Fairness with an Honest Minority and a Rational Majority[⋆]

Shien Jin Ong[1⋆⋆], David C. Parkes[2], Alon Rosen[3⋆⋆⋆], and Salil Vadhan[4†]

[1] Goldman, Sachs & Co., New York, NY.
shienjin@alum.mit.edu.

[2] Harvard School of Engineering and Applied Sciences, Cambridge, MA.
parkes@eecs.harvard.edu

[3] Herzliya Interdisciplinary Center, Herzliya, Israel.
alon.rosen@idc.ac.il

[4] Harvard School of Engineering and Applied Sciences and
Center for Research on Computation and Society, Cambridge, MA.
salil@eecs.harvard.edu.

Abstract. We provide a simple protocol for secret reconstruction in any threshold secret sharing scheme, and prove that it is *fair* when executed with many *rational* parties together with a small minority of *honest* parties. That is, all parties will learn the secret with high probability when the honest parties follow the protocol and the rational parties act in their own self-interest (as captured by a set-Nash analogue of trembling hand perfect equilibrium). The protocol only requires a *standard* (synchronous) broadcast channel, tolerates both early stopping and incorrectly computed messages, and only requires 2 rounds of communication.
Previous protocols for this problem in the cryptographic or economic models have either required an honest majority, used strong communication channels that enable simultaneous exchange of information, or settled for approximate notions of security/equilibria. They all also required a nonconstant number of rounds of communication.

Keywords: game theory, fairness, secret sharing.

1 Introduction

A major concern in the design of distributed protocols is the possibility that parties may deviate from the protocol. Historically, there have been two main paradigms for modeling this possibility. One is the cryptographic paradigm, where some parties are honest, meaning they will always follow the specified

[⋆] Earlier versions of this paper are [34, 35].

[⋆⋆] Work done while author was a graduate student at Harvard School of Engineering and Applied Sciences.

[⋆⋆⋆] Part of this work done while the author was a postdoctoral fellow at Harvard University's Center for Research on Computation and Society. Work supported in part by ISF Grant 334/08.

[†] Supported by NSF grant CNS-0831289.

protocol, and others are malicious, meaning they can deviate arbitrarily from the protocol. The other is the economic paradigm, where all parties are considered to be rational, meaning that they will deviate from the protocol if and only if it is in their interest to do so.

Recently, some researchers have proposed studying mixtures of these traditional cryptographic and economic models, with various combinations of honest, malicious, and rational participants. One motivation for this that it may allow a more accurate modeling of the diversity of participants in real-life executions of protocols. Along these lines, the papers of Aiyer et al. [2], Lysyanskaya and Triandopoulos [30], and Abraham et al. [1] construct protocols that achieve the best of both worlds. Specifically, they take protocol properties that are known to be achievable in both the cryptographic model (with honest and malicious parties) and the economic model (with only rational parties), and show that protocols with the same properties can still be achieved in a more general model consisting of malicious and rational parties.

Our work is of the opposite flavor. We consider properties that are not achievable in either the cryptographic or economic models alone, and show that they can be achieved in a model consisting of both honest and rational parties. Specifically, we consider the task of secret reconstruction in *secret sharing*, and provide a protocol that is *fair*, meaning that all parties will receive the output, given many rational participants together with a small minority of honest participants. In standard communication models, fairness is impossible in a purely economic model (with only rational participants) [20, 25] or in a purely cryptographic model (with a majority of malicious participants) [10]. Previous works in the individual models achieved fairness by assuming strong communication primitives that allow simultaneous exchange of information [20, 19, 1, 25, 28, 29, 21][5] or settled for approximate notions of security/equilibria [12, 7, 17, 38, 25], whereas we only use a standard (i.e. synchronous but not simultaneous) broadcast channel and achieve a standard notion of game-theoretic equilibrium (namely, a trembling hand perfect equilibrium).

Thus, our work illustrates the potential power of a small number of honest parties to maintain equilibria in protocols. These parties follow the specified strategy even when it is not in their interest to do so, whether out of altruism or laziness. While we study a very specific problem (secret sharing reconstruction, as opposed to general secure function evaluation), we hope that eventually the understanding developed in this clean setting will be leveraged to handle more complex settings (as has been the case in the past).

Below, we review the cryptographic and economic paradigms in more detail. We then introduce the secret-sharing problem we study and survey recent works on this problem in the purely economic model. We then describe our results and compare them to what was achieved before.

[5] Actually, the impossibility results of [20, 25] also hold in the presence of a simultaneous broadcast channel and thus the works of [20, 19, 1, 25] use additional relaxations, such as allowing the number of rounds and/or the sizes of the shares to be unbounded random variables.

1.1 The Cryptographic Paradigm

In the cryptographic paradigm, we allow for a subset of the parties to deviate from the protocol in an arbitrary, malicious manner (possibly restricted to computationally feasible strategies), and the actions of these parties are viewed as being controlled by a single adversary. Intuitively, this captures worst-case deviations from the protocol, so protocols protecting against such malicious and monolithic adversaries provide a very high level of security. Remarkably, this kind of security can be achieved for essentially every multiparty functionality, as shown by a series of beautiful results from the 1980's [45, 18, 8, 5, 39]. However, considering arbitrary (and coordinated) malicious behavior does have some important limitations. For example, it is necessary to either assume that a majority of the participants are honest (i.e. not controlled by the adversary) or allow for protocols that are unfair (i.e. the adversary can prevent some parties from getting the output). This follows from a classic result of Cleve [10], who first showed that there is no fair 2-party protocol for coin-tossing (even with computational security), and then deduced the general version by viewing a multiparty protocol an interaction between two super-parties, each of which controls half of the original parties. Lepinski et al. [28] bypass this impossibility result by assuming a strong communication primitive ("ideal envelopes") which allow simultaneous exchange of information, but it remains of interest to find ways of achieving fairness without changing the communication model.

