Verified Influence Auctions for LLM Markets: Proper-Scoring Audits for Truthful Distribution Reports

Liz Lemma Future Detective

January 15, 2026

Abstract

Token-auction mechanisms for aggregating LLM outputs typically assume the platform has truthful access to each bidder's next-token distributions. In 2026, that assumption is fragile: advertisers can strategically modify prompts, adapters, or served APIs, and regulators increasingly demand verifiable claims about sponsored generation. We introduce a clean verification layer for influence auctions: advertisers commit to a reporting interface, the platform occasionally audits that interface on held-out contexts, and uses strictly proper scoring rules to penalize misreporting. We show that under log-score auditing, the expected penalty equals a KL divergence term, yielding a quadratic ($\Omega(\varepsilon^2)$) loss for ε-scale misreports (Pinsker). When combined with a monotone aggregation rule (e.g., linear pooling) and second-price-style influence payments from prior token-auction work, we obtain approximate dominant-strategy truthfulness for distribution reporting under mild Lipschitz assumptions on the value of influence. In a tractable linear-utility benchmark we give a closed-form best-response characterization, quantify misreport distortion as $\Theta(1/(\alpha\gamma))$, and derive comparative statics for audit probability, penalty weight, and number of audit samples. We also discuss practical enforcement via TEEs/cryptographic commitments and outline how audits compose over multi-token generation.

Table of Contents

3. Model: contexts, committed true distributions, reporting interfaces, monotone aggregation, influence payments, and audit protocol.

4. Proper scoring rules and audit penalties: log score / Brier score; expected penalty identities and divergence lower bounds.

5. Main results (general): approximate truthfulness bounds for reporting; equilibrium misreport magnitude; how audit intensity trades off with manipulation gains.

6. Closed-form benchmark: linear valuations + linear pooling; explicit best-response report; distortion rate $\Theta(1/(\alpha\gamma))$; welfare/revenue implications.

7. Implementation discussion: enforcing single committed reporting function (TEE, remote attestation, cryptographic commitments); when numerical methods are required (nonlinear constraints, nonconvex safety layers).

8. Extensions: multi-step generation, adaptive audits, bond/slashing variants, strategic selection of contexts, and collusion-resistance considerations.

9. Empirical sketch (optional): simulation plan on open LLMs with synthetic advertiser prompts; calibration of (α, γ, m) to achieve a target distortion bound.

10. Conclusion: design recommendations for verified influence markets and open problems.

1 Introduction and motivation: why 'truthful LLM access' fails in 2026; threat models (prompt/adaptor manipulation, API shading), and why audits are needed.

In 2026, the premise that a platform can obtain *truthful* access to a bidder's language model—in the sense of reliably learning the bidder's intended conditional behavior at the exact context where a decision is made—is no longer a benign engineering assumption. It is an equilibrium object. When multiple parties have a monetary stake in how an LLM completes a prompt, the model becomes an instrument of influence, and any interface that exposes "probabilities" or "token-level preferences" is a natural surface for strategic distortion. The motivating question for our design is therefore not whether the platform *can* query an advertiser's system, but whether it can do so in a way that is incentive compatible: if we ask for a distribution over next tokens, do we get the distribution that the advertiser's committed model would actually induce, or do we get a strategically shaded report that is optimized against the platform's aggregation and pricing rules?

A useful analogy is to classic ad auctions: we do not take an advertiser's statement "my click-through rate will be x" at face value, because the advertiser can benefit from misstatements unless there is verification, reputational capital, or a mechanism that makes the claim self-enforcing. The difference in the LLM setting is that the relevant object is not a scalar but an entire *context-dependent distribution* (or a policy), and the action space is correspondingly larger. Even when a bidder genuinely deploys a fixed base model, there is typically a thick layer of orchestration—prompting, retrieval, sampling, tool calls, and post-processing—that can be adjusted to produce behaviors that are hard to detect ex post from a single realized completion. As a result, a platform that relies only on non-audited API access is implicitly trusting a complex pipeline that the counterparty has both the ability and the incentive to manipulate.

The failure mode becomes sharp in any mechanism that aggregates bidders' probabilistic inputs. Token-level auctions and mixture-style aggregators require probability vectors because they must trade off competing objectives (e.g., safety, helpfulness, brand tone, or topical preferences) at the margin. Once payments depend on the reported distribution, bidders face a familiar temptation: they can tilt the report toward tokens that are advantageous under the pricing rule while still producing superficially plausible outputs. This is precisely the kind of "shading" that auctions are designed to discipline, but the discipline in standard auctions comes from the fact that the platform can *observe* the allocation and charge payments accordingly. In the LLM setting, the relevant allocation is a *distribution* used internally by the platform to sample a token; if the bidder can distort the distribution it

reports, it can distort the platform's internal allocation rule itself.

We can organize the strategic threats into three families that have become operationally salient in modern LLM stacks. The first is *prompt and context manipulation*. In principle, the platform controls the prompt, so one might think the context is fixed. In practice, the "context" is the full state passed into the bidder's system, and bidders can influence that state through seemingly innocuous channels: providing suggested system prompts, "compatibility" templates, or safety policies that must be prepended; requesting additional metadata fields; or offering a retrieval plugin that changes which documents are injected. Even if the platform insists on a fixed interface, bidders can define reporting functions that are highly non-smooth in the context, behaving truthfully on common or audited prefixes while switching behavior on rare or commercially valuable states. Because LLM policies are extremely high-dimensional, such conditional manipulations are hard to detect without a deliberate stress-testing procedure. Moreover, the platform's own incentives can exacerbate the problem: when latency and user experience matter, platforms standardize prompts and reduce variability, which makes it easier for an adversary to overfit its manipulation to the typical distribution of contexts.

The second family is *adapter and fine-tuning manipulation*. By 2026, it is routine to ship a base model plus a collection of low-rank adapters, soft prompts, routing rules, or mixture-of-experts gates. These components can be swapped cheaply and can condition on latent features of the prompt. From a mechanism-design perspective, this means that a bidder can commit to a base model (or even to a hash of a model) while retaining substantial degrees of freedom in how the model is actually invoked. For example, the bidder can implement one adapter for "normal" behavior and another for "competitive" contexts where influencing the platform's output is especially valuable. Even when the platform requires attestation of the model weights, the bidder may still control the sampling temperature, truncation scheme, or decoding constraints; each of these changes the induced token distribution in systematic ways. The strategic concern is not merely that the bidder can change behavior, but that it can change behavior *selectively* in ways that exploit the platform's aggregation rule and the distribution of prompts encountered in production.

The third family is *API shading*, by which we mean any divergence between (i) the distribution the bidder claims it would sample from at a given context and (ii) the distribution it actually uses when audited or when the platform relies on it to construct an aggregate. There are several concrete instantiations. A bidder may return a probability vector that is not the true next-token distribution of any underlying model (e.g., after applying a proprietary reweighting that is designed to increase its influence under the aggregator). A bidder may report a distribution but then sample from a different one (for example, to preserve some internal objective while still

earning influence payments). Or the bidder may implement an interface that is non-committal in subtler ways: returning probabilities at low precision, clipping small probabilities to zero, or smoothing in a way that makes the reported distribution appear well-behaved while masking targeted distortions on a small subset of tokens. These are not hypothetical concerns; they are direct analogues of misreporting quality in procurement and of miscalibration in prediction markets, except that here the mechanism's input is a full conditional distribution, so there is far more room to hide economically meaningful deviations in corners of the simplex.

