COBRA: CONTEXTUAL BANDIT ALGORITHM FOR ENSURING TRUTHFUL STRATEGIC AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

This paper considers a contextual bandit problem involving multiple agents, where a learner sequentially observes the contexts and the agents' reported arms, and then selects the arm that maximizes the system's overall reward. Existing work in contextual bandits assumes that agents always truthfully report their arms, which is unrealistic in many real-life applications. For instance, consider an online platform with multiple sellers; some sellers may misrepresent product features to maximize reward, such as having the platform preferentially recommend their products to its users. To address this challenge, we propose an algorithm, COBRA, for contextual bandit settings involving strategic agents that disincentivize their strategic behavior without using any monetary incentives, while having an approximate Bayesian incentive compatibility and a sub-linear regret guarantee. Experimental results also validate our theoretical results and the different performance aspects of COBRA.

1 INTRODUCTION

Contextual bandit (Slivkins, 2019; Lattimore & Szepesvári, 2020) is a sequential decision-making framework in which a learner selects an arm for a given context to maximize its total reward. Unlike traditional multi-armed bandits (Auer et al., 2002; Garivier & Cappé, 2011; Agrawal & Goyal, 2012), contextual bandit (Chu et al., 2011; Krause & Ong, 2011; Valko et al., 2013; Zhou et al., 2020) use additional information, such as user profile, location, and purchase history, to make more informed and personalized decisions (Li et al., 2010). Contextual bandits have many real-life applications in personalized decision-making. Prominent examples include online recommendation systems (Li et al., 2010), online advertising (Geng et al., 2021), and clinical trials (Chow & Chang, 2006; Aziz et al., 2021), where the best recommendation depends on the context.

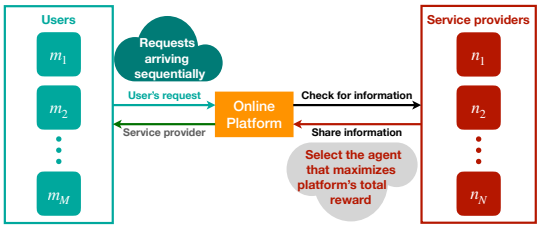


Figure 1: Example of a contextual bandit problem with strategic agents: Consider an online platform recommending service providers (agents) to users (context) who arrive sequentially. As service providers can misreport their private information to receive favorable recommendations, the platform must implement a mechanism incentivizing truthful reporting. With accurate private information, the platform can recommend the best service provider, thereby improving the overall user experience.

Many real-life applications of contextual bandits involve multiple strategic agents, from which the learner must select one to recommend based on the given context. As illustrated in Fig. 1, consider an online platform with multiple service providers (agents), where the platform must recommend one provider to a user (context). In such settings, service providers can strategically misreport their information to influence the platform's decisions and increase their utilities by increasing their chances of being recommended (Resnick & Sami, 2007; Zhang et al., 2019). For example, an online food delivery platform wants to maximize the overall user experience by selecting and presenting the best restaurant options when a user searches for a specific type of food. Since users tend to order from restaurants listed at the top of search results (Malaga, 2008), restaurants are incentivized to misrepresent their menu offerings to appear more prominently for specific food categories. This misreporting creates a significant challenge: If users consistently encounter misleading restaurant listings that fail to align with their true preferences, their experience with the platform will worsen; in the worst case, they

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

may switch to competing platforms. Similar examples also include personalized pricing (Liu et al., 2024); algorithmic trading (Zeng et al., 2024a); and firms allocating resources (Zeng et al., 2024b).

These real-life applications highlight the importance of designing contextual bandit algorithms that discourage strategic misreporting by agents. However, most existing contextual bandit algorithms do not account for strategic agent behavior, leading to suboptimal agent selection. We bridge this gap by designing a contextual bandit algorithm that accounts for potential misreporting and ensures that reporting arm features truthfully is the best response for agents. Specifically, this paper answers the following question: ***How to design an approximate Bayesian incentive-compatible contextual bandit algorithm for settings where strategic agents may misreport their true features?*** To answer this, we propose a contextual bandit algorithm, COBRA, that discourages strategic misreporting without relying on monetary incentives, while providing approximate Bayesian incentive compatibility (BIC) and sub-linear regret guarantees under appropriate assumptions. Designing a BIC contextual bandit algorithm has the following key challenges, which we address using novel techniques.

① **Detecting strategic misreporting.** Existing contextual bandit algorithms (Slivkins, 2019; Lattimore & Szepesvári, 2020) typically assume that agents truthfully report the features of their arms, which may not hold in many practical applications to gain an advantage. To ensure that agents are discouraged from strategically misreporting arm features, we introduce a novel method, the *Leave-One-Out-based Mechanism (LOOM) for identifying misreporting agents*, which draws inspiration from the Vickrey-Clarke-Groves (VCG) mechanism (Vickrey, 1961; Clarke, 1971; Groves, 1973) and uses the reported arm features of other agents to identify the misreporting agent.

② **Arm-selection under strategic misreporting.** Strategic misreporting of arm features introduces bias into the reward estimator, breaking the theoretical guarantees of existing contextual bandit algorithm (Kleine Buening et al., 2024). To address this bias in the reward estimator, we first introduce the notion of an LOOM-compatible contextual bandit algorithm. An LOOM-compatible contextual bandit algorithm (e.g., algorithms based on upper confidence bound (Abbasi-Yadkori et al., 2011; Chowdhury & Gopalan, 2017)) can integrate *LOOM as a post-processing step after each round to identify the misreporting agent* by using suitable statistical estimates. This integration ensures that the underlying LOOM-compatible contextual bandit algorithm performs under truthful reporting, while adaptively correcting for bias once LOOM detects a strategic agent.

③ **Bounding regret due to strategic misreporting.** The arm-selection strategy relies on a reward estimator that may be biased by using strategically misreported arm features. This bias can lead to the selection of suboptimal arms, resulting in additional loss from not choosing the optimal arms (i.e., increased regret). To quantify this additional regret, we derive high-probability bounds on the estimation error of LOOM's reward estimators and use them to characterize the regret of the underlying contextual bandit algorithm when combined with LOOM.

We summarize the key contributions of this paper as follows:

- **Bayesian Incentive-compatible mechanism.** In Sec. 3, we introduce LOOM, a novel mechanism for contextual bandits with strategic agents inspired by the VCG framework. Unlike VCG, LOOM discourages strategic misreporting without monetary incentives. We also show that truthful agents are not misidentified as a misreporting agent with high probability (see Theorem 1).
- **BIC contextual bandit algorithm.** We propose COBRA in Sec. 4 that uses LOOM to disincentivize agents from misreporting. We prove that COBRA achieves $\tilde{O}(d\sqrt{T})$-NE (i.e., truthfulness leads to an approximate Bayesian Nash equilibrium) and regret $\tilde{O}(d\sqrt{T})$ when agents report truthfully (see Theorem 2 for linear and Theorem 3 for non-linear reward function). Under appropriate assumptions, we prove that COBRA has regret at most $\tilde{O}(Cd\sqrt{T} + \sqrt{NT})$ under every Nash equilibrium (see Theorem 4 for linear and Theorem 5 for non-linear reward function), where N is the number of agents, d is the dimension of the context vector, T is the number of rounds, and $C > 1$ is the maximum advantage any agent holds over the coverage provided by other agents.
- **Empirical results.** In Sec. 5, our experimental results on synthetic instances with strategic agents corroborate our theoretical results and validate the different performance aspects of COBRA.

1.1 Related Work

This section focuses on the most relevant work to our setting, i.e., strategic multi-armed and contextual bandits. We discuss related topics, such as contextual bandits and strategic learning, in Sec. A.

Strategic multi-armed bandits. The strategic variant of the multi-armed bandit problem was first studied by Braverman et al. (2019), in which the selected arm shares a fraction of its reward with the learner. Within this setting, they designed an incentive-compatible mechanism. Building on this work, Yahmed et al. (2024) proposed an algorithm that rewards arms based on their reported values, achieving both incentive compatibility and sublinear regret. A related line of work by Yin et al. (2022) studied an online allocation problem that maximizes social welfare under fairness constraints in a strategic setting, assuming that valuations are unknown to the algorithm but follow an IID distribution. Their results show that when agents truthfully reveal their information, the mechanism maximizes social welfare while achieving sublinear regret relative to the offline-optimal policy. Our mechanism design follows a similar spirit but is applied to a different problem setting. The robustness of bandit learning against strategic manipulation has also been explored, assuming a bounded manipulation budget Feng et al. (2020); Dong et al. (2022). A distinct but related direction considers multi-armed bandits with replicas, where strategic agents can submit multiple copies of the same arm Esmaeili et al. (2023); Shin et al. (2022). Finally, the integration of multi-armed bandits with mechanism design for online recommendations has been studied in Kleine Buening et al. (2023).

Strategic contextual bandits. Our work is closely related to Kleine Buening et al. (2024), which considers strategic agent behavior within a linear contextual bandit framework. Their method uses past allocation history to design agent-specific estimators that identify misreporting with high probability. In contrast, our method is inspired by the VCG mechanism (Vickrey, 1961; Clarke, 1971; Groves, 1973), using the reported arm features of other agents to identify misreports and supports non-linear reward function. Recent work by Hu & Duan (2025) introduce a Bayesian contextual linear bandit framework in a similar spirit, with non-repeated interactions with agents, employing a linear programming-based approach to design an incentive-compatible mechanism. However, our work differs by explicitly considering repeated interactions with strategic agents in contextual bandits.

2 Contextual Bandits with Strategic Agents

Contextual bandits. This paper studies a contextual bandit problem with homogeneous strategic agents who aim to maximize their utility (i.e., number of pulls) by strategically misreporting their arm's feature to the learner, while the learner's goal is to select the agent for a given context that maximizes the total reward. Our problem setting differs from standard contextual bandits as the arm features can be strategically manipulated by the agents to maximize their own utility. Let $\mathcal{C}$ be the set of all contexts and $\mathcal{A}$ be the set of all arms of all agents. Let $\mathcal{N}$ be the set of all agents and $N_t \in (2, |\mathcal{N}|]$ denote the number of active agents at time t. For brevity, we use $\mathcal{X} \subset \mathbb{R}^d$ to denote the set of all context-arm feature vectors, and $x_{t,a} = \varphi(c_t, a) \in \mathcal{X}$ to represent the feature vector associated for context c_t and arm $a \in \mathcal{A}$, where $\varphi : \mathcal{C} \times \mathcal{A} \to \mathcal{X}$ is a feature map and assume that $\|x\|_2^2 \leq L,\ \forall x \in \mathcal{X}$, i.e., the context space is bounded. At the start of round t, the environment generates a context $c_t \in \mathcal{C}$ and each agent $n \in \mathcal{N}_t \subseteq \mathcal{N}$ reports an arm, denoted by $a_t^{(n)} \in \mathcal{A}_t \subset \mathcal{A}$, where $\mathcal{N}_t$ is the set of active agents in round t and $\mathcal{A}_t = \{a_t^{(n)}\}_{n \in \mathcal{N}_t}$. The learner then selects an arm $a_t \in \mathcal{A}_t$ to recommend and observes a stochastic reward, denoted by $y_t \doteq f(x_{t,a_t}) + \varepsilon_t$, where $y_t \in \mathbb{R}$, $f : \mathcal{X} \to \mathbb{R}$ is an unknown reward function, and ε_t is a R-sub-Gaussian noise. We further assume that the function is bounded i.e., $|f(x_{t,a_t})| \leq B$, for some $B < \infty$. For simplicity, the agent only reports one arm in each round so that we can use 'agent' and 'arm' interchangeably in the paper.[1]

Strategic manipulations by agents. A strategic agent can misreport the features of their arm by manipulating them such that the agent is selected more often, thereby maximizing its utility. Let $x_{t,a}^\star$ be the true arm feature vector and $x_{t,a}$ be the reported arm feature vector for context c_t and arm a. Although an agent can strategically manipulate the arm feature vector; we assume the observed reward only depends on the true arm feature vector.[2] Our goal is to design a contextual bandit algorithm with an approximately Bayesian incentive-compatible mechanism that guarantees truthful reporting (i.e., $x_{t,a} = x_{t,a}^\star$ for all $t \geq 1$ and $a \in \mathcal{A}$) as the agents' best response.

[1] Our framework is more general and can also apply to settings where agents can report multiple arms per round, e.g., sellers offering multiple variants of the same product on an online platform. We also want to highlight that all agent-related computations can be performed in parallel, as they are independent of one another. The multiple-arms-per-agent case is treated in Sec. B.1 ('Agents with multiple arms').

[2] Sellers can misrepresent product features on the platform such that it becomes a top recommendation. But it cannot change the actual physical quality and nature of the product, i.e., observed reward $y_t = f(x_{t,a_t}^\star) + \varepsilon_t$.

Bayesian Incentive-Compatible algorithm. Let σ_n denote the strategy of agent $n \in \mathcal{N}$, which is history-dependent and maps the true features of their arms to reported features. We use $\boldsymbol{\sigma}_{-n}$ to denote the strategies of all agents other than agent n, and $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \ldots, \sigma_N)$ to represent the full strategy profile of all agents. We first define what it means for an agent to be truthful.

Definition 1 (Truthful). *An agent $n \in \mathcal{N}$ is said to be truthful if the agent reports the true features of their arms to the learner in each round, i.e., $x_{t,a} = x^\star_{t,a}$ for all $t \geq 1$ and a denotes agent's arm.*

We use σ^*_n to denote the truthful strategy for the agent n and $\boldsymbol{\sigma}^* = (\sigma^*_1, \sigma^*_2, \ldots, \sigma^*_N)$ to represent the vector of the truthful strategy for all agents. Next, we formally define the utility of an agent n in our setting. Let $S_T(n) \doteq \sum_{t=1}^{T} \mathbb{1}(\text{arm } a_t \text{ belongs to agent } n)$ denote the number of times agent n is selected by the learner up to round T. Each agent's objective is to maximize the expected number of $S_T(n)$. Therefore, the utility of agent n is given by $\mathbb{E}\left[S_T(a) \mid \boldsymbol{\sigma}\right]$, where we conditioned on all agents strategies $\boldsymbol{\sigma}$. In the following, we define the notion of ε-Nash equilibrium (NE), in which no agent has more than ε incentive to deviate from the truthful reporting strategy.

Definition 2 (ε-Nash Equilibrium). *Let $\varepsilon > 0$ and $T > 0$. We say that $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \ldots, \sigma_N)$ forms a ε-Nash equilibrium if any deviating strategy $\sigma'_a (\neq \sigma_a)$ for any agent $a \in \mathcal{A}$, the following holds: $\mathbb{E}\big[S_T(a) \mid \sigma_a, \boldsymbol{\sigma}_{-a}\big] \geq \mathbb{E}\big[S_T(a) \mid \sigma'_a, \boldsymbol{\sigma}_{-a}\big] - \varepsilon$.*

We next define Bayesian Nash incentive compatibility for a contextual bandit algorithm.

Definition 3 (Bayesian Nash Incentive Compatible). *A contextual bandit algorithm is Bayesian Nash incentive compatible if truthfulness is a Bayesian Nash equilibrium, i.e., reporting the true arm features is the best response for agents.*

Performance measure. Let $a^\star_t$ denote the optimal arm (agent) for context c_t having the maximum expected reward, i.e., $a^\star_t = \text{argmax}_{a \in \mathcal{A}_t} f(x^\star_{t,a_t})$. After selecting arm a_t, the learner incurs a penalty r_t, where $r_t = f(x^\star_{t,a^\star_t}) - f(x^\star_{t,a_t})$. Our aim is to learn a sequential policy that selects an arm for a given context such that the learner's total penalty for not selecting the optimal arm (or *cumulative regret*) is as minimal as possible. However, the performance of the contextual bandit algorithm depends on the incentive-compatible mechanism for the strategic agents whose strategy profile is represented by $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_N)$. We use strategic regret as a performance measure of a sequential policy π for which the agents act according to a Nash equilibrium under policy π. Specifically, for T rounds and $\boldsymbol{\sigma} \in \text{NE}(\pi)$, the strategic regret of a policy π that selects arm a_t in the round t is

$$\mathfrak{R}_T(\pi, \boldsymbol{\sigma}) \doteq \sum_{t=1}^{T} \left(f(x^\star_{t,a^\star_t}) - f(x^\star_{t,a_t}) \right). \tag{1}$$

A policy π is a good policy if it has sub-linear regret, i.e., $\lim_{T \to \infty} \mathfrak{R}_T(\pi, \boldsymbol{\sigma})/T = 0$. This implies that, as T increases, the policy π will eventually start selecting optimal arms for the given contexts.

3 Leave-One-Out-based Mechanism (LOOM)

In the contextual bandit setting, designing a Bayesian incentive-compatible mechanism that ensures truthful reporting of arm features by strategic agents is challenging due to limited access to true contexts, the potential for strategic misreporting, noisy reward feedback, and unknown reward function parameters. These challenges naturally raise the question: *How to design a mechanism that effectively incentivizes strategic agents to report truthfully while simultaneously achieving low regret?*

To answer this, we propose a method, *Leave-One-Out-based Mechanism (LOOM) for identifying misreporting agents*, inspired by the Vickrey-Clarke-Groves (VCG) framework (Vickrey, 1961; Clarke, 1971; Groves, 1973) and uses the reported arm features of other agents to identify misreporting agents. To identify whether an agent a is misreporting (i.e., *over-reporting* arm features to increase its expected reward, such that $f(x_{t,a}) > f(x^\star_{t,a})$), LOOM uses three key components: ① a pessimistic estimate of the agent's total expected reward, derived from past data of all other agents, ② an optimistic estimate of the agent's total reward, based on observed rewards when agent a is selected, ③ a statistical test that uses these estimates to identify if agent a is over-reporting with high probability.

① Pessimistic estimate of the agent's total expected reward. Since the true reward function is unknown, LOOM estimates it using past observations (context-arm features and rewards) from

all agents except agent a, ensuring the estimator is not influenced by agent a. Next, we formally introduce the notion of a *LOOM-compatible* contextual bandit algorithm, which refers to a class of algorithms that can integrate LOOM as a post-processing step after each round to identify the over-reporting agent. Let $\mathcal{O}_t$ be the past observations from all agents at the beginning of round t and $\mathcal{O}_{t,-a}$ represent the past observations from all agents except agent a. Let f_t and $f_{t,-a}$ represent the estimate of reward function f using observations $\mathcal{O}_t$ and $\mathcal{O}_{t,-a}$, respectively, at the end of round t.

Definition 4 (**LOOM-Compatible Contextual Bandit Algorithm**). *Any contextual bandit algorithm $\mathfrak{A}$ is* LOOM-compatible *if the following holds: (i) The estimated function $f_t^{\mathfrak{A}}$ from $\mathcal{O}_t$, with probability $1-\delta$, satisfies: For any $x \in \mathcal{X}:\ |f_t^{\mathfrak{A}}(x) - f(x)| \leq h(x, \mathcal{O}_t)$ and (ii) The estimated function $f_{t,-a}^{\mathfrak{A}}$ from $\mathcal{O}_{t,-a}$, with probability $1-\delta$, satisfies: For any $x \in \mathcal{X}:\ |f_{t,-a}^{\mathfrak{A}}(x) - f(x)| \leq h(x, \delta, \mathcal{O}_{t,-a})$. Here, the value of $h(x, \cdot)$ depends on x, δ past observations ($\mathcal{O}_t$ or $\mathcal{O}_{t,-a}$), and $\mathfrak{A}$.*

Many contextual bandit algorithms like Lin-UCB (Chu et al., 2011) (as shown in Sec. 4), UCB-GLM (Li et al., 2017), IGP-UCB (Chowdhury & Gopalan, 2017), GP-TS (Chowdhury & Gopalan, 2017), Neural-UCB (Zhou et al., 2020), and Neural-TS (Zhang et al., 2021) are LOOM-compatible. Note that the assumptions required by contextual bandit algorithms must also hold in our setting, as they directly influence the performance of our proposed algorithm through $h(x, \mathcal{O}_t)$ and $h(x, \mathcal{O}_{t,-a})$. See Tab. 1 in Sec. D for different values of $h(x, \mathcal{O}_t)$.

Henceforth, we assume that the underlying contextual bandit algorithm is LOOM-compatible and omit the superscript $\mathfrak{A}$ in estimators ($f_t^{\mathfrak{A}}$ and $f_{t,-a}^{\mathfrak{A}}$) superscript for notational simplicity. For any $x \in \mathcal{X}$, if $|f_{t,-a}(x) - f(x)| \leq h(x, \mathcal{O}_{t,-a})$ holds with probability $1-\delta_{t,a}$, then $\text{LCB}_{t,-a}(x) = f_{t,-a}(x) - h(x, \mathcal{O}_{t,-a})$ is a pessimistic estimates of the expected reward for x, which also holds with probability $1-\delta_{t,a}$ (see Lemma 8 in Sec. D). We further define $\text{LCB}_{t,a}^{(x)} = \sum_{s=1,a_s=a}^{t} \text{LCB}_{t,-a}(x_{s,a_s})$ to denote the pessimistic estimates of the agent a's total expected reward. We assume that $\text{LCB}_{t,a}^{(x)}$ holds with probability at least $1-\delta_{t,a}^{x}$ (more details are provided in Sec. D).

② Optimistic estimate of the agent's total reward. Since the observed reward depends only on the true feature vector, the learner receives a noisy reward, where the noise is sub-Gaussian. Our following result provides an optimistic estimate of the agent a's total expected reward.

Lemma 1. *Let $S_t(a)$ be the number of times that agent a is selected until round t, and ε_s be R-sub-Gaussian noise in the observed reward y_s, where $1 \leq s \leq t$. Then, $\sum_{s \leq t, a_s=a} f(x_{s,a_s}^{\star}) \leq \sum_{s \leq t, a_s=a} y_s + \sqrt{2R^2 S_t(a) \log(1/\delta_{t,a}^{y})}$. holds with probability at least $1-\delta_{t,a}^{y}$.*

Proof outline. This result follows from applying Hoeffding's inequality to a sum of sub-Gaussian random variables; detailed and omitted proofs are provided in Sec. C.

③ Statistical test for finding whether the agent is over-reporting. For simplicity, consider the case where the reward function is known. In this case, we say that an agent is over-reporting if the total expected reward for reported arm features exceeds the total noiseless expected reward, i.e., $\sum_{s=1,a_s=a}^{t} f(x_{s,a_s}) > \sum_{s \leq t, a_s=a} \bar{y}_s$ for any $t \geq 1$, where $\bar{y}_s$ is the noiseless expected reward. However, since the reward function is unknown and the observed reward is noisy in practice, we assess over-reporting using optimistic and pessimistic estimates of the expected rewards. We define $\text{UCB}_{t,a}^{(y)} = \sum_{s=1,a_s=a}^{t} y_s + \sqrt{2S_t(a) \log(1/\delta_{t,a}^{y})}$ as the optimistic estimate of the sum of the agent a's expected rewards that holds with probability at least $1-\delta_{t,a}^{y}$. Therefore, an agent a over reporting the true arm features with probability at least $1-\delta_{t,a}^{x}-\delta_{t,a}^{y}$ if the following condition holds:

$$\boxed{\textbf{LOOM Condition: } \text{LCB}_{t,a}^{(x)} > \text{UCB}_{t,a}^{(y)}.} \tag{2}$$

By eliminating the agents who satisfy Eq. (2) from future rounds, the LOOM condition incentivizes agents to report truthfully. Our next result shows that when an agent a always reports truthfully, i.e., $x_{t,a} = x_{t,a}^{\star}$ for all $t \geq 1$, it does not get eliminated with high probability at least $1-\delta_{t,a}^{x}-\delta_{t,a}^{y}$.