1.2 The Economic Paradigm

In the economic paradigm, parties are modeled as rational agents with individual preferences, and will only deviate from the protocol if this is in their own self interest. This approach has become very popular in the CS literature in recent years, with many beautiful results. There are two aspects of this approach:

1. Design computationally efficient mechanisms (i.e. functionalities that can be implemented by a trusted mediator) that give parties an incentive to be truthful about their private inputs, while optimizing some *social choice function*, which measures the benefit to society and/or the mechanism designer [32, 27, 3].
2. Implement these mechanisms by distributed protocols, with computational efficiency emphasized in *distributed algorithmic mechanism design* [13–15] and extended to also emphasize additional equilibrium considerations in *distributed implementation* [42, 36, 37], so that parties are "faithful" and choose to perform message passing and computational tasks in *ex post* Nash equilibrium. More recent works achieve a strong form of distributed implementation, with provably no additional equilibria [29, 21], but require strong communication primitives.

Note that distributed algorithmic mechanism design is different in spirit from the traditional problem considered in cryptographic protocols, in that parties have "true" private inputs (whereas in cryptography all inputs are considered equally valid) and there is freedom to change how these inputs are mapped to outcomes through choosing appropriate social choice functions to implement (whereas in cryptography, the functionality is pre-specified.) Nevertheless, recent works have explored whether we can use the economic model to obtain 'better' solutions to traditionally cryptographic problems, namely to compute some pre-specified functionalities. One potential benefit is that we may be able to incentivize parties to provide their "true" private inputs along the lines of Item 1 above; the papers [31, 43] explore for what functionalities and kinds of utility functions this is possible in the presence of game-theoretic agents.

A second potential benefit is that rational deviations may be easier to handle than malicious deviations (thus possibly leading to protocols with better properties), while also preferable to assuming a mixture of players at the honest and malicious extremes. This has led to a line of work, started by Halpern and Teague [20] and followed by [19, 1, 25], studying the problems of secret sharing and multiparty computation in the purely economic model, with all rational participants. One can also require notions of equilibria that are robust against coalitions of rational players [1]. While this approach has proved to be quite fruitful, it too has limitations. Specifically, as pointed out in [19, 25], it seems difficult to construct rational protocols that are fair in the standard communication model, because parties may have an incentive to stop participating once they receive their own output. The works [20, 19, 1, 25], as well as [29, 21] applied to appropriately designed mediated games, achieve fairness by using strong communication primitives (simultaneous broadcast, "ballot boxes") that allow simultaneous exchange of information.

As mentioned above, we achieve fairness in the standard communication model by considering a mix of many rational participants together with a *small* minority of honest participants. Note that Cleve's [10] proof that an honest majority is necessary in the cryptographic setting, by reduction to the two-party case, no longer applies. The reason is that we cannot view a subset of the rational parties as being controlled by a single super-party. Even when considering coalitions, it seems that each individual in that subset would only agree to a coordinated (joint) deviation if it is in its own interest to do so.

Our protocol is for the share reconstruction problem in secret sharing, which we now describe in more detail.

1.3 Secret Sharing

In a *t-out-of-n secret-sharing scheme* [41, 6], a dealer takes a secret s and computes n (randomized) *shares* $s_1, \ldots, s_n$ of s, which are distributed among n parties. The required properties are that (a) any set of t parties can reconstruct the secret s from their shares, but (b) any set of fewer than t parties has no information about s (i.e. they would have been equally likely to receive the same shares for every possible value of s).

Secret sharing is a fundamental building block for cryptographic protocols [18, 5, 8, 39]. Typically, these protocols are structured as follows. First, every party shares its private input among all the parties. Then the computation of the functionality is done on shares (to maintain privacy). And at the end, the parties reveal their shares of the output so that everyone can reconstruct it. Our focus is on this final reconstruction step. Typically, it is assumed that there are enough honest parties in the protocol to ensure that the secret can be reconstructed from the revealed shares, even if some parties refuse to reveal their shares or even reveal incorrect values. This turns out to be achievable if and only if more than a 2/3 fraction of the players are honest [9]. (In previous versions of the paper, we restricted attention to *fail-stop deviations* where a party may stop participating in the protocol early but otherwise follows the prescribed strategy, in which case only an honest majority is needed in the traditional cryptographic model.)

1.4 Rational Secret Sharing

It is natural to ask whether we can bypass this need for an honest majority by considering only *rational* deviations from the protocol. As noted above, the study of secret sharing with only rational participants was initiated by Halpern and Teague [20], and there have been several subsequent works [19, 25, 1]. In these works, it is assumed that participants prefer to learn the secret over not learning the secret, and secondarily, prefer that as few other agents as possible learn it. As pointed out in Gordon and Katz [19], any protocol where rational participants reveal their shares sequentially will not yield a Nash equilibrium. This is because it is rational for the t'th player to stop participating, as she can already compute the secret from the shares of the first $t-1$ players and her own, and stopping may prevent the first $t-1$ players from learning it.

One way to get around this difficulty is to assume a *simultaneous broadcast channel*, where all parties can broadcast values at the same time, without the option of waiting to see what values the other parties are broadcasting. All parties simultaneously revealing their shares is a Nash equilibrium. That is, assuming all of the other parties are simultaneously revealing their shares, no party can increase her utility by aborting (stopping early) instead of revealing. This basic protocol is instructive because it has several deficiences:

1. A simultaneous broadcast channel is a strong (and perhaps unrealistic) communication primitive, particularly in the context of trying to achieve fairness, where the typical difficulties are due to asymmetries in the times that parties get information. For example, fair coin-tossing is trivial with a simultaneous broadcast channel (everyone broadcasts a bit, and the output is the exclusive-or), in contrast to Cleve's impossibility result for synchronous broadcast channels [10].
2. Nash Equilibrium in this context is a very weak guarantee. As argued by Halpern and Teague [20], it seems likely that rational parties would actually abort. The reason is that aborting is never worse than revealing, and is sometimes better (if $t-1$ other parties reveal, then the tth party will always learn the secret and can prevent the other parties from doing so by aborting.)