A natural reaction is to demand a stronger form of verification: require that bidders run inside a trusted execution environment, disclose code, or submit to third-party certification. But such requirements are neither costless nor fully sufficient. TEEs are promising precisely because they can attest to code and produce verifiable samples, yet they do not automatically solve the mechanism problem: if the platform only uses the bidder's *reported* distribution to compute allocations and payments, the bidder can still choose what to report, and TEEs only help if the platform has a way to compare the report to a verifiable ground truth. Code disclosure faces obvious commercial constraints and does not prevent context-dependent behavior unless the full pipeline (including retrieval and orchestration) is inside scope. Certification can establish baseline safety properties, but it is not designed to enforce *truthfulness* with respect to a platform-specific aggregation and pricing scheme.

This leads to the central motivation for audits. We want a mechanism that (a) allows bidders to retain their proprietary models and pipelines, (b) makes it costly to misreport the relevant distribution, and (c) does so in a way that scales to the token-by-token nature of generation. Auditing provides a conceptually clean approach: rather than trying to prevent every form of manipulation technologically, we make misreporting *economically dominated* by attaching expected penalties to deviations from truth. In this view, the platform does not need omniscient access; it needs a credible procedure that occasionally checks the bidder's claimed distribution against verifiable outcomes generated by the bidder's committed model. When the penalty is derived from a strictly proper scoring rule, truthful reporting becomes uniquely optimal in expectation for the audited instances, and the magnitude of the incentive can be tuned by audit frequency and penalty scale.

The key design constraint is that audits must be compatible with real-time generation. Token auctions and related aggregation rules are meant to operate at inference speed. We therefore cannot assume heavy ex post investigations or long adjudication processes. Instead, we need a lightweight audit primitive: draw a modest number of held-out contexts, generate outcomes from a committed model instance in a verifiable way (e.g., inside a TEE), and score the bidder's reported distributions on those outcomes. This

separates two roles that are too often conflated. The platform's *production* interface may remain a black-box API, but the *audit* interface must be tied to something that can be trusted as ground truth. Importantly, this does not require the platform to know the model's internal probabilities; it only requires the ability to sample from the committed model and to evaluate the bidder's reported probabilities on the realized samples.

Audits also discipline the more subtle threat models described above. Prompt and adapter manipulation become less attractive when the bidder cannot predict which contexts will be audited and when the audit contexts are drawn from a distribution that reflects the platform's relevant use cases. API shading becomes directly penalized: if a bidder inflates probability mass on tokens that increase its influence under aggregation, it must accept that the same inflated report will be scored against samples from the committed model, and proper scoring rules convert such discrepancies into expected losses. Even if the bidder tries to game the system by being truthful on some subset of contexts and manipulative elsewhere, the audit distribution creates an explicit tradeoff: the more the manipulation is targeted to contexts that matter in production, the more it risks being caught in audit, unless the bidder can exploit distribution shift between production and audit.

This last point highlights an essential limitation and a policy-relevant implication. The effectiveness of auditing depends on the relationship between the platform's operational contexts and the audit distribution. If audits are drawn from an unrepresentative or predictable set of prompts, sophisticated bidders can learn to behave truthfully on audited inputs while shading on the rest. In other words, verification is only as strong as the coverage of the audit process. This is a familiar lesson from tax enforcement, product compliance, and financial auditing: sampling and inspection work when the inspected set is sufficiently correlated with the set where malfeasance generates value. For LLM platforms, this suggests that audit-context design is not a minor implementation detail but a governance choice. Platforms may need to rotate audit suites, stratify audits across product surfaces, and incorporate adversarial prompt generation to keep the audit distribution aligned with emerging manipulation strategies.

A second limitation concerns measurement and calibration. Proper scoring rules can be unbounded (as with log scores when a bidder assigns near-zero probability to an event that occurs), which is good for incentives but delicate for implementation. Real systems must handle numerical issues, bounded liability, and dispute resolution when a bidder claims the audit environment differed from production (e.g., different tokenizer versions, nondeterministic tool outputs, or retrieval corpora). These practicalities push us toward a hybrid view: audits are a mechanism-design tool whose details must be engineered with robustness in mind, including version pinning, reproducible execution, and clear commitments about what constitutes the "committed" model behavior.

Despite these challenges, the argument for audits is ultimately an argument about aligning incentives under incomplete technological control. In a world where bidders can cheaply alter model behavior through orchestration and decoding, and where platforms cannot feasibly inspect every execution path, purely contractual notions of "truthful access" are brittle. Auditing converts a hard-to-enforce property (truthful reporting of a complex object) into a standard enforcement paradigm: occasional verification with penalties that scale with the severity of misreporting. The result is not perfect truthfulness in every state, but a tunable approximation whose tightness improves as audits become more frequent or more severe. That is the tradeoff our model is meant to illuminate: we can preserve the performance and flexibility benefits of multi-party LLM aggregation while introducing an enforcement layer that makes strategic misreporting an unattractive equilibrium behavior.

2 Related work: token auctions/LLM aggregation, proper scoring rules, mechanisms with verification/audits, and practical ML auditing/TEEs.

Related work. Our setting sits at the intersection of (i) mechanisms for aggregating probabilistic or policy-like inputs, (ii) proper scoring rules and information elicitation, (iii) classic "verification and auditing" models in economics and mechanism design, and (iv) practical systems work on trustworthy execution and ML auditing. Because each literature addresses a different part of the problem, it is useful to be explicit about which pieces we borrow and where the LLM-token setting forces new design choices.

Token auctions and LLM aggregation mechanisms. The immediate backdrop is the emerging line of work that treats next-token generation as an allocation problem in which multiple agents have preferences over a probability distribution on tokens, and the platform selects a distribution (or action) to sample from. This "distributional allocation" perspective is reminiscent of mixture modeling and ensemble methods in machine learning, but the mechanism-design point is that the mixture weights, the pooling rule, and any pricing rule together define incentives over an object far richer than a single scalar bid. Existing proposals for token-level auctions and influence-style payments emphasize monotonicity of allocation with respect to bids and the feasibility of second-price analogues in settings where the outcome is a randomized policy rather than a deterministic slot assignment (see, e.g., **???**). Our contribution is complementary: we treat those allocation-and-pricing rules as a black box on the bid side and focus on a separate strategic margin that becomes salient only when bidders also control (or can misrepresent) the probabilistic inputs used by the platform.

More broadly, our model relates to classical work on randomized allocations and lotteries in mechanism design, where an allocation is a distribution over outcomes and agents are risk-neutral with quasi-linear utilities. In that tradition, incentive properties typically hinge on the platform being able to compute outcomes and payments as specified by the mechanism. In the LLM setting, however, the platform often does *not* directly observe the relevant primitives (the bidders' token distributions or policies); it queries them through an interface. This interface layer is what turns a standard randomized mechanism into a mechanism-with-reports problem. One can view our audit design as a way to restore the usual implementability assumptions (that the mechanism has access to the inputs it needs) without requiring full transparency of proprietary models.