Theorem 1. *Let agent a report truthfully and all other agents are assumed to be truthful. Then, LOOM does not eliminate agent a with high probability at least $1-\delta_{t,a}^{x}-\delta_{t,a}^{y}$.*

Proof outline. The key idea of the proof is to apply the confidence ellipsoid lemma alongside high-probability upper bounds on the noisy reward and lower bounds on the expected reward for agent a's reported arm features. Additional details are provided in the supplementary material.

The performance of LOOM depends on how well the remaining agents represent the distribution of an over-reporting agent's arm feature. This structural property is central to our setting, as it allows estimating an agent's expected reward using the past observations from other agents under the shared reward function f (an illustrative example is provided in Sec. B.1, along with discussion of LOOM, including its failure cases, such as heterogeneous agents, multiple strategic agents, and collusion, as well as its connections to existing work (Kleine Buening et al., 2024)).

Since LOOM's estimation depends on the information provided by other agents, we next define the maximum advantage that any agent can have over the remaining agents.

Definition 5 (Maximum Advantage for any Agent)**.** *For an agent $a \in \mathcal{N}$ and round $t \geq 1$, the advantage of agent a at round t is $C_{t,a} \doteq \sup_{x \in \mathcal{X}} \frac{h(x, \mathcal{O}_{t,-a})}{h(x, \mathcal{O}_t)} \geq 1$. The maximum advantage is any constant C satisfying $C \geq \max_{t \geq 1, a \in \mathcal{N}} C_{t,a} \geq 1$, which uniformly bounds this contribution across all agents and rounds.*

The term $C_{t,a}$ measures the largest multiplicative increase in confidence width caused by removing agent a's observations, and thus quantifies how much those observations contribute relative to the coverage provided by the remaining agents. Both $C_{t,a} \geq 1$ and $C \geq 1$ follow from $\mathcal{O}_{t,-a} \subseteq \mathcal{O}_t$, since removing agent a's observations can only increase the confidence width. In the linear case of Sec. 4, where $h(x, \mathcal{O}_{t,-a}) = \alpha_{t,-a} \|x\| \, Vt, -a^{-1}$ and $h(x, \mathcal{O}_t) = \alpha_t \|x\|_{V_t^{-1}}$, the definition becomes the ratio of the two design norms, $C_{t,a} = \sup_{x \neq 0} \frac{\|x\|_{V_{t,-a}^{-1}}}{\|x\|_{V_t^{-1}}}$ (see Sec. C.2.2).

4 Incentive-Compatible Contextual Bandit Algorithm: COBRA

In this section, we present our contextual bandit algorithm, COBRA, which is specifically designed to ensure strategic agents report truthfully. To present our key ideas and results, we restrict our setting to linear reward functions and later extend them to non-linear reward functions.

Linear reward function. We first consider the setting where the underlying reward function is linear, i.e., $f(x) = \theta_\star^\top x$ in which $\theta_\star \in \mathbb{R}^d$ is the unknown parameter. At the beginning of round t, the learner observes the randomly generated context $c_t \in \mathcal{C}$ and the set of reported arms $\mathcal{A}_t$. After selecting the arm a_t, the learner observes stochastic reward $y_t = \theta_\star^\top x_{t,a_t} + \varepsilon_t$, where $x_{t,a_t} = \varphi(c_t, a_t)$ and ε_t is R-sub-Gaussian. We estimate the unknown parameter $\theta_\star$ using the available observations of context-arm features and corresponding rewards at the beginning of round t, denoted by $\mathcal{O}_t \doteq \{(x_{s,a_s}, y_s)\}_{s=1}^{t-1}$, as follows: $\hat{\theta}_t \doteq V_t^{-1} \sum_{s=1}^{t-1} x_{s,a_s} y_s$, where $V_t \doteq \lambda I_d + \sum_{s=1}^{t-1} x_{s,a_s} x_{s,a_s}^\top$, I_d is the $d \times d$ identity matrix, and $\lambda > 0$ ensures the Gram matrix V_t is positive definite.

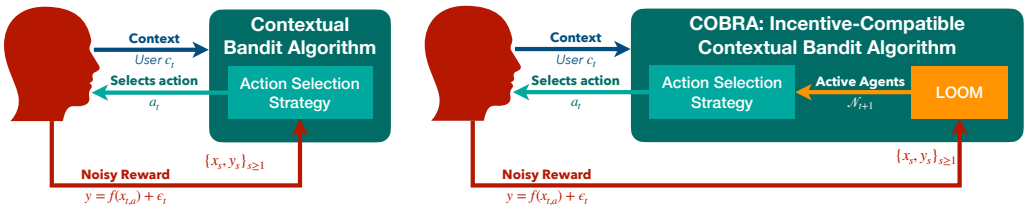
Optimistic reward estimate. In the round t, the optimistic reward estimate/ upper confidence bound (UCB) of any context-arm feature vector $x_{t,a}$ is computed as follows: $\text{UCB}_{t,a}(x_{t,a}) \doteq \hat{\theta}_t^\top x_{t,a} + \alpha_t \|x_{t,a}\|_{V_t^{-1}}$, where $\hat{\theta}_t^\top x_{t,a}$ denotes the estimated reward for $x_{t,a}$ and $\alpha_t \|x_{t,a}\|_{V_t^{-1}}$ is the confidence bonus in which $\alpha_t \doteq R \left(d \log \left(\frac{1 + tL^2/\lambda}{\delta} \right) \right)^{1/2} + \lambda^{\frac{1}{2}} S$ is a increasing function in t and the value of $\|x_{t,a}\|_{V_t^{-1}}$, weighted l_2-norm of $x_{t,a}$ with respect to V_t^{-1}), goes to zero as t increases.

COBRA Algorithm for **CO**ntextual **B**andits with St**RA**tegic Agents

Input: $\mathcal{N}_1$: set of agents before the round $t = 1$, $\delta \in (0, 1)$, $\mathcal{B}_t = \emptyset$, and $\lambda > 0$
for $t = 1, 2, \ldots$ **do**
  Observe context c_t and then receive set of arms $\mathcal{A}_t$ reported by agents in $\mathcal{N}_t$, with $\{x_{t,a}\}_{a \in \mathcal{A}_t}$
  Select an arm $a_t = \text{argmax}_{a \in \mathcal{A}_t} \text{UCB}_t(x_{t,a})$
  while $\text{UCB}_t(x_{t,a_t}) > \text{UCB}_{t,-a_t}(x_{t,a_t})$ **do**
    Set $\mathcal{B}_t = \mathcal{B}_t \cup \{a_t\}$ and select an arm $a_t = \text{argmax}_{a \in \mathcal{A}_t \setminus B_t} \text{UCB}_t(x_{t,a})$
  end while
  Play arm a_t and observe noisy reward y_t. Set $\mathcal{B}_t = \emptyset$.
  Check LOOM condition in Eq. (2) for each agent in $\mathcal{N}_t$. If it holds for any agent a, then update $\mathcal{N}_{t+1} = \mathcal{N}_t \setminus \{a\}$. If $\mathcal{N}_{t+1} = \emptyset$, stop and receive 0 reward thereafter.
end for

UCB-based algorithm. The upper confidence bound (Chu et al., 2011; Zhou et al., 2020; Li et al., 2010) is a widely used technique for addressing the exploration-exploitation trade-off in contextual bandit problems. Our UCB-based algorithm, COBRA (UCB), for linear contextual bandit problems works as follows. At the start of round t (see Fig. 2), the learner observes the context and reported arm features $x_{t,a}$, and then selects an arm $a_t = \text{argmax}_{a \in \mathcal{A}_t} \text{UCB}_{t,a}(x_{t,a})$ (Line 4). Importantly, COBRA (UCB) does not have access to the true arm features or the true reward function parameter $\theta_\star$. As a result, over-reporting by agents can lead COBRA (UCB) to make suboptimal arm selections. To address this, we introduce a validity check (Line 5) that triggers only when an agent over-reports significantly while other agents report truthfully. We incorporate LOOM (Line 9, more details on how we adapt LOOM to linear contextual bandits are provided on the next page) to identify the over-reporting agent. Eliminating agents who satisfy the LOOM condition defined in Eq. (2) from future rounds ensures agents report truthfully.

LOOM in COBRA (UCB). To check the LOOM condition defined in Eq. (2), we need to compute $\text{LCB}_{t,a}$, which requires estimating the reward function parameters using observations from all agents except agent a. To construct the aforementioned estimate, we exclude the observations from agent a, which is given as follows: $\hat{\theta}_{t,-a} = V_{t,-a}^{-1} \sum_{s=1, a_s \neq a}^{t} x_{s,a_s} y_s$, where $V_{t,-a} = \lambda I_d + \sum_{s=1, a_s \neq a}^{t} x_{s,a_s} x_{s,a_s}^\top$. We now formally define the pessimistic estimate of the total expected reward for an agent a as: $\text{LCB}_{t,a}^{(x)} = \sum_{s=1, a_s = a}^{t} \text{LCB}_{t,-a}(x_{s,a_s})$, where $\text{LCB}_{t,-a}(x_{s,a_s}) = x_{s,a_s}^\top \hat{\theta}_{t,-a} - \alpha_{t,-a} \|x_{s,a_s}\|_{V_{t,-a}^{-1}}$ for $1 \leq s \leq t$, with $\alpha_{t,-a} = R\left(d \log\left(\frac{1+(t+1-S_t(a))L^2/\lambda}{\delta}\right)\right)^{\frac{1}{2}} + \lambda^{\frac{1}{2}} S$ and $S_t(a)$ denoting the number of times agent a has been selected up to round t. Using the upper bound $\text{UCB}_{t,a}^{(y)} = \sum_{s \leq t, a_s = a} y_s + \sqrt{2 S_t(a) \log(1/\delta_{t,a}^y)}$ from Lemma 1, we can apply LOOM condition to identify whether any agent is over-reporting their arm features.



(a) Standard Contextual Bandit Algorithm (b) COBRA: Incentive Compatible Contextual Bandit Algorithm

Figure 2: COBRA integrates LOOM as a post-processing step after each interaction round to identify over-reporting agents in a LOOM-compatible contextual bandit algorithm.

Non-linear reward function. We now consider contextual bandit problems with non-linear reward functions. COBRA naturally generalizes to this setting by adopting any suitable *LOOM-compatible* contextual bandit algorithm to get optimistic reward estimates for context–arm feature vectors. These estimates are then used to select the best arm for a given context (as in Line 4), after which LOOM is applied as a post-processing step to identify over-reporting agents.

NE guarantee and regret analysis. We derive NE and regret guarantees for COBRA and establish its desirable properties, including Bayesian incentive compatibility (i.e., reporting truthfully is the best response) and a sublinear regret guarantee. We assume that the agent only over-reports their arm features so that the corresponding reward is higher, i.e., for all $x^\star \in \mathcal{X}: \ f(x) > f(x^\star)$, where x is the reported arm features for the true arm feature $x^\star$. Notably, we impose no restrictions on how agents report their arm features, aside from the no collusion assumption, which is common in VCG-type mechanisms (Vickrey, 1961; Clarke, 1971; Groves, 1973). Let $\tilde{O}$ hide the logarithmic factors and constants. Our next result shows that when arms report truthfully, COBRA approximately incentivizes truthful behavior and achieves a regret bound of at most $\tilde{O}(d\sqrt{T})$ under this approximate NE. Next, we present the results for COBRA (UCB) under linear and non-linear reward functions.

Theorem 2 (Linear)**.** *When agents report truthfully, being truthful is a* $\tilde{O}(d\sqrt{T})$*-*NE *under* COBRA*. The regret of* COBRA *under this approximate* NE *is at most* $\mathfrak{R}_T(\text{COBRA}, \boldsymbol{\sigma}^\star) = \tilde{O}(d\sqrt{T})$.

Let $\tilde{d}$ be the effective dimension associated with contextual bandit problems with non-linear reward functions. Let $\mathfrak{A}$ be a LOOM-compatible contextual bandit algorithm for which $|f_t(x) - f(x)| \leq$

$h(x, \mathcal{O}_t)$ holds with probability at least $1 - \delta$ for any $x \in \mathcal{X}$ and $\sqrt{\sum_{t=1}^{T} \left[h(x_{t,a_t}, \mathcal{O}_t)\right]^2} = \tilde{O}(\tilde{d} \log T)$. We assume that this bound holds for the algorithm $\mathfrak{A}$ used by COBRA.

Theorem 3 (Non-linear)**.** *Let $\mathfrak{A}$ be a LOOM-compatible contextual bandit algorithm used by* COBRA. *When agents report truthfully, being truthful is a $\tilde{O}(\tilde{d}\sqrt{T})$-*NE *under* COBRA. *With probability at least $1 - \delta^x_{t,a} - \delta^y_{t,a}$, the regret of* COBRA *under this approximate* NE *is* $\mathfrak{R}_T\left(\text{COBRA}(\mathfrak{A}), \boldsymbol{\sigma}^\star\right) = \tilde{O}(\tilde{d}\sqrt{T})$.

When multiple agents over-report, all COBRA estimators become biased (Lemma 2 in the Appendix) because the over-reported arm features used to estimate the reward function no longer reflect the true distribution. Our next results hold only under the conditions specified in the following assumptions.

Assumption 1. *Let x and $x^\star$ be the reported and true context-arm feature vector, respectively. Then, we assume $\forall t \geq 1, a \in \mathcal{A}_t : f(x_{t,a}) \leq \text{UCB}_t(x_{t,a})$, where $\text{UCB}_t(x) = f_t(x) + h(x, \mathcal{O}_t)$.*

The assumption states that each agent's expected true reward for the reported features is upper bounded by the optimistic reward estimate, $\text{UCB}_t(x_{t,a})$, that uses all available context-arm features to estimate $\theta_\star$. Next, we prove a strategic regret bound that holds for every NE.

Theorem 4 (Linear)**.** *If Assumption 1 hold then, with C as in Definition 5, the regret of* COBRA *is $\mathfrak{R}_T(\text{COBRA}, \boldsymbol{\sigma}) = \tilde{O}(Cd\sqrt{T} + \sqrt{NT})$ for every $\boldsymbol{\sigma} \in$ NE(COBRA). Hence,* $\max_{\sigma \in \text{NE(COBRA)}} \mathfrak{R}_T(\text{COBRA}, \boldsymbol{\sigma}) = \tilde{O}(Cd\sqrt{T} + \sqrt{NT})$.

Our next result extends the previous result to the general setting with non-linear reward functions.

Theorem 5 (Non-linear)**.** *Let $\mathfrak{A}$ be a LOOM-compatible contextual bandit algorithm used by* COBRA. *If Assumption 1 hold then, with C as in Definition 5, the regret of* COBRA *is $\mathfrak{R}_T(\text{COBRA}(\mathfrak{A}), \boldsymbol{\sigma}) = \tilde{O}(C\tilde{d}\sqrt{T} + \sqrt{NT})$ for every $\boldsymbol{\sigma} \in$ NE(COBRA($\mathfrak{A}$)). Hence,* $\max_{\sigma \in \text{NE(COBRA}(\mathfrak{A}))} \mathfrak{R}_T(\text{COBRA}(\mathfrak{A}), \boldsymbol{\sigma}) = \tilde{O}(C\tilde{d}\sqrt{T} + \sqrt{NT})$.

Outline of the proofs. The proofs of Theorem 2 and Theorem 4 depend on the LOOM mechanism to identify agents who are over-reporting. LOOM ensures that optimistic estimates are tightly bounded, thereby limiting the potential benefit from over-reporting and reinforcing truthfulness as the best response for agents. The $\sqrt{NT}$ term in Theorem 4 arises due to the strategic nature of the agents who can exploit initial noisy estimates of COBRA. The detailed proofs are provided in Sec. C. The proofs of Theorem 3 and Theorem 5 rely on the LOOM mechanism to identify over-reporting agents (via C). The remainder of the proof proceeds as before, making use of Definition 4 and Assumption 1.

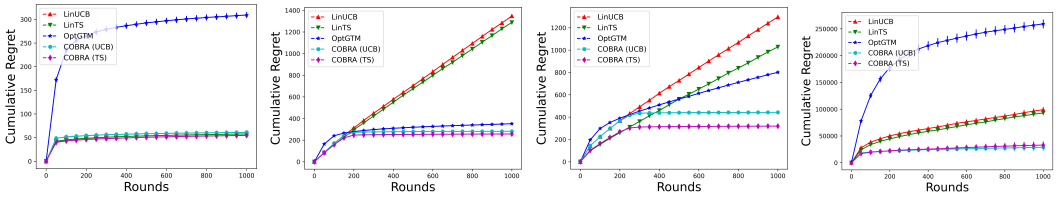
5 Experiments

In this section, we aim to corroborate our theoretical results and empirically demonstrate the performance of our proposed algorithm in different strategic contextual bandit problems. We repeat all our experiments 20 times and show the regret (as defined in Eq. (1)) with a 95% confidence interval (the vertical line on each curve shows the confidence interval). To demonstrate the different performance aspects of our proposed algorithm, we have used different synthetic problem instances.

Experiment setting. We use a d_c-dimensional space to generate the sample features of each context, where context c_t is represented by $c_t = (x_{c_t,1}, \ldots, x_{c_t,d_c})$ for $t \geq 1$. Similarly, we use a d_n-dimensional space to generate the sample of each agent's arm features, where agent $n \in \mathcal{N}$ is represented by $a_t^{(n)} = \left(x^{(n)}_{a_t,1}, \ldots, x^{(n)}_{a_t,d_n}\right)$. The value of the i-th feature $x_{c_t,i}$ (or $x^{(n)}_{a_t,i}$) is sampled uniformly at random from $(0, 2)$. To get the context-arm feature vectors for context c_t in the round t, we concatenate the context features c_t with all arm feature vectors. For context c_t and agent n, the concatenated feature vector is denoted by $x_{t,n}$, which is an d-dimensional vector with $d = d_c + d_n$. We sample a d-dimensional vector $\theta_\star$ uniformly from $(0, 2)^d$ and normalize it to have unit l_2-norm. Across all experiments, we set $\lambda = 0.01$, $R = 0.1$, $\delta = 0.05$, and $d_c = d_n$. We have provided details about different strategic manipulations in Sec. E.

Regret comparison with baselines. We compare the regret of the proposed algorithms with three baselines: Lin-UCB (Li et al., 2010), Lin-TS (Agrawal & Goyal, 2013), and OptGTM (Kleine Buening et al., 2024). For our experiments, we consider two reward functions: **Linear**, $f(x) = 5x^\top \theta_\star$, and **Square**, $f(x) = 10(x^\top \theta_\star)^2$. We use 1000 contexts, 5 agents, and $d_c = d_n = 5$, resulting in a

context-arm feature dimension of $d = 10$. We evaluate four problem instances with the same setup, except using two different types of strategic over-reporting: **Agent type I:** Feature adaptation with a learning rate of $\eta = 0.05$. **Agent type II:** Systematic over-reporting, where $\hat{\Delta}_x \sim \mathbb{N}(\Delta_x^\star, \sigma_\Delta^2)$. Here, $\mathbb{N}$ denotes a normal distribution, $\Delta_x^\star$ is the optimal scaling factor such that $f((1 + \Delta_x^\star)x^\star)$ gives the highest reward among all arms, and σ_Δ represents the standard deviation. We assume that only one agent over-reports, and the maximum perturbation in each round is bounded by $\Delta_{\max} = 1.0$.

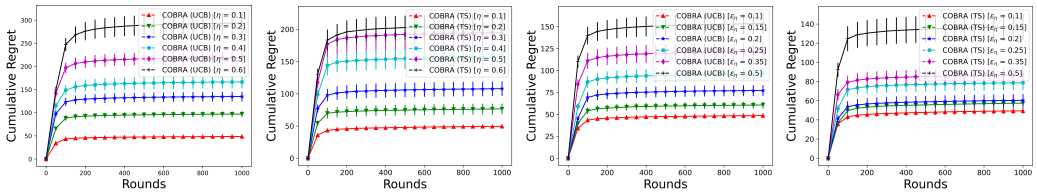


(a) Linear (Truthful setting) (b) Linear (Agent type I) (c) Linear (Agent type II) (d) Square (Agent type II)

Figure 3: Comparing cumulative regret of COBRA with baselines using different problem instances.

In Fig. 3a, all agents report truthfully under the **Linear** reward function. Even in this setting, our algorithm outperforms the state-of-the-art OptGTM and matches the performance of standard contextual bandit algorithms (LinUCB and LinTS). As expected, our proposed algorithm COBRA, based on UCB and TS variants of contextual bandits, outperforms all baselines across different problem instances with **Linear** and **Square** reward functions (Fig. 3b-3d). For the **Square** reward function, we estimate it using kernel regression with a polynomial kernel of degree 2. We observe that the TS-based variants of COBRA consistently outperform their UCB-based counterparts.

Regret of COBRA vs. different types of strategic manipulations. To simulate the strategic manipulations, we define the over-report feature vector as $x = (1 + a)x^\star$. Since all features and parameters ($\theta_\star$) are positive, scaling true feature vector $x^\star$ by a factor of $(1+a)$ ensures over-reporting. To control the over-reporting, we uniformly sample a from $(\eta, \eta + \varepsilon_\eta)$, where changing η or ε_η leads to different types of manipulations. Specifically, an increased η or ε_η implies increasing the misreporting amount. As shown in Fig. 4a-4d, the regret bound of our COBRA UCB- and TS-based algorithms increases as noise levels increase. However, in instances with higher manipulation, the COBRA TS-based algorithm consistently outperforms the COBRA UCB-based algorithm. Additional experimental results and ablations are provided in Sec. E.



(a) Varying η (UCB) (b) Varying η (TS) (c) Varying ε_η (UCB) (d) Varying ε_η (TS)

Figure 4: Cumulative regret of COBRA vs. different types of strategic manipulations.

Due to space constraints, additional experiments on regret versus number of agents, context-arm feature dimension, two strategic agents, and nonlinear reward functions are provided in Sec. E.

6 Conclusion

This paper addresses a contextual bandit problem involving strategic agents who may misreport arm features to increase their own utility. To tackle this, we propose LOOM, a mechanism that identifies over-reporting agents by leveraging the reported arm features from other agents. Building on LOOM, we introduce an algorithm, COBRA, for contextual bandit problems with strategic agents. COBRA disincentivizes strategic behavior without relying on monetary incentives, while ensuring Bayesian incentive compatibility and achieving a sub-linear regret guarantee. Our experimental results across different problem instances further demonstrate the performance advantages of our algorithm. A few promising directions for future work include incorporating fairness constraints into the arm selection process and handling more complex forms of strategic behavior within this setting.

AI USE STATEMENT

In this work, we used generative AI tools solely to assist with writing and language refinement, specifically, improving the clarity, grammar, phrasing, and readability of author-drafted text. All technical content, including the problem formulation, methodology, theoretical results and proofs, experimental design, and analysis, was conceived, developed, and written by the authors. AI tools were not used to generate research ideas, derive or verify proofs, produce experimental results, write or debug code, conduct the literature review, or create any figures, tables, or data. Any such tasks requiring disclosure under the policy are therefore not applicable to this work.