Halpern and Teague [20] and follow-up works [19, 1, 25] focus on the second issue. That is, they allow simultaneous broadcast, and explore whether stronger solution concepts than plain Nash equilibrium can be achieved. Halpern and Teague [20] propose looking for an equilibrium that survives "iterated deletion of weakly dominated strategies." They prove that no bounded-round protocol can achieve a fair outcome in equilibrium when adopting this solution concept. However, they and subsequent works by Gordon and Katz [19] and Abraham et al. [1] show that fair outcomes are possible even with this equilibrium refinement using a probabilistic protocol whose number of rounds has finite expectation. Moreover, Abraham et al. [1] show how to achieve an equilibrium that is resistant to deviations by coalitions of limited size. Kol and Naor [25] argue that "strict equilibria" is a preferable solution concept to the iterated deletion notion used by Halpern and Teague [20], and show how to achieve it with a protocol where the size of shares dealt is an unbounded random variable with finite expectation. (They also show that a strict equilibrium cannot be achieved if the shares are of bounded size.) In all of the above works, the protocols' prescribed instructions crucially depend on the utilities of the various players.

The works of Lepinski et al. [29] and Izmalkov et al. [21] also can be used to obtain fair protocols for secret sharing by making an even stronger physical assumption than a simultaneous broadcast channel, namely "ballot boxes." Specifically, they show how to compile any game with a trusted mediator into a fair ballot-box protocol with the same incentive structure. Since the share-reconstruction problem has a simple fair solution with a trusted mediator (the mediator takes all the inputs, and broadcasts the secret iff *all* players reveal their share), we can apply their compiler to obtain a fair ballot-box protocol. But our interest in this paper is on retaining standard communication models.

1.5 Our Results

In this paper, we address both issues above. Specifically, we assume that there is at least some *small* number k of honest participants, and along with many rational players. In this setting, we exhibit a simple protocol that only requires a standard communication model, namely *synchronous broadcast*, and in cases where the total number of players is sufficiently large, achieves fair outcomes with high probability with respect to a strong solution concept, namely (a set-Nash analogue of) *trembling hand perfect equilibrium.* We describe both aspects of our result in more detail below.

Synchronous Broadcast. With a *synchronous* (as opposed to *simultaneous*) broadcast channel, the protocol proceeds in rounds, and only one party can broadcast in each round.[6] When all parties are rational, the only previous positive results in this model are in works by Kol and Naor [25, 24], who achieve a fair solution with an approximate notion of Nash equilibrium — no party can improve her utility by ε by deviating from the protocol. However, it is unclear whether

[6] For round efficiency, sometimes people use a slightly more general channel where many parties can broadcast in a single round, but deviating parties are can perform 'rushing' — wait to see what others have broadcast before broadcasting their own values. We describe how to extend our results to this setting below.

such ε-Nash equilibria are satisfactory solution concepts because they may be unstable. In particular, how can everyone be sure that some parties will not try to improve their utility by ε? Once this possibility is allowed, it may snowball into opportunities for even greater gains by deviation. Indeed, Kol and Naor argue in favor of *strict* Nash equilibria, where parties will obtain strictly less utility by deviating (and show how to achieve strict equilibria in the presence of a simultaneous broadcast channel).

In our work, we achieve an exact notion of equilibrium (i.e. $\varepsilon = 0$). However, we allow a negligible probability that the honest parties will fail to compute the secret correctly, so our notion of "fairness" is also approximate. Nevertheless, we feel that the kind of error we achieve is preferable to ε-Nash. Indeed, the equilibrium concept is supposed to ensure that parties have an incentive to behave in a particular manner; if it is too weak, then parties may ignore it entirely and whatever analysis we do may be rendered irrelevant. On the other hand, if we achieve a sufficiently strong notion of exact equilibrium, then we may be confident that players will behave as predicted, and we are unlikely to see any bad events that are shown to occur with small probability under equilibrium play.

Trembling-Hand Equilibrium. In order to establish the equilibrium properties of the protocol, we introduce a framework of "extensive form games with public actions and private outputs," and use the formalism of incomplete information games to model players' uncertainties about the inputs (i.e. shares) of other players as well as uncertainty about which players are honest and which are rational. (For simplicity, we assume that each player is honest independently with some probability p, but with small modifications, the result should extend to other distributions on the set of honest players.) The solution concept of *Bayesian Nash equilibrium* handles the uncertainty that a player has about the shares dealt to other players and requires that beliefs are updated according to Bayes rule "whenever possible," meaning that this occurs when the observed actions are consistent with the equilibrium. A standard refinement is that of Bayesian *subgame perfect* Nash equilibrium, which captures the idea that the strategy is rational to follow regardless of the previous history of messages; intuitively, this means that the equilibrium does not rely on irrational empty threats (where a player will punish another player for deviating even at his own expense). In fact, we achieve the additional refinement of *trembling hand perfect equilibrium* [40], which strengthens this notion by requiring that players update their beliefs in a consistent and meaningful manner even when out-of-equilibrium play occurs. It is one of the strongest solution concepts studied for extensive form games; related notions were advocated in this context by Peter Bro Miltersen (personal communication) and Jonathan Katz [23].

Our Protocol. The protocol that we instruct honest players to follow is simple to describe. The participants take turns broadcasting their shares in sequence. However, if any of the first $t-1$ parties deviates from the protocol by stopping and refusing to broadcast her share, then the protocol instructs all parties subsequent to the first $t - 1$ to do the same. The intuition behind this protocol is that if there is likely to be at least one *honest* party after the first $t-1$ parties, then each rational party in the first $t-1$ parties will also have an incentive to reveal its share because by doing so, the honest party will also reveal her share and enable the rational parties to reconstruct the secret. Then we observe that as long as the set of honest parties is a random subset of $k = \omega(\log n)$ parties, and assuming that the total number, n, of players is sufficiently large, then there will be an honest party after party $t-1$ with all but negligible probability, as long as $t \leq (1-\Omega(1)) \cdot n$. Thus, assuming that parties have a nonnegligible preference to learn the secret, we obtain an *exact* equilibrium in which *everyone* learns the secret with all but negligible probability.

In order to deal with the possibility that some players may try to reveal incorrect shares, we use information-theoretic message authentication codes (MACs) to authenticate the shares, following Kol and Naor [25]. Intuitively, we can tolerate the (negligible) forgery probability of the MACs (without getting an ε-Nash equilibrium) because the first $t-1$ players actually achieve *strictly* higher utility by revealing a valid share than by not doing so.

In addition, the incentives in our protocol hold regardless of what information the first $t-1$ players have about each others' actions, and similarly for the last $n-t+1$ players. Thus, our protocol can actually be implemented with only 2 rounds of communication (in contrast to all previous protocols, which required a super-constant number of rounds); we discuss how to formalize this below.