Linear pooling, ensemble learning, and probabilistic opinion aggregation. On the purely statistical side, linear pooling and related aggregation rules have a long history in forecast combination and Bayesian opinion pooling **?**. These methods provide a natural vocabulary for what platforms are already doing operationally when they blend model outputs from multiple sources (mixture-of-experts routing, weighted ensembles, and policy interpolation). What changes in our setting is not the mathematics of pooling per se, but the introduction of strategic agents who can *choose* what distribution to feed into the pooling operator. Once reports are strategic, the platform cannot treat a pooling rule as merely a predictive heuristic; it is a mechanism that must be evaluated by equilibrium behavior. This distinction is easy to miss because the objects (probability vectors) look the same in both worlds, but the normative question changes from "which pooling rule predicts well?" to "which pooling rule, together with payments and verification, induces the reports we want?"

Proper scoring rules and truthful probability reports. The most direct tool we use is the theory of strictly proper scoring rules, which characterizes loss functions under which a forecaster maximizes expected score by reporting its true predictive distribution **????**. Proper scoring rules are by now standard in probabilistic forecasting and in machine learning evaluation because they operationalize calibration and sharpness. In mechanism design, they also appear as the canonical way to elicit beliefs when the realized outcome will be observed. The log score in particular plays two roles that are especially convenient for our purposes: it has an information-theoretic interpretation (expected log loss equals cross-entropy, differing from entropy by a KL divergence term), and it generates strong marginal incentives against assigning near-zero probability to events that can occur.

Our use of proper scoring rules is closer in spirit to the "decision markets" or "prediction market scoring rule" literature **??** than to evaluation-

as-benchmarking. The platform is not merely measuring quality; it is using scoring-rule losses as *penalties* to shape equilibrium reports. That said, an important implementation nuance in LLM systems is that unbounded scores create bounded-liability and numerical-stability issues. Several practical variants in forecasting (clipped log scores, capped losses, or convex combinations with bounded scores) trade off incentive strength for robustness. We keep the analysis clean by presenting the log-score benchmark, but in practice the same logic can be carried through with bounded strictly proper rules at the cost of weaker quadratic bounds and more explicit calibration of maximum penalties.

Information elicitation without verification versus with verification. A large mechanism-design literature studies eliciting information when the outcome is not directly verifiable, including peer prediction, Bayesian truth serum, and related methods **???**. These mechanisms are attractive when no trusted ground truth exists, but they typically require multiple reports about the same latent event and impose distributional assumptions to pin down equilibria. Our setting is different in a way that matters operationally: the platform can, at least in principle, create a verifiable channel by sampling from a committed model instance (via a TEE or equivalent attestation mechanism). Once verification is available, the design space is closer to standard proper scoring (single-agent truth-telling against realized outcomes) and avoids the multiplicity and coordination concerns that arise in peer-prediction equilibria. Put differently, we are not trying to infer truth from cross-consistency of agents; we are trying to make a particular interface report self-enforcing by occasionally checking it against a trusted source.

This comparison also clarifies a limitation: if verification is weak or ambiguous (e.g., the "true" model behavior depends on non-deterministic tools, external retrieval, or hidden state), then the platform is pushed back toward the no-verification regime, and peer-prediction-style tools may again be relevant. Our emphasis on committed distributions and reproducible sampling is precisely an attempt to keep the problem in the verification regime where incentives are simpler and sharper.

Mechanisms with verification, auditing, and monitoring. The economic core of our approach is the idea that rare but credible audits can discipline behavior in rich action spaces. This is a recurring theme in contract theory and public finance: costly monitoring can be used selectively to induce effort or truthful reporting, with equilibrium distortions determined by audit frequency and penalty severity **???**. In auctions and procurement, related ideas appear as mechanisms with verification or ex post checks, where misreports can be punished if evidence arises **?**. In algorithmic mechanism design, there is a parallel line on "mechanism design with monitoring" and

"truthful mechanisms with verification," where limited verification can substitute for full observability of types or outcomes **??**.

Our twist is that the audited object is not a scalar claim (cost, value, quality) but a *context-dependent distribution* over tokens, and the audit must be compatible with high-frequency generation. This changes the engineering constraints (audits must be lightweight and sample-based), but it also changes the economic geometry: the deviation set is infinite-dimensional, and the platform needs a penalty that grows smoothly with the magnitude of misreporting. Proper scoring rules are natural here precisely because they convert distributional deviations into divergences (KL, and via Pinsker, quadratic lower bounds in total variation). This is analogous to how convex regularization disciplines high-dimensional optimization: we are effectively adding an "entropic barrier" around the truthful distribution, with strength controlled by (α, γ).

Robustness, approximate incentive compatibility, and Lipschitz environments. A separate but related literature studies approximate incentive compatibility when agents can only slightly affect outcomes or when the mechanism is implemented with noise, discretization, or sampling error. In large markets, for example, an individual agent's effect on prices can vanish, making truth-telling approximately optimal even without exact IC. In our setting, the analogous scaling parameter is the bidder's influence weight $w_i(b)$ under the aggregation rule: when $w_i(b)$ is small, a bidder's report has limited leverage, so the benefit from shading is at most linear in $\|\hat{p}_i - p_i\|_1$. Combining this with a quadratic audit penalty yields a clean approximate-truthfulness conclusion. Conceptually, this is close to the spirit of smoothness and Lipschitz-based analyses in algorithmic game theory: one bounds how much a deviation can move allocations and payoffs, and then chooses enforcement strength so that the deviation is not worth it.

We also view this as a bridge between theory and practice. Platform designers rarely need literal truth-telling in every state; they need deviations small enough that downstream properties (monotonicity, fairness constraints, safety filters) are not materially undermined. Approximate IC bounds provide a way to translate engineering tolerances (how much can q shift before something breaks?) into audit parameters.

Trusted execution environments, verifiable inference, and reproducible sampling. On the systems side, our audit primitive presumes the platform can obtain verifiable samples from a committed model behavior. TEEs such as Intel SGX and related enclave technologies were designed to support precisely this kind of remote attestation: a remote party can verify that some code is running in an isolated environment and that outputs are produced by that code **?**. A growing applied cryptography and systems

literature explores secure and verifiable ML inference using enclaves, secure multiparty computation, or zero-knowledge proofs **??**. These tools differ in performance and trust assumptions, but they share the goal of making it costly (or impossible) to deviate from an agreed computation without detection.

We deliberately do not take a stand on which primitive dominates, because the right choice depends on threat models and latency budgets. What matters for our mechanism is the existence of *some* channel that (i) pins down what the "committed model" is for audit purposes, and (ii) allows the platform to sample from it on demand. In practice, this points to a checklist of design requirements that are often glossed over: version-pinned tokenizers, deterministic decoding and sampling procedures (or at least auditable randomness), fixed retrieval corpora or logged retrieval traces, and clear boundaries around which parts of an orchestration stack are in-scope for attestation. The more ambiguity remains about what is being attested, the more room remains for strategic behavior that is technically "compliant" but economically manipulative.