We have reviewed all AI-assisted text. Every passage edited or rephrased with AI assistance was checked by the authors for factual accuracy, correct representation of our claims, and consistency with the underlying technical content, and was revised as needed. We take full responsibility for the final content of this work, including all text, claims, and artifacts.

ETHICS STATEMENT

This work is theoretical, focusing on the design and analysis of algorithms. The proposed methods do not directly involve human subjects, personal data, or real-world deployments. While the framework could potentially be applied in systems that interact with users, we emphasize that ethical considerations, such as fairness, privacy, and informed consent, must be addressed in practical deployments. Our primary goal is to advance the theoretical understanding of Bayesian incentive-compatible contextual bandit algorithms, and we do not anticipate any negative societal impacts.

REPRODUCIBILITY STATEMENT

This paper presents theoretical results, including formal proofs of Bayesian incentive compatibility and regret upper bounds. All assumptions, definitions, and derivations are stated explicitly in the main text (see Sec. 4 and the Appendix). The details of our experimental setup are provided in Sec. 5 and the Appendix. Additionally, the code used in our experiments has been included in the supplementary material, enabling full reproduction of the results reported in this paper.

REFERENCES

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved Algorithms for Linear Stochastic Bandits. In *Proc. NeurIPS*, pp. 2312–2320, 2011.

Shipra Agrawal and Navin Goyal. Analysis of Thompson Sampling for the Multi-armed Bandit Problem. In *Proc. COLT*, pp. 39.1–39.26, 2012.

Shipra Agrawal and Navin Goyal. Thompson Sampling for Contextual Bandits with Linear Payoffs. In *Proc. ICML*, pp. 127–135, 2013.

Noga Alon, Nicolo Cesa-Bianchi, Ofer Dekel, and Tomer Koren. Online Learning with Feedback Graphs: Beyond Bandits. In *Proc. COLT*, pp. 23–35, 2015.

Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time Analysis of the Multiarmed Bandit Problem. *Machine Learning*, pp. 235–256, 2002.

Maryam Aziz, Emilie Kaufmann, and Marie-Karelle Riviere. On multi-armed bandit designs for dose-finding trials. *Journal of Machine Learning Research*, pp. 1–38, 2021.

Mark Braverman, Jieming Mao, Jon Schneider, and S Matthew Weinberg. Multi-armed bandit problems with strategic arms. In *Conference on Learning Theory*, pp. 383–416. PMLR, 2019.

Trenton Chang, Lindsay Warrenburg, Sae-Hwan Park, Ravi Parikh, Maggie Makar, and Jenna Wiens. Who's gaming the system? a causally-motivated approach for detecting strategic adaptation. *Advances in Neural Information Processing Systems*, 37:42311–42348, 2024.

Olivier Chapelle and Lihong Li. An Empirical Evaluation of Thompson Sampling. In *Proc. NeurIPS*, pp. 2249–2257, 2011.

Shein-Chung Chow and Mark Chang. *Adaptive design methods in clinical trials*. Chapman and Hall/CRC, 2006.

Sayak Ray Chowdhury and Aditya Gopalan. On Kernelized Multi-armed Bandits. In *Proc. ICML*, pp. 844–853, 2017.

Wei Chu, Lihong Li, Lev Reyzin, and Robert E Schapire. Contextual Bandits with Linear Payoff Functions. In *Proc. AISTATS*, pp. 208–214, 2011.

Edward H Clarke. Multipart pricing of public goods. *Public choice*, pp. 17–33, 1971.

Jing Dong, Ke Li, Shuai Li, and Baoxiang Wang. Combinatorial bandits under strategic manipulations. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, pp. 219–229, 2022.

Jinshuo Dong, Aaron Roth, Zachary Schutzman, Bo Waggoner, and Zhiwei Steven Wu. Strategic classification from revealed preferences. In *Proc. EC*, pp. 55–70, 2018.

Seyed Esmaeili, MohammadTaghi Hajiaghayi, and Suho Shin. Replication-proof bandit mechanism design. *arXiv:2312.16896*, 2023.

Zhe Feng, David Parkes, and Haifeng Xu. The intrinsic robustness of stochastic bandits to strategic manipulation. In *Proc. ICML*, pp. 3092–3101. PMLR, 2020.

Sarah Filippi, Olivier Cappe, Aurélien Garivier, and Csaba Szepesvári. Parametric Bandits: The Generalized Linear Case. In *Proc. NeurIPS*, pp. 586–594, 2010.

Rupert Freeman, David Pennock, Chara Podimata, and Jennifer Wortman Vaughan. No-regret and incentive-compatible online learning. In *Proc. ICML*, pp. 3270–3279. PMLR, 2020.

Aurélien Garivier and Olivier Cappé. The KL-UCB Algorithm for Bounded Stochastic Bandits and Beyond. In *Proc. COLT*, pp. 359–376, 2011.

Nicolas Gast, Stratis Ioannidis, Patrick Loiseau, and Benjamin Roussillon. Linear regression from strategic data sources. *ACM Transactions on Economics and Computation (TEAC)*, pp. 1–24, 2020.

Tong Geng, Xiliang Lin, Harikesh S Nair, Jun Hao, Bin Xiang, and Shurui Fan. Comparison lift: Bandit-based experimentation system for online advertising. In *Proc. AAAI*, pp. 15117–15126, 2021.

Theodore Groves. Incentives in teams. *Econometrica: Journal of the Econometric Society*, pp. 617–631, 1973.

Moritz Hardt, Nimrod Megiddo, Christos Papadimitriou, and Mary Wootters. Strategic classification. In *Proceedings of the 2016 ACM conference on innovations in theoretical computer science*, pp. 111–122, 2016.

Keegan Harris, Dung Daniel T Ngo, Logan Stapleton, Hoda Heidari, and Steven Wu. Strategic instrumental variable regression: Recovering causal relationships from strategic responses. In *Proc. ICML*, pp. 8502–8522. PMLR, 2022.

Keegan Harris, Chara Podimata, and Steven Z Wu. Strategic apple tasting. *Proc. NeurIPS*, 36: 79918–79945, 2023.

Yiting Hu and Lingjie Duan. Truthful mechanisms for linear bandit games with private contexts, 2025.

Kwang-Sung Jun, Aniruddha Bhargava, Robert Nowak, and Rebecca Willett. Scalable Generalized Linear Bandits: Online Computation and Hashing. In *Proc. NeurIPS*, pp. 99–109, 2017.

Thomas Kleine Buening, Aadirupa Saha, Christos Dimitrakakis, and Haifeng Xu. Bandits meet mechanism design to combat clickbait in online recommendation. *arXiv e-prints*, pp. arXiv–2311, 2023.

Thomas Kleine Buening, Aadirupa Saha, Christos Dimitrakakis, and Haifeng Xu. Strategic linear contextual bandits. *Advances in Neural Information Processing Systems*, 37:116638–116675, 2024.

Andreas Krause and Cheng S Ong. Contextual Gaussian Process Bandit Optimization. In *Proc. NeurIPS*, pp. 2447–2455, 2011.

Tor Lattimore and Csaba Szepesvári. *Bandit Algorithms*. Cambridge University Press, 2020.

Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A Contextual-Bandit Approach to Personalized News Article Recommendation. In *Proc. WWW*, pp. 661–670, 2010.

Lihong Li, Yu Lu, and Dengyong Zhou. Provably Optimal Algorithms for Generalized Linear Contextual Bandits. In *Proc. ICML*, pp. 2071–2080, 2017.

Pangpang Liu, Zhuoran Yang, Zhaoran Wang, and Will Wei Sun. Contextual dynamic pricing with strategic buyers. *Journal of the American Statistical Association*, pp. 1–13, 2024.

Yang Liu and Yiling Chen. A bandit framework for strategic regression. *Proc. NeurIPS*, 29, 2016.

Ross A Malaga. Worst practices in search engine optimization. *Communications of the ACM*, 51(12): 147–150, 2008.

Paul Resnick and Rahul Sami. The influence limiter: provably manipulation-resistant recommender systems. In *Proceedings of the 2007 ACM conference on Recommender systems*, pp. 25–32, 2007.

Suho Shin, Seungjoon Lee, and Jungseul Ok. Multi-armed bandit algorithm against strategic replication. In *Proc. AISTATS*, pp. 403–431, 2022.

Aleksandrs Slivkins. Introduction to Multi-Armed Bandits. *Foundations and Trends® in Machine Learning*, 2019.

Ravi Sundaram, Anil Vullikanti, Haifeng Xu, and Fan Yao. Pac-learning for strategic classification. *Journal of Machine Learning Research*, 24(192):1–38, 2023.

Michal Valko, Nathan Korda, Rémi Munos, Ilias Flaounas, and Nello Cristianini. Finite-time Analysis of Kernelised Contextual Bandits. In *Proc. UAI*, pp. 654–663, 2013.

Arun Verma and Manjesh K Hanawal. Stochastic Multi-Armed Bandits with Control Variates. In *Proc. NeurIPS*, pp. 27592–27603, 2021.

Arun Verma, Zhongxiang Dai, Yao Shu, and Bryan Kian Hsiang Low. Exploiting correlated auxiliary feedback in parameterized bandits. In *Proc. NeurIPS*, 2023.

Arun Verma, Zhongxiang Dai, Xiaoqiang Lin, Patrick Jaillet, and Bryan Kian Hsiang Low. Neural dueling bandits: Preference-based optimization with human feedback. In *Proc. ICLR*, 2025.

William Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of finance*, pp. 8–37, 1961.

Yifan Wu, András György, and Csaba Szepesvári. Online Learning with Gaussian Payoffs and Side Observations. In *Proc. NeurIPS*, pp. 1360–1368, 2015.

Ahmed Ben Yahmed, Clément Calauzènes, and Vianney Perchet. Strategic multi-armed bandit problems under debt-free reporting. In *Proc. NeurIPS*, 2024.

Steven Yin, Shipra Agrawal, and Assaf Zeevi. Online allocation and learning in the presence of strategic agents. *Proc. NeurIPS*, 35:6333–6344, 2022.

Sihan Zeng, Sujay Bhatt, Alec Koppel, and Sumitra Ganesh. Partially observable contextual bandits with linear payoffs. *arXiv:2409.11521*, 2024a.

Sihan Zeng, Sujay Bhatt, Eleonora Kreacic, Parisa Hassanzadeh, Alec Koppel, and Sumitra Ganesh. Learning payment-free resource allocation mechanisms. In *2024 Winter Simulation Conference (WSC)*, pp. 2667–2678. IEEE, 2024b.

Hanrui Zhang and Vincent Conitzer. Incentive-aware pac learning. In *Proc. AAAI*, pp. 5797–5804, 2021.

Weitong Zhang, Dongruo Zhou, Lihong Li, and Quanquan Gu. Neural Thompson sampling. In *Proc. ICLR*, 2021.

Yubao Zhang, Jidong Xiao, Shuai Hao, Haining Wang, Sencun Zhu, and Sushil Jajodia. Understanding the manipulation on recommender systems through web injection. *IEEE Transactions on Information Forensics and Security*, 15:3807–3818, 2019.

Dongruo Zhou, Lihong Li, and Quanquan Gu. Neural contextual bandits with UCB-based exploration. In *Proc. ICML*, pp. 11492–11502, 2020.

A Additional related work

Contextual bandits. Contextual bandits (Slivkins, 2019; Lattimore & Szepesvári, 2020) have many real-life applications, such as online recommendations, advertising, web search, and e-commerce. In this framework, a learner selects an arm and receives a reward for that choice. Given the potentially large or infinite set of arms, the mean reward for each arm is typically modeled as an unknown function, which may be linear (Li et al., 2010; Chu et al., 2011; Abbasi-Yadkori et al., 2011; Agrawal & Goyal, 2013), generalized linear model (GLM) (Filippi et al., 2010; Li et al., 2017; Jun et al., 2017; Verma et al., 2025), or non-linear (Valko et al., 2013; Chowdhury & Gopalan, 2017; Zhou et al., 2020; Zhang et al., 2021). The learner's objective is to identify the optimal action as efficiently as possible, which depends on how tightly the confidence bounds for the reward-function mapping actions to rewards are defined. Several works have explored various sources of information and side observations to enhance the learning process (Li et al., 2010; Agrawal & Goyal, 2013; Alon et al., 2015; Wu et al., 2015; Li et al., 2017; Verma & Hanawal, 2021; Verma et al., 2023).

Strategic learning. There are several works on strategic learning (Liu & Chen, 2016; Freeman et al., 2020; Gast et al., 2020; Zhang & Conitzer, 2021; Harris et al., 2022; 2023) and strategic classification (Hardt et al., 2016; Dong et al., 2018; Sundaram et al., 2023). The strategic classification problem was first introduced in Hardt et al. (2016). The authors considered a sequential game between a decision-maker selecting a classifier, and a strategic agent who responds by modifying their features. Chang et al. (2024) address the problem of identifying agents who exhibit the highest degree of strategic manipulation in their inputs, given a dataset of agents and their observed model inputs in an offline setting. Our work aligns with this research direction, as it explores the interaction between a strategic agent and a learning algorithm. However, unlike prior studies where agents interact with the learner only once to achieve a desired outcome, our setting involves repeated interactions, forming a repeated game without monetary transactions. Our main contribution is the development of an incentive-compatible mechanism designed to handle repeated interactions with strategic agents, specifically tailored for contextual bandit problems.

B Discussion about LOOM and Assumption 1

B.1 LOOM-related Discussion

Connection between LOOM and VCG mechanism. The connection we intend is *structural*, rather than monetary, and we state it precisely to avoid over-claiming. In VCG (Vickrey, 1961; Clarke, 1971; Groves, 1973), agent a is evaluated against the outcome computed *without* a, preventing a from influencing the quantity used to determine its payment. LOOM adopts this same agent-excluding construction: it evaluates agent a against $\mathrm{LCB}_{t,-a}$, which is constructed from $\mathcal{O}_{t,-a}$, the observations of all agents except a, for the same reason (see Sec. B.2). Beyond this exclusion principle, however, the mechanisms differ. LOOM involves no monetary transfers or externality pricing; instead, it creates incentives through *elimination*, and its leave-one-out quantity is a learned statistical estimate rather than an exactly computed welfare measure. Thus, LOOM is best viewed as a leave-one-out, externality-*style* test that adopts VCG's exclusion principle without its payment mechanism.

Homogeneous setting. This work considers a homogeneous multi-agent setting where all agents share a common reward function f, and each agent's arm feature space is spanned by those of the remaining agents. Our proposed LOOM mechanism leverages the reports and rewards of other agents to construct a baseline for detecting misreports. In a heterogeneous setting, where agents may have different reward functions or non-overlapping arm feature space, the pessimistic estimate $LCB_{t,-a}$ derived from other agents may not accurately bound the expected reward of agent a. Consequently, LOOM may become less effective or unable to reliably detect strategic misreporting in such scenarios. In Sec. E, we conduct additional experimental studies to illustrate these issues and also see Sec. B.1 for a motivating practical example of our proposed mechanism.

Impact of arm feature distribution on LOOM identification. The performance of LOOM depends on how well the remaining agents represent the distribution of an over-reporting agent's arm feature, as captured by the term $h(x, \mathcal{O}_{t,-a})$ in $\mathrm{LCB}_{t,-a}(x)$. If the over-reporting agent's distribution is well represented, $h(x, \mathcal{O}_{t,-a})$ is going to be small, making the over-reporting agent easier to identify; otherwise, a large $h(x, \mathcal{O}_{t,-a})$ makes identification harder. This structural property is central to our setting, as it allows estimating an agent's expected reward using the past observations from other

agents under the shared reward function f (an illustrative example is provided in Sec. B.1, along with discussion of LOOM, including its failure cases, such as heterogeneous agents, multiple strategic agents, and collusion, as well as its connections to existing work (Kleine Buening et al., 2024)).

Example showing impact of arm feature distribution on LOOM. To illustrate how the arm feature distribution of agents plays out in our setting, consider the example of an online e-commerce platform recommending sellers. When multiple sellers offer similar products, such as round-neck T-shirts priced between \$5 and \$15, their corresponding arms (i.e., T-shirts) will have similar feature vectors. In contrast, sellers offering distinct products, such as one selling T-shirts and another selling smartphones, will likely have arms with very different feature representations. Suppose a seller offers a unique product, for example, being the only seller of Apple products on the online e-commerce platform. There is no incentive to misreport their features, as no competitors exist. Thus, misreporting becomes strategically beneficial only when sellers offer similar products, in which case their arm features are drawn from similar distributions, allowing LOOM to identify the misreporting agent.

Agent under-reporting. Agents have no incentive to under-report, as it reduces their likelihood of being selected by the algorithm. Instead, they are more inclined to over-report to increase their chances of being selected. However, as noted in (Kleine Buening et al., 2024), there are some cases where under-reporting may yield a small gain. Our proposed method, LOOM, is designed to detect over-reporting and does not capture under-reporting. Developing a mechanism that can reliably detect both under-reporting and over-reporting remains an open problem.

Failure Examples of LOOM. We consider two examples in which our mechanism, LOOM, is ineffective at detecting the misreport.

Example 1 (Unidentifiable arm direction). Let $\theta_\star = (1, 1)$ and consider two arms with true contexts $x^\star_{t,1} = (1/4,\ 0)$ and $x^\star_{t,2} = (0,\ v_t)$, where $v_t \sim \mathrm{Uniform}(0, 1)$ i.i.d. Samples from arm 1 follow a Gaussian with mean $1/4$, while arm 2 is never selected and thus yields no observations. Suppose agent 2 persistently misreports $x_{t,2} = (0, 1)$. Since the learner receives no samples from arm 2, it cannot detect the misreport and incorrectly estimates its expected reward as 1. Consequently, the learner plays arm 2 in every round, whereas under the true contexts it would prefer arm 1 whenever $v_t < 1/4$. This misreport goes undetected and induces linear regret. The same failure mode arises if the two arms' feature vectors are swapped with any fixed positive probability at each round.

Example 2 (Collinear arm contexts). Let $\theta_\star = 1$, $x^\star_{t,1} = 1/2$, and $x^\star_{t,2} = 1/4$ for all t. Suppose agent 2 misreports $x_{t,2} = 1$. The learner always selects arm 2 because it achieves the highest UCB across the reported contexts. Since arm 1 is never played, the learner collects no samples from it and cannot detect the misreport.

What the failure examples cost quantitatively. The above two examples characterize the regime in which the advantage constant C in Definition 5 becomes large. In Example 2 ($d = 1$, agent 2 misreports and is selected in every round), we have $V_{t,-2} = \lambda$ and $V_t = \lambda + t - 1$, so

$$C_{t,2} = \sqrt{\frac{\|x\|^2_{V^{-1}_{t,-2}}}{\|x\|^2_{V^{-1}_t}}} = \sqrt{\frac{\lambda + t - 1}{\lambda}} = \Theta(\sqrt{t}),$$

implying that any valid C is $\Theta(\sqrt{T})$ and the bound in Theorem 4 degrades to the vacuous $\tilde{O}(dT)$. This illustrates why we keep C explicit in the theorem statement: the guarantee degrades continuously with the severity of the coverage gap and can become vacuous when peer coverage is insufficient for detection. More generally, an agent that concentrates its over-reporting in directions never explored by the remaining agents is undetectable by any peer-based test, including LOOM, because there is no independent peer signal against which to compare its report. The incentive to exploit such a blind spot may nevertheless be limited: over-reporting is beneficial only if it can displace a competitor, whereas a direction that no other agent explores provides no competitor to displace. This does not eliminate the limitation, however, since a deviator may deliberately target weakly covered directions while still competing for selection; such coverage-aware deviations remain outside our guarantees.

An agent that concentrates its over-reporting in directions never explored by the remaining agents is undetectable by any peer-based test, including LOOM: without peer coverage, there is no independent signal against which to test the report. We state this limitation explicitly. It is illustrated by Examples 1 and 2 and is captured by a large confidence width $h(x, \mathcal{O}_{t,-a})$. Two observations qualify this limitation. First, the loss of coverage is continuous and quantifiable. The same quantity that captures

this blind spot is the advantage constant C in Definition 5, which appears explicitly in Theorem 4 and Theorem 5. As peer coverage deteriorates, C increases and the corresponding guarantee degrades smoothly; in Example 2, $C = \Theta(\sqrt{T})$, causing the bound to degrade to the trivial $\tilde{O}(dT)$. Second, the incentive to exploit a weakly covered direction depends on competition. Over-reporting is beneficial when it changes the agent's selection relative to its competitors. Thus, an agent operating in a direction that no other agent covers may have limited incentive to misreport, since there is no competitor to displace. The above example about the seller illustrates this distinction: a sole supplier of a product category has no rival against which misreporting can improve its selection, whereas sellers of similar products, with similar feature directions, provide precisely the peer coverage that LOOM exploits. This tension does not eliminate the coverage gap. An agent may deliberately target weakly covered directions while still competing for selection, which remains outside our guarantees. Designing a mechanism with power against such coverage-aware deviations is an open problem.

Alternative to complete removal of over-reporting agent. When an agent over-reports its arm features and is completely eliminated, such an event would be extremely rare in practice. If every agent were to persistently misreport, any mechanism prioritizing user experience would eventually stop recommending such agents, thereby driving the platform's reward to zero. This outcome is not a flaw but rather a *safeguard preserving user trust*: continuing to recommend only strategic, untrustworthy agents would ultimately diminish both reward and user engagement. An alternative mechanism is that when an agent systematically over-reports, COBRA flags and temporarily removes it. This mechanism discourages agents from gaming the system while restoring recommendation quality and benefiting users and the platform. However, deriving theoretical guarantees for such a mechanism may be non-trivial.

Agents with multiple arms. Our results generalize to the setting where each agent controls multiple arms. LOOM maintains individual records of each agent's reported arm feature vectors and reward history for their respective arms, applying LOOM's statistical test independently to each agent is possible. This allows LOOM to identify over-reporting agents whose optimistic/pessimistic reward gaps exceed the threshold, even if the agent over-reports its different arms in distinct ways. As long as Assumption 1 holds for each active agent, the regret and approximate equilibrium guarantees continue to apply. Specifically, the regret bound remains $\tilde{O}(d\sqrt{T} + \sqrt{NT})$, where N denotes the number of agents. For a very large number of agents (N), it may increase computational complexity and affect regret bounds due to the $\sqrt{N}$ term. Note that we can perform all agent-related computations in parallel, as they are independent of each other.

Sub-optimal agent. We highlight that, for agents with suboptimal arms, truthful reporting and over-reporting may lead to similar outcomes: either being ignored by the learner or being eliminated. To address this challenge, a promising direction for future work is to incorporate fairness constraints into the arm selection, thereby ensuring that even suboptimal agents have a chance of being selected.

Colluding agents. In the paper, we assume that agents choose their reporting strategies independently. For each agent $n \in \mathcal{N}$, deviations are restricted to unilateral best responses to $\boldsymbol{\sigma}_{-n}$, so the only deviations considered are of the form $(\sigma'_n, \boldsymbol{\sigma}_{-n})$. In particular, no coalition $\mathcal{K} \subseteq \mathcal{N}$ with $|\mathcal{K}| \geq 2$ jointly deviates. This assumption is essential for LOOM's peer-based verification. If agents can coordinate, the reference set $\mathcal{O}_{t,-a}$ for a deviating agent a may contain data systematically biased by colluding agents, thereby weakening or preventing detection. This limitation is inherent to peer-based verification; see Sec. F (Question 2) and Sec. B.2 for a discussion of additional conditions, such as requiring a minimum fraction of truthful agents, that can restore guarantees under coordinated deviations.