Modeling Contributions. While the intuition for our protocol is quite natural, modeling it game-theoretically turns out to be quite delicate. As discussed above, we introduce a Bayesian framework for capturing the uncertainty that players have about each others' secrets and which other players are honest vs. rational. Additional modeling contributions include:

Set Nash. We find it useful to avoid specifying the exact actions that rational players should take in situations where the choice is irrelevant to the overall strategic and fairness properties of our protocol. We do this by developing a variant of the notions of Set-Nash [26] and CURB (Closed Under Rational Behavior) Sets [4] for extensive-form games and trembling-hand perfect equilibrium. Roughly speaking, this notion allows us to specify the equilibrium actions only in cases that we care about, and argue that players have no incentive to deviate from the specified actions provided that all other players are playing according to the specified actions (even if they may act arbitrarily when the action is unspecified) and given the existence of a small number of honest players. Since the honest strategy is consistent with the specified equilibrium actions, this solution concept ensures that even repeated rational deviations from the honest strategy (which we envision to be initial "program" distributed to all players) by all but a small number of players will keep everyone consistent with the specified actions. When this occurs as predicted, we show that all honest players will learn the secret with all but negligible probability, and thus fairness is maintained.

Modeling Rushing. To save on rounds, the cryptography literature often allows protocols that specify messages for several players at once, but allows the possibility that deviating players may wait to see other players' messages before computing their own (i.e. simultaneity is allowed but not enforced). Modelling such "rushing" game-theoretically was posed as a challenge in the survey talk of Katz [23]. As mentioned above, we argue that our protocol can be collapsed to two rounds of communication. To capture the possibility of rushing game-theoretically, we follow an idea of Kalai [22], and argue that the specified strategy remains an equilibrium for every ordering of players within each round. Thus, players have no incentive to wait for other players' messages; sending the same message will maximize their utility regardless of what other players send in the same round.

1.6 Future Directions and Independent Work

We view our work as but one more step in the line of work understanding the benefits of bringing together cryptography and algorithmic mechanism design. (See the survey [23].) While our main theorem is admittedly far from achieving an end goal that one would want to implement as is, we hope that our high-level message (regarding the benefit of a few honest players with many rational players) and our game-theoretic modelling (e.g. the Bayesian framework, the use of set-Nash, and the modelling of rushing) prove useful in subsequent work. Some specific ways in which our results could be improved are:

- Handling other distributions on (i.e. beliefs about) the set of honest players. Intuitively, this should be possible by having the dealer randomly permuting the order of the players and including the permutation in the shares (or publishing it).
- Achieving solution concepts that are robust even to coalitional deviations from the protocol. In an earlier version of our paper [34], we demonstrated coalition-proofness (against "stable" coalitions) in a model that is even more simplified than the fail-stop one. As we have mentioned, Abraham et al. [1] show how to handle arbitrary, not necessarily stable, collusions of a small number of players with a simultaneous broadcast channel.
- Generalizing from secret sharing to secure multiparty computation. Indeed, this is the main application for secret sharing and their reconstruction protocols.
- Getting stronger impossibility results for the entirely rational setting (prior impossibility results either require players to learn the secret with probability 1 [25], or suffered other restrictive constraints [35]) or, alternatively, finding a purely rational protocol.

O'Neill and Sangwan [33] extend the results from a preliminary version of our paper [35] in several ways, including achieving a strict trembling-hand perfect equilibrium for a restricted deviation model (which is still more general than the fail-stop deviation model we considered in [35]) and handling a small number of malicious players in this model. Fuchsbauer et al. [16] have recently shown how to obtain a computational analogue of trembling-hand ε-equilibrium on a standard communication channel when all players are rational.

2 Definitions

2.1 Games with Public Actions and Private Outputs

To cast protocol executions into a game-theoretic setting, we introduce the notion of *extensive games with public actions and private outputs.* The basis of this new notion is the more standard definition of *extensive form games with perfect information.* Extensive form games enable us to model the *sequential* aspect of protocols, where each player considers his plan of action only following some of the other players' messages (the "*actions*" of the game-theoretic model). The perfect information property captures the fact that each player, when making any decision in the public phase of the protocol, is perfectly informed of all the actions by other players that have previously occurred. Thus, extensive form games with perfect information are a good model for communication on a *synchronous broadcast channel.*

We build upon extensive form games with perfect information and augment them with an additional final *private* stage. This additional stage models the fact that at the end of the game, each player is allowed to take some arbitrary action as a function of the "terminal" history $h \in Z$ of messages so far. This action, along with the "non-terminal history" $h \in H \setminus Z$ of public actions that have taken place during the execution of the game (as well as the players' inputs) has a direct effect on players' payoffs.

Working in the framework of Bayesian games of incomplete information, players $i \in N$ are handed private inputs θ_i (a.k.a. "*types*") that belong to some pre-specified set Θ_i and specify a distribution μ according to which the inputs are chosen. Players' inputs can be thought of as the shares for the secret-sharing scheme, and are generated jointly with the secret. The secret is thought of as a "reference" value $\delta \in \Delta$ that is not given to the players at the beginning of the protocol (but may be determined by them through messages exchanged), and is used at the output stage along with private actions to determine player utilities.

A game $\Gamma = (N, H, P, A, L, \Delta, \Theta, \mu, u)$ proceeds as follows: the reference value and the types are selected according to a joint distribution μ. The type $\theta_i \in \Theta_i$ is handed to player $i \in N$ and the value $\delta \in \Delta$ remains secret and affects the players' utilities. This is followed by a sequence of actions that are visible by all players. After any history $h \in H$, player $i = P(h)$ whose turn to play is next chooses a public action $a \in A_i(\theta_i, h)$. This choice determines the next actions of the players, and so on until a terminal history $h \in Z$ is reached. At this point, all players $i \in N$ simultaneously pick an action, $b_i \in L_i(\theta_i, h)$, where $L_i(\theta_i, h) \subseteq \Delta$. The utility (or payoff) of player i for an execution of the game is then determined to be the value $u_i(\delta, \theta, h, b_1, \ldots, b_n)$.