ML auditing practice: evaluation harnesses, red-teaming, and governance. Finally, we connect to the fast-growing practice of ML auditing and evaluation. Frameworks such as model cards and datasheets were introduced to standardize documentation and disclosure **??**. Benchmark suites and evaluation harnesses for LLMs (e.g., broad-coverage evaluation and red-teaming methodologies) aim to measure performance and safety across heterogeneous tasks **??**. Regulatory and standards efforts (e.g., risk-management frameworks) increasingly emphasize auditability as a governance requirement rather than a purely technical desideratum.

Our mechanism-theoretic view does not substitute for these efforts; it complements them by clarifying what audits can and cannot accomplish when agents are strategic. In particular, evaluation-oriented audits typically assume the model is trying to do well on the benchmark. Our audits assume the opposite: the bidder may want to *look* well-calibrated while covertly shifting probability mass to increase its influence under a platform's aggregation and pricing rules. That adversarial posture changes how one should select audit contexts and how one should interpret "coverage." It also suggests a policy implication: audit design (the choice of $\mathcal{D}$, context sampling, and update cadence) is not merely a technical detail but an institutional decision akin to how tax authorities design audit selection or how financial regulators design stress tests. If the audited distribution becomes predictable or too narrow, sophisticated agents can learn to comply on-audit and deviate off-audit.

Positioning and limitations. Taken together, these literatures motivate the structure we analyze: an influence-based allocation-and-pricing mechanism on the bid side, combined with a proper-scoring audit layer that makes distributional reports approximately self-enforcing. The model is intentionally stylized in two respects. First, we treat the audit channel as capable of producing draws from a well-defined $p_i(\cdot \mid x)$, whereas real LLM stacks may have non-stationary components (retrieval drift, tool APIs, or adaptive safety layers). Second, we focus on incentives to misreport distributions given an aggregation rule, rather than on collusion, sybil attacks, or endogenous entry. These omissions are not innocuous, but they are a useful starting point: they let us isolate the core economic tradeoff between the *benefit* of distorting the platform's internal allocation rule and the *expected cost* imposed by verifiable audits. In that sense, our contribution is not to claim that audits solve every strategic problem in multi-model generation, but to formalize when a lightweight audit primitive can restore the assumptions under which token-auction-style mechanisms are meant to operate.

3 Model

We model a platform that generates tokens by sampling from a context-dependent distribution, where multiple strategic "advertisers" (or model providers) can both *bid* for influence and *report* probabilistic outputs. The central friction is that the platform needs access to bidders' token distributions in order to run an influence-based allocation and pricing rule, but those distributions are typically exposed only through an interface that the bidder controls. Our goal in this section is to make that interface layer explicit and to separate (i) the bid-side mechanism that determines influence and payments from (ii) the audit layer that makes reported distributions credible.

Contexts and tokens. Fix a finite token (or action) set T. Generation occurs in contexts $x \in \mathcal{X}$, where x should be interpreted broadly: a prompt prefix, a dialogue state, a tool-augmented state, or any sufficient statistic of the platform's generation process. In a multi-step generation, the context evolves endogenously as tokens are produced; we deliberately do not restrict the context dynamics, since our incentive results will be stated pointwise in x or in expectation over an external audit distribution. What matters is that at each step the mechanism takes as input a context x and returns a distribution $q(\cdot \mid x) \in \Delta(T)$ from which the next token y is sampled.

Advertisers and committed "true" distributions. There are n advertisers indexed by $i \in [n]$. Each advertiser has an underlying model (or policy) that induces a conditional distribution over tokens at every context.

We denote this by

$$p_i(\cdot \mid x) \in \Delta(T).$$

We treat p_i as the economically relevant primitive: it is the behavior the advertiser would exhibit when queried honestly. Operationally, we assume the platform can obtain verifiable samples from $p_i(\cdot \mid x)$ on demand, for example by querying an attested model instance in a trusted execution environment (TEE), or by using another verification channel that pins down the model version, tokenizer, randomness source, and decoding/sampling procedure. The economic role of this assumption is not to require that the platform can *inspect* the model, but rather that it can occasionally *test* a reported distribution against draws generated by a committed, reproducible implementation.

This commitment requirement deserves emphasis. If the "true" distribution can drift or be selectively altered in response to audits, then audits cease to measure a stable object. In practice, maintaining a well-defined $p_i(\cdot \mid x)$ typically entails version-pinning (weights, tokenizer, system prompt), controlling stochasticity (auditable randomness seeds or a verifiable RNG), and specifying what external calls (retrieval, tools) are in scope. We abstract from these systems details and treat them as part of the enforcement primitive that makes p_i meaningful.

Reporting functions as interface control. In addition to having a true distribution p_i, advertiser i chooses a *reporting function*

$$\hat{p}_i : \mathcal{X} \to \Delta(T),$$

which is the object the platform actually observes and uses in its aggregation rule. We impose that $\hat{p}_i(\cdot \mid x)$ is a valid distribution for every x (nonnegative and summing to one). Economically, $\hat{p}_i$ is the bidder-controlled interface: it may coincide with p_i, but it may also be strategically shaded, simplified, truncated, or otherwise distorted. The key design choice in our timing is that $\hat{p}_i$ is committed *once per session* (e.g., by committing to code whose hash is attested), and the same committed $\hat{p}_i$ is used both during generation and during any subsequent audit. This coupling is what prevents a bidder from behaving one way on audited queries and another way on production queries *within the same session*. It does not eliminate all forms of strategic behavior (e.g., exploiting predictable audit selection across sessions), but it rules out the most direct on-the-fly evasion.

We allow $\hat{p}_i$ to be rich: it may depend arbitrarily on x, and we do not require it to be derived from a model that can be sampled. This is intentional. In many real systems, the platform queries an API that returns logits or probabilities, and nothing prevents an advertiser from post-processing those values before returning them. Our mechanism therefore treats the report as a potentially adversarial function, and uses audits to discipline it.

Bids and influence weights. Each advertiser also submits a scalar bid $b_i \geq 0$. The bid affects the advertiser's influence on the platform's aggregate distribution, and it affects the payment charged by the mechanism. We summarize the influence of bids through *weights* $w_i(b)$, where $b = (b_1, \ldots, b_n)$. The canonical example is proportional weighting,

$$w_i(b) = \frac{b_i}{\sum_{j=1}^{n} b_j},$$

defined whenever $\sum_j b_j > 0$; if all bids are zero we may set a default rule (e.g., uniform weights or an outside option model). What we require at the level of abstraction used here is that influence is *monotone* in bids: increasing b_i should not reduce advertiser i's weight, holding others fixed. This is the bid-side analogue of monotone allocation in classic auctions, and it is the condition under which second-price-style payments can be used to support desirable bidding incentives.