LOOM-compatible contextual bandit algorithm. LOOM-compatibility is a property of the *estimator conditional on the observation set it receives*; it does not require concentration to hold under arbitrary corruption of that set. This distinction is important. When analyzing a deviation by agent a, the remaining agents are truthful by hypothesis. Hence, $\mathcal{O}_{t,-a}$ contains only truthful observations, and part (ii) applies directly: $f_{t,-a}$ concentrates without any assumption on the deviating agent's reports. This is precisely the setting used in Theorem 1 and explains why LOOM evaluates agent a against a statistic that agent a cannot influence. In contrast, part (i) is applied to $\mathcal{O}_t$, which includes agent a's reports. We therefore do not assume that the concentration guarantee in part (i) continues to hold under over-reporting. Lemma 2 quantifies the resulting effect directly: over-reporting introduces the bias $\big(\lambda I + \sum_s x_{s,a_s} x_{s,a_s}^\top\big)^{-1}\big(\sum_s x_{s,a_s}\eta_s\big)$ into the ridge estimator, so $\theta_\star$

may lie outside the resulting confidence ellipsoid. For multiple deviators, the additional condition required to recover part (i) is isolated in Assumption 1 and invoked only in Theorem 4 and Theorem 5.

B.2 Discussion about Assumption 1

Two clarifications about the status of this assumption. First, we do *not* assume it away: Lemma 3 shows that Assumption 1 *together with* Step 5 of COBRA has the same theoretical consequence (Eq. (4)) as replacing the arm-selection rule of Step 4 by the leave-one-out rule $a_t = \mathrm{argmax}_{a\in\mathcal{A}_t} \mathrm{UCB}_{t,-a}(x_{t,a})$, which delivers the identical inequality (Eq. (5)) with *no* assumption at all. Assumption 1 is therefore realizable by an algorithm; we do not adopt the leave-one-out rule as the default only because leave-one-out bounds are looser and underperform the standard rule when all agents are truthful. Lemma 3 is stated for a single over-reporting agent. Second, we state the remaining scope restriction directly: when several agents deviate simultaneously, $\mathcal{O}_{t,-a}$ still contains the other deviators' manipulated data even though they are not colluding, so the leave-one-out estimator is no longer clean. This is Case 3 (given below), and it is exactly the regime in which Assumption 1 is doing work.

We believe that addressing strategic behavior in a contextual bandit setting without relying on monetary incentives is a challenging and underexplored problem. There is limited prior work in the literature on this topic, despite its many practical applications, e.g., online platforms where sellers may attempt to manipulate the contextual information of their products to gain an advantage. Our key contribution is the development of an approximately Bayesian incentive-compatible property inspired by the VCG mechanism. However, when an agent overreports arm features, all estimators used by LOOM become biased due to the over-reported arm feature vectors as inputs. It happens because the misreported features distort the overall feature distribution, creating mismatches between features and their corresponding rewards, which in turn induces bias in the estimators.

In contextual bandits, estimators (such as those for the reward function parameters) rely on the assumption that observed features and rewards are generated according to an honest, stationary process. When agents systematically over-report by deliberately inflating feature values to increase their selection probability, the samples collected by the algorithm are corrupted: the feature vectors in the data do not match the ground truth. Our next result demonstrates that over-reported arm features increase the bias in estimators used by LOOM that take them as input.

Lemma 2 (Biased-ness due over-reporting.)**.** *Using over-reported arm features increases the bias in all estimators used by LOOM that take these arm features as input.*

Proof. Without loss of generality, consider a linear reward function with an unknown true parameter vector $\theta_\star \in \mathbb{R}^d$, where $d \geq 1$ is the dimension of the context-arm feature vector. At each round t, the true feature vector associated with agent a is denoted by $x^\star_{t,a}$. However, the agent may strategically misreport their features as $x_{t,a}$ in an attempt to appear more favorable, i.e., to give the impression of a higher expected reward, satisfying $\theta_\star^\top x_{t,a} \geq \theta_\star^\top x^\star_{t,a}$ Let $\eta_s = \theta_\star^\top x_{s,a_s} - \theta_\star^\top x^\star_{s,a_s} \geq 0$ denote the difference in reward between the misreported features and the true features, where a_s is the agent selected in round s. Note that $\eta_s = 0$ when the agent reports truthfully. The ridge regression estimator based on the observed data $\mathcal{H}_t = \{(x_{s,a_s}, y_s)\}_{s<t}$ is given by

$$\hat{\theta}_t = \left(\lambda I + \sum_{s<t} x_{s,a_s} x_{s,a_s}^\top\right)^{-1} \left(\sum_{s<t} x_{s,a_s} y_s\right),$$

where $\lambda > 0$ is the regularization parameter that ensures the matrix $\sum_{s<t} x_{s,a_s} x_{s,a_s}^\top$ is invertible.

Since $y_s = \theta_\star^\top x^\star_{s,a_s} + \varepsilon_s$, which depends only on the true context-arm feature vector, we can rewrite it as $y_s = \theta_\star^\top x_{s,a_s} - \eta_s + \varepsilon_s$. Substituting this into the ridge regression estimator, we obtain:

$$\hat{\theta}_t = \left(\lambda I + \sum_{s<t} x_{s,a_s} x_{s,a_s}^\top\right)^{-1} \left(\sum_{s<t} x_{s,a_s} \left(\theta_\star^\top x_{s,a_s} - \eta_s + \varepsilon_s\right)\right).$$

Let $y'_s = \theta_\star^\top x_{s,a_s} + \varepsilon_s$ denote the noisy reward for the (possibly misreported) context-arm feature vector x_{s,a_s}, generated by the true reward function. Using this notation, we can rewrite the estimator

as:

$$\hat{\theta}_t = \left(\lambda I + \sum_{s<t} x_{s,a_s} x_{s,a_s}^\top\right)^{-1} \left(\sum_{s<t} x_{s,a_s}(y'_s - \eta_s)\right).$$

Expanding the expression, we obtain:

$$\hat{\theta}_t = \left(\lambda I + \sum_{s<t} x_{s,a_s} x_{s,a_s}^\top\right)^{-1} \left(\sum_{s<t} x_{s,a_s} y'_s\right) - \left(\lambda I + \sum_{s<t} x_{s,a_s} x_{s,a_s}^\top\right)^{-1} \left(\sum_{s<t} x_{s,a_s}\eta_s\right).$$

Let $\tilde{\theta}_t$ denote the ridge regression estimator computed using the misreported feature vectors $\{x_{s,\cdot}\}_{s<t}$ and the corresponding noisy rewards $\{y'_s\}_{s<t}$, i.e.,

$$\tilde{\theta}_t = \left(\lambda I + \sum_{s<t} x_{s,a_s} x_{s,a_s}^\top\right)^{-1} \left(\sum_{s<t} x_{s,a_s} y'_s\right).$$

Then, we can express $\hat{\theta}_t$ as:

$$\hat{\theta}_t = \tilde{\theta}_t - \left(\lambda I + \sum_{s<t} x_{s,a_s} x_{s,a_s}^\top\right)^{-1} \sum_{s<t} \left(x_{s,a_s}\eta_s\right),$$

where the term $\left(\lambda I + \sum_{s<t} x_{s,a_s} x_{s,a_s}^\top\right)^{-1} \sum_{s<t} \left(x_{s,a_s}\eta_s\right)$ represents the additional bias introduced due to the misreporting of features. □

Some special cases of Assumption 1. The bias due to over-reported arm features makes it impossible to derive theoretical guarantees without introducing additional assumptions, such as Assumption 1. To validate its practicality, we consider the following three cases:

Case 1. All agents report truthfully: When reported features are the same as true features, i.e., $x = x^*$ for all $x \in \mathcal{X}$, $\text{UCB}_t(x)$ is an upper bound of $\theta_*^\top x$ with probability at least $1-\delta$ (or with high probability, Lemma 4). As a result, the first part of Assumption 1 holds, which is only needed to prove our Theorem 2, and hence the NE and regret bounds of our proposed algorithm, COBRA, are improved by a factor of $\sqrt{N}$ compared to Theorem 5.1 in Kleine Buening et al. (2024).

Case 2. One agent can over-report while other agents report truthfully and linear reward function: In Lemma 4, α_t is a non-decreasing function of t that grows logarithmically, while $||x||_{V_t^{-1}}$ converges at a rate of $1/\sqrt{t}$, leading to tighter confidence ellipsoid as t increases. Thus, $\text{UCB}_t(x)$ is smaller than $\text{UCB}_{t,-a}(x)$ for any x due to the use of additional observations from agent a. However, when an agent a overreports its features, it leads to biased estimates of $\theta_\star$. Since the agent over-reports, $\hat{\theta}_t$ becomes a downward-biased estimator of $\theta_\star$ (Example 4.7 in Wooldridge, in which over-reporting features can be treated as under-reporting rewards[3]). As a downward biased estimator leads to under-estimation with the fact that $\alpha_t||x||_{V_t^{-1}}$ is smaller than $\alpha_{t,-a}||x||_{V_{t,-a}^{-1}}$, $\text{UCB}_t(x)$ is smaller than $\text{UCB}_{t,-a}(x)$ with high probability. If an agent keeps over-reporting, our proposed method, LOOM, will detect this behavior and remove the agent from the active selection pool. Consequently, Assumption 1 will hold as the remaining agents are truthful.

Case 3. Multiple agents can over-report: In this case, all estimators used by COBRA become biased, making it impossible to derive theoretical guarantees without additional constraints. Our Theorem 4 and Theorem 5 hold as long as Assumption 1 is satisfied. Notably, we impose no restrictions on how agents report their features, aside from the no collusion assumption, which is a common assumption in VCG-type mechanisms (Vickrey, 1961; Clarke, 1971; Groves, 1973). We state the scope restriction of this case explicitly, as it is the key limitation of the leave-one-out construction. Even under no collusion, when multiple agents deviate independently, the reference set $\mathcal{O}_{t,-a}$ for agent a still contains the manipulated reports of the other deviators. Thus, the leave-one-out estimator $f_{t,-a}$ is no longer based solely on truthful data, and the bias characterized in Lemma 2 can affect it as well. Consequently, detection is guaranteed only for unilateral deviations (Case 2). In Case 3,

[3]Wooldridge, J. M. (2010). Econometric analysis of cross-section and panel data. MIT press.

Assumption 1 serves a different purpose: it ensures a valid optimistic bound $f(x_{t,a}) \leq \text{UCB}_t(x_{t,a})$ despite corrupted observations, which is the property required by the regret analysis. Moreover, Lemma 3 shows that this condition can be realized by an algorithm rather than assumed abstractly.

Relaxing this assumption remains an interesting direction for future work.

Comparison from existing literature. Kleine Buening et al. (2024) use agent-specific estimators that detect over-reporting in linear contextual bandits. In contrast, our method takes inspiration from the VCG mechanism (Vickrey, 1961; Clarke, 1971; Groves, 1973) and uses the observations associated with other agents to identify the over-reporting of an agent. This key difference leads us to use $\text{LCB}_{t,-a}(x_{t,a})$ (pessimistic reward estimate using observations of all agents except agent a) while Kleine Buening et al. (2024) use $\text{LCB}_{t,a}(x_{t,a})$ (pessimistic reward estimate only using observation associated with agent a) for detection.

Theorem 5.2 of Kleine Buening et al. (2024) holds under their Assumption 2 (holds only for linear reward functions), which has the following consequence:

$$f(x^*_{t,a^*}) \leq \text{UCB}_{t,a^*_t}(x_{t,a^*}) \leq \text{UCB}_{t,a_t}(x_{t,a_t})$$

(see proof of Lemma E.5 on Page 27 in Kleine Buening et al. (2024)).

In contrast, our assumptions imply the following:

$$f(x^*_{t,a^*}) \leq f(x_{t,a^*}) \leq \text{UCB}_t(x_{t,a^*_t}) \leq \text{UCB}_t(x_{t,a_t}) \leq \text{UCB}_{t,-a}(x_{t,a_t})$$

which gives:

$$f(x^*_{t,a^*}) \leq \text{UCB}_t(x_{t,a^*_t}) \leq \text{UCB}_{t,-a_t}(x_{t,a_t}) \implies f(x^*_{t,a^*}) \leq \text{UCB}_{t,-a_t}(x_{t,a_t}). \tag{3}$$

Assumption 2 of (Kleine Buening et al., 2024) and our Step 5 of COBRA share a key similarity: they define the conditions under which some theoretical results hold (their Theorem 5.2 and ours Theorem 4). These assumptions also lead to similar consequences, i.e., the maximum expected reward in any round is upper-bounded by the optimistic reward estimate of the selected arm (or agent) computed using the same agent(s) (i.e., $\text{UCB}_{t,a_t}(x_{t,a})$ and $\text{UCB}_{t,-a_t}(x_{t,a})$) as used in the mechanism for identifying over-reporting agents. We emphasize that our Step 5 of COBRA is not directly comparable to that of Kleine Buening et al. (2024), as they provide conditions for the theoretical guarantees of algorithms based on different underlying mechanisms.

We would like to highlight that detecting over-reporting using only an agent's own observations may be ineffective in practice, particularly when the true parameter θ_* is unknown due to the absence of any external baseline for comparison. In contrast, our VCG-inspired approach leverages observations from other agents to identify over-reporting, making it more practical, as the targeted agent cannot directly influence the detection mechanism.

Furthermore, *we extend our analysis to a class of non-linear contextual bandit algorithms, where the confidence ellipsoid around the unknown parameter $\theta_\star$ satisfies certain assumptions, and LOOM can be used as a subroutine in linear contextual bandit algorithms to identify strategic agents*. This constitutes a new contribution within this setting.

Equivalence between Step 5 of COBRA and Leave-one-out arm selection strategy. Our Step 5 of COBRA is equivalent to replacing the arm selection strategy in Step 4 of our proposed algorithm, COBRA, with a leave-one-out (LOO) arm selection strategy defined as:

$$a_t = \arg\max_{a \in \mathcal{A}} \text{UCB}_{t,-a}(x_{t,a}),$$

where $\text{UCB}_{t,-a}(x_{t,a})$ denotes the upper confidence bound computed using a LOO estimator that excludes the historical data from agent a, i.e., while calculating the UCBvalue for agent a, we use only the data from other agents. However, the confidence bounds derived from these LOO estimators are generally looser than those obtained using the data from all agents. We next prove the equivalence between Assumption 1 and the use of the LOO arm selection strategy.

Lemma 3. *When one of the agents overreports, having Assumption 1 together with Step 5 of* COBRA *results in the same regret as using the LOO arm selection strategy.*

This equivalence is stated for a *single* over-reporting agent. Its significance is that Assumption 1 need not be imposed as an external assumption: the leave-one-out arm-selection rule

$a_t = \text{argmax}_{a \in \mathcal{A}_t} \text{UCB}_{t,-a}(x_{t,a})$ satisfies the same inequality in Eq. (5) without any additional assumption. Thus, Assumption 1 is realizable by the algorithm itself. We retain the standard selection rule as the default only because it offers better performance under truthful reporting, as discussed after the proof.

Proof. For completeness, we first derive the consequences of over-reporting under Assumption 1 (also mentioned at the top of Page 19 in the Appendix) as follows. When an agent over-reports, the following inequality holds for the optimal arm $x_{t,a_t^\star}$ in round t: $f(x^\star_{t,a_t^\star}) \leq f(x_{t,a_t^\star})$. Using Assumption 1, i.e., $f(x_{t,a_t^\star}) \leq \text{UCB}_t(x_{t,a_t^\star})$, which states that the upper confidence bounds remain valid even when agents over-report, we obtain:

$$f(x^\star_{t,a_t^\star}) \leq f(x_{t,a_t^\star}) \leq \text{UCB}_t(x_{t,a_t^\star}).$$

Since COBRA selects arm a_t over $a_t^\star$, i.e., $\text{UCB}_t(x_{t,a_t^\star}) \leq \text{UCB}_t(x_{t,a_t})$, we obtain:

$$f(x^\star_{t,a_t^\star}) \leq \text{UCB}_t(x_{t,a_t}).$$

Using Step 5 of COBRA, i.e., $\text{UCB}_t(x_{t,a_t}) \leq \text{UCB}_{t,-a_t}(x_{t,a_t})$, we get:

$$f(x^\star_{t,a_t^\star}) \leq \text{UCB}_{t,-a_t}(x_{t,a_t}). \tag{4}$$

When we use the LOO arm selection strategy $a_t = \arg\max_{a \in \mathcal{A}} \text{UCB}_{t,-a}(x_{t,a})$, the following chain of inequalities holds under over-reporting:

$$f(x^\star_{t,a_t^\star}) \leq f(x_{t,a_t^\star} \leq \text{UCB}_{t,-a_t^\star}(x_{t,a_t^\star}).$$

Since arm a_t is selected, i.e., $\text{UCB}_{t,-a_t^\star}(x_{t,a_t^\star}) \leq \text{UCB}_{t,-a_t}(x_{t,a_t})$, we conclude:

$$f(x^\star_{t,a_t^\star}) \leq \text{UCB}_{t,-a_t}(x_{t,a_t}). \tag{5}$$

Note that Eq. (4) and Eq. (5) are equivalent, Assumption 1 and the leave-one-out (LOO) arm selection strategy lead to the same theoretical consequences, hence having the same NE and regret guarantees. □

To obtain an upper bound on the regret, we derive a key result that bounds the difference $\text{UCB}_{t,-a_t}(x_{t,a_t}) - f(x_{t,a_t^\star})$. These results are formalized in Lemma 6 for the linear reward function and in Lemma 9 for the non-linear case. Importantly, our mechanism, LOOM, ensures that truthful reporting is a dominant strategy for each agent when all others report truthfully. Therefore, we do not adopt the LOO arm selection strategy as the default in COBRA, as the LOO strategy underperforms the standard arm selection strategy currently used in COBRA when all agents report truthfully. LOOM does not work in cases involving complex strategic behavior by agents, such as when multiple agents collude or misreport together. Addressing these challenges by designing new mechanisms will be a promising area of research at the intersection of mechanism design and contextual bandits. We will leave studying these complex settings to future work, for which our work can serve as a foundation.

C LEFTOVER PROOFS

Scope of the guarantees. We first state the scope of our results: (i) LOOM is designed to detect *over-reporting* and does not detect under-reporting; (ii) our incentive-compatibility guarantees consider only *unilateral* deviations, and thus do not allow collusion between agents; (iii) we consider a *homogeneous* setting in which all agents share a common reward function f; and (iv) because LOOM is a peer-based test, its power depends on how well the remaining agents *cover* the deviating agent's arm-feature directions. This coverage requirement is the binding condition in our guarantees and is captured by the constant C. When coverage is poor, C can grow as large as $\Theta(\sqrt{T})$, making the bound in Theorems 4 and 5 vacuous (Sec. B.1). We discuss this residual gap, including the substantial overlap between regions where misreporting is beneficial and regions where LOOM works, in Sec. 3 and Sec. B.1; we discuss possible extensions to heterogeneous agents in Sec. 6 and Sec. F.

C.1 LEFTOVER PROOFS FROM SEC. 3

Lemma 1. *Let $S_t(a)$ be the number of times that agent a is selected until round t, and ε_s be R-sub-Gaussian noise in the observed reward y_s, where $1 \leq s \leq t$. Then, $\sum_{s \leq t, a_s = a} f(x^\star_{s,a_s}) \leq \sum_{s \leq t, a_s = a} y_s + \sqrt{2R^2 S_t(a) \log(1/\delta^y_{t,a})}$. holds with probability at least $1 - \delta^y_{t,a}$.*

Proof. Recall, the observed reward in round t is $y_t = f(x^\star_{t,a_t}) + \varepsilon_t$, where ε_t is R-sub-Gaussian noise. We want to get the upper bound for the sum of observed rewards in terms of the sum of true rewards, i.e., $\sum_{s \leq t, a_s = a} \left(y_s - f(x^\star_{s,a_s})\right)$. Note that $\varepsilon_s = y_s - f(x^\star_{s,a_s})$ is a R-sub-Gaussian random variable. Using Hoeffding's inequality for sums of sub-Gaussian random variables, we obtain

$$\text{For any } \tau > 0,\ \mathbb{P}\left\{ \sum_{s \leq t, a_s = a} \varepsilon_s \geq \tau \right\} \leq \exp\left(-\frac{\tau^2}{2R^2 S_t(a)} \right).$$

Setting $\tau = \sqrt{2R^2 S_t(a) \log(1/\delta^y_{t,a})}$, we get

$$\mathbb{P}\left\{ \sum_{s \leq t, a_s = a} \varepsilon_s \geq \sqrt{2R^2 S_t(a) \log(1/\delta^y_{t,a})} \right\} \leq \delta^y_{t,a}.$$

Expanding $\varepsilon_s = y_s - f(x^\star_{s,a_s})$ in the above equation, we can have the following results with probability at least $1 - \delta^y_{t,a}$,

$$\sum_{s \leq t, a_s = a} y_s \geq \sum_{s \leq t, a_s = a} f(x^\star_{s,a_s}) - \sqrt{2R^2 S_t(a) \log(1/\delta^y_{t,a})}. \tag{6}$$

Similarly, with probability at least $1 - \delta^y_{t,a}$,

$$\sum_{s \leq t, a_s = a} y_s \leq \sum_{s \leq t, a_s = a} f(x^\star_{s,a_s}) + \sqrt{2R^2 S_t(a) \log(1/\delta^y_{t,a})}. \tag{7}$$

After re-arrangements of some terms in Eq. (6), the sum of true rewards must be less than the upper bound of observed rewards with probability at least $1 - \delta^y_{t,a}$, i.e.,

$$\sum_{s \leq t, a_s = a} f(x^\star_{s,a_s}) \leq \sum_{s \leq t, a_s = a} y_s + \sqrt{2R^2 S_t(a) \log(1/\delta^y_{t,a})}. \qquad \square$$

Theorem 1. *Let agent a report truthfully and all other agents are assumed to be truthful. Then, LOOM does not eliminate agent a with high probability at least $1 - \delta^x_{t,a} - \delta^y_{t,a}$.*

Proof. Since all agents report truthfully, for all $t \geq 1, a \in \mathcal{A}_t : x^\star = x$. Note that we are estimating the reward function f using available observations, context-arm features, and rewards. Recall that we use $\mathcal{O}_{t,-a}$ to denote the observations from all agents except agent a and $f_{t,-a}$ represents the estimate of reward function f using $\mathcal{O}_{t,-a}$ at the end of round t. Even if other agents report truthfully, noisy reward feedback may lead to an inaccurate estimator. Let the confidence ellipsoid $|f_{t,-a}(x) - f(x)| \leq h(x, \mathcal{O}_{t,-a})$ hold with probability $1 - \delta_{t,a}$. Then, for any $x \in \mathcal{X}$, $\mathrm{LCB}_{t,-a}(x) = f_{t,-a}(x) - h(x, \mathcal{O}_{t,-a})$ is the pessimistic estimates of the expected reward for x that also holds with probability $1 - \delta_{t,a}$. Furthermore, $f(x) \geq \mathrm{LCB}_{t,-a}(x)$ (see Lemma 8 in Sec. D.1 for more details). Using this, for any $x_{t,a_t} \in \mathcal{X}$, we have

$$f(x^\star_{t,a_t}) = f(x_{t,a_t}) \geq \mathrm{LCB}_{t,-a}(x_{t,a_t}) \implies f(x^\star_{t,a_t}) \geq \mathrm{LCB}_{t,-a}(x_{t,a_t}).$$

Next, we can lower bound the sum of true rewards in terms of the lower confidence bound on estimated rewards using observed context-arm feature vectors as follows:

$$\sum_{s \leq t, a_s = a} f(x^\star_{s,a_s}) \geq \sum_{s \leq t, a_s = a} \mathrm{LCB}_{t,-a}(x_{s,a_s})$$

$$\implies \sum_{s\le t, a_s=a} \mathrm{LCB}_{t,-a}(x_{s,a_s}) \le \sum_{s\le t, a_s=a} f(x^\star_{s,a_s}). \tag{8}$$

For brevity, we assume the above bound holds with probability at least $1-\delta^x_{t,a}$ in the round t. Note that $\delta^x_{t,a}$ can be computed exactly when applying the union bound. Since the true reward is unknown, we instead first use the upper bound provided in Lemma 1, which holds with probability at least $\delta^y_{t,a}$, to modify Eq. (8). We use the definitions of $\mathrm{LCB}^{(x)}_{t,a}$ and $\mathrm{UCB}^{(y)}_{t,a}$ to get:

$$\sum_{s\le t, a_s=a} \mathrm{LCB}_{t,-a}(x_{s,a_s}) \le \sum_{s\le t, a_s=a} y_s + \sqrt{2R^2 S_t(a)\log(1/\delta^y_{t,a})}$$
$$\implies \mathrm{LCB}^{(x)}_{t,a} \le \mathrm{UCB}^{(y)}_{t,a}. \tag{9}$$

If the sum of the lower bound of estimated rewards is less than the upper bound of observed rewards for an agent, then that agent is not misreporting. However, if any agent violates Eq. (9), i.e., $\mathrm{LCB}^{(x)}_{t,a} > \mathrm{UCB}^{(y)}_{t,a}$, then that agent is not truthful. The probability of failing this LOOM condition is upper bounded by $\delta^x_{t,a}+\delta^y_{t,a}$. Since this condition is used as a criterion in COBRA to identify the strategic agent, COBRA does not eliminate a truthful agent with probability at least $1-\delta^x_{t,a}-\delta^y_{t,a}$. □

We want to emphasize that the quantifier $\boldsymbol{\sigma}_{-a} = \boldsymbol{\sigma}^\star_{-a}$ is exactly the one required by Nash incentive compatibility, under which $\mathcal{O}_{t,-a}$ contains only truthful observations. Accordingly, the proof (Sec. C) relies only on $\mathrm{LCB}_{t,-a}$, Lemma 1, and Lemma 8, all evaluated at the truthful profile, and makes no assumption about corrupted observations. Moreover, the proof never compares $h(x,\mathcal{O}_t)$ and $h(x,\mathcal{O}_{t,-a})$, and therefore does not depend on the advantage constant C in Definition 5.

