Optimally Auditing Adversarial Agents

Sanmay Das, Fang-Yi Yu, Yuang Zhang

VIRGINIA TECH

GEORGE MASON UNIVERSITY

Motivation Questions

- How can we design audits when agents adversarially misreport their private type?
- Examples include
  - **IRS:** Verifying tax deductions and self-reported income.
  - **Insurance:** Checking claims on property or medical damages.
  - **Market Surveillance:** Detecting fraudulent trading activities.

Summary

- **Optimal Non-Adaptive Audit:** Algorithm finds an ε-optimal audit policy in $O(m^2)$ time.
- **Social Welfare & Policy Levers:** An $\boldsymbol{O(m^2)}$ algorithm for social welfare; moreover, higher penalties or lower audit costs weakly improve both the principal's utility and welfare.
- **Learning without a prior :**We design a no-regret online method (EXP3-based) with Regret $\boldsymbol{O(n\sqrt{T}m^2\log m)}$
- **Adaptive adds no value:** Best adaptive = best non-adaptive; the same $\boldsymbol{O(m^2)}$ search.

Model

Agents

- n agents has a private type $i \sim q \in \Delta_m$. Each report type $k \in [m]$.
- Misreporting strategy $\boldsymbol{Q} \in [0,1]^{m\times m}$, where $Q_{i,k} = \Pr[i \to k]$
- The principal's audit policy $\boldsymbol{p} \in [0,1]^m$
- Agents' utility

$$U_{i,k}(\boldsymbol{p}) = \mathrm{pay}(k) - p_k \mathrm{pen}(i,k)$$

For instance, $\mathrm{pen}(i,k) = 1[i \neq k](\mathrm{pay}(k) + c))$ (affine penalty)

Principal's Strategy:

- Non-adaptive: commit a fix audit vector $\boldsymbol{p}$,
- Adaptive: $\boldsymbol{p} = \boldsymbol{\pi}(\hat{\boldsymbol{q}})$ as a function of observed report distribution $\hat{\boldsymbol{q}}$

Principal's utility (n=1)

- Revenue : $V(\boldsymbol{p},\boldsymbol{Q}) = \sum_{i,k} q_i\, Q_{i,k}(\mathrm{val}(i,k) - \mathrm{pay}(k) + p_k \mathrm{pen}(i,k))$
- Audit cost:$C(\boldsymbol{p},\boldsymbol{Q}) = \sum_{i,k} q_i\, Q_{i,k} p_k$
- Costly setting:

$$V_\lambda(\boldsymbol{p},\boldsymbol{Q}) = V(\boldsymbol{p},\boldsymbol{Q}) - \lambda C(\boldsymbol{p},\boldsymbol{Q}) \text{ with cost per audit } \boldsymbol{\lambda}$$

- Budget setting:

$$V_B(\boldsymbol{p},\boldsymbol{Q}) = \begin{cases} V(\boldsymbol{p},\boldsymbol{Q}) & \text{if } C(\boldsymbol{p},\boldsymbol{Q}) \le B \\ -\infty & \text{otherwise} \end{cases}$$

- Agents best-respond to p(or π); the principal maximizes utility under the worst equilibrium (pessimistic/Wardrop).

$$\max_{p} \min_{Q:Eqi(p)} V(p,Q)$$

Bayes-Nash Equilibrium (BNE)

- $\boldsymbol{Q}$ is (non-atomic) BNE under $\boldsymbol{p}$ if, for all types i and reports $k, l \in [m]$ with $Q_{i,k} > 0$:

$$U_{i,k}(\boldsymbol{p}) \ge U_{i,l}(\boldsymbol{p})$$

We set $Eqi(\boldsymbol{p})$ as the set of BNE.