Aggregation rule over reported distributions. Given bids and reports, the platform chooses an aggregate distribution

$$q(\cdot \mid x) = q\big(b, (\hat{p}_1(\cdot \mid x), \ldots, \hat{p}_n(\cdot \mid x))\big) \in \Delta(T) \quad \text{for each } x \in \mathcal{X}.$$

We allow q to be any aggregation operator that maps $(b, \hat{p}(\cdot \mid x))$ into a distribution on T. The running benchmark is *linear pooling*,

$$q(\cdot \mid x) = \sum_{i=1}^{n} w_i(b)\, \hat{p}_i(\cdot \mid x),$$

which is attractive because it is transparent, easily implementable, and makes each advertiser's influence scale directly with its weight. But our analysis is meant to accommodate other monotone aggregation schemes used in practice (e.g., temperature-scaled mixtures, capped weights, or rules that interpolate between a baseline model and bidders' models). The economic point is that q is the mechanism's "allocation": instead of assigning a deterministic slot, it assigns probability mass over tokens, and bidders care about this randomized policy.

Generation process and realized outcomes. At each generation step k, the platform observes the current context x_k, queries each committed reporting function $\hat{p}_i(\cdot \mid x_k)$, forms the aggregate distribution $q_k(\cdot \mid x_k)$ according to the aggregation rule, and samples an outcome

$$y_k \sim q_k(\cdot \mid x_k).$$

The context may then update to x_{k+1} according to the platform's environment (for example by appending y_k to a prompt, updating a dialogue state,

or incorporating tool outputs). Because our primary objective is incentive control over the reported distributions, we do not need to specify the law of motion for contexts; we will either reason pointwise in x or take expectations over an externally specified audit distribution.

Utilities from the aggregate distribution. Advertiser i receives a gross per-step utility $U_i(q, x)$ that depends on the chosen aggregate distribution and the context. This utility captures whatever the advertiser values about the platform's behavior: probability assigned to its preferred tokens, downstream user actions induced by the sampled token, brand-safe language, or other context-dependent objectives. We take advertisers to be risk neutral and to have quasi-linear preferences over money, so total expected utility is gross utility minus payments and penalties.

To make the mechanism analyzable in a high-dimensional outcome space, we impose a regularity condition that is natural in distributional allocation problems: U_i is Lipschitz in the induced distribution. Concretely, there exists $L_i < \infty$ such that for all x and all distributions $q, q' \in \Delta(T)$,

$$|U_i(q, x) - U_i(q', x)| \le L_i \, \|q(\cdot \mid x) - q'(\cdot \mid x)\|_1.$$

This assumption says that small perturbations of token probabilities cannot cause unbounded jumps in value. It is a modeling choice that rules out knife-edge discontinuities (e.g., value that depends on whether a token's probability crosses an exact threshold), but it matches many practical objectives that are smooth in probabilities, including expected click-through, expected conversion, or expected compliance with soft constraints.

Influence payments (bid-side pricing). Alongside the allocation rule, the platform charges an influence payment $z_i(b, \hat{p})$ to advertiser i. We keep the payment rule abstract, with two motivating requirements. First, payments should be compatible with monotone influence: a bidder who increases its bid and thereby increases its weight should face (weakly) higher expected payment, in the same way that higher bids in a second-price auction can increase the chance of winning and the expected price paid. Second, payments should be implementable given the randomized nature of q; in practice this often means using sampling-based estimators (as in "stable sampling" implementations) that approximate counterfactual influence or marginal contribution.

Because our focus is on the interaction between reporting and auditing, we summarize how payments enter incentives through a Lipschitz condition analogous to the one for utility. Let $z_i(q, x)$ denote the per-step payment as a function of the induced aggregate distribution (suppressing dependence on $(b, \hat{p})$ through q). We assume there exists $L_i^z < \infty$ such that

$$|z_i(q, x) - z_i(q', x)| \le L_i^z \, \|q(\cdot \mid x) - q'(\cdot \mid x)\|_1.$$

This condition captures the idea that if a bidder can only slightly perturb the platform's distribution, it can only slightly perturb its own payment. It is also the key technical bridge that lets us treat reporting incentives through how reports move q.

Audit contexts and audit protocol. The platform can audit advertisers' reports using a held-out distribution $\mathcal{D}$ over contexts. We interpret $\mathcal{D}$ as an "audit harness" or evaluation distribution: it is selected by the platform (or a regulator, or a third-party auditor) and is not under advertisers' direct control. An audit consists of drawing m i.i.d. contexts

$$x_1^a, \ldots, x_m^a \sim \mathcal{D}$$

and then, for each advertiser i, obtaining verifiable samples

$$y_j^a \sim p_i(\cdot \mid x_j^a)$$

from the committed model behavior. The platform then evaluates the committed report $\hat{p}_i(\cdot \mid x_j^a)$ against the realized y_j^a using a scoring function $S(\hat{p}, y)$ that assigns a loss (or negative score) to probabilistic reports. The per-audit penalty for advertiser i is scaled by $\gamma > 0$ and averaged over contexts:

$$\text{Penalty}_i \;=\; \gamma \cdot \frac{1}{m} \sum_{j=1}^{m} S\big(\hat{p}_i(\cdot \mid x_j^a), y_j^a\big).$$

Audits occur with probability $\alpha \in (0, 1)$ per session (or per block of interactions). The two parameters (α, m) capture distinct operational levers: α controls how often an audit is triggered, while m controls how informative a triggered audit is. The scale γ controls the economic magnitude of penalties relative to the gains from influencing q.

Two practical remarks are useful here. First, the coupling of generation and audits through the same committed $\hat{p}_i$ matters: it ensures that the object being penalized is exactly what the platform uses to form q. Second, the audit uses samples from p_i, not from q. This is deliberate: the purpose is not to evaluate the platform's overall output, but to verify whether advertiser i's reported distribution matches its committed model behavior on the audited contexts.

Session payoff and equilibrium notion. A session consists of K generation steps followed by settlement. Let q_k denote the aggregate distribution used at step k, and let $z_{i,k}$ denote the corresponding payment component. Advertiser i's expected payoff can be written as

$$\Pi_i(b, \hat{p}) = \mathbb{E}\Big[\sum_{k=1}^{K} U_i\big(q_k, x_k\big)\Big] - \mathbb{E}\Big[\sum_{k=1}^{K} z_{i,k}(b, \hat{p})\Big] - \alpha\,\gamma\,\mathbb{E}\Big[\frac{1}{m}\sum_{j=1}^{m} S\big(\hat{p}_i(\cdot \mid x_j^a), y_j^a\big)\Big],$$

where the last expectation is taken over audit contexts $x_j^a \sim \mathcal{D}$ and audit outcomes $y_j^a \sim p_i(\cdot \mid x_j^a)$, as well as any randomness in generation and payment computation. We study Nash behavior: each advertiser chooses $(b_i, \hat{p}_i)$ to maximize Π_i given others' choices. The platform is treated as a designer that commits to $q(\cdot)$, $z_i(\cdot)$, and the audit protocol.

What the model isolates, and what it does not. This formulation isolates a specific strategic margin that is easy to overlook if one starts from standard auction logic: the bidder is not only bidding for influence; it is also supplying the probabilistic input that determines what influence means. In other words, even if bid-side incentives are well-behaved under monotone influence pricing, the mechanism can fail if bidders can cheaply misrepresent $\hat{p}_i$ to steer q while still paying the "right" price for the *reported* influence.