2.2 Set Valued Strategies and Set Nash Equilibrium

The action chosen by a player for every history after which it is her turn to move, is determined by her *strategy* function. As is required in extensive-form games, the strategy is defined for all histories, even ones that would not be reached if the strategy is followed.

To enable a simple description of our protocol, and in order to minimize the difference between the description of the behavior of an honest player and that of a rational player, we allow each player to have a strategy that actually maps each information set (i.e. view of the player) into a *set* of possible actions. More precisely, a *set-valued strategy* for player $i \in N$ is a pair $S_i = (M_i, F_i)$, where:

- The *public set-valued strategy* M_i is a function that takes a pair $(\theta_i, h) \in \Theta_i \times (H \setminus Z)$ and defines a set of public messages, $M_i(\theta_i, h)$.
- The *private set-valued strategy* F_i is a function that takes a pair $(\theta_i, h) \in \Theta_i \times Z$ and defines a set of private outputs $F_i(\theta_i, h)$.

We write $s_i = (m_i, f_i) \in S_i$ to indicate that strategy s_i is consistent with S_i, i.e. with $m_i(\theta_i, h) \in M_i(\theta_i, h)$ for all $(\theta_i, h) \in \Theta_i \times (H \setminus Z)$ and $f_i(\theta_i, h) \in F_i(\theta_i, h)$ for all $(\theta_i, h) \in \Theta_i \times Z$ (where these inclusions should hold with probability 1 in case s_i is a mixed strategy).

We will allow the public and private strategy functions to be mixed, where the randomization of the strategy is interpreted to be done independently at each application of the function, if a player has multiple moves in the game. Strategies that consist of deterministic functions are called *pure*, whereas strategies whose functions have full support on the player's action set are said to be *fully mixed*. We achieve fairness (with high probability) in a pure strategy equilibrium but use fully mixed strategies in defining the concept of trembling hand equilibrium.

The *outcome* o of the game Γ under a strategy $s \in S$ is the random variable $(\delta, \theta, h, b_1, \ldots, b_n)$, where $(\delta, \theta) \in \Delta \times \Theta$ are sampled according to μ, $h \in Z$ is the terminal history that results when each player $i \in N$ is given her type $\theta_i \in \Theta_i$, publicly follows the actions chosen by m_i, and computes her final private output b_i using f_i. The value of player i's utility is totally determined by o. The initial distribution, μ, of the secret and the shares, along with the strategies $s_i = (m_i, f_i)$ induce a distribution on o, and thus on the utilities. Define $u_i(\mu, s)$ to be the expected value of the utility of player $i \in N$, when the types are selected according to the distribution μ and all players follow strategy s. We assume that rational players seek to maximize expected utility.

Definition 2.1 (Set Nash equilibrium) *A profile $S = (S_1, \ldots, S_n)$ of set-valued strategies is a* (Bayesian) Set Nash equilibrium *for a game Γ if for all $i \in N$, every (possibly mixed) $s_{-i} \in S_{-i}$, there exists a strategy $s_i \in S_i$ so that for all strategies s'_i, $u_i(\mu, (s_{-i}, s'_i)) \leq u_i(\mu, s)$.*

Our definition of set-Nash equilibrium is stronger than the set-Nash equilibrium definition introduced by Lavi and Nisan [26], who require only that for every *pure* $s_{-i} \in S_{-i}$ there exists some $s_i \in S_i$ for which $u_i(\mu, s) \geq u_i(\mu, (s_{-i}, s'_i))$ for all possible strategies s'_i. This earlier definition of set-Nash is insufficient to ensure that there is a Nash equilibrium consistent with the set-valued strategy profile S. The problem is that upon restricting the game to S the only Nash equilibrium may be a mixed equilibrium, yet there may be some strategy s'_i outside of S_i that is strictly better than all $s_i \in S_i$ given that players $\neq i$ play a mixed strategy consistent with S_{-i}. On the other hand, our definition of set-Nash is weaker than the CURB (Closed Under Rational Behavior) sets of Basu and Weibull [4]; see also a recent discussion in Benisch et al. [11]. A CURB set requires that for every mixed strategy s_{-i} consistent with S_{-i}, *all* best-responses for player i are consistent with set-valued strategy S_i whereas we require only that there is *some* best-response that is consistent with S_i.

2.3 Trembling-Hand Perfect Set Equilibrium

At the heart of a solution concept for extensive-form games with incomplete information is a requirement about the way in which the players update their *beliefs* about the values of other players' types. The beliefs are distributions from which players think that the types of other players were drawn. At the beginning of the game, the belief corresponds to the initial distribution μ conditioned on the player's knowledge of her own type. As the game progresses, players update their beliefs as a function of other players' actions.

A straightforward approach for a player to update her beliefs is to use Bayes rule to condition on her own view of the actions taken in the game. This is the basic approach taken in the game theory literature, and the one pursued in a previous version of this paper [35]. But such an approach suffers from the drawback that updating is not well-defined for views that occur with zero probability, i.e. following out of equilibrium play.

A stronger approach, also discussed in the game theory literature, is the one of *trembling-hand perfect equilibrium* [40]. The idea behind trembling-hand perfect equilibrium is that updating is not problematic if the strategies under consideration are fully mixed (since such strategies would never incur an updating that conditions on a zero probability event). It thus becomes natural to require that the equilibrium strategy is a best response in every subgame to some sequence of fully mixed strategies that converge to equilibrium, and this is indeed the definition of trembling hand equilibria.

The trembling-hand solution concept builds on the notion of a subgame $\Gamma(s, h)$, which is defined in the natural way to be the restriction of an extensive form game with public actions and private outputs Γ, at a history $h \in H$. This definition implicitly captures the way in which players update their beliefs as a result of past players' actions, assuming previous play according to strategy s.

Given a strategy profile s and a strategy s'_i for player i, we denote by $u_i|_h(\mu|_h, s_{-i}|_h, s'_i|_h)$ the expected value of player i's utility under strategy vector $(s_{-i}|_h, s'_i|_h)$ in the game $\Gamma(s, h)$. This is interpreted as considering player i's expected utility when all players except player i follow strategies s_{-i}, and assuming that until the history h has been reached player i has played according to strategy s_i, and from that point on according to strategy s'_i.

