Rational Secret Sharing Under Fairness

Jessica Shi, Evan Wildenhain
Princeton University, COS 521

January 18, 2018

Introduction

Problem

- Nash equilibrium: Players choose moves indep. based on best response
- Correlated equilibrium: Given a **trusted mediator**, can obtain better expected utility
  - Problem: Such a mediator may not exist, and players may not trust each other

Problem Example

- **Battle of the Sexes**:

	A	B
A	2, 1	0, 0
B	0, 0	1, 2

- Nash equilibrium:
  - (A, A): Payoff $(2, 1)$
  - (B, B): Payoff $(1, 2)$
  - $((\frac{2}{3}, \frac{1}{3}), (\frac{1}{3}, \frac{2}{3}))$: Payoff $(\frac{2}{3}, \frac{2}{3})$
- Correlated equilibrium:
  - Mediator flips coin: (A, A) if H, (B, B) if T
  - Payoff $(\frac{3}{2}, \frac{3}{2})$

Solution (sort of) + Prior Work

- Cryptography: Use **multiparty computation** (MPC)
  - Players have secret inputs + collectively compute functions w/o revealing secrets
  - Replace trusted mediator
- Prior work:
  - 2000: Dodis *et al.*: Success given **fair** + secure MPC
  - 1986: Cleve: Fairness is impossible (in gen) w/o honest majority
  - 2000: Dodis *et al.*: Success in 2-PC w/**rational** players
    - Doesn't extend to $>$ 2-PC
  - 2004: Halpern + Teague: Deterministic secret sharing is impossible under iterated deletion of weakly dominated strats

Our Work

- Focus on Halpern + Teague: deterministic secret sharing
- Fair + secure MPC schemes outside of classic cryptography:
  - **Gradual release**: Penalize unfair actions using resources
  - **Compensation**: Penalize unfair actions using money
- Iterated deletion of weakly dominated strats in each subgame:
  - **Gradual release**: Impossible
  - **Compensation**: Success

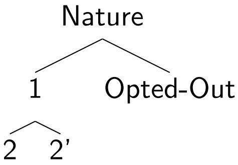
Preliminaries

Game theory

View secret sharing as an finite extensive-form game.

- Players $N = [n]$
- Histories
- Available actions
- Information sets
- Payoffs

Can represent as a *game tree*, where each node is a history and each edge is available actions.



Weak domination

- For player i, a strategy σ_i weakly dominates σ_i' if $U_i(\sigma_{-i}, \sigma_i) \geq U_i(\sigma_{-i}, \sigma_i')$ for all σ_{-i} and $U_i(\sigma_{-i}, \sigma_i) > U_i(\sigma_{-i}, \sigma_i')$ for some σ_{-i}
- We consider a refinement of Nash equilibria: iterated domination of weakly dominated strategies
- Intuitively, we should want no player to play weakly dominated strategies in our protocols, as they have no reason not to play something else
- Iteratively delete through backward induction: start at end of the game, at each info set delete all weakly dominated strategies, iterate up the game tree

t-out-of-n secret sharing

- n parties have shares $(m_1, \ldots, m_n)$ of a secret message m such that anyone with t shares can reconstruct the message, but anyone with $(t-1)$ shares learns nothing about the secret
- Example: dealer chooses random $(t-1)$-degree polynomial f such that $f(0) = m$, distributes shares $m_i := f(i)$

Multiparty computation (MPC)

n parties each have private input x_i and want to jointly compute $f(x_1, \ldots x_n)$ without revealing more than the function output does

Many MPC protocol designs rely on secret sharing

E.g. *BGW* protocol:

- Players "deal" secret shares to other players such that players can use them to compute shares of final output
- Then t players can can combine their shares to receive the final output
- MPC protocols are designed with the assumption of "honest-but-curious" (aka *semi-honest*) players: players correctly execute protocol but attempt to compute as much as possible with the information they get
- Protocols based on t-out-of-n secret sharing are secure against a group of $< t$ passive adversaries: follow protocol but can collude to gain more information
- We will instead consider players that are "rational-but-not-malicious": players either send honest information or send no message at all

Classic Setting

Setup

- ℓ rounds, where at each round player i can:
  - Give j their share m_i
  - Give j a share they have received m_k
  - Give j a share they have received m_k signed by h
- Any player with $\geq t$ shares receives m
- Utilities:
  1. Want to know m
  2. Want as few other players to know m

Classic Setting Impossibility

Theorem

Deterministic secret sharing is impossible assuming a commonly known bound and using iterated deletion of weakly dominated strategies in every subgame.