At the same time, the model abstracts from several complications. We do not model collusion (e.g., bidders coordinating reports), sybil attacks (splitting identity across multiple bidders), or endogenous entry. We also treat $\mathcal{D}$ as fixed and exogenous, whereas in practice the choice of audit contexts is itself a policy decision and may be strategically anticipated. Finally, we present the audit as producing verifiable samples from p_i; when real-world inference stacks include retrieval drift, tool calls, or adaptive safety layers, pinning down a stable p_i may require additional engineering constraints. These limitations matter for deployment, but the model is designed to make one tradeoff transparent: auditing turns misreporting into an expected monetary cost, and the strength of that discipline is governed by (α, m, γ) together with how much a bidder can move q through its weight and the aggregation rule.

In the next section we specialize to standard strictly proper scoring rules (notably the log score and the Brier score) and derive the expected-penalty identities and lower bounds that make this tradeoff quantitatively sharp.

4 Proper-scoring audits and divergence penalties

Our audit layer is meant to solve a very specific credibility problem: the platform needs to treat each $\hat{p}_i(\cdot \mid x)$ as a meaningful probabilistic object when it aggregates reports into q, yet the interface that produces $\hat{p}_i$ is bidder-controlled. The standard mechanism-design response is to attach a monetary consequence to the report that is minimized (in expectation) by truth-telling. Proper scoring rules provide exactly this instrument. In this section we (i) recall the properness property in the present "distribution over tokens" setting, (ii) specialize to the log score and the Brier score as canonical choices, and (iii) extract the quantitative lower bounds that convert an expected misreport into an expected monetary loss. These identities are the input to our approximate-truthfulness guarantees in the next section.

4.1 Strictly proper scoring rules as audit primitives

A (loss-based) scoring rule is a function

$$S : \Delta(T) \times T \to \mathbb{R},$$

where $S(\hat{p}, y)$ is interpreted as the loss charged when the agent reports $\hat{p}$ and the realized outcome is $y \in T$. We say that S is *proper* if for every true distribution $p \in \Delta(T)$,

$$\hat{p} \in \arg\min_{r \in \Delta(T)} \ \mathbb{E}_{y \sim p}\big[S(r, y)\big],$$

and *strictly proper* if the minimizer is unique and equals p. Properness is the probabilistic analogue of dominant-strategy truthfulness: it says that, holding fixed how the audit outcome y is generated (here, $y \sim p_i(\cdot \mid x)$ via the committed model), the bidder minimizes its *expected* audit loss by reporting the true conditional distribution.

Two remarks connect this property to deployment. First, the audit needs draws from the bidder-specific truth p_i, not from the platform aggregate q. Otherwise the scoring rule would incentivize conforming to the platform rather than revealing the bidder's model behavior. Second, strict properness is inherently an *in-expectation* statement: it guarantees that the expected audit penalty is minimized at truth, but any finite audit can be noisy. This is why we separate the *shape* of incentives (properness) from the *strength* of incentives (how α, m, and γ scale realized losses).

A useful structural fact is that every strictly proper scoring rule induces a divergence measuring the expected cost of misreporting. Concretely, for many common scoring rules there exists a convex potential $\Phi : \Delta(T) \to \mathbb{R}$ such that, for any $p, \hat{p} \in \Delta(T)$,

$$\mathbb{E}_{y \sim p}\big[S(\hat{p}, y)\big] = \mathbb{E}_{y \sim p}\big[S(p, y)\big] + D_\Phi(p, \hat{p}),$$

where D_Φ is a (nonnegative) Bregman divergence with $D_\Phi(p, \hat{p}) = 0$ iff $\hat{p} = p$. In our setting this decomposition is especially valuable: the first term is a constant with respect to the report and therefore irrelevant for incentives, while the second term is a clean, geometry-aware penalty for misreporting.

4.2 Log score: cross-entropy and KL divergence

The log score is the workhorse scoring rule in probabilistic forecasting and in information-theoretic mechanism design:

$$S_{\log}(\hat{p}, y) = -\log \hat{p}(y).$$

When we evaluate $S_{\log}$ under a true distribution p, we obtain the cross-entropy:

$$\mathbb{E}_{y \sim p}\big[-\log \hat{p}(y)\big] = \sum_{t \in T} p(t)\big(-\log \hat{p}(t)\big) =: H(p, \hat{p}).$$

The key identity is that cross-entropy decomposes into entropy plus a KL term.

Proposition 4.1 (Log-score audit identity)**.** *For any $p, \hat{p} \in \Delta(T)$ with $\hat{p}(t) > 0$ whenever $p(t) > 0$,*

$$\mathbb{E}_{y\sim p}\big[-\log \hat{p}(y)\big] = H(p) + \mathrm{KL}(p\|\hat{p}),$$

where $H(p) = -\sum_{t\in T} p(t)\log p(t)$ is Shannon entropy and $\mathrm{KL}(p\|\hat{p}) = \sum_{t\in T} p(t)\log\frac{p(t)}{\hat{p}(t)}$.

The proof is immediate algebra, but the economic interpretation is worth stating explicitly. Under log-score audits, the *incremental* expected penalty from reporting $\hat{p}$ rather than the truth p is exactly $\mathrm{KL}(p\|\hat{p})$. Thus, when we scale audits by $\alpha\gamma$, we are literally pricing misreporting in units of relative entropy. This is attractive because KL has several properties that map well to interface credibility: it is zero only at equality; it is sensitive to "hiding" probability mass (placing too little mass on outcomes that occur under p); and it is additive across independent draws, which aligns with sampling m independent audit contexts and outcomes.

At the same time, the log score has an operational sharp edge: if $\hat{p}(y) = 0$ for an outcome that occurs under p, the penalty is infinite. In an abstract model this is a feature (it makes support-misreporting prohibitively costly), but in deployed systems it requires care. The platform may need to enforce a minimum probability floor (e.g. $\hat{p}(t) \geq \varepsilon/|T|$) or to clip scores (replace $-\log\hat{p}(y)$ with $-\log(\max\{\hat{p}(y), \varepsilon\})$). Clipping preserves approximate properness while restoring boundedness and therefore cleaner finite-sample concentration; we return briefly to this point below.

4.3 Brier score: squared error geometry

A second canonical choice is the Brier score (quadratic score). In loss form, for a realized token y,

$$S_{\mathrm{Br}}(\hat{p}, y) = \sum_{t\in T}\big(\hat{p}(t) - \mathbf{1}\{t = y\}\big)^2.$$

Taking expectations under $y \sim p$ yields a simple Euclidean decomposition:

$$\mathbb{E}_{y\sim p}\big[S_{\mathrm{Br}}(\hat{p}, y)\big] = \sum_{t\in T}\Big(\hat{p}(t)^2 - 2\hat{p}(t)p(t) + p(t)\Big) = \|\hat{p} - p\|_2^2 + \big(1 - \|p\|_2^2\big),$$

where $\|\cdot\|_2$ is the standard Euclidean norm on $\mathbb{R}^{|T|}$. Hence, up to a constant independent of the report, the expected Brier loss is exactly $\|\hat{p} - p\|_2^2$, and strict properness follows because this squared distance is uniquely minimized at $\hat{p} = p$.