Non-adaptive audit

Theorem 1

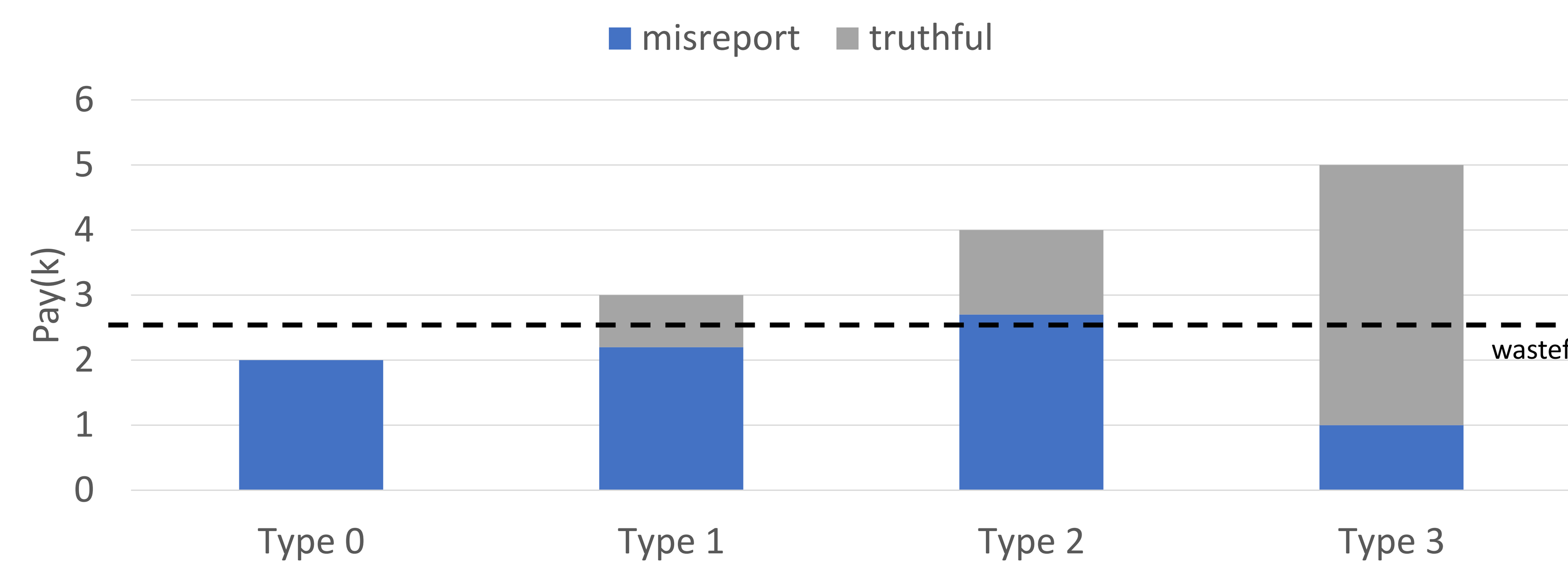
For any small enough $\boldsymbol{\varepsilon > 0}$ and any non-adaptive game with cost λ and parameters (n, m, q, val, pay, pen), there is an algorithm that computes a $2n\varepsilon$ -optimal audit policy in $\boldsymbol{O(m^2)}$ time.

Welfare objective: The same audit-search methods.
Monotonicity: Higher penalties or lower audit costs improve utility and welfare.
Relaxed priors: Even without exact knowledge of $\boldsymbol{q}$ **,** a no-regret online learner achieves near-optimal audits.

Equilibrium Characterization

- Idea: agents' best response is a threshold strategy

$$\hat{U}_k = pay(k) - p_k pen(k)$$



- Choose the audit vector that equalizes $\hat{U}_k$ for all k

$$\rho_k(u) = \frac{pay(k) - u}{pen(k)}$$

- The principal chooses from a family of audit policy, $\boldsymbol{p} = \mathrm{equa}(u, A, \varepsilon)$ to make agents' highest misreport utility is u:

$$p_k = \begin{cases} 0 & k < \iota \\ \rho_k(u) & k \in A \\ \rho_k(u-\varepsilon) & \text{otherwise} \end{cases}$$

Adaptive setting

Assumption on the penalty

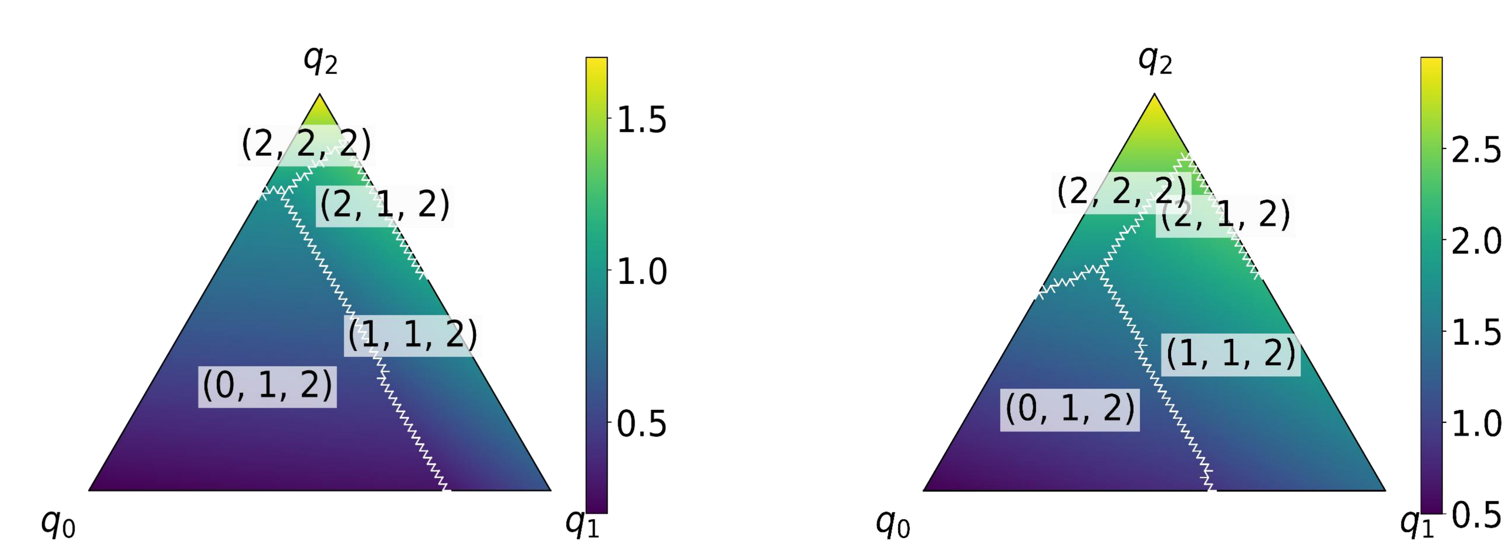
- Type independent: $\mathrm{pen}(i,k) = 1[i \neq k]\mathrm{pen}(k)$
- Insensitive : $\frac{pay(l)}{pay(k)} \ge \frac{pen(l)}{pen(k)}$