- Proof sketch:
  - **Backwards induction**: at each level,
  - In every info set:
    - Doing nothing is never weakly dominated
  - In the info set containing everyone doing nothing:
    - Only weakly dominating strat is doing nothing
    - Strictly better than sending a share to j, and having $t-2$ ppl send shares to j

Gradual Release Setting

Gradual Release

- **Gradual release**: Release secrets over time, s.t. if a party aborts at any stage, remaining parties can compute secret in same time as aborting party (approx.)
- Scheme: **Commit-prove-fair-open**:
  - Commit phase: i broadcasts commitment to value x_i
  - Prove phase: i broadcasts proof y_i s.t. $R(x_i, y_i) = 1$
  - Open phase: everyone opens $x_1, \ldots, x_n$ simultaneously (over k rounds)

Timelines

- $N = pq$ is a Blum integer (p, q prime, $= 3 \mod 4$)
- $G = (g, g^2, g^{2^2}, \ldots, g^{2^k})$ in $\mathbb{Z}_N$, $g \in \mathbb{Z}_N^*$; $G[i] = g^{2^i}$
  - Given g, easy to compute $G[i]$ given factorization of N, hard o.w.
- **Yet-more-general BBS assumption**: (YMG-BBS)
  - Let $a_1, \ldots, a_{\ell+1}$ s.t. $|a_{\ell+1} - a_i| \geq 2^\ell \; \forall \; i$
  - Given $(G[a_1], \ldots, G[a_\ell])$, $G[a_\ell]$ appears pseudorandom
- **Decreasing timeline**: $T = \langle N, g, \vec{u} \rangle$ where $u[i] = G[2^k - 2^{k-i}]$
  - $u[k]$ appears pseudorandom by YMG-BBS
- **Derived timeline** of T: $T' = \langle N, h, \vec{v} \rangle$ where $h = g^\alpha$ and $v[i] = (u[i])^\alpha$ for $\alpha \in \mathbb{Z}_{[1,(N-1)/2]}$
  - $v[k]$ appears pseudorandom given T (as long as α is secret)

Implementing CPFO

- T is a **common reference string**
- Commit phase: i derives a timeline $T_i = \langle N, g_i, \vec{u_i} \rangle$ + commits to $(g_i, m_i \cdot u_i[k])$
  - j can **force-open** $m_i \cdot u_i[k]$ by repeatedly squaring g_i; however, exp time
- Prove phase: i gives zero-knowledge pf that they know α_i
- Open phase: In round ℓ, i broadcasts $u_i[\ell]$ (with zero-knowledge pf)
  - If a player aborts, in the next round all players abort + force-open if feasible
  - If not feasible to force-open, aborting player cannot force-open either

Theorem

The commit-prove-fair-open scheme implemented with timelines are fair.[1]

[1] Garay, MacKenzie, and Yang. 2004.

Setup

- $k + 1$ rounds: first for commit-prove phases, rest for open phase
- At each round, player i can:
    - Give j their corresponding timeline-commitment
    - Abort + force-open
- Any player with $\geq t$ shares in a round can force-open
- Utilities:
    1. Want to know m
    2. Want to know m as quickly as possible
    3. Want other players to know m as slowly as possible

Gradual Release Impossibility

Theorem

Deterministic secret sharing under gradual release is impossible assuming a commonly known bound and using iterated deletion of weakly dominated strategies in every subgame.

- Proof is the same as that for the classic setting:
  - **Backwards induction**: at each level,
  - Utilities are s.t. doing nothing is always preferable

Compensation Setting

Intuition

- If a player already has their desired output in an MPC protocol, why continue participating?
- Solution: pay them for participating (or fine them for exiting early)
- Real world implementation: Ethereum smart contracts allow you to create transactions that execute under time restrictions and under certain conditions
- Can construct a *composable* compensation framework: take a semi-honest MPC protocol π_{SH} and use compensation to fine malicious players

Commitment ledger

- Need a ledger that supports special transactions with conditions on how the transferred coins can be spent
- For a transfer of coins from player i to j, can specify:
  - Time restriction
  - State-dependent condition: validation function from current ledger-state, ledger-buffer, and transaction to $\{$valid, not valid$\}$

Compensation protocol for secret sharing: setup

- Every player checks that they have at least $(n-1)c$ coins and chooses whether to participate in protocol
- Every player i that **opts in** makes a "commitment" transaction for every player $j \neq i$: player j can claim c coins from i in round r iff player j sends player i their share of the secret (by embedding it in a "claiming" transaction's `aux` field)

Compensation protocol for secret sharing: claiming committed transactions

From times $\tau = 1, \ldots, \ell + 1$, each player i:

- Reads the ledger's state and computes the state of the protocol π_{SH} given transcript of participants' messages so far
- If protocol has not aborted or terminated, i calculates the messages they need to send to claim coins, and posts those messages in a claiming transaction
- If the protocol has aborted or terminated, post transactions reclaiming the funds from commitment transactions that have not been claimed

Utility assumptions

- First, want to learn the secret
- Second, want to maximize their net coin profit
- Third, want fewer other people to learn the secret

Secret sharing with compensation is dominant-strategy honest participation

Theorem

Utility assumptions imply that the only strategies that survive iterated deletion of weakly dominated strategies are strategies in which every player opts-in at setup and sends its secret share to all players before the end of the final round.

Proof sketch:

- Utility assumptions mean that sending any remaining secrets in the final round strictly dominates not sending them
- After deleting all non-"all-send" strategies, opting-out at setup is weakly dominated by opting in and playing "all-send"

Conclusion

- Deterministic secret sharing under:
  - Gradual release: Impossible
  - Compensation: Success
- MPC using secret sharing:
  - Gradual release: Impossible
  - Compensation: ???
- Future work:
  - Issues w/valuing coins in compensation framework
  - Extend successful result to MPC
  - Iterated deletion of weakly dominated strats in general