The Brier score is often attractive in practice because it is bounded: $S_{\mathrm{Br}}(\hat{p}, y) \in [0, 2]$ for distributions on a finite alphabet. This boundedness

makes finite-sample audits easier to calibrate using standard concentration inequalities. The tradeoff is geometric: unlike the log score, the Brier score does not assign disproportionately large penalties to under-reporting rare-but-possible events. If the platform's primary concern is preventing bidders from strategically "zeroing out" tokens that occur with small probability under p_i, the log score (or a clipped variant) provides a stronger deterrent.

4.4 From divergence to quantitative deterrence: lower bounds in total variation

Properness alone says that misreporting raises expected loss, but to obtain sharp equilibrium implications we need a *rate*: how quickly does the expected penalty increase as $\hat{p}$ departs from p? Since our later incentive bounds are stated in $\|\cdot\|_1$ (total variation) because aggregation and utilities are Lipschitz in that metric, we want lower bounds that convert divergence or squared error into $\|p - \hat{p}\|_1^2$.

For log-score audits, the link is Pinsker's inequality. Writing $\mathrm{TV}(p, \hat{p}) = \frac{1}{2}\|p - \hat{p}\|_1$, Pinsker implies

$$\mathrm{KL}(p\|\hat{p}) \;\geq\; c\,\mathrm{TV}(p, \hat{p})^2$$

for a universal constant $c > 0$; in particular one can use the conservative bound

$$\mathrm{KL}(p\|\hat{p}) \;\geq\; \frac{1}{8}\,\|p - \hat{p}\|_1^2.$$

(We emphasize that constants are not the main issue for our comparative statics; what matters is the quadratic dependence on $\|p - \hat{p}\|_1$.) Combining with Proposition 4.1, we obtain a direct quadratic lower bound on the incremental expected audit loss under the log score:

$$\mathbb{E}_{y\sim p}\big[S_{\log}(\hat{p}, y) - S_{\log}(p, y)\big] = \mathrm{KL}(p\|\hat{p}) \;\geq\; \frac{1}{8}\,\|p - \hat{p}\|_1^2.$$

This inequality is the technical hinge of our later results: it says that any attempt to shift the report by δ in ℓ_1 distance necessarily pays at least on the order of δ^2 in expected audit penalties, scaled by $\alpha\gamma$.

For the Brier score, the analogous conversion uses norm inequalities. Since $\|v\|_2 \geq \|v\|_1/\sqrt{|T|}$ for any $v \in \mathbb{R}^{|T|}$, we have

$$\|\hat{p} - p\|_2^2 \;\geq\; \frac{1}{|T|}\,\|\hat{p} - p\|_1^2.$$

Thus Brier audits also impose a quadratic cost in ℓ_1 distance, with a factor that depends on the alphabet size:

$$\mathbb{E}_{y\sim p}\big[S_{\mathrm{Br}}(\hat{p}, y) - S_{\mathrm{Br}}(p, y)\big] = \|\hat{p} - p\|_2^2 \;\geq\; \frac{1}{|T|}\,\|\hat{p} - p\|_1^2.$$

When $|T|$ is very large (as in token vocabularies), this bound is looser than the log-score Pinsker bound, reflecting again that quadratic scoring rules are "gentler" in the tails. In applications with large T, one practical response is to audit on a coarsened outcome space (e.g. token classes, safety categories, or top-k plus an "other" bin) so that both reporting and auditing focus on the parts of the distribution that matter for influence and welfare.

4.5 Context dependence and averaging over the audit distribution

Because our objects are conditional distributions, the preceding identities apply pointwise in context: for each fixed x, the expected audit loss under $y \sim p_i(\cdot \mid x)$ is minimized by reporting $\hat{p}_i(\cdot \mid x) = p_i(\cdot \mid x)$. In our protocol, however, the platform samples contexts $x \sim \mathcal{D}$ and averages scores over m draws. Taking expectations over $x \sim \mathcal{D}$, the incremental expected penalty from a reporting function $\hat{p}_i$ relative to truth p_i becomes

$$\alpha\gamma\,\mathbb{E}_{x\sim\mathcal{D}}\Big[\mathbb{E}_{y\sim p_i(\cdot|x)}S\big(\hat{p}_i(\cdot \mid x), y\big)\Big] - \alpha\gamma\,\mathbb{E}_{x\sim\mathcal{D}}\Big[\mathbb{E}_{y\sim p_i(\cdot|x)}S\big(p_i(\cdot \mid x), y\big)\Big],$$

which equals $\alpha\gamma\,\mathbb{E}_{x\sim\mathcal{D}}\mathrm{KL}(p_i(\cdot \mid x)\|\hat{p}_i(\cdot \mid x))$ under the log score, and equals $\alpha\gamma\,\mathbb{E}_{x\sim\mathcal{D}}\|p_i(\cdot \mid x) - \hat{p}_i(\cdot \mid x)\|_2^2$ under the Brier score.

To connect these expressions to a single scalar notion of "how much misreporting occurs on audited contexts," it is convenient to define

$$\delta_i := \mathbb{E}_{x\sim\mathcal{D}}\big\|p_i(\cdot \mid x) - \hat{p}_i(\cdot \mid x)\big\|_1.$$

Then, under log-score audits and using Pinsker pointwise in x followed by Jensen's inequality,

$$\mathbb{E}_{x\sim\mathcal{D}}\mathrm{KL}\big(p_i(\cdot \mid x)\|\hat{p}_i(\cdot \mid x)\big) \;\geq\; \frac{1}{8}\,\mathbb{E}_{x\sim\mathcal{D}}\big\|p_i(\cdot \mid x) - \hat{p}_i(\cdot \mid x)\big\|_1^2 \;\geq\; \frac{1}{8}\,\delta_i^2.$$

This is the cleanest way to see the mechanism's discipline: the expected audit loss grows at least quadratically in the average ℓ_1 deviation on the audit distribution. In the next section we will combine this quadratic cost with the (at most) linear benefit a bidder can obtain by moving the platform aggregate q through its weight, yielding the familiar "linear gain versus quadratic cost" tradeoff that pins down an $O(1/(\alpha\gamma))$ deviation scale.

4.6 Finite-sample audits and calibration considerations

Although our main incentive statements are in expectation, it is useful to note what m buys the platform. When S is bounded (as with the Brier score), standard concentration implies that the realized average score $\frac{1}{m}\sum_{j=1}^{m} S(\hat{p}, y_j)$ is close to its expectation uniformly over draws, so a bidder cannot rely on "getting lucky" in a small audit. When S is unbounded

(as with the log score), analogous concentration requires additional steps—most commonly clipping, flooring probabilities, or restricting attention to events with controlled likelihood ratios. These are not merely technicalities: they are the operational knobs that determine how harshly the system treats near-zero reports, how robust audits are to numerical underflow, and how predictable penalties are to participants.