Theorem 2 (informal). Under the type independent and insensitive condition, adaptive audits do not outperform non-adaptive ones:

$$\max_{\pi} \min_{Q \in Eqi(\pi)} V_\lambda(\pi, Q) = \max_{p} \min_{Q \in Eqi(p)} V_\lambda(p, Q).$$

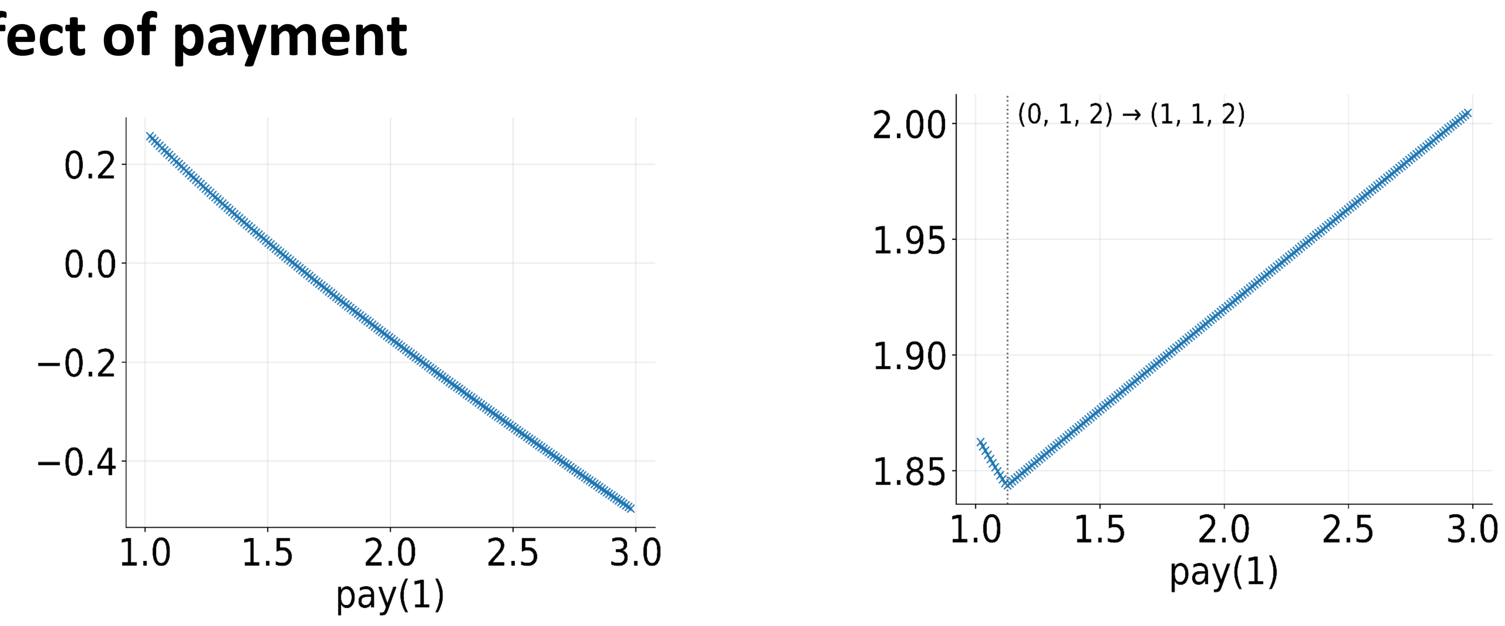
Simulations

Effect of prior



Principal Utility   Social Welfare

Effect of payment



Principal Utility   Social Welfare

Future work

- Generalize to finite agents, noisy or partial verification, and richer penalty structures.
- Design problem of payment and penalty function.
- Connect to security games and tolling games