We define a set-Nash analogue of trembling-hand perfect equilibrium. To the best of our knowledge, such a combination has not been previously considered in the literature. Given a history $h \in H$ and a set valued strategy S_i, we define $S_i|_h$ in the natural way (i.e., if $S_i = (M_i, F_i)$ then $S_i|_h = (M_i|_h, F_i|_h)$ where $M_i|_h(\theta_i, h') = M_i(\theta_i, (h, h'))$ and $F_i|_h(\theta_i, h') = F_i(\theta_i, (h, h'))$).

Definition 2.2 (Trembling-hand perfect set equilibrium) *Let Γ be an extensive form game with public actions and private outputs. A profile of set-valued strategies $S = (S_1, \ldots, S_n)$ is said to be a* trembling-hand perfect set equilibrium *for Γ if for every $s \in S_1 \times \cdots \times S_n$ there exists a sequence of fully mixed strategy profiles $(s^k)_{k=0}^{\infty}$ converging to s so that for every history $h \in H$ and every $i \in N$, there exists a strategy $s'_i \in S_i|_h$ such that for all strategies s''_i in the game $\Gamma(s^k, h)$ it holds that $u_i|_h(\mu|_h, s^k_{-i}|_h, s'_i) \geq u_i|_h(\mu|_h, s^k_{-i}|_h, s''_i)$ in the game $\Gamma(s^k, h)$ for all k.*

3 Secret Sharing

A secret sharing scheme $(N, t, \Delta, \Theta, \mu, g)$ is implemented by letting a trusted *dealer* jointly pick the secret and shares according to the distribution μ, and then distributing share $\theta_i \in \Theta_i$ to player $i \in N$. The reconstruction functions are what enables any set S of at least t players to use their shares $(\theta_i)_{i \in S}$ in order to jointly reconstruct the secret (by using a function $g_S \in g$). The scheme should also guarantee secrecy against any subset S of less than t players.

To prevent players from revealing shares that are different than the ones they were dealt, we will want to work with a secret sharing scheme that is *authenticated*. One can use standard “information theoretic” techniques for authenticating shares in any (plain) secret sharing scheme (cf. [44, 39, 25]).

3.1 Reconstruction Protocols

Once shares are distributed among the players, it is required to specify a protocol according to which the players can jointly reconstruct the secret at a later stage. The reconstruction protocol prescribes a way in which the players compute their “messages”, which are chosen from a given fixed “alphabet,” and are then broadcast to all other players. The protocol also specifies an output function that is used by the players to compute their (private) output.

A reconstruction protocol $\Pi = (\Sigma, H, P, m^*, f^*)$ for a given secret sharing scheme is implemented under the assumption that the secret and shares $(\delta, \theta_1, \ldots, \theta_n)$ are chosen according to the distribution μ. Player i's type is θ_i. The protocol is interpreted as follows: player $i = P(h)$ chooses a message $m = m^*_i(\theta_i, h) \in \Sigma$; this choice determines the next player to move, and so on until a terminal history $h \in Z$ is reached. At this point all players can determine the value of their private output functions, $f^*_i(\theta_i, h)$.

3.2 Reconstruction Games

A secret sharing protocol induces a *reconstruction game* in a natural way. Loosely speaking, this is an interpretation of a reconstruction protocol as an extensive form game with public messages and private outputs, in which arbitrary deviations from the protocol's instructions are allowed. The interpretation of the protocol as a game is straightforward: protocol histories correspond to game histories, messages in the protocol correspond to actions, next message functions correspond to strategies, and the outputs correspond to output actions.

The reconstruction game allows player i the choice between continuing with the protocol's prescribed instructions (and in particular choosing an action according to m_i^*), and deviating from Π (by sending some other message from Σ).

We require that the utility functions are *linear* in the sense there are (real) parameters $\{a_{ij}\}$ such that the utility received by player i is equal to the sum of a_{ij} over all players j that correctly compute the secret at the end of the protocol. We define the players' *preference* for learning the secret to be $\rho = \min_i a_{ii}/(-\sum_{j \neq i} a_{ij})$. We require that $a_{ii} > 0$, $a_{ij} < 0$ for $i \neq j$, and $\rho > 1$. These assumptions correspond to the assumptions (also made in previous works) that players prefer to learn the secret over all else, and secondarily prefer that as few other players learn the secret as possible.

4 Our Protocol

4.1 Introducing an honest minority

Our goal is to show that every authenticated secret-sharing scheme has a reconstruction protocol so that any reconstruction game that corresponds to it has an equilibrium strategy in which all players learn the secret. To do this, we require that a small subset of *honest* players in the reconstruction game always follows the strategy prescribed by the reconstruction protocol (whether or not this is the best response to other players' actions). We model this scenario by assuming that the set of honest players is selected according to some distribution that specifies to each player whether she is to act honestly or rationally. The set of actions of an honest player coincides with the strategy prescribed by the reconstruction protocol. The set of actions of a rational player remains unchanged.

The private type of player $i \in N$ in a reconstruction game with honest players consists of a pair $(\theta_i, \omega_i) \in \Theta_i \times \Omega_i$ that is drawn along with other player's types and the reference value δ according to the distribution $\mu \times \zeta$. The value of $\omega_i \in \{\texttt{honest}, \texttt{rational}\}$ determines whether player i is bound to follow the honest strategy (as prescribed by Π), or will be allowed to deviate from it. We constraint the set of actions of each player in order to create a situation in which rational players are indeed free to deviate from the public strategy vector m^* (since they are allowed to choose any action in Σ), whereas the honest players are in fact restricted to the single action prescribed by m^*.