C.2 Leftover proofs from Sec. 4

The following lemmas are fundamental to the proof of our theoretical results. We use the following notation throughout the proof: the arm is represented by a, and $-a$ represents other than arm a's estimate. We use $\|x\|_A$ to denote the weighted l_2-norm of vector x with respect to matrix A. We next state the result that gives the confidence ellipsoid with center at $\hat{\theta}_t$ or confidence set for the case when the reward function is linear. We will use this result to prove our bounds in Sec. 4.

Lemma 4. *Let $\delta\in(0,1)$, $\lambda>0$, $R>0$, $\hat{\theta}_t = V_t^{-1}\sum_{s=1}^{t-1} x_{s,a_s}y_s$, $V_t = \lambda I + \sum_{s=1}^{t-1} x_{s,a_s}x_{s,a_s}^\top$. Then, with probability at least $1-\delta$, for all $t\ge 1$, $\theta_\star$ lies in the following confidence set:*

$$C_t = \left\{\theta\in\mathbb{R}^d:\ \left\|\hat{\theta}_t-\theta\right\|_{V_t}\le\alpha_t\right\},\ \textit{where}\ \alpha_t = \left(R\sqrt{d\log\left(\frac{1+(tL^2/\lambda)}{\delta}\right)}+\lambda^{\frac{1}{2}}S\right).$$

Furthermore, with probability at least $1-\delta$,

$$\forall x\in\mathcal{X}: \theta_\star^\top x \le \mathrm{UCB}_t(x) = \hat{\theta}_t^\top x + \alpha_t\|x\|_{V_t^{-1}}.$$

Similarly, with probability at least $1-\delta$,

$$\forall x\in\mathcal{X}: \theta_\star^\top x \ge \mathrm{LCB}_t(x) = \hat{\theta}_t^\top x - \alpha_t\|x\|_{V_t^{-1}}.$$

Proof. The proof of the first part of the results directly follows from Theorem 2 of Abbasi-Yadkori et al. (2011). The proof of the second part follows from the first part with some simple algebraic manipulations as follows:

$$\theta_\star^\top x - \hat{\theta}_t^\top x \le |\hat{\theta}_t^\top x - \theta_\star^\top x|$$
$$\implies \theta_\star^\top x - \hat{\theta}_t^\top x \le \left\|\hat{\theta}_t-\theta_\star\right\|_{V_t}\|x\|_{V_t^{-1}}$$
$$\implies \theta_\star^\top x \le \hat{\theta}_t^\top x + \alpha_t\|x\|_{V_t^{-1}}$$
$$\implies \theta_\star^\top x \le \mathrm{UCB}_t(x).$$

Similarly, the last part also follows from the first part with some simple algebraic manipulations as follows:

$$|\hat{\theta}_t^\top x - \theta_\star^\top x| \leq \|x\|_{V_t^{-1}} \left\|\hat{\theta}_t - \theta_\star\right\|_{V_t}.$$

After reversing the above inequality, we have

$$\begin{aligned}
&\|x\|_{V_t^{-1}} \left\|\hat{\theta}_t - \theta_\star\right\|_{V_t} \geq |\hat{\theta}_t^\top x - \theta_\star^\top x| \geq \hat{\theta}_t^\top x - \theta_\star^\top x \\
\implies &\|x\|_{V_t^{-1}} \left\|\hat{\theta}_t - \theta_\star\right\|_{V_t} \geq \hat{\theta}_t^\top x - \theta_\star^\top x \\
\implies &\theta_\star^\top x \geq \hat{\theta}_t^\top x - \left\|\hat{\theta}_t - \theta_\star\right\|_{V_t} \|x\|_{V_t^{-1}} \\
\implies &\theta_\star^\top x \geq \hat{\theta}_t^\top x - \alpha_t \|x\|_{V_t^{-1}} \\
\implies &\theta_\star^\top x \geq \mathrm{LCB}_t(x).
\end{aligned}$$

□

Note that it is possible $\theta_\star$ may not belong to the confidence ellipsoid of θ. However, when all agents are truthful, i.e., $x = x^\star$, then $\theta = \theta_\star$ is trivially satisfied. We make the manipulation setting explicit because it is the regime for which LOOM is designed. Lemma 4 is derived for observations generated by the reported features, and we do not assume that its confidence guarantee remains valid when those features are corrupted. In particular, when an agent over-reports, Lemma 2 in Sec. B.2 shows that the ridge estimator acquires the bias $\left(\lambda I + \sum_{s<t} x_{s,a_s} x_{s,a_s}^\top\right)^{-1}\left(\sum_{s<t} x_{s,a_s}\eta_s\right)$, with $\eta_s \geq 0$, so $\theta_\star$ may lie outside the resulting confidence ellipsoid. Thus, the confidence guarantee for the full observation set relies on truthful reporting. In contrast, Lemma 5 is stated for $\mathcal{O}_{t,-a}$ and is the key ingredient in the detection guarantee: under a unilateral deviation, the remaining agents are truthful, so its assumptions hold. For multiple deviators, the additional condition required for Lemma 4 is isolated explicitly in Assumption 1.

Recall the following definitions from the main paper (note that we estimated the ordinary least square (OLS) closed-form solution excluding the information of agent a):

$$\hat{\theta}_{t,-a} = V_{t,-a}^{-1} \sum_{s=1, a_s \neq a}^{t-1} x_{s,a_s} y_s, \text{ with } V_{t,-a} = \lambda I + \sum_{s=1, a_s \neq a}^{t-1} x_{s,a_s} x_{s,a_s}^\top.$$

Lemma 5. *Let $\delta \in (0,1)$, $\lambda > 0$, and $R > 0$. Then, with probability $1-\delta$,*

$$\left\|\hat{\theta}_{t,-a} - \theta_\star\right\|_{V_{t,-a}} \leq \left(R\sqrt{d\log\left(\frac{1 + (t - S_t(a))L^2/\lambda}{\delta}\right)} + \lambda^{\frac{1}{2}} S\right) = \alpha_{t,-a}.$$

Furthermore, with probability at least $1-\delta$, the upper bound of $\theta_\star^\top x$ is given by

$$\forall x \in \mathcal{X} : \theta_\star^\top x \leq \mathrm{UCB}_{t,-a}(x) = \hat{\theta}_{t,-a}^\top x + \alpha_{t,-a} \|x\|_{V_{t,-a}^{-1}}.$$

Similarly, with probability at least $1-\delta$, the lower bound of $\theta_\star^\top x$ is given by

$$\forall x \in \mathcal{X} : \theta_\star^\top x \geq \mathrm{LCB}_{t,-a}(x) = \hat{\theta}_{t,-a}^\top x - \alpha_{t,-a} \|x\|_{V_{t,-a}^{-1}}.$$

Proof. The first part of the proof follows from Lemma 4 as we are not using observations associated with agent a, reducing to the standard confidence bound restricted to observations of all agents except a. The proof of the second part follows from the first part with some simple algebraic manipulations as follows:

$$\begin{aligned}
&\theta_\star^\top x - \hat{\theta}_{t,-a}^\top x \leq |\hat{\theta}_{t,-a}^\top x - \theta_\star^\top x| \\
\implies &\theta_\star^\top x - \hat{\theta}_{t,-a}^\top x \leq \left\|\hat{\theta}_{t,-a} - \theta_\star\right\|_{V_{t,-a}} \|x\|_{V_{t,-a}^{-1}} \\
\implies &\theta_\star^\top x \leq \hat{\theta}_{t,-a}^\top x + \alpha_{t,-a} \|x\|_{V_{t,-a}^{-1}} \\
\implies &\theta_\star^\top x \leq \mathrm{UCB}_{t,-a}(x).
\end{aligned}$$

Similarly, the last part follows from the first part with some algebraic manipulations as follows:

$$|\hat{\theta}_{t,-a}^\top x - \theta_\star^\top x| \leq \|x\|_{V_{t,-a}^{-1}} \left\|\hat{\theta}_{t,-a} - \theta_\star\right\|_{V_{t,-a}}.$$

After reversing the above inequality, we have

$$\begin{aligned}
&\|x\|_{V_{t,-a}^{-1}} \left\|\hat{\theta}_{t,-a} - \theta_\star\right\|_{V_{t,-a}} \geq |\hat{\theta}_{t,-a}^\top x - \theta_\star^\top x| \geq \hat{\theta}_{t,-a}^\top x - \theta_\star^\top x \\
\implies &\|x\|_{V_{t,-a}^{-1}} \left\|\hat{\theta}_{t,-a} - \theta_\star\right\|_{V_{t,-a}} \geq \hat{\theta}_{t,-a}^\top x - \theta_\star^\top x \\
\implies &\theta_\star^\top x \geq \hat{\theta}_{t,-a}^\top x - \left\|\hat{\theta}_{t,-a} - \theta_\star\right\|_{V_{t,-a}} \|x\|_{V_{t,-a}^{-1}} \\
\implies &\theta_\star^\top x \geq \hat{\theta}_{t,-a}^\top x - \alpha_{t,-a} \|x\|_{V_{t,-a}^{-1}} \\
\implies &\theta_\star^\top x \geq \mathrm{LCB}_{t,-a}(x).
\end{aligned}$$

□

C.2.1 Proof of Theorem 2

Theorem 2 (Linear)**.** *When agents report truthfully, being truthful is a* $\tilde{O}(d\sqrt{T})$-NE *under* COBRA. *The regret of* COBRA *under this approximate* NE *is at most* $\mathfrak{R}_T(\mathrm{COBRA}, \boldsymbol{\sigma}^\star) = \tilde{O}(d\sqrt{T})$.

Proof. When all agents report truthfully, our algorithm is the same as Lin-UCB (Chu et al., 2011) with a mechanism for identifying strategic agents that holds with probability $1 - \delta_x - \delta_y$. For completeness, we first prove the regret upper bound of COBRA as follows:

$$\mathfrak{R}_T\left(\mathrm{COBRA}, \boldsymbol{\sigma}^\star\right) = \sum_{t=1}^{T} (\theta_\star^\top x_{t,a_t^\star}^\star - \theta_\star^\top x_{t,a_t}^\star). \tag{10}$$

Since the true feature vector is the same as the reported context-arm feature vector (i.e., $x_{t,a}^\star = x_{t,a}$), we can start with upper bounding the difference $\theta_\star^\top x_{t,a_t^\star}^\star - \theta_\star^\top x_{t,a_t}^\star$ as follows:

$$\begin{aligned}
\theta_\star^\top x_{t,a_t^\star}^\star - \theta_\star^\top x_{t,a_t}^\star &= \theta_\star^\top x_{t,a_t^\star} - \theta_\star^\top x_{t,a_t} \\
&\leq \mathrm{UCB}(x_{t,a_t^\star}) - \theta_\star^\top x_{t,a_t} \\
&\leq \mathrm{UCB}(x_{t,a_t}) - \theta_\star^\top x_{t,a_t} \qquad \left(\text{as } \mathrm{UCB}(x_{t,a_t^\star}) \leq \mathrm{UCB}(x_{t,a_t})\right) \\
&= \hat{\theta}_t^\top x_{t,a_t} + \alpha_t \|x_{t,a_t}\|_{V_t^{-1}} - \theta_\star^\top x_{t,a_t} \\
&= \hat{\theta}_t^\top x_{t,a_t} - \theta_\star^\top x_{t,a_t} + \alpha_t \|x_{t,a_t}\|_{V_t^{-1}} \\
&\leq \left\|\theta_\star - \hat{\theta}_t\right\|_{V_t} \|x_{t,a_t}\|_{V_t^{-1}} + \alpha_t \|x_{t,a_t}\|_{V_t^{-1}} \\
&\leq \alpha_t \|x_{t,a_t}\|_{V_t^{-1}} + \alpha_t \|x_{t,a_t}\|_{V_t^{-1}} \\
\implies \theta_\star^\top x_{t,a_t^\star}^\star - \theta_\star^\top x_{t,a_t}^\star &\leq 2\alpha_t \|x_{t,a_t}\|_{V_t^{-1}}.
\end{aligned} \tag{11}$$

Note that $\hat{\theta}_t$ is an estimator of $\theta_\star$ as the true feature vector is the same as the reported context-arm feature vector. After using the upper bound given in Eq. (11) into Eq. (10), we get an upper bound on the regret as follows:

$$\begin{aligned}
\mathfrak{R}_T\left(\mathrm{COBRA}, \boldsymbol{\sigma}^\star\right) &= \theta_\star^\top x_{t,a_t^\star}^\star - \theta_\star^\top x_{t,a_t}^\star \\
&\leq \sum_{t=1}^{T} 2\alpha_t \|x_{t,a_t}\|_{V_t^{-1}} \\
&= 2\sum_{t=1}^{T} \alpha_t \|x_{t,a_t}\|_{V_t^{-1}} \\
&\leq 2\sqrt{T}\sqrt{\sum_{t=1}^{T} \left[\alpha_t \|x_{t,a_t}\|_{V_t^{-1}}\right]^2}
\end{aligned}$$

$$\leq 2\sqrt{T}\sqrt{\sum_{t=1}^{T}\left[\alpha_T \left\|x_{t,a_t}\right\|_{V_t^{-1}}\right]^2}$$

$$= 2\sqrt{T}\sqrt{\alpha_T^2 \sum_{t=1}^{T}\left\|x_{t,a_t}\right\|_{V_t^{-1}}^2}$$

$$= 2\alpha_T\sqrt{T}\sqrt{\sum_{t=1}^{T}\left\|x_{t,a_t}\right\|_{V_t^{-1}}^2}$$

$$\leq 2\alpha_T\sqrt{T}\sqrt{2\log\frac{\det(V_T)}{\det(\lambda I_d)}}$$

$$\implies \mathfrak{R}_T\left(\text{COBRA}, \boldsymbol{\sigma}^\star\right) \leq 2\alpha_T\sqrt{2dT\log(\lambda + TL/d)} = \tilde{O}(d\sqrt{T}). \tag{12}$$

The first inequality directly follows from Eq. (11). The second inequality is due to using the Cauchy-Schwarz inequality, where the third inequality follows from the fact that α_t increases with t. The last two inequalities follow from Lemma 11 and Lemma 10 of Abbasi-Yadkori et al. (2011), respectively, and $\alpha_T = \tilde{O}(d\log T)$.

We now prove that being truthful is an approximate Nash equilibrium for COBRA. Recall, $S_T(a)$ denotes the number of times an agent being selected by COBRA, which is given as follows:

$$\begin{aligned}
S_T(a) &= \sum_{t=1}^{T}\mathbb{1}(a_t = a) \\
&= \sum_{t=1}^{T}\mathbb{1}(a_t = a, a_t^\star = a) + \sum_{t=1}^{T}\mathbb{1}(a_t = a, a_t^\star \neq a) \\
&\geq \sum_{t=1}^{T}\mathbb{1}(a_t^\star = a) - \sum_{t=1}^{T}\mathbb{1}(a_t^\star = a, a_t \neq a) \\
&\geq \sum_{t=1}^{T}\mathbb{1}(a_t^\star = a) - \sum_{t=1}^{T}\mathbb{1}(a_t \neq a_t^\star) \\
\implies S_T(a) &\geq S_T^\star(a) - \sum_{t=1}^{T}\mathbb{1}(a_t \neq a_t^\star).
\end{aligned} \tag{13}$$

To get the lower bound $S_T(a)$, we get an upper bound $\sum_{t=1}^{T}\mathbb{1}(a_t \neq a_t^\star)$. Let $\Delta_{a_t} = \left(\theta_\star^\top x_{t,a_t^\star}^\star - \theta_\star^\top x_{t,a_t}^\star\right) > 0$ for $a_t \neq a_t^\star$. We multiply and divide $\mathbb{1}(a_t \neq a_t^\star)$ by Δ_{a_t} and then use inequality in Eq. (11), i.e., $\Delta_{a_t} \leq 2\alpha_t \left\|x_{t,a_t}\right\|_{V_t^{-1}}$ as follows:

$$\begin{aligned}
\sum_{t=1}^{T}\mathbb{1}(a_t \neq a_t^\star) &= \sum_{t=1}^{T}\mathbb{1}(a_t \neq a_t^\star)\frac{\Delta_{a_t}}{\Delta_{a_t}} \\
&\leq \sum_{t=1}^{T}\mathbb{1}(a_t \neq a_t^\star)\frac{2\alpha_t \left\|x_{t,a_t}\right\|_{V_t^{-1}}}{\Delta_{a_t}} \qquad (\text{as } x_{t,a}^\star = x_{t,a}) \\
&\leq \sum_{t=1}^{T}\frac{2\alpha_t \left\|x_{t,a_t}\right\|_{V_t^{-1}}}{\Delta_{a_t}} \\
&= \sum_{t=1}^{T}\frac{1}{\Delta_{a_t}} 2\alpha_t \left\|x_{t,a_t}\right\|_{V_t^{-1}} \\
&\leq \sqrt{\sum_{t=1}^{T}\left(\frac{1}{\Delta_{a_t}}\right)^2 \sum_{t=1}^{T}\left(2\alpha_t \left\|x_{t,a_t}\right\|_{V_t^{-1}}\right)^2}
\end{aligned}$$

$$
\begin{aligned}
&\leq \sqrt{\sum_{t=1}^{T}\left(\frac{1}{\Delta_{a_t}}\right)^2 \sum_{t=1}^{T}\left(2\alpha_T \left\|x_{t,a_t}\right\|_{V_t^{-1}}\right)^2} \\
&= \sqrt{\sum_{t=1}^{T}\left(\frac{1}{\Delta_{a_t}}\right)^2} \sqrt{\sum_{t=1}^{T}\left(2\alpha_T \left\|x_{t,a_t}\right\|_{V_t^{-1}}\right)^2} \\
&= \sqrt{\sum_{t=1}^{T}\left(\frac{1}{\Delta_{a_t}}\right)^2} \sqrt{(2\alpha_T)^2 \sum_{t=1}^{T}\left\|x_{t,a_t}\right\|^2_{V_t^{-1}}} \\
&= 2\alpha_T \sqrt{\sum_{t=1}^{T}\left(\frac{1}{\Delta_{a_t}}\right)^2} \sqrt{\sum_{t=1}^{T}\left\|x_{t,a_t}\right\|^2_{V_t^{-1}}} \\
&\leq 2\alpha_T \sqrt{\sum_{t=1}^{T}\left(\frac{1}{\Delta_{a_t}}\right)^2} \sqrt{2\log\frac{\det(V_T)}{\det(\lambda I_d)}} \\
&\leq 2\alpha_T \sqrt{\sum_{t=1}^{T}\left(\frac{1}{\Delta_{a_t}}\right)^2} \sqrt{2d\log(\lambda + TL/d)} \\
&\leq 2\alpha_T \sqrt{\sum_{t=1}^{T}\left(\frac{1}{\Delta_{\min}}\right)^2} \sqrt{2d\log(\lambda + TL/d)} \\
&\leq \frac{2}{\Delta_{\min}} \alpha_T \sqrt{T} \sqrt{2d\log(\lambda + TL/d)} \\
\implies \sum_{t=1}^{T} \mathbb{1}(a_t \neq a_t^\star) &\leq \frac{2}{\Delta_{\min}} \left(R\sqrt{d\log\left(\frac{1 + (tL^2/\lambda)}{\delta}\right)} + \lambda^{\frac{1}{2}} S \right) \sqrt{2dT\log(\lambda + TL/d)}.
\end{aligned}
$$

Note that $\Delta_{\min} = \min_{a_t \neq a_t^\star} \Delta_{a_t}$. Although using $\Delta_{\min}$ loosens the upper bound, we use this to get dependence on T. Let $\tilde{O}$ hide the dependence on logarithmic terms, then we have the following result:

$$
\sum_{t=1}^{T} \mathbb{1}(a_t \neq a_t^\star) \leq \tilde{O}\left(d\sqrt{T}\right). \tag{14}
$$

Using this upper bound in Eq. (13), we get the following bound for any agent $a \in \mathcal{A}$:

$$
S_T(a) \geq S_T^\star(a) - \tilde{O}\left(d\sqrt{T}\right). \tag{15}
$$

Now we consider the case where an agent a deviates from the truthful strategy. The number of times an agent being selected by COBRA is given as follows:

$$
\begin{aligned}
S_T(a) &= \sum_{t=1}^{T} \mathbb{1}(a_t = a) \\
&= \sum_{t=1}^{T} \mathbb{1}(a_t = a, a_t^\star = a) + \sum_{t=1}^{T} \mathbb{1}(a_t = a, a_t^\star \neq a) \\
&\leq \sum_{t=1}^{T} \mathbb{1}(a_t^\star = a) + \sum_{t=1}^{T} \mathbb{1}(a_t = a, a_t^\star \neq a) \\
&\leq \sum_{t=1}^{T} \mathbb{1}(a_t^\star = a) + \sum_{t=1}^{T} \mathbb{1}(a_t \neq a_t^\star).
\end{aligned} \tag{16}
$$

Using Eq. (14) in Eq. (16), we get

$$S_T(a) \leq S_T^\star(a) + \tilde{O}\left(d\sqrt{T}\right). \tag{17}$$

Combining Eq. (15) and Eq. (17) completes our proof that COBRA is $\tilde{O}(d\sqrt{T})$-NE. □

Theorem 2 considers a unilateral deviation from the truthful profile: the remaining agents report truthfully ($\boldsymbol{\sigma}_{-a} = \boldsymbol{\sigma}^\star_{-a}$), while the deviating agent's reports are unrestricted. Accordingly, the deviation step of the proof (Eq. (16) in Sec. C) bounds the deviator's gain through the number of suboptimal selections. The argument is entirely in terms of V_t and never compares V_t with $V_{t,-a}$; consequently, it does not involve the advantage constant C of Definition 5.

