 2m_xs_x + 2m_ys_y - s_x^2 - s_y^2 \ .\end{aligned}$$

Thus, we evaluate (5):

$$\begin{aligned} &1 - 2s_x + 2m_x - f(\vec{x},\vec{y}) \\ &\quad \geq 1 - 2s_x + 2m_x - 2m_x s_x - 2m_y s_y + s_x^2 + s_y^2 \\ &\quad = 1 - 2s_x + s_x^2 + s_y^2 + 2m_x(1-s_x) - 2m_y s_y \\ &\quad = 2s_y^2 + 2m_x s_y - 2m_y s_y \\ &\quad = 2s_y(s_y + m_x - m_y) \geq 0 \ . \end{aligned}$$

The last inequality follows from $s_y - m_y \geq 0$. Replacing the roles of x and y, a similar calculation shows that $s_k(\mu) \geq s_y - m_y$, and thus $s_k(\mu) \geq \operatorname{succ}(\mu, \pi_0)$. □

PROPOSITION 4.12. *For any allowed plane, s_k restricted to that plane is concave.*

PROOF. By symmetry, we can restrict ourselves to PLR$_1$-allowed planes. That is, all distributions μ' that are proportional to μ on $\{0,1\} \times ([k] \setminus \{1\})$. Let μ be any distribution and let $(\mu_t)_{t\in\mathbb{R}}$ be a line through μ that is contained in a PLR$_1$-allowed plane. It suffices to show that s_k is concave along all such lines.

First suppose that in our line, each μ_t is not only proportional to μ on $\{0,1\} \times ([k] \setminus \{1\})$, but actually identical to it. Then μ_t looks as follows:

	0	1
P_1	$x_1 + t$	$y_1 - t$
P_2	x_2	y_2
$\vdots$	$\vdots$	$\vdots$
P_n	x_n	y_n

(6)

and $f(\mu_t)$ is quadratic in t with leading monomial $2t^2$. Therefore, it is convex, and s_k is concave, along $(\mu_t)_{t\in\mathbb{R}}$.

Suppose from now on that μ_t is not identical to μ on $\{0,1\} \times ([k] \setminus \{1\})$. How does a line $(\mu_t)_{t\in\mathbb{R}}$ through μ in a PLR$_1$-allowed plane look? The probabilities $x_2, \ldots, x_n$ and $y_2, \ldots, y_n$ get multiplied by a factor $(1-t)$. Let $b_0 := x_2 + \cdots + x_n$, $b_1 := y_2 + \cdots + y_n$, and $b = b_0 + b_1$. Note that $b > 0$, otherwise all μ_t are 0 on $\{0,1\} \times ([k] \setminus \{1\})$, and this belongs to the above case. The distribution μ_t on the PLR$_1$-allowed plane containing μ has the form

	0	1
P_1	$x_1 + ctb$	$y_1 + \bar{c}tb$
P_2	$x_2(1-t)$	$y_2(1-t)$
$\vdots$	$\vdots$	$\vdots$
P_n	$x_n(1-t)$	$y_n(1-t)$

(7)

where $c \in \mathbb{R}$ is some parameter specific to the line $(\mu_t)_{t\in\mathbb{R}}$, and $\bar{c} := 1 - c$. For fixed $\vec{x}, \vec{y}, c$, all μ_t lie on a line. It remains to show that f is convex along this line. We evaluate $f(\mu_t)$, which is a quadratic polynomial in t, and analyze the coefficient of the monomial t^2: In the terms $\|\vec{x}\|_2^2$, $\|\vec{y}\|_2^2$, $\|\vec{x}\|_1^2$, $\|\vec{y}\|_1^2$, evaluated at μ_t, the monomial t^2 has the following coefficients:

$$\begin{aligned} \|\vec{x}\|_2^2 &\longrightarrow c^2b^2 + x_2^2 + \cdots + x_k^2 \geq c^2b^2 \\ \|\vec{y}\|_2^2 &\longrightarrow \bar{c}^2b^2 + y_2^2 + \cdots + y_k^2 \geq \bar{c}^2b^2 \\ \|\vec{x}\|_1^2 &\longrightarrow (cb - x_2 - \cdots - x_k)^2 = (cb - b_0)^2 = c^2b^2 - 2cbb_0 + b_0^2 \\ \|\vec{y}\|_1^2 &\longrightarrow (\bar{c}b - y_2 - \cdots - y_k)^2 = (\bar{c}b - b_1)^2 = \bar{c}^2b^2 - 2\bar{c}bb_1 + b_1^2 \end{aligned}$$

Thus, the coefficient of t^2 of $f(\vec{x},\vec{y}) = 2\|\vec{x}\|_2^2 + 2\|\vec{y}\|_2^2 - \|x\|_1^2 - \|y\|_1^2$ is at least

$$b^2(c^2 + \bar{c}^2) - b_0^2 - b_1^2 + 2b(cb_0 + \bar{c}b_1) \ . \qquad (8)$$

It remains to show that this is non-negative. Recall that b is the probability that PLR$_1$ does not own the bit. Since we assume $b > 0$, the expression in (8) is quadratic in c with leading monomial $2b^2c^2$ (note that $\bar{c}^2 = (1-c)^2 = 1 - 2c + c^2$). Thus, (8) is minimized if its derivate with respect to c is 0:

$$\begin{aligned} \frac{\partial(8)}{\partial c} &= 2b^2(c - \bar{c}) + 2b(b_0 - b_1) \\ &= 2b^2(2c - 1) + 2b(b - 2b_1) \\ &= 4b^2c - 4bb_1 \ . \end{aligned}$$