4.2 Main Result

We show that assuming the existence of a small number of honest players, there is a reconstruction protocol such that every corresponding reconstruction game has an equilibrium such that with high probability all players learn the secret, provided that the set of honest players is uniform among all sets of a sufficiently large size and every player has a nonnegligible preference for learning the secret. Specifically, our theorem is the following:

Theorem 4.1 *Every authenticated secret-sharing scheme* $(N, t, \Delta, \Theta, \mu, g)$*, with* $t < |N| = n$*, has a reconstruction protocol* Π *such that the following holds. Let* $\Gamma = (N, H, P, A, L, \Delta, \Theta \times \Omega, \mu \times \zeta_m, u)$ *be a reconstruction game that corresponds* *to Π with honest players and linear utility functions, where ζ_m is a distribution over tuples $(\omega_1, \ldots, \omega_n) \in \Omega$ for which $\omega_i =$ `honest` with probability m/n independently for all $i \in N$, for some real number $m \in [0, n]$. Suppose further that the players' preference ρ for learning the secret satisfies:*

$$\rho > \frac{1 - 1/|\Delta|}{1 - 1/|\Delta| - p(n,m) - \gamma} \tag{1}$$

where $p(n,m) = (1 - m/n)^{n-t+1} \leq \exp(-m \cdot (n-t)/n)$ and γ is the forgery probability of the authenticated secret-sharing scheme. Then Γ has a profile $S = (S_1, \ldots, S_n)$ of set-valued "rational" strategies such that:

1. *The honest strategy profile $s^* = (m^*, f^*)$ is consistent with S,*
2. *S is a trembling-hand perfect set equilibrium in Γ,*
3. *For every strategy vector $s \in S$, the probability that all* honest *players compute the secret correctly in Γ is at least $1 - (n-t+1)\gamma - p(n,m)$, when the players' types are chosen according to μ and they follow strategy vector s.*
4. *For every Nash equilibrium $s \in S$, the probability that* all *players compute the secret correctly in Γ is at least $1 - (n-t+1)\gamma - p(n,m)$, when players' types are chosen according to μ and they follow strategy s.*
5. *S does not depend on the utility functions u in Γ (provided they satisfy (1)).*

To interpret this theorem, consider a setting in which we distribute to each player software that is programmed to play the honest strategy, which is in consistent with S by Item 1. Rational players may then decide to deviate from this strategy (i.e. reprogram their software) in order to improve their utility. The fact that S is a trembling-hand perfect set equilibrium (Item 2), however, guarantees that there is no incentive for the rational players to deviate from S, even if this process is iterated. As long as all players remain within S, Items 3 and 4 say that fairness is maintained (with high probability).

In case that $t \leq (1–\Omega(1))\cdot n$, observe that $p(n,m) = \exp(-\Omega(m))$ is negligible provided that $m = \omega(\log n)$, i.e. the expected number of honest players is super-logarithmic. If, in addition, the forgery probability γ is negligible then (1) simply says that a player's preference for learning the secret should not be negligible.

4.3 The reconstruction protocol

The protocol proceeds in two stages, where in the first stage a subset of $t-1$ players is instructed to reveal their share to all other parties in some sequence, and in the second stage the remaining $n-t+1$ players are instructed to reveal their share in some sequence, provided that none of the $t-1$ players in the first stage has failed to reveal her share. The individual parties will reveal their share using a *synchronous broadcast* channel.

The stage in which a player is instructed to broadcast is fixed in some arbitrary manner. For concreteness, suppose that at stage 1 of the protocol, it is the turn of players $1, \ldots, t-1$ to broadcast, and that at stage 2 it is the turn of players $t, \ldots, n$. The protocol will instruct player i to either *reveal* her share $\theta_i \in \Theta_i$ or not to reveal anything (symbolized by a special action denoted $\perp \in \Sigma$).

Specifically, we will require that player i reveals θ_i unless she is one of the stage 2 players and one of the first $t-1$ parties to speak has chosen not to reveal their share. In the latter case player i does not reveal her share either.

In addition to revealing her share, player i is required to send along the authentication information that was provided to her by the dealer. In case that either the authentication fails, or that the player has refused to broadcast her message,[7] player i will be considered as having failed the authentication and chosen the special $\perp$ action.

After the two stages are completed, each player locally uses a reconstruction function $g_S \in g$ in order to try and compute the secret given the shares that have been revealed during the protocol's execution. By the properties of secret sharing, it follows that a party will be able to compute the secret at the end of the protocol if a set $S \subseteq N$ of at least $t-1$ *other* parties have revealed their shares, and otherwise she has no information about the secret.

The protocol requires that the players in each stage reveal their shares in sequential order. However, the order in which the first $t-1$ players broadcast has no effect on the strategic properties of the protocol, and similarly for the last $n-t+1$ players. Thus, the protocol can effectively be implemented with two rounds of communication (see full version for details).

4.4 Rational Strategies for Corresponding Reconstruction Games

The rational set-valued strategy S instructs both honest and rational players to follow the strategy prescribed by Π, except that it does not specify how rational players should act in cases when the honest strategy may not be in their self interest. Specifically, we allow arbitrary action by a rational player $i \geq t$ when the first $t-1$ players have all revealed valid shares (whereas honest players must reveal in this case). The honest strategy (equivalent to the earlier 'protocol') is itself consistent with the rational set-valued strategy profile. In the full version of the paper, we show that S satisfies all the requirements of Theorem 4.1.

5 Acknowledgements

We thank Drew Fudenberg, Jonathan Katz, Silvio Micali, Peter Bro Miltersen, Moni Naor, Adam O'Neill and the anonymous reviewers for helpful discussions and comments.

References

1. I. Abraham, D. Dolev, R. Gonen, and J. Y. Halpern. Distributed computing meets game theory: robust mechanisms for rational secret sharing and multiparty computation. In *PODC'06*, pages 53–62, 2006.
2. A. S. Aiyer, L. Alvisi, A. Clement, M. Dahlin, J.-P. Martin, and C. Porth. Bar fault tolerance for cooperative services. In *SOSP*, pages 45–58, 2005.

[7] In an implementation, a player that fails to broadcast her value within some predetermined amount of time might be considered to have refused to broadcast.