C.2.2 Proof of Theorem 4

To prove Theorem 4, we will use the following result that upper bounds the total amount of regret that an agent a can exert before being identified by LOOM.

Lemma 6. *Let* $\mathrm{UCB}_{t,-a}(x_{s,a}) = \hat{\theta}_{t,-a}^\top x_{s,a} + \alpha_{t,-a} \|x_{s,a}\|_{V_{t,-a}^{-1}}$, *where* $x_{s,a}$ *is the arm feature vector associated with agent* a *in the round* s. *Then, with probability at least* $1 - \delta_{t,a}^x - \delta_{t,a}^y$,

$$\sum_{s \leq t:\ a_s = a} \left(\mathrm{UCB}_{t,-a}(x_{s,a}) - \theta_\star^\top x_{s,a}^\star\right) \leq \sum_{s \leq t, a_s = a} 2\alpha_{t,-a} \|x_{s,a}\|_{V_{t,-a}^{-1}} + 2\sqrt{2R^2 S_t(a) \log(1/\delta_{t,a}^y)}.$$

Proof. Using Eq. (9) with Lemma 5 for linear reward function that holds with probability at least $1 - \delta_{t,a}^x$, we get:

$$\begin{aligned}
\sum_{s \leq t, a_s = a} \left(\hat{\theta}_{t,-a}^\top x_{s,a} - \alpha_{t,-a} \|x_{s,a}\|_{V_{t,-a}^{-1}}\right) &\leq \sum_{s \leq t, a_s = a} y_s + \sqrt{2R^2 S_t(a) \log(1/\delta_{t,a}^y)} \\
&\leq \sum_{s \leq t, a_s = a} \theta_\star^\top x_{s,a}^\star + \sqrt{2R^2 S_t(a) \log(1/\delta_{t,a}^y)} \\
&\quad + \sqrt{2R^2 S_t(a) \log(1/\delta_{t,a}^y)} \\
\implies \sum_{s \leq t, a_s = a} \left(\hat{\theta}_{t,-a}^\top x_{s,a} - \theta_\star^\top x_{s,a}^\star\right) &\leq \sum_{s \leq t, a_s = a} \alpha_{t,-a} \|x_{s,a}\|_{V_{t,-a}^{-1}} + 2\sqrt{2R^2 S_t(a) \log(1/\delta_{t,a}^y)}.
\end{aligned}$$

The second inequality follows from Eq. (7) by using upper bound (as the reward function is linear) on $\sum_{s \leq t, a_s = a} y_s$ that holds with probability $1 - \delta_{t,a}^y$. Now we prove the second part of the result by replacing $\hat{\theta}_{t,-a}^\top x_{s,a}$ by $\mathrm{UCB}_{t,-a}(x_{s,a}) - \alpha_{t,-a} \|x_{s,a}\|_{V_{t,-a}^{-1}}$ and we get

$$\begin{aligned}
&\sum_{s \leq t, a_s = a} \left(\mathrm{UCB}_{t,-a}(x_{s,a}) - \alpha_{t,-a} \|x_{s,a}\|_{V_{t,-a}^{-1}} - \theta_\star^\top x_{s,a}^\star\right) \\
&\qquad \leq \sum_{s \leq t, a_s = a} \alpha_{t,-a} \|x_{s,a}\|_{V_{t,-a}^{-1}} + 2\sqrt{2R^2 S_t(a) \log(1/\delta_{t,a}^y)} \\
\implies &\sum_{s \leq t, a_s = a} \left(\mathrm{UCB}_{t,-a}(x_{s,a}) - \theta_\star^\top x_{s,a}^\star\right) \\
&\qquad \leq \sum_{s \leq t, a_s = a} 2\alpha_{t,-a} \|x_{s,a}\|_{V_{t,-a}^{-1}} + 2\sqrt{2R^2 S_t(a) \log(1/\delta_{t,a}^y)}.
\end{aligned}$$

□

We first restate the main assumptions needed to prove Theorem 4.

Assumption 1. *Let* x *and* $x^\star$ *be the reported and true context-arm feature vector, respectively. Then, we assume* $\forall t \geq 1, a \in \mathcal{A}_t : f(x_{t,a}) \leq \mathrm{UCB}_t(x_{t,a})$, *where* $\mathrm{UCB}_t(x) = f_t(x) + h(x, \mathcal{O}_t)$.

We want to highlight that the assumption requirement is stronger than what LOOM actually needs. LOOM only requires the *directions* played by an over-reporting agent to be sufficiently covered by

the reports of the remaining agents. This is a property of the *design*, rather than of agent identity. Accordingly, we formulate detectability in terms of the confidence width $h(x, \mathcal{O}_{t,-a})$, rather than the identity of the agent, and the same quantity determines the advantage constant C in Definition 5. Possible extensions to genuinely heterogeneous agents, including clustered agent types, similarity-based weighting, and shared latent representations, are discussed in Sec. F (Question 3).

We now have all the results that will be used to prove Theorem 4.

Theorem 4 (Linear)**.** *If Assumption 1 hold then, with C as in Definition 5, the regret of* COBRA *is* $\mathfrak{R}_T(\text{COBRA}, \boldsymbol{\sigma}) = \tilde{O}(Cd\sqrt{T} + \sqrt{NT})$ *for every* $\boldsymbol{\sigma} \in \text{NE}(\text{COBRA})$. *Hence,* $\max_{\sigma \in \text{NE(COBRA)}} \mathfrak{R}_T(\text{COBRA}, \boldsymbol{\sigma}) = \tilde{O}(Cd\sqrt{T} + \sqrt{NT})$.

Proof. Recall $\mathcal{A}_t$ denotes the set of arms reported by active agents in the round t, whose feature vectors are $\{x_{t,a}\}_{a \in \mathcal{A}_t}$. The regret of COBRA for $\boldsymbol{\sigma} \in \text{NE}(\text{COBRA})$ is given as follows:

$$\mathfrak{R}_T\left(\text{COBRA}, \boldsymbol{\sigma}\right) = \sum_{t=1}^{T} \left(\theta_\star^\top x_{t,a_t^\star}^\star - \theta_\star^\top x_{t,a_t}^\star \right) = \sum_{t=1}^{T} \left(\max_{a \in \mathcal{A}_t} \theta_\star^\top x_{t,a}^\star - \theta_\star^\top x_{t,a_t}^\star \right). \tag{18}$$

Under Assumption 1, if COBRA selects $a_t \in \mathcal{A}_t \colon a_t \neq a_t^\star$, we have $\theta_\star^\top x_{t,a_t^\star}^\star \leq \theta_\star^\top x_{t,a_t^\star} \leq \text{UCB}_t(x_{t,a_t^\star}) \leq \text{UCB}_t(x_{t,a_t})$. Using $\theta_\star^\top x_{t,a_t^\star}^\star \leq \text{UCB}_t(x_{t,a_t})$ inequality with Lemma 6, we have

$$\begin{aligned}
\mathfrak{R}_T\left(\text{COBRA}, \boldsymbol{\sigma}\right) &= \sum_{t=1}^{T} \left(\theta_\star^\top x_{t,a_t^\star}^\star - \theta_\star^\top x_{t,a_t}^\star \right) \\
&\leq \sum_{t=1}^{T} \left(\theta_\star^\top x_{t,a_t^\star} - \theta_\star^\top x_{t,a_t}^\star \right) \qquad \text{(agents are over-reporting)} \\
&\leq \sum_{t=1}^{T} \left(\text{UCB}_t(x_{t,a_t^\star}) - \theta_\star^\top x_{t,a_t}^\star \right) \qquad \text{(Assumption 1)} \\
&\leq \sum_{t=1}^{T} \left(\text{UCB}_t(x_{t,a_t}) - \theta_\star^\top x_{t,a_t}^\star \right) \qquad \text{(as selected arm is } a_t\text{)} \\
&\leq \sum_{t=1}^{T} \left(\text{UCB}_{t,-a_t}(x_{t,a_t}) - \theta_\star^\top x_{t,a_t}^\star \right) \qquad \text{(ensured by the Step 5 of COBRA)} \\
&= \sum_{t=1}^{T} \mathbb{1}(a_t = a) \left(\text{UCB}_{t,-a}(x_{t,a}) - \theta_\star^\top x_{t,a}^\star \right) \\
&= \sum_{a=1}^{N} \sum_{t \leq T, a_t = a} \left(\text{UCB}_{t,-a}(x_{t,a}) - \theta_\star^\top x_{t,a}^\star \right) \\
&\leq \sum_{a=1}^{N} \left(\sum_{t \leq T, a_t = a} 2\alpha_{t,-a} \left\| x_{t,a} \right\|_{V_{t,-a}^{-1}} + 2\sqrt{2R^2 S_t(a) \log(1/\delta_{t,a}^y)} \right) \qquad \text{(Lemma 6)} \\
&= \sum_{a=1}^{N} \sum_{t \leq T, a_t = a} 2\alpha_{t,-a} \left\| x_{t,a} \right\|_{V_{t,-a}^{-1}} + \sum_{a=1}^{N} 2\sqrt{2R^2 S_t(a) \log(1/\delta_{t,a}^y)} \\
&= \sum_{t=1}^{T} 2\alpha_{t,-a} \left\| x_{t,a_t} \right\|_{V_{t,-a}^{-1}} + \sum_{a=1}^{N} 2\sqrt{2R^2 S_t(a) \log(1/\delta_{t,a}^y)}. \qquad (19)
\end{aligned}$$

First, we will upper bound the first part of the above inequality, i.e., $\sum_{t=1}^{T} \alpha_{t,-a} \left\| x_{t,a_t} \right\|_{V_{t,-a}^{-1}}$, as follows:

$$\sum_{t=1}^{T} 2\alpha_{t,-a} \left\| x_{t,a_t} \right\|_{V_{t,-a}^{-1}} \leq 2\sqrt{T} \sqrt{\sum_{t=1}^{T} \left[\alpha_{t,-a} \left\| x_{t,a_t} \right\|_{V_{t,-a}^{-1}} \right]^2}$$

$$
\begin{aligned}
&\leq 2\sqrt{T}\sqrt{\sum_{t=1}^{T}\left[\alpha_T \left\|x_{t,a_t}\right\|_{V_{t,-a}^{-1}}\right]^2} \\
&= 2\sqrt{T}\sqrt{\alpha_T^2 \sum_{t=1}^{T}\left\|x_{t,a_t}\right\|_{V_{t,-a}^{-1}}^2} \\
&= 2\alpha_T\sqrt{T}\sqrt{\sum_{t=1}^{T}\left\|x_{t,a_t}\right\|_{V_{t,-a}^{-1}}^2} \\
&= 2\alpha_T\sqrt{T}\sqrt{\sum_{t=1}^{T}\left\|x_{t,a_t}\right\|_{V_{t,-a}^{-1}}^2 \frac{\left\|x_{t,a_t}\right\|_{V_t^{-1}}^2}{\left\|x_{t,a_t}\right\|_{V_t^{-1}}^2}} \\
&= 2\alpha_T\sqrt{T}\sqrt{\sum_{t=1}^{T}\left\|x_{t,a_t}\right\|_{V_t^{-1}}^2 \frac{\left\|x_{t,a_t}\right\|_{V_{t,-a}^{-1}}^2}{\left\|x_{t,a_t}\right\|_{V_t^{-1}}^2}} \\
&\leq 2\alpha_T C\sqrt{T}\sqrt{\sum_{t=1}^{T}\left\|x_{t,a_t}\right\|_{V_t^{-1}}^2} \\
&\leq 2\alpha_T C\sqrt{T}\sqrt{2\log\frac{\det(V_T)}{\det(\lambda I_d)}} \\
\implies \mathfrak{R}_T\left(\text{COBRA}, \boldsymbol{\sigma}\right) &\leq 2\alpha_T C\sqrt{2dT\log(\lambda + TL/d)} = \tilde{O}(Cd\sqrt{T}).
\end{aligned} \tag{20}
$$

The first inequality is due to using Cauchy-Schwarz inequality, where the second inequality follows from the fact that $\alpha_{t,-a}$ increases with t. The third inequality follows from Lemma 12 of Abbasi-Yadkori et al. (2011), by adapting it to our setting. The fourth inequality follows from the fact that there exists an universal constant C (Definition 5) such that $C \geq \max_a \frac{\left\|x_{t,a_t}\right\|_{V_{t,-a}^{-1}}}{\left\|x_{t,a_t}\right\|_{V_t^{-1}}}$ for all $t \geq 1$. The last two inequalities follow from Lemma 10 and Lemma 11 of Abbasi-Yadkori et al. (2011), respectively, and $\alpha_T = \tilde{O}(d\log T)$. For first part of Eq. (19), we have $\sum_{t=1}^{T} 2\alpha_{t,-a}\left\|x\right\|_{V_{t,-a}^{-1}} \leq 2\alpha_T\sqrt{2dT(1+C)}\log(\lambda + TL/d)$ from Eq. (20), and then using the Jensen's inequality for the second part with the fact that $\sum_{a=1}^{N} S_t(a) \leq T$. Then, we have

$$
\begin{aligned}
&\mathfrak{R}_T\left(\text{COBRA}, \boldsymbol{\sigma}\right) \leq 2\alpha_T C\sqrt{2dT\log(\lambda + TL/d)} + 2\sqrt{2R^2 NT\log(1/\delta_{t,a}^{y})} \\
\implies &\mathfrak{R}_T\left(\text{COBRA}, \boldsymbol{\sigma}\right) \leq \tilde{O}(Cd\sqrt{T} + \sqrt{NT}).
\end{aligned} \tag{21}
$$

We now prove that being truthful is an approximate Nash equilibrium for COBRA. Recall Eq. (13), $S_T(a)$ denotes the number of times an agent being selected by COBRA, which is given as follows:

$$
S_T(a) = \sum_{t=1}^{T}\mathbb{1}(a_t = a) \geq S_T^{\star}(a) - \sum_{t=1}^{T}\mathbb{1}(a_t \neq a_t^{\star}). \tag{22}
$$

To get the lower bound $S_T(a)$, we get the upper bound $\sum_{t=1}^{T}\mathbb{1}(a_t \neq a_t^{\star})$ when any agent can behave strategically. Recall $\Delta_{a_t} = \left(\theta_{\star}^{\top} x_{t,a_t^{\star}}^{\star} - \theta_{\star}^{\top} x_{t,a_t}^{\star}\right) \leq \text{UCB}_{t,-a}(x_{t,a}) - \theta_{\star}^{\top} x_{t,a}^{\star}$ for $a_t \neq a_t^{\star}$. We multiply and divide $\mathbb{1}(a_t \neq a_t^{\star})$ by Δ_{a_t} as follows:

$$
\begin{aligned}
\sum_{t=1}^{T}\mathbb{1}(a_t \neq a_t^{\star}) &= \sum_{t=1}^{T}\mathbb{1}(a_t \neq a_t^{\star})\frac{\Delta_{a_t}}{\Delta_{a_t}} \\
&= \sum_{k=1}^{N}\sum_{t\leq T, a_t = a}\mathbb{1}(a_t \neq a_t^{\star}, a_t = a)\frac{\Delta_{a_t}}{\Delta_{a_t}}
\end{aligned}
$$

$$\leq \sum_{k=1}^{N} \sum_{t \leq T, a_t = a} \mathbb{1}(a_t \neq a_t^\star, a_t = a) \frac{\mathrm{UCB}_{t,-a}(x_{t,a}) - \theta_\star^\top x_{t,a}^\star}{\Delta_{a_t}}$$

$$\leq \sum_{k=1}^{N} \sum_{t \leq T, a_t = a} \frac{\mathrm{UCB}_{t,-a} - \theta_\star^\top x_{t,a}^\star}{\Delta_{a_t}}$$

Assuming there exists a $\Delta_{\min}$ such that $\Delta_{\min} = \min_{a_t \neq a_t^\star} \Delta_{a_t}$, we get

$$\leq \frac{1}{\Delta_{\min}} \sum_{k=1}^{N} \sum_{t \leq T, a_t = a} \mathrm{UCB}_{t,-a}(x_{t,a}) - \theta_\star^\top x_{t,a}^\star$$

Using Eq. (19) with its upper bound, we have

$$\sum_{t=1}^{T} \mathbb{1}(a_t \neq a_t^\star) \leq \frac{2}{\Delta_{\min}} \left(2\alpha_T C \sqrt{2dT \log(\lambda + TL/d)} + 2\sqrt{2R^2 NT \log(1/\delta_{t,a}^y)} \right)$$

$$\implies \sum_{t=1}^{T} \mathbb{1}(a_t \neq a_t^\star) \leq \tilde{O}\left(Cd\sqrt{T} + \sqrt{NT} \right). \tag{23}$$

Using this upper bound in Eq. (22), we get the following bound for any agent $a \in \mathcal{A}$:

$$S_T(a) \geq S_T^\star(a) - \tilde{O}\left(Cd\sqrt{T} + \sqrt{NT} \right). \tag{24}$$

Now, we consider the case where an agent a deviates from the truthful strategy. Recall Eq. (16), the number of times an agent being selected by COBRA is given as follows:

$$S_T(a) = \sum_{t=1}^{T} \mathbb{1}(a_t = a) \leq \sum_{t=1}^{T} \mathbb{1}(a_t^\star = a) + \sum_{t=1}^{T} \mathbb{1}(a_t \neq a_t^\star). \tag{25}$$

Using Eq. (23) in Eq. (25), we get

$$S_T(a) \leq S_T^\star(a) + \tilde{O}\left(Cd\sqrt{T} + \sqrt{NT} \right). \tag{26}$$

Combining Eq. (24) and Eq. (26) completes our proof that COBRA is $\tilde{O}\left(Cd\sqrt{T} + \sqrt{NT} \right)$-NE. □

The constant C is used in the fourth inequality in Eq. (20), where $\sum_t \|x_{t,a_t}\|^2_{V_{t,-a}^{-1}}$ is bounded in terms of $\sum_t \|x_{t,a_t}\|^2_{V_t^{-1}}$. It then propagates into the ε-NE bound through Eq. (23). We keep C explicit at this step rather than absorbing it into $\tilde{O}(\cdot)$, making the dependence on the mechanism's blind spot transparent: as peer coverage deteriorates, the guarantee degrades continuously. In Example 2 of Sec. B.1, this dependence is tight in the sense that $V_{t,-2} = \lambda$ and $V_t = \lambda + t - 1$, yielding $C_{t,2} = \Theta(\sqrt{t})$ and hence $C = \Theta(\sqrt{T})$; the resulting bound degrades to the trivial $\tilde{O}(dT)$.

Advantage constant C in the linear setting. The following results hold for every Nash equilibrium $\boldsymbol{\sigma} \in \mathrm{NE}(\mathrm{COBRA})$, allowing multiple agents to deviate simultaneously. In this setting, converting the leave-one-out norm to the full-design norm introduces a factor that was previously absorbed into $\tilde{O}(\cdot)$. We instead make this dependence explicit through the advantage constants $C_{t,a}$ and C from Definition 5. In the linear setting considered here, $C_{t,a} = \sup_{x \neq 0} \frac{\|x\|_{Vt,-a^{-1}}}{\|x\|_{V_t^{-1}}}$ is the advantage of agent a at round t; any $C_a \geq Ct, a$ for all $t \geq 1$ is an advantage of agent a; and $C \geq \max_{a \in \mathcal{N}} C_a \geq 1$ is the maximum advantage.

Assume $G_{t,a} = \sum_{s<t,, a_s = a} x_{s,a_s} x_{s,a_s}^\top$, so that $V_t = V_{t,-a} + G_{t,a}$, the linear case admits the exact characterization

$$C_{t,a}^2 = \lambda_{\max}\left(V_{t,-a}^{-1} V_t \right) = 1 + \lambda_{\max}\left(V_{t,-a}^{-1/2} G_{t,a} V_{t,-a}^{-1/2} \right). \tag{27}$$

As $V_{t,-a} \succeq \lambda I$ and $\|x\| 2^2 \leq L$, we have $V_t \preceq (1 + S_t(a)L/\lambda)\, Vt, -a$, yielding the crude bound $C_{t,a} \leq \sqrt{1 + S_t(a)L/\lambda}$. Thus, C is large precisely when the directions explored by agent a are poorly covered by the remaining agents' observations, capturing the coverage condition underlying LOOM's detection power in Sec. 3. Obtaining a sharper theoretical bound on C is challenging because $G_{t,a}$ is policy-dependent, noise-dependent, and generally non-i.i.d. We therefore evaluate C directly in our experiments (Fig. 7e–7h for two over-reporting agents, and Fig. 8e–8h for $N \in \{100, \ldots, 500\}$ and $d \in \{60, \ldots, 100\}$). Across these settings, C grows slowly with T and remains nearly flat. These observations are empirical and do not establish a theoretical bound; deriving such a bound remains an open problem.

D Non-linear Reward Function

Table 1: Examples of different $h(x, \mathcal{O}_t)$ values for some LOOM-compatible contextual bandit algorithms, using notations from the original papers.

Contextual bandit algorithm	$h(x, \mathcal{O}_t)$
Lin-UCB (Chu et al., 2011)	$\left(R\sqrt{d\log\left(\frac{1+\frac{tL^2}{\lambda}}{\delta}\right)} + \lambda^{\frac{1}{2}} S\right) \|x\|_{V_t^{-1}}$
GLM-UCB (Li et al., 2017)	$\sqrt{\frac{d}{2}\log(1 + 2t/d) + \log(1/\delta)}\, \frac{\|x\|_{V_t^{-1}}}{\kappa}$
IGP-UCB (Chowdhury & Gopalan, 2017)	$\left(\sqrt{2(\gamma_{t-1} + 1 + \log(1/\delta))} + B\right) \sigma_{t-1}(x)$

D.1 Theoretical Results

We first derive results similar to Lemma 4 and Lemma 5 for contextual bandit problems with non-linear reward functions. For brevity, we ignore $\mathfrak{A}$ in $f_t^{\mathfrak{A}}$ and use only f_t.