This is 0 if and only if $c = \frac{b_1}{b}$. At that point, $\bar{c} = \frac{b_0}{b}$. In particular, $c, \bar{c} \geq 0$. This is not a priori clear, since c is a parameter of the line (μ_t), not a probability. Let us evaluate (8) at $c = \frac{b_1}{b}$:

$$\begin{aligned} (8) &= \ b^2(c^2 + \bar{c}^2) - b_0^2 - b_1^2 + 2b(cb_0 + \bar{c}b_1) \\ &\geq \ b^2(c^2 + \bar{c}^2) - b_0^2 - b_1^2 \\ &= \ b^2\left(\left(\frac{b_1}{b}\right)^2 + \left(\frac{b_0}{b}\right)^2\right) - b_0^2 - b_1^2 = 0 \ . \end{aligned}$$

This shows that f is convex along the line $(\mu_t)_{t\in\mathbb{R}}$, and thus on whole PLR$_1$-allowed plane containing μ. Thus, s_k is concave along those planes, which proves the proposition. □

5. CONCLUSIONS AND OPEN PROBLEMS

In this work we considered a game where a group of people is collaborating, to let one of them publish one bit of information, while minimising the probability that an observer will guess who leaked the bit. This problem is easy to analyse under standard cryptographic assumption, and assuming that the observer has bounded computational power, but we wanted to analyse the problem with the assumption that the observer has unbounded computational power. We gave a characterisation of the optimal probability that the secret-holder is not guessed as a function of the prior distribution of secret-value and secret-holder, and using this characterisation we showed that no matter how many people are in the group, they will always lose the game with probability at least $\frac{1}{4}$. We also gave a general protocol that ensures the group wins with probability > 0.5644. In the case with only two players, we gave a protocol that ensures that the group wins with probability $\frac{1}{3}$, and we showed that they cannot win with probability more than $\frac{3}{8}$.

There are several interesting open questions to look at next. First of all, it is still an open problem to determine $\operatorname{succ}(\mu_2)$ and $\lim_{k\to\infty} \operatorname{succ}(\mu_k)$. More interestingly, we could ask the same problem, but where more than one player knows the information and/or the information is more than one bit. How fast does the number of player who know the information have to grow as a function of the amount of information to make it possible to leak the information?

Finally we see that a version of Theorem 4.4 holds for any game where a group of collaborating players receive some utility depending on the posterior distribution at the end of a communication round. A special case is the result in information complexity in [3] and [8], where the utility received in the end is either internal privacy or external privacy. We believe that it would be interesting to study this class of problems in general.

6. ACKNOWLEDGEMENTS

We thank Søren Riis for several stimulating discussions during the development of this work. Joshua Brody and Dominik Scheder acknowledge support from the Danish National Research Foundation and The National Science Foundation of China (under the grant 61061130540) for the Sino-Danish Center for the Theory of Interactive Computation, within which this work was performed. Dominik Scheder acknowledges support from the Simons Institute for the Theory of Computing, UC Berkeley.

7. REFERENCES

[1] R. Ahlswede and I. Csiszar. Common randomness in information theory and cryptography. I. Secret sharing. *IEEE Trans. Inf. Theory*, 39(4):1121–1132, 1993.

[2] Donald Beaver, Stuart Haber, and Peter Winkler. On the isolation of a common secret. In R.L. Graham and J. Nešetřil, editors, *The Mathematics of Paul Erdős Vol. II*, pages 121–135, Berlin, 1996. Springer-Verlag. (Reprinted and updated 2013.).

[3] Mark Braverman, Ankit Garg, Denis Pankratov, and Omri Weinstein. From information to exact communication. In *Proc. 45th Annual ACM Symposium on the Theory of Computing*, 2013.

[4] W. Diffie and M.E. Hellman. New directions in cryptography. *IEEE Trans. Inf. Theory*, 22(6):644–654, 1976.

[5] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *J. Comput. Syst. Sci.*, 28(2):270 – 299, 1984.

[6] Sune Jakobsen. Greatest function satisfying some convexity requirements. mathoverflow.net/questions/81753. (2011).

[7] N. Ma and P. Ishwar. Interactive source coding for function computation in collocated networks. *IEEE Trans. Inf. Theory*, 58(7):4289–4305, 2012.

[8] N. Ma and P. Ishwar. The infinite-message limit of two-terminal interactive source coding. *IEEE Trans. Inf. Theory*, 59(7):4071–4094, 2013.

[9] Ralph C. Merkle. Secure communications over insecure channels. *Commun. ACM*, 21(4):294–299, 1978.

[10] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM*, 21(2):120–126, 1978.

[11] Claude E Shannon. Communication theory of secrecy systems. *Bell system technical journal*, 28(4):656–715, 1949.

[12] Wikipedia. 2013 mass surveillance disclosures. http://en.wikipedia.org/wiki/2013_mass_surveillance_-disclosures.
(2013).

[13] Wikipedia. United states diplomatic cables leak. http://en.wikipedia.org/wiki/united_states_-diplomatic_cables_leak.
(2010).