3. M. Babaioff, R. Lavi, and E. Pavlov. Mechanism design for single-value domains. In *Proc. Nat. Conf. on Artificial Intelligence, AAAI05*, 2005.
4. K. Basu and J. W. Weibull. Strategy subsets closed under rational behavior. *Economics Letters*, 36:141–146, 1991.
5. M. Ben-Or, S. Goldwasser, and A. Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation (extended abstract). In *STOC'88*, pages 1–10, 1988.
6. G. Blakely. Safeguarding cryptographic keys. In *AFIPS*, volume 48, p. 313, 1979.
7. D. Boneh and M. Naor. Timed commitments. In *CRYPTO'00, p. 236-254*, 2000.
8. D. Chaum, C. Crépeau, and I. Damgård. Multiparty unconditionally secure protocols (extended abstract). In *STOC'88*, pages 11–19, 1988.
9. B. Chor, S. Goldwasser, S. Micali, and B. Awerbuch. Verifiable secret sharing and achieving simultaneity in the presence of faults (extended abstract). In *FOCS*, pages 383–395. IEEE, 1985.
10. R. Cleve. Limits on the security of coin flips when half the processors are faulty (extended abstract). In *STOC*, pages 364–369. ACM, 1986.
11. G. B. Davis and T. W. Sandholm. Algorithms for Rationalizability and CURB Sets. In *AAAI'06*, 2006.
12. S. Even, O. Goldreich, and A. Lempel. A randomized protocol for signing contracts. *Commun. ACM*, 28(6):637–647, 1985.
13. J. Feigenbaum, C. Papadimitriou, R. Sami, and S. Shenker. A BGP-based mechanism for lowest-cost routing. In *PODC*, pages 173–182, 2002.
14. J. Feigenbaum, C. H. Papadimitriou, and S. Shenker. Sharing the cost of multicast transmissions. *Journal of Computer and System Sciences*, 63:21–41, 2001.
15. J. Feigenbaum and S. Shenker. Distributed Algorithmic Mechanism Design: Recent Results and Future Directions. In *Proc. 6th Int'l Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, pages 1–13, 2002.
16. G. Fuchsbauer, J. Katz, E. Levieil, and D. Naccache. Efficient rational secret sharing in the standard communication model. Cryptology ePrint Archive, Report 2008/488, 2008. http://eprint.iacr.org/.
17. J. A. Garay and M. Jakobsson. Timed release of standard digital signatures. In *Proc. Financial Cryptography 2002*, pages 168–182, 2002.
18. O. Goldreich, S. Micali, and A. Wigderson. How to play any mental game or a completeness theorem for protocols with honest majority. In *STOC*, pages 218–229. ACM, 1987.
19. S. D. Gordon and J. Katz. Rational secret sharing, revisited. In R. D. Prisco and M. Yung, editors, *Proc. SCN*, pages 229–241, 2006.
20. J. Y. Halpern and V. Teague. Rational secret sharing and multiparty computation: extended abstract. In L. Babai, editor, *STOC*, pages 623–632. ACM, 2004.
21. S. Izmalkov, S. Micali, and M. Lepinski. Rational secure computation and ideal mechanism design. In *FOCS*, pages 585–595. IEEE Computer Society, 2005.
22. E. Kalai. Large robust games. *Econometrica*, 72(6):1631–1665, November 2004.
23. J. Katz. Bridging game theory and cryptography: Recent results and future directions. In *TCC'08*, pages 251–272, 2008.
24. G. Kol and M. Naor. Cryptography and game theory: Designing protocols for exchanging information. In *TCC'08*, pages 320–339, 2008.
25. G. Kol and M. Naor. Games for exchanging information. In *STOC*, pages 423–432. ACM, 2008.
26. R. Lavi and N. Nisan. Online ascending auctions for gradually expiring goods. In *SODA'05*, 2005.

27. D. Lehmann, L. I. O'Callaghan, and Y. Shoham. Truth revelation in approximately efficient combinatorial auctions. *Journal of the ACM*, 49(5).
28. M. Lepinski, S. Micali, C. Peikert, and A. Shelat. Completely fair sfe and coalition-safe cheap talk. In *PODC'04*, pages 1–10, 2004.
29. M. Lepinski, S. Micali, and A. Shelat. Collusion-free protocols. In H. N. Gabow and R. Fagin, editors, *STOC*, pages 543–552. ACM, 2005.
30. A. Lysyanskaya and N. Triandopoulos. Rationality and adversarial behavior in multi-party computation. In *CRYPTO*, pages 180–197, 2006.
31. R. McGrew, R. Porter, and Y. Shoham. Towards a general theory of non-cooperative computation. In *TARK*, pages 59–71, 2003.
32. N. Nisan and A. Ronen. Algorithmic mechanism design. *Games and Economic Behavior*, 35:166–196, 2001.
33. A. O'Neill and A. Sangwan. Honesty, rationality, and malice in secret sharing and MPC: Robust protocols for real-world populations. Manuscript, 2008.
34. S. J. Ong, D. Parkes, A. Rosen, and S. Vadhan. Fairness with an honest minority and a rational majority. Available from `http://eecs.harvard.edu/~salil/Fairness-abs.html`, April 2007.
35. S. J. Ong, D. Parkes, A. Rosen, and S. Vadhan. Fairness with an honest minority and a rational majority. Cryptology ePrint Archive, Report 2008/097, March 2008. http://eprint.iacr.org/.
36. D. C. Parkes and J. Shneidman. Distributed implementations of Vickrey-Clarke-Groves mechanisms. In *Proc. 3rd AAMAS*, pages 261–268, 2004.
37. A. Petcu, B. Faltings, and D. Parkes. M-dpop: Faithful distributed implementation of efficient social choice problems. In *AAMAS'06*, pages 1397–1404, May 2006.
38. B. Pinkas. Fair secure two-party computation. In *Proc. EUROCRYPT 2003*, pages 87–105, 2003.
39. T. Rabin and M. Ben-Or. Verifiable secret sharing and multiparty protocols with honest majority (extended abstract). In *STOC*, pages 73–85. ACM, 1989.
40. R. Selten. A reexamination of the perfectness concept for equilibrium points in extensive games. *International Journal of Game Theory*, 4:25–55, 1975.
41. A. Shamir. How to share a secret. *Commun. ACM*, 22(11):612–613, 1979.
42. J. Shneidman and D. C. Parkes. Specification faithfulness in networks with rational nodes. In *PODC'04*, St. John's, Canada, 2004.
43. Y. Shoham and M. Tennenholtz. Non-cooperative computation: Boolean functions with correctness and exclusivity. *Theor. Comput. Sci.*, 343(1-2):97–113, 2005.
44. M. N. Wegman and L. Carter. New hash functions and their use in authentication and set equality. *J. Comput. Syst. Sci.*, 22(3):265–279, 1981.
45. A. C.-C. Yao. How to generate and exchange secrets (extended abstract). In *FOCS*, pages 162–167. IEEE, 1986.