Lemma 7. *Let $\mathfrak{A}$ be a LOOM-compatible contextual bandit algorithm for which $|f_t(x) - f(x)| \leq h(x, \mathcal{O}_t)$ holds with probability at least $1 - \delta$ for any $x \in \mathcal{X}$. Then, for all $t \geq 1$,*

1. *With probability at least $1 - \delta$,*

$$\forall x \in \mathcal{X} : f(x) \leq \mathrm{UCB}_t(x) = f_t(x) + h(x, \mathcal{O}_t).$$

2. *Similarly, with probability at least $1 - \delta$,*

$$\forall x \in \mathcal{X} : f(x) \geq \mathrm{LCB}_t(x) = f_t(x) - h(x, \mathcal{O}_t).$$

Proof. The proofs of these results follow directly from the first part of Definition 4. For completeness, we provide the proof of the first part, which follows from straightforward algebraic simplifications of $|f_t(x) - f(x)| \leq h(x, \mathcal{O}_t)$:

$$\begin{aligned} & |f_t(x) - f(x)| \leq h(x, \mathcal{O}_t) \\ \implies & |f(x) - f_t(x)| \leq h(x, \mathcal{O}_t) \\ \implies & f(x) - f_t(x) \leq h(x, \mathcal{O}_t) \\ \implies & f(x) \leq f_t(x) + h(x, \mathcal{O}_t). \end{aligned}$$

Similarly, the proof of the second part follows with some simple algebraic simplifications of $|f_t(x) - f(x)| \leq h(x, \mathcal{O}_t)$:

$$\begin{aligned} & |f_t(x) - f(x)| \leq h(x, \mathcal{O}_t) \\ \implies & f_t(x) - f(x) \leq h(x, \mathcal{O}_t) \\ \implies & f_t(x) - h(x, \mathcal{O}_t) \leq f(x) \\ \implies & f(x) \geq f_t(x) - h(x, \mathcal{O}_t). \end{aligned}$$

□

Lemma 8. *Let $\mathfrak{A}$ be a LOOM-compatible contextual bandit algorithm for which $|f_{t,-a}(x) - f(x)| \leq h(x, \mathcal{O}_{t,-a})$ holds with probability at least $1-\delta$ for any $x \in \mathcal{X}$. Then, for all $t \geq 1$,*

1. *With probability at least $1-\delta$,*

$$\forall x \in \mathcal{X} : f(x) \leq \mathrm{UCB}_{t,-a}(x) = f_{t,-a}(x) + h(x, \mathcal{O}_{t,-a}).$$

2. *Similarly, with probability at least $1-\delta$,*

$$\forall x \in \mathcal{X} : f(x) \geq \mathrm{LCB}_{t,-a}(x) = f_{t,-a}(x) - h(x, \mathcal{O}_{t,-a}).$$

Proof. The proofs of these results follow directly from the second part of Definition 4. For completeness, we provide the proof of the first part, which follows from straightforward algebraic simplifications of $|f_t(x) - f(x)| \leq h(x, \mathcal{O}_t)$:

$$\begin{aligned} & |f_{t,-a}(x) - f(x)| \leq h(x, \mathcal{O}_{t,-a}) \\ \implies & |f(x) - f_{t,-a}(x)| \leq h(x, \mathcal{O}_{t,-a}) \\ \implies & f(x) - f_{t,-a}(x) \leq h(x, \mathcal{O}_{t,-a}) \\ \implies & f(x) \leq f_{t,-a}(x) + h(x, \mathcal{O}_{t,-a}). \end{aligned}$$

Similarly, the proof of the second part follows with some simple algebraic simplifications of $|f_t(x) - f(x)| \leq h(x, \mathcal{O}_t)$:

$$\begin{aligned} & |f_{t,-a}(x) - f(x)| \leq h(x, \mathcal{O}_{t,-a}) \\ \implies & f_{t,-a}(x) - f(x) \leq h(x, \mathcal{O}_{t,-a}) \\ \implies & f_{t,-a}(x) - h(x, \mathcal{O}_{t,-a}) \leq f(x) \\ \implies & f(x) \geq f_{t,-a}(x) - h(x, \mathcal{O}_{t,-a}). \end{aligned}$$

□

Theorem 3 (Non-linear)**.** *Let $\mathfrak{A}$ be a LOOM-compatible contextual bandit algorithm used by* COBRA. *When agents report truthfully, being truthful is a $\tilde{O}(\tilde{d}\sqrt{T})$-*NE *under* COBRA. *With probability at least $1 - \delta^x_{t,a} - \delta^y_{t,a}$, the regret of* COBRA *under this approximate* NE *is* $\mathfrak{R}_T\left(\mathrm{COBRA}(\mathfrak{A}), \boldsymbol{\sigma}^\star\right) = \tilde{O}(\tilde{d}\sqrt{T})$.

Proof. When all agents report truthfully, our algorithm is the same as the contextual bandit algorithm $\mathfrak{A}$ with a mechanism for identifying strategic agents that holds with probability at least $1 - \delta_x - \delta_y$. For completeness, we first recall the definition of the regret of COBRA as follows:

$$\mathfrak{R}_T\left(\mathrm{COBRA}(\mathfrak{A}), \boldsymbol{\sigma}^\star\right) = \sum_{t=1}^{T} \left(f(x^\star_{t,a^\star_t}) - f(x^\star_{t,a_t}) \right). \tag{28}$$

Since the true feature vector is the same as the reported context-arm feature vector (i.e., $x^\star_{t,a} = x_{t,a}$), we can start by upper-bounding the difference $f(x^\star_{t,a^\star_t}) - f(x^\star_{t,a_t})$ as follows:

$$\begin{aligned} f(x^\star_{t,a^\star_t}) - f(x^\star_{t,a_t}) &\leq \mathrm{UCB}(x_{t,a^\star_t}) - f(x^\star_{t,a_t}) \\ &\leq \mathrm{UCB}(x_{t,a_t}) - f(x^\star_{t,a_t}) \qquad \left(\text{as } \mathrm{UCB}(x_{t,a^\star_t}) \leq \mathrm{UCB}(x_{t,a_t})\right) \\ &= f_t(x_{t,a_t}) + h(x_{t,a_t}, \mathcal{O}_t) - f(x^\star_{t,a_t}) \\ &\leq |f_t(x_{t,a_t}) - f(x^\star_{t,a_t})| + h(x_{t,a_t}, \mathcal{O}_t) \\ &\leq h(x_{t,a_t}, \mathcal{O}_t) + h(x_{t,a_t}, \mathcal{O}_t) \\ \implies f(x^\star_{t,a^\star_t}) - f(x^\star_{t,a_t}) &\leq 2h(x_{t,a_t}, \mathcal{O}_t). \end{aligned} \tag{29}$$

Note that f_t is an estimator of the reward function f as the true feature vector is the same as the reported context-arm feature vector. After using the upper bound given in Eq. (29) into Eq. (28), we get an upper bound on the regret as follows:

$$\mathfrak{R}_T\left(\mathrm{COBRA}(\mathfrak{A}), \boldsymbol{\sigma}^\star\right) = \sum_{t=1}^{T} \left(f(x^\star_{t,a^\star_t}) - f(x^\star_{t,a_t}) \right)$$

$$\leq \sum_{t=1}^{T} 2h(x_{t,a_t}, \mathcal{O}_t)$$
$$= 2\sum_{t=1}^{T} h(x_{t,a_t}, \mathcal{O}_t)$$
$$\leq 2\sum_{t=1}^{T} \sqrt{T\sum_{t=1}^{T} \left[h(x_{t,a_t}, \mathcal{O}_t)\right]^2}$$
$$\implies \mathfrak{R}_T\left(\text{COBRA}(\mathfrak{A}), \boldsymbol{\sigma}^\star\right) \leq 2\sqrt{T}\sqrt{\sum_{t=1}^{T} \left[h(x_{t,a_t}, \mathcal{O}_t)\right]^2} = \tilde{O}\left(\tilde{d}\sqrt{T}\right). \tag{30}$$

The first inequality directly follows from Eq. (29). The second inequality is due to using the Cauchy-Schwarz inequality. The last equality is due to $\sqrt{\sum_{t=1}^{T} \left[h(x_{t,a_t}, \mathcal{O}_t)\right]^2} = \tilde{O}(\tilde{d}\log T)$.

We now prove that being truthful is an approximate Nash equilibrium for COBRA. Recall, $S_T(a)$ denotes the number of times an agent being selected by COBRA, which is given as follows:

$$S_T(a) = \sum_{t=1}^{T} \mathbb{1}(a_t = a)$$
$$= \sum_{t=1}^{T} \mathbb{1}(a_t = a, a_t^\star = a) + \sum_{t=1}^{T} \mathbb{1}(a_t = a, a_t^\star \neq a)$$
$$\geq \sum_{t=1}^{T} \mathbb{1}(a_t^\star = a) - \sum_{t=1}^{T} \mathbb{1}(a_t^\star = a, a_t \neq a)$$
$$\geq \sum_{t=1}^{T} \mathbb{1}(a_t^\star = a) - \sum_{t=1}^{T} \mathbb{1}(a_t \neq a_t^\star)$$
$$\implies S_T(a) \geq S_T^\star(a) - \sum_{t=1}^{T} \mathbb{1}(a_t \neq a_t^\star). \tag{31}$$

To get the lower bound $S_T(a)$, we get an upper bound $\sum_{t=1}^{T} \mathbb{1}(a_t \neq a_t^\star)$. Let $\Delta_{a_t} = \left(f(x_{t,a_t^\star}^\star) - f(x_{t,a_t}^\star)\right) > 0$ for $a_t \neq a_t^\star$. We multiply and divide $\mathbb{1}(a_t \neq a_t^\star)$ by Δ_{a_t} and then use inequality in Eq. (29), i.e., $\Delta_{a_t} \leq 2h(x_{t,a_t}, \mathcal{O}_t)$ as follows:

$$\sum_{t=1}^{T} \mathbb{1}(a_t \neq a_t^\star) = \sum_{t=1}^{T} \mathbb{1}(a_t \neq a_t^\star)\frac{\Delta_{a_t}}{\Delta_{a_t}}$$
$$\leq \sum_{t=1}^{T} \mathbb{1}(a_t \neq a_t^\star)\frac{2h(x_{t,a_t}, \mathcal{O}_t)}{\Delta_{a_t}} \qquad (\text{as } x_{t,a}^\star = x_{t,a})$$
$$\leq \sum_{t=1}^{T} \frac{2h(x_{t,a_t}, \mathcal{O}_t)}{\Delta_{a_t}}$$
$$= \sum_{t=1}^{T} \frac{1}{\Delta_{a_t}} 2h(x_{t,a_t}, \mathcal{O}_t)$$
$$\leq \sum_{t=1}^{T} \frac{1}{\Delta_{\min}} 2h(x_{t,a_t}, \mathcal{O}_t)$$
$$= \frac{2}{\Delta_{\min}} \sum_{t=1}^{T} h(x_{t,a_t}, \mathcal{O}_t)$$

$$\leq \frac{2}{\Delta_{\min}}\sqrt{T}\sqrt{\sum_{t=1}^{T}[h(x_{t,a_t},\mathcal{O}_t)]^2}$$

$$\implies \sum_{t=1}^{T}\mathbb{1}(a_t \neq a_t^\star) \leq \tilde{O}\left(\tilde{d}\sqrt{T}\right). \tag{32}$$

Note that $\Delta_{\min} = \min_{a_t \neq a_t^\star} \Delta_{a_t}$. Although using $\Delta_{\min}$ loosens the upper bound, we use this to get dependence on T. Using this upper bound in Eq. (31), we get the following bound for any agent $a \in \mathcal{A}$:

$$S_T(a) \geq S_T^\star(a) - \tilde{O}\left(\tilde{d}\sqrt{T}\right). \tag{33}$$

Now we consider the case where an agent a deviates from the truthful strategy. The number of times an agent is selected by COBRA is given as follows:

$$\begin{aligned}
S_T(a) &= \sum_{t=1}^{T}\mathbb{1}(a_t = a) \\
&= \sum_{t=1}^{T}\mathbb{1}(a_t = a, a_t^\star = a) + \sum_{t=1}^{T}\mathbb{1}(a_t = a, a_t^\star \neq a) \\
&\leq \sum_{t=1}^{T}\mathbb{1}(a_t^\star = a) + \sum_{t=1}^{T}\mathbb{1}(a_t = a, a_t^\star \neq a) \\
&\leq \sum_{t=1}^{T}\mathbb{1}(a_t^\star = a) + \sum_{t=1}^{T}\mathbb{1}(a_t \neq a_t^\star).
\end{aligned} \tag{34}$$

Using Eq. (32) in Eq. (34), we get

$$S_T(a) \leq S_T^\star(a) + \tilde{O}\left(\tilde{d}\sqrt{T}\right). \tag{35}$$

Combining Eq. (33) and Eq. (35) completes our proof that COBRA is $\tilde{O}(\tilde{d}\sqrt{T})$-NE. □

The proof of Theorem 3 relies entirely on $h(x, \mathcal{O}_t)$ and, like the proof of Theorem 2, never compares it with $h(x, \mathcal{O}_{t,-a})$. Thus, Theorem 1, Theorem 2, and Theorem 3, which establish the soundness of the LOOM test and both incentive-compatibility guarantees, are independent of the advantage constant C in Definition 5. We need the following result that upper bound the total amount of regret that an agent a can exert before being identified by LOOM.

Lemma 9. *Let* $\mathrm{UCB}_{t,-a}(x_{s,a}) = f_{t,-a}(x_{s,a}) + h(x_{s,a}, \mathcal{O}_{t,-a})$*, where* $x_{s,a}$ *is the arm feature vector associated with agent* a *in the round* s*. Then, with probability at least* $1 - \delta_{t,a}^x - \delta_{t,a}^y$*,*

$$\sum_{s\leq t:\ a_s=a}\left(\mathrm{UCB}_{t,-a}(x_{s,a}) - f(x_{s,a}^\star)\right) \leq \sum_{s\leq t, a_s=a} 2h(x_{s,a},\mathcal{O}_{t,-a}) + 2\sqrt{2R^2 S_t(a)\log(1/\delta_{t,a}^y)}.$$

Proof. Using Eq. (9) with Lemma 5 for non-linear reward function that holds with probability at least $1 - \delta_{t,a}^x$, we get:

$$\begin{aligned}
\sum_{s\leq t, a_s=a}(f_{t,-a}(x_{s,a}) + h(x_{s,a},\mathcal{O}_{t,-a})) &\leq \sum_{s\leq t, a_s=a} y_s + \sqrt{2R^2 S_t(a)\log(1/\delta_{t,a}^y)} \\
&\leq \sum_{s\leq t, a_s=a} f(x_{s,a}^\star) + \sqrt{2R^2 S_t(a)\log(1/\delta_{t,a}^y)} \\
&\quad + \sqrt{2R^2 S_t(a)\log(1/\delta_{t,a}^y)} \\
\implies \sum_{s\leq t, a_s=a}\left(f_{t,-a}(x_{s,a}) - f(x_{s,a}^\star)\right) &\leq \sum_{s\leq t, a_s=a} h(x_{s,a},\mathcal{O}_{t,-a}) + 2\sqrt{2R^2 S_t(a)\log(1/\delta_{t,a}^y)}.
\end{aligned}$$

The second inequality follows from Eq. (7) by using an upper bound on $\sum_{s\le t,a_s=a} y_s$ that holds with probability $1-\delta^y_{t,a}$. Now we prove the second part of the result by replacing $f_{t,-a}(x_{s,a})$ by $\mathrm{UCB}_{t,-a}(x_{s,a}) - h(x_{s,a},\mathcal{O}_{t,-a})$ and we get

$$\begin{aligned}
&\sum_{s\le t,a_s=a}\left(\mathrm{UCB}_{t,-a}(x_{s,a}) - h(x,\mathcal{O}_{t,-a}) - f(x^\star_{s,a_s})\right)\\
&\qquad\le \sum_{s\le t,a_s=a} h(x_{s,a},\mathcal{O}_{t,-a}) + 2\sqrt{2R^2S_t(a)\log(1/\delta^y_{t,a})}\\
\implies &\sum_{s\le t,a_s=a}\left(\mathrm{UCB}_{t,-a}(x_{s,a}) - f(x^\star_{s,a_s})\right)\\
&\qquad\le \sum_{s\le t,a_s=a} 2h(x_{s,a},\mathcal{O}_{t,-a}) + 2\sqrt{2R^2S_t(a)\log(1/\delta^y_{t,a})}. \qquad \square
\end{aligned}$$

Theorem 5 (Non-linear)**.** *Let $\mathfrak{A}$ be a LOOM-compatible contextual bandit algorithm used by* COBRA*. If Assumption 1 hold then, with C as in Definition 5, the regret of* COBRA *is $\mathfrak{R}_T(\mathrm{COBRA}(\mathfrak{A}),\boldsymbol{\sigma}) = \tilde{O}(C\tilde{d}\sqrt{T}+\sqrt{NT})$ for every $\boldsymbol{\sigma}\in\mathrm{NE}(\mathrm{COBRA}(\mathfrak{A}))$. Hence, $\max_{\sigma\in\mathrm{NE}(\mathrm{COBRA}(\mathfrak{A}))}\mathfrak{R}_T(\mathrm{COBRA}(\mathfrak{A}),\boldsymbol{\sigma}) = \tilde{O}(C\tilde{d}\sqrt{T}+\sqrt{NT})$.*

Proof. Recall $\mathcal{A}_t$ denotes the set of arms' features corresponding to the active agents in the round t. The regret of COBRA for $\boldsymbol{\sigma}\in\mathrm{NE}(\mathrm{COBRA}(\mathfrak{A}))$ is given as follows:

$$\mathfrak{R}_T\left(\mathrm{COBRA}(\mathfrak{A}),\boldsymbol{\sigma}\right) = \sum_{t=1}^{T}\left(f(x^\star_{t,a^\star_t}) - f(x^\star_{t,a_t})\right) = \sum_{t=1}^{T}\left(\max_{a\in\mathcal{A}_t} f(x^\star_{t,a}) - f(x^\star_{t,a_t})\right). \tag{36}$$

Under Assumption 1, if COBRA selects $a_t\in\mathcal{A}_t\colon a_t\neq a^\star_t$, we have $f(x^\star_{t,a^\star_t})\le f(x_{t,a^\star_t})\le \mathrm{UCB}_t(x_{t,a^\star_t})\le \mathrm{UCB}_t(x_{t,a_t})$. Using $f(x^\star_{t,a^\star_t})\le \mathrm{UCB}_t(x_{t,a_t})$ inequality with Lemma 9, we have

$$\begin{aligned}
\mathfrak{R}_T\left(\mathrm{COBRA}(\mathfrak{A}),\boldsymbol{\sigma}\right) &= \sum_{t=1}^{T}\left(f(x^\star_{t,a^\star_t}) - f(x^\star_{t,a_t})\right)\\
&\le \sum_{t=1}^{T}\left(f(x_{t,a^\star_t}) - f(x^\star_{t,a_t})\right) \qquad \text{(agents are over-reporting)}\\
&\le \sum_{t=1}^{T}\left(\mathrm{UCB}_t(x_{t,a^\star_t}) - f(x^\star_{t,a_t})\right) \qquad \text{(Assumption 1)}\\
&\le \sum_{t=1}^{T}\left(\mathrm{UCB}_t(x_{t,a_t}) - f(x^\star_{t,a_t})\right) \qquad \text{(as selected arm is } a_t)\\
&\le \sum_{t=1}^{T}\left(\mathrm{UCB}_{t,-a}(x_{t,a_t}) - f(x^\star_{t,a_t})\right) \qquad \text{(ensured by the Step 5 of COBRA)}\\
&= \sum_{t=1}^{T}\mathbb{1}(a_t=a)\left(\mathrm{UCB}_{t,-a}(x_{t,a}) - f(x^\star_{t,a})\right)\\
&= \sum_{a=1}^{N}\sum_{t\le T,a_t=a}\left(\mathrm{UCB}_{t,-a}(x_{t,a}) - f(x^\star_{t,a})\right)\\
&\le \sum_{a=1}^{N}\left(\sum_{t\le T,a_t=a} 2h(x_{t,a},\mathcal{O}_{t,-a}) + 2\sqrt{2R^2S_t(a)\log(1/\delta^y_{t,a})}\right)\\
&= \sum_{a=1}^{N}\sum_{t\le T,a_t=a} 2h(x_{t,a},\mathcal{O}_{t,-a}) + \sum_{a=1}^{N} 2\sqrt{2R^2S_t(a)\log(1/\delta^y_{t,a})}
\end{aligned}$$

$$
\begin{aligned}
&= \sum_{t=1}^{T} 2h(x_{t,a_t}, \mathcal{O}_{t,-a}) + \sum_{a=1}^{N} 2\sqrt{2R^2 S_t(a)\log(1/\delta_{t,a}^y)} \\
&\leq 2\sqrt{T}\sqrt{\sum_{t=1}^{T}\left[h(x_{t,a_t}, \mathcal{O}_{t,-a})\right]^2} + 2\sum_{a=1}^{N}\sqrt{2R^2 S_t(a)\log(1/\delta_{t,a}^y)} \\
&\leq 2\sqrt{T}\sqrt{\sum_{t=1}^{T}\left[h(x_{t,a_t}, \mathcal{O}_{t,-a})\right]^2} + 2\sqrt{2R^2 NT\log(1/\delta_{t,a}^y)} \\
&\leq 2C\sqrt{T}\sqrt{\sum_{t=1}^{T}\left[h(x_{t,a_t}, \mathcal{O}_{t,a})\right]^2} + 2\sqrt{2R^2 NT\log(1/\delta_{t,a}^y)} \\
\implies \mathfrak{R}_T\left(\text{COBRA}(\mathfrak{A}), \boldsymbol{\sigma}\right) &= \tilde{O}\left(C\tilde{d}\sqrt{T} + \sqrt{NT}\right). \qquad (37)
\end{aligned}
$$

The third-last inequality follows from Lemma 9. The second-last inequality is due to using Cauchy-Schwarz inequality, where the last inequality follows from Jensen's inequality with the fact that $\sum_{a=1}^{N} S_t(a) \leq T$. The last inequality is the non-linear counterpart of the fourth inequality in Eq. (20) and is the *only* step in the proof that compares the leave-one-out confidence width with the full-design width. It therefore introduces the advantage constant C from Definition 5. Specifically, for all a and $t \geq 1$, $C \geq \sup_{x\in\mathcal{X}} \frac{h(x,\mathcal{O}_{t,-a})}{h(x,\mathcal{O}_t)}$, which implies $\sum_t \left[h(x_{t,a_t}, \mathcal{O}_{t,-a})\right]^2 \leq C^2 \sum_t \left[h(x_{t,a_t}, \mathcal{O}_t)\right]^2$. As in the linear case, we keep this dependence on C explicit in the final regret bound.

We now prove that being truthful is an approximate Nash equilibrium for COBRA. Recall Eq. (31), $S_T(a)$ denotes the number of times an agent being selected by COBRA, which is given as follows:

$$
S_T(a) = \sum_{t=1}^{T} \mathbb{1}(a_t = a) \geq S_T^\star(a) - \sum_{t=1}^{T} \mathbb{1}(a_t \neq a_t^\star). \qquad (38)
$$

To get the lower bound $S_T(a)$, we get the upper bound $\sum_{t=1}^{T} \mathbb{1}(a_t \neq a_t^\star)$ when any agent can behave strategically. Recall $\Delta_{a_t} = \left(f(x_{t,a_t^\star}^\star) - f(x_{t,a_t}^\star)\right) \leq \text{UCB}_{t,-a}(x_{t,a}) - f(x_{t,a}^\star)$ for $a_t \neq a_t^\star$. We multiply and divide $\mathbb{1}(a_t \neq a_t^\star)$ by Δ_{a_t} as follows:

$$
\begin{aligned}
\sum_{t=1}^{T} \mathbb{1}(a_t \neq a_t^\star) &= \sum_{t=1}^{T} \mathbb{1}(a_t \neq a_t^\star)\frac{\Delta_{a_t}}{\Delta_{a_t}} \\
&= \sum_{k=1}^{N}\sum_{t\leq T, a_t=a} \mathbb{1}(a_t \neq a_t^\star, a_t = a)\frac{\Delta_{a_t}}{\Delta_{a_t}} \\
&\leq \sum_{k=1}^{N}\sum_{t\leq T, a_t=a} \mathbb{1}(a_t \neq a_t^\star, a_t = a)\frac{\text{UCB}_{t,-a}(x_{t,a}) - f(x_{t,a}^\star)}{\Delta_{a_t}} \\
&\leq \sum_{k=1}^{N}\sum_{t\leq T, a_t=a} \frac{\text{UCB}_{t,-a} - f(x_{t,a}^\star)}{\Delta_{a_t}}
\end{aligned}
$$

Assuming there exists a $\Delta_{\min}$ such that $\Delta_{\min} = \min_{a_t \neq a_t^\star} \Delta_{a_t}$, we get

$$
\leq \frac{1}{\Delta_{\min}} \sum_{k=1}^{N}\sum_{t\leq T, a_t=a} \text{UCB}_{t,-a}(x_{t,a}) - f(x_{t,a}^\star)
$$

Using Eq. (37) with its upper bound, we have

$$
\sum_{t=1}^{T} \mathbb{1}(a_t \neq a_t^\star) \leq \frac{2}{\Delta_{\min}}\left(2\sqrt{T}\sqrt{\sum_{t=1}^{T}\left[h(x_{t,a_t}, \mathcal{O}_{t,-a})\right]^2} + 2\sqrt{2R^2 NT\log(1/\delta_{t,a}^y)}\right) \qquad (39)
$$

$$\implies \sum_{t=1}^{T} \mathbb{1}(a_t \neq a_t^\star) = \tilde{O}\left(Cd\sqrt{T} + \sqrt{NT}\right).$$

Using this upper bound in Eq. (38), we get the following bound for any agent $a \in \mathcal{A}$:

$$S_T(a) \geq S_T^\star(a) - \tilde{O}\left(C\tilde{d}\sqrt{T} + \sqrt{NT}\right). \tag{40}$$

Now, we consider the case where an agent a deviates from the truthful strategy. Recall Eq. (34), the number of times an agent being selected by COBRA is given as follows:

$$S_T(a) = \sum_{t=1}^{T} \mathbb{1}(a_t = a) \leq \sum_{t=1}^{T} \mathbb{1}(a_t^\star = a) + \sum_{t=1}^{T} \mathbb{1}(a_t \neq a_t^\star). \tag{41}$$

Using Eq. (39) in Eq. (41), we get

$$S_T(a) \leq S_T^\star(a) + \tilde{O}\left(C\tilde{d}\sqrt{T} + \sqrt{NT}\right). \tag{42}$$

Combining Eq. (40) and Eq. (42) completes our proof that COBRA is $\tilde{O}\left(C\tilde{d}\sqrt{T} + \sqrt{NT}\right)$-NE. □

Remark 1. *A strategic agent may attempt to introduce a small but persistent misreport that remains within LOOM's confidence bounds over finite horizons. However, any systematic bias accumulates linearly in the LOOM test statistic, so to avoid elimination with high probability, the misreport magnitude must decay as $O(1/\sqrt{S_t(a)})$. This implies that the total additional utility gained through such deviations is at most $\tilde{O}(\sqrt{T})$. Consequently, although arbitrarily small misreports cannot be statistically ruled out, they yield vanishing per-round gains, preserving approximate BIC.*

In the non-linear case, the constant C enters at the last inequality of Eq. (37), where $\sum_t [h(x_{t,a_t}, \mathcal{O}_{t,-a})]^2$ is bounded in terms of $\sum_t [h(x_{t,a_t}, \mathcal{O}_t)]^2$. This is the direct non-linear analog of the corresponding step in the linear analysis above. In summary, C is needed only for the equilibrium regret guarantees Theorem 4 and Theorem 5. It does not enter Theorem 1, which establishes soundness of the test, or Theorem 2 and Theorem 3, which establish incentive compatibility and regret under truthful reporting. Their proofs depend entirely on V_t in the linear case and on $h(x, \mathcal{O}_t)$ in the non-linear case, without comparing either quantity to its leave-one-out counterpart.

E Additional Experiments and Details

This section provides additional details from Sec. 5, followed by further experimental results.

TS-based algorithm. Motivated by the empirical advantages of Thompson Sampling (TS) over UCB-based bandit algorithms (Zhang et al., 2021; Chapelle & Li, 2011; Agrawal & Goyal, 2013), we also propose a TS-based variant, COBRA (TS). This algorithm closely mirrors COBRA (UCB), differing only in the arm selection step (Line 4). To obtain a TS-based reward estimate, the algorithm first samples a reward function parameter $\tilde{\theta}_t \sim \mathbb{N}\left(\hat{\theta}_t, \beta_t^2 V_t^{-1}\right)$, where $\mathbb{N}$ denotes the normal distribution and $\beta_t = R\sqrt{9d\log(t/\delta)}$ (Agrawal & Goyal, 2013). Using $\tilde{\theta}_t$, the TS-based reward estimate, i.e., $\text{TS}_t(x_{t,a}) = x_{t,a}^\top \tilde{\theta}_t$, replaces $\text{UCB}_t(x_{t,a})$ when computing the optimistic reward in Line 4. We compare and demonstrate the superior empirical performance of COBRA (TS) in Sec. 5.

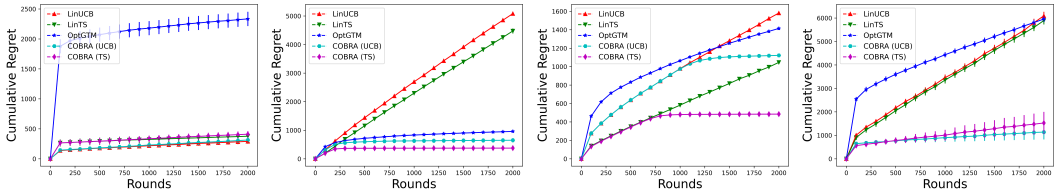
Why synthetic problem instances. Evaluating whether COBRA actually incentivizes truthful reporting requires simulating strategic behavior, including access to each agent's true features $x_{t,a}^\star$ and the ability to choose a misreport $x_{t,a}$ in response to the deployed algorithm. To the best of our knowledge, no publicly available real-world dataset contains the counterfactual information required for this evaluation, namely, both the true and reported features of a strategic agent. Synthetic instances, which are standard in the bandit literature, therefore provide a controlled setting for isolating and validating our theoretical predictions. We evaluate regret against Lin-UCB, Lin-TS, and OptGTM (Fig. 3), the effect of manipulation strength (Fig. 4), scaling with the number of agents N and dimension d (Fig. 6 and Fig. 8), two simultaneously strategic agents and the corresponding measured advantage C (Fig. 7), and non-linear reward functions (Fig. 5).

Strategic over-reporting. We consider two types of strategic manipulations: *(I) Feature adaptation:* The strategic agent updates the arm features it reports based on past selection outcomes using a finite-difference stochastic gradient ascent update. Specifically, the agent receives a binary feedback signal: 1 if it was selected in the previous round and 0 otherwise. The agent uses this feedback to iteratively adjust its reported features to increase the probability of being selected in future rounds. *(II) Systematic over-reporting:* To maximize the likelihood of selection, the strategic agent over-reports its feature vector according to $x = (1 + \Delta_x)x^\star$, where $x^\star$ denotes the true arm features. The agent maintains an estimate of the optimal over-reporting factor, $\hat{\Delta}_x$, which guides the extent of over-reporting.

Agents' observation space. Each agent observes only a binary signal indicating whether it was selected in the previous round, together with its share of the realized reward (e.g., an e-commerce platform may retain a fee before passing the remaining revenue to the agent). An agent does *not* observe other agents' reports, the learner's estimates f_t or $f_{t,-a}$, or the underlying reward function f. This is precisely the information available to the strategic agents in our experiments: the type I agent in Sec. 5 adapts its reports using finite-difference stochastic gradient ascent based solely on the binary selection signal (see Sec. E). Importantly, f is unknown to both the learner and the agents and is not required as part of the agents' information. Hence, our guarantees apply even to agents with strictly less information, including the type *I* agent described above.

Strategic manipulations via feature adaptation. We want to highlight that our proposed algorithm, COBRA, operates without prior knowledge of the specific nature of these manipulations, which we model as equivalent to over-reporting. Under the assumption that the agent engages only in strategic over-reporting, the objective is to identify any such over-reporting behavior. For feature adaptation, the agent can strategically manipulate and optimize its features against the deployed algorithm using only binary feedback (whether it was selected or not) in each round as follows: Assume $x_n^\star$ be the true arm-features of agent n. The agent can over-report a feature $\tilde{x} = x_n^\star + \eta\Delta$, where $\Delta \in \mathbb{R}^d$ is a bounded perturbation such that $||\Delta||_2 \leq \Delta_{\max}$ and η is the learning rate. The agent's goal is to learn a manipulation strategy Δ using binary feedback that increases its probability of being selected. To do so, the agent uses a finite-difference stochastic gradient ascent update.

Experiments with non-linear reward. We also compare the performance of our proposed algorithm, COBRA, for contextual bandit problems with non-linear reward functions. For this experiment, we adapt problem instances with non-linear reward functions from those used for linear functions in Sec. 5. We apply a polynomial kernel of degree 2 to transform the item-agent feature vectors to introduce non-linearity. The constant terms (i.e., the 1's) resulting from this transformation are removed. As an example, a sample 4-d feature vector $x = (x_1, x_2, x_3, x_4)$ is transformed into a 14-d feature vector: $x' = (x_1, x_2, x_3, x_4, x_1x_2, x_1x_3, x_1x_4, x_2x_3, x_2x_4, x_3x_4, x_1x_2x_3, x_1x_2x_4, x_1x_3x_4, x_2x_3x_4)$. We also remove 1's, which appear in the transformed samples. As expected, our algorithms COBRA based on UCB and TS-based contextual nonlinear bandit algorithms (prefixed with 'n') outperform all the baselines (adapted to non-linear setting, also prefixed with 'n') as shown in Fig. 5. These results are observed across various problem instances, where only the reward function is varied while all other parameters remain unchanged, except for the number of rounds, which is set to $T = 2000$. We further observe that COBRA with TS outperforms its UCB-based counterpart.

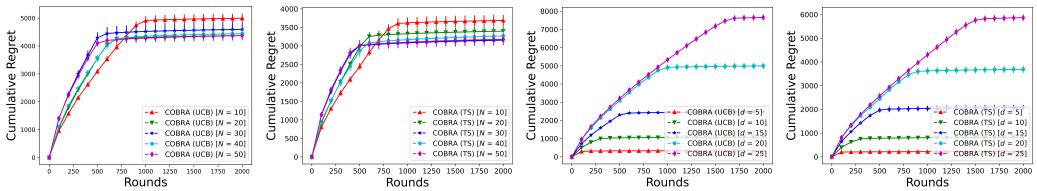


(a) Linear (Truthful setting) (b) Linear (Agent type I) (c) Linear (Agent type II) (d) Square (Agent type II)

Figure 5: Comparing the cumulative regret of COBRA with different baselines for problem instances with non-linear reward functions.

Regret of COBRA vs. number of agents (N) and dimension (d). The number of agents (N) and the dimension of the context-agent feature vector ($2d$) in the contextual bandit problem control the difficulty. As their values increase, the problem becomes more difficult, making it harder to allocate the context to the best agent. We want to verify this by observing how the regret of our proposed

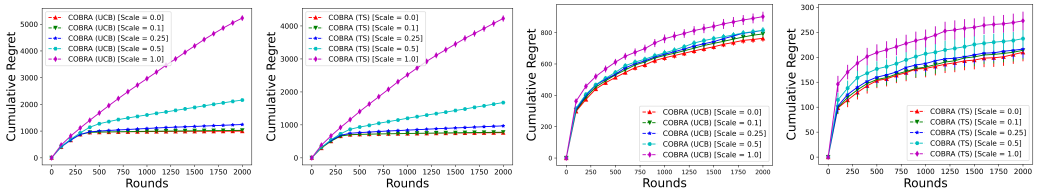
algorithms changes while varying N and d in the contextual bandit problem. To see this in our experiments, we use the linear reward function (i.e., $f(x) = 5x^\top \theta_\star$), 2000 contexts, $N = 10$ when varying dimension, $d = 20$ while varying the number of agents. As shown in Fig. 6a and Fig. 6b, the regret bound of our COBRA UCB- and TS- based algorithms increases as we increase the number of agents, i.e., $N = \{10, 20, 30, 40, 50\}$. We also observe the same trend when we increase the dimension of the context-agent feature vector from $d = \{5, 10, 15, 20, 25\}$ as shown in Fig. 6c and Fig. 6d. In all experiments, we also observe that the COBRA TS-based algorithm performs better than its COBRA UCB-based counterpart (as seen in Fig. 6a-6d by comparing the regret of both algorithms).



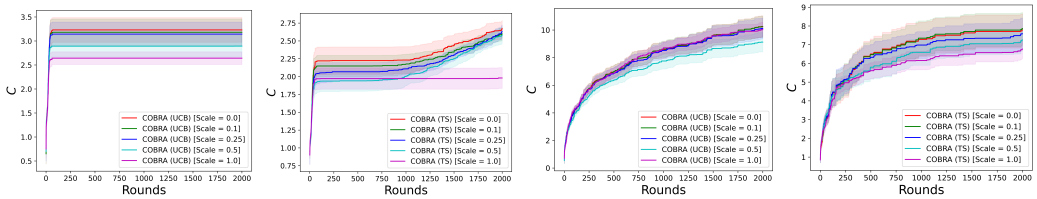
(a) Vary agents/arms (UCB) (b) Vary agents/arms (TS) (c) Vary dimension (UCB) (d) Vary dimension (TS)

Figure 6: Cumulative regret of COBRA vs. different values of N and d.

Experiments with two strategic agents. We also conducted additional experiments involving two type II strategic agents. In these experiments, we control the degree of over-reporting for one of the two strategic agents as $\hat{\Delta}_x \sim \mathbb{N}(\text{Scale} * \Delta_x^\star, \sigma_\Delta^2)$. The results (Fig. 7a–7d) show that as the controlled over-reporting of the agent increases, the regret of COBRA also increases. The higher regret is because the strategic behavior becomes harder to detect, and the over-reporting biases the estimator used in arm selection. For the same settings, we also plot the maximum value of C (the maximum advantage of any agent, Definition 5) under different levels of over-reporting for the two strategic agents. As expected, C grows very slowly with the number of rounds and remains almost constant, as shown in Fig. 7e–7h. This is the constant carried explicitly in Theorem 4 and Theorem 5. These measurements provide empirical evidence that C remains small in the homogeneous setting, but do not constitute a theoretical bound.



(a) Vary agents/arms (UCB) (b) Vary agents/arms (TS) (c) Fix agents/arms (UCB) (d) Fix agents/arms (TS)

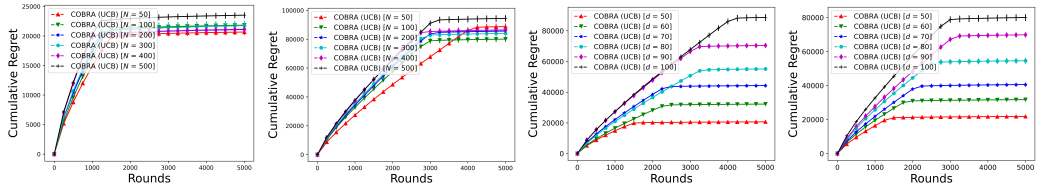


(e) Vary agents/arms (UCB) (f) Vary agents/arms (TS) (g) Fix agents/arms (UCB) (h) Fix agents/arms (TS)

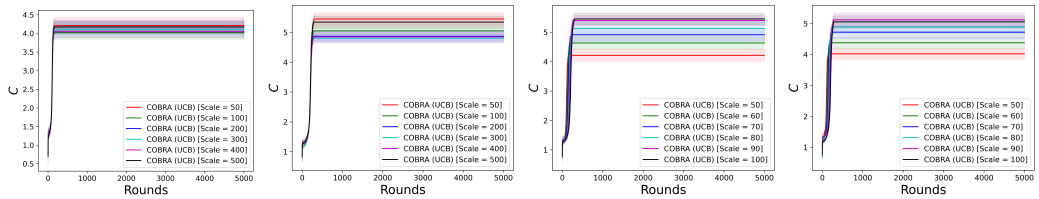
Figure 7: **Top row:** Cumulative regret of COBRA vs. different level of two over-reporting agents. **Bottom row:** Maximum value of C vs. different level of two over-reporting agents.

Regret of COBRA vs. large number of agents (N) and dimension (d). We conducted additional experiments to evaluate the performance of our LOOM-based mechanism in higher-dimensional settings and with larger numbers of strategic agents. Specifically, we considered the following scales: number of agents $N \in \{100, 200, 300, 400, 500\}$ and feature dimension $d \in \{60, 70, 80, 90, 100\}$. Note that the overall dimension of the context-arm feature is $2d$. When varying the number of agents, we fixed the feature dimension to 50 and 100. As expected, the regret of COBRA remains sub-linear as the number of agents increases (Fig. 8a-8b). Similarly, when varying the feature dimension, we

set the number of agents to 50 and 100. As anticipated, the regret of COBRA increases with the dimension, since higher-dimensional problems require substantially more samples to estimate the reward function accurately (Fig. 8c-8d). These results demonstrate the robustness of our theoretical guarantees, even in complex, large-scale environments. For the same settings, we also plot the maximum value of C under different N and d. As expected, C grows very slowly with the number of rounds and remains almost constant, as shown in Fig. 8e–8h.



(a) Vary agents/arms (50) (b) Vary agents/arms (100) (c) Vary dimension (50) (d) Vary dimension (100)



(e) Vary agents/arms (50) (f) Vary agents/arms (100) (g) Vary dimension (50) (h) Vary dimension (100)

Figure 8: **Top row:** Cumulative regret of COBRA vs. different values of N and d. **Bottom row:** Maximum value of C vs. different values of N and d. (A) in captions implies either $N = A$ or $d = A$.

Computational resources. All the experiments are run on a Apple M3 Pro with 18GB memory.

Time and space complexity of COBRA. The computational complexity of COBRA is comparable to that of standard contextual bandit algorithms, and it scales efficiently even when agents control multiple arms, as discussed below:
Time complexity: The overall time complexity is dominated by the underlying estimation procedure, which is identical to that of standard contextual bandit algorithms. LOOM requires computing a Leave-One-Out (LOO) estimator for each agent. These LOO estimators are inherently parallelizable, i.e., one can independently estimate each agent-specific estimator (trained on data excluding agent i). This parallelism substantially alleviates computational burden, although the total computational resources required still scale linearly with the number of agents.
Space complexity: The space complexity is likewise similar to standard contextual bandit approaches, with the main additional overhead arising from maintaining N separate LOO estimators.

F Frequently Asked Questions

Question 1. How sensitive is LOOM to the number of agents? What happens when only a few agents are active?

Answer. LOOM uses data contributed by other agents to construct reliable leave-one-out estimates, and its detection power improves as the number of truthful agents increases. When only a small number of agents are active, excluding a single agent removes a substantial portion of the available data, resulting in wider confidence intervals and more conservative behavior. In this regime, LOOM may delay agent elimination but does not introduce false positives. Consequently, COBRA smoothly degrades to standard contextual bandit behavior, and strategic agents can obtain at most a limited and transient advantage. LOOM is therefore most effective in the multi-agent regimes that motivate our study.

Question 2. How does LOOM behave when two agents collude or misreport symmetrically?

Answer. LOOM detects unilateral over-reporting through leave-one-out comparisons. If two agents collude or misreport symmetrically, each agent's reference set may include the other's biased data, potentially delaying or preventing detection; this reflects a fundamental limitation of peer-based mechanisms. Nevertheless, such collusion does not yield cumulative gains exceeding $\tilde{O}(\sqrt{T})$, and the learner's regret remains sublinear under our assumptions. Addressing more general forms of collusion would require additional structural assumptions (e.g., a minimum fraction of truthful agents), which we leave for future work.

Question 3. Can the LOOM idea be extended to heterogeneous agents without requiring a shared reward structure?

Answer. Our LOOM mechanism relies on a shared reward structure, since the leave-one-out estimator uses observations from other agents to estimate the target agent's reward. In heterogeneous settings, this transferability may no longer hold. However, the core idea can potentially be extended under additional structure, such as clustered agent types, similarity-based weighting, or shared latent representations. More generally, LOOM mainly requires sufficient coverage of an agent's feature distribution by other agents rather than exact reward homogeneity. Extending LOOM to fully heterogeneous settings is an important direction for future work. We also note that the conditions that create an incentive to misreport often coincide with those that provide peer coverage: as illustrated in Sec. B.1, a seller who is the sole supplier of a product category has no competitor to displace and hence little scope to benefit from misreporting, whereas sellers of similar products, whose features are drawn from similar distributions, provide the peer coverage that LOOM can exploit.

Question 4. Why not exploit user feedback directly to identify strategic agents, e.g., ask the user whether a recommended restaurant actually had the vegetarian menu it advertised?

Answer. LOOM already does this, but at the *reward* level rather than the attribute level. The optimistic side of the LOOM test, $\mathrm{UCB}_{t,a}^{(y)} = \sum_{s \le t,\, a_s = a} y_s + \sqrt{2S_t(a)\log(1/\delta_{t,a}^{y})}$ (see Lemma 1), is constructed entirely from the realized user feedback collected when agent a is selected. Because rewards depend only on the true features (Footnote 2), an agent that over-reports a feature can receive realized feedback that is inconsistent with its report. The LOOM condition Eq. (2) therefore compares what the agent's reports promise, through the peer-based pessimistic estimate, with what its realized feedback delivers. What LOOM does not use is attribute-level verification, such as asking the user whether a specific declared feature was accurate. This is a promising direction for heterogeneous settings because attribute-level feedback would not require cross-agent comparability and hence would not be subject to the coverage limitation captured by C in Definition 5. We leave this extension to future work.

G Scope and Future Work

LOOM detects over-reporting, but not under-reporting, under unilateral deviations, and its power depends on how well the remaining agents cover the deviating agent's arm-feature directions. We make this dependence explicit through the maximum advantage C (Definition 5), which appears in Theorem 4 and Theorem 5. As peer coverage deteriorates, the corresponding guarantees degrade continuously and can become vacuous in the failure cases illustrated in Sec. B.1.

These limitations suggest several directions for future work. *(i) Theoretical control of C*: identifying design or exploration conditions under which C can be provably bounded, rather than measured empirically as in our experiments. *(ii) Heterogeneous agents*: relaxing the shared reward-function assumption in a homogeneous setting through clustered agent types, similarity-based weighting, or shared latent representations (Sec. F, Question 3), since LOOM fundamentally requires peer coverage rather than exact reward homogeneity. *(iii) Attribute-level user feedback*: extending LOOM beyond reward-level feedback to settings where users directly verify declared attributes, which would remove the need for cross-agent comparability and could naturally accommodate heterogeneous agents (Sec. F, Question 4). *(iv) Coverage-aware and colluding agents*: designing mechanisms that remain effective when agents deliberately misreport in weakly covered directions or coordinate their reports.