Finally, the choice of $\mathcal{D}$ interacts with every statement above. Properness guarantees truthfulness *on the support of the audit distribution.* If audited contexts are systematically easier, more templated, or otherwise unrepresentative of production contexts, bidders may remain truthful on $\mathcal{D}$ while shading reports elsewhere. This is not a failure of scoring rules; it is a reminder that auditing is only as strong as the harness it is evaluated on. For this reason, in practice we view the selection and refresh of $\mathcal{D}$ as a policy instrument: by broadening coverage and limiting predictability, the platform increases the effective domain on which proper-scoring penalties bind.

The next section takes these identities as primitives and studies how they interact with influence aggregation and payments. The high-level message will be that, under Lipschitz utilities and monotone aggregation, misreporting can create at most linear gains through its effect on q, while proper-scoring audits impose quadratic expected costs; the resulting equilibrium distortions shrink at rate $1/(\alpha\gamma)$, with m controlling the reliability of enforcement in finite samples.

5 Approximate truthfulness from linear influence and quadratic audit costs

The incentive problem in our environment has a simple economic structure once we separate *how much a report can move outcomes* from *how harshly audits punish misreporting.* A bidder reports an entire conditional distribution $\hat{p}_i(\cdot \mid x)$, and this report influences the platform only through the aggregate $q(\cdot \mid x)$. If bidder i has a small influence weight (as induced by its bid relative to others), then even a large distortion in $\hat{p}_i$ can move q only slightly; conversely, a high-weight bidder can move q more and therefore has a stronger incentive to shade its report toward tokens that it privately values. Audits counteract this channel by imposing an expected cost that grows *quadratically* with the misreport (Section 4), creating the familiar "linear gains versus quadratic costs" tradeoff that underlies our approximate-truthfulness bounds.

To make this tradeoff explicit we work with aggregation rules that are Lipschitz in each bidder's report. Linear pooling is the cleanest case, but the logic extends to any rule that satisfies a comparable sensitivity bound.

5.1 The influence channel: how reports move the aggregate

Fix a bid vector b and reports $\hat{p}_{-i}$ from bidders other than i. Let $q(\cdot \mid x)$ denote the aggregate when bidder i reports $\hat{p}_i(\cdot \mid x)$, and let $q^{\mathrm{tr}}(\cdot \mid x)$ denote the aggregate when bidder i reports truthfully $p_i(\cdot \mid x)$, holding $(b, \hat{p}_{-i})$ fixed. Under linear pooling,

$$q(\cdot \mid x) = w_i(b)\,\hat{p}_i(\cdot \mid x) + \sum_{j \neq i} w_j(b)\,\hat{p}_j(\cdot \mid x), \qquad q^{\mathrm{tr}}(\cdot \mid x) = w_i(b)\,p_i(\cdot \mid x) + \sum_{j \neq i} w_j(b)\,\hat{p}_j(\cdot \mid x),$$

and therefore

$$\big\|q(\cdot \mid x) - q^{\mathrm{tr}}(\cdot \mid x)\big\|_1 = w_i(b)\,\big\|\hat{p}_i(\cdot \mid x) - p_i(\cdot \mid x)\big\|_1. \tag{1}$$

Equation (1) captures the core comparative static: report distortion is "attenuated" by the bidder's weight.

For later use we summarize the property we need as an assumption that can be verified for many monotone aggregators beyond linear pooling.

Assumption 5.1 (Individual Lipschitzness of aggregation)**.** *For each bidder i and bid vector b there exists a coefficient $\kappa_i(b) \in [0,1]$ such that for all contexts x and all alternative reports $\hat{p}_i, \hat{p}'_i$,*

$$\big\|q\big(b, (\hat{p}_i(\cdot \mid x), \hat{p}_{-i}(\cdot \mid x))\big) - q\big(b, (\hat{p}'_i(\cdot \mid x), \hat{p}_{-i}(\cdot \mid x))\big)\big\|_1 \leq \kappa_i(b)\,\big\|\hat{p}_i(\cdot \mid x) - \hat{p}'_i(\cdot \mid x)\big\|_1.$$

Linear pooling satisfies Assumption 5.1 with $\kappa_i(b) = w_i(b)$ and equality in (1). In what follows we state results for $\kappa_i(b) = w_i(b)$ to keep formulas interpretable; replacing $w_i(b)$ by $\kappa_i(b)$ gives the corresponding extension.

5.2 Approximate truthfulness under Lipschitz utilities and payments

We now translate the bound (1) into a bound on the bidder's *non-audit* gains from misreporting. By assumption, for each context x,

$$\big|U_i(q,x) - U_i(q^{\mathrm{tr}},x)\big| \leq L_i\,\big\|q(\cdot \mid x) - q^{\mathrm{tr}}(\cdot \mid x)\big\|_1, \qquad \big|z_i(q,x) - z_i(q^{\mathrm{tr}},x)\big| \leq L_i^z\,\big\|q(\cdot \mid x) - q^{\mathrm{tr}}(\cdot \mid x)\big\|_1.$$

Combining these with (1) gives a pointwise bound on the maximal improvement in the bidder's instantaneous objective (utility net of influence payment) from shifting its report at context x:

$$\Big(U_i(q,x) - z_i(q,x)\Big) - \Big(U_i(q^{\mathrm{tr}},x) - z_i(q^{\mathrm{tr}},x)\Big) \leq (L_i + L_i^z)\,w_i(b)\,\big\|\hat{p}_i(\cdot \mid x) - p_i(\cdot \mid x)\big\|_1. \tag{2}$$

Audits impose the opposing force. Under the log score, Section 4 showed that the incremental expected audit penalty from reporting $\hat{p}_i(\cdot \mid x)$ instead of $p_i(\cdot \mid x)$ is exactly $\mathrm{KL}(p_i(\cdot \mid x) \| \hat{p}_i(\cdot \mid x))$, and Pinsker yields a quadratic

lower bound in ℓ_1. Averaging over the audit distribution $\mathcal{D}$ and defining $\delta_i := \mathbb{E}_{x\sim\mathcal{D}}\|\hat{p}_i(\cdot \mid x) - p_i(\cdot \mid x)\|_1$, we have

$$\alpha\gamma\, \mathbb{E}_{x\sim\mathcal{D}}\mathrm{KL}\big(p_i(\cdot \mid x)\,\|\,\hat{p}_i(\cdot \mid x)\big) \;\geq\; \frac{\alpha\gamma}{8}\,\delta_i^2. \tag{3}$$

Putting (2) and (3) together yields our main general incentive inequality: any deviation can generate at most linear gains through influence on q, but necessarily pays at least quadratic costs through the audit layer. Formally, when we evaluate the bidder's payoff difference between an arbitrary report function $\hat{p}_i$ and truthful reporting p_i, holding $(b, \hat{p}_{-i})$ fixed, we obtain the upper envelope

$$\Pi_i(b, \hat{p}_i, \hat{p}_{-i}) - \Pi_i(b, p_i, \hat{p}_{-i}) \;\leq\; (L_i + L_i^z)\, w_i(b)\, \delta_i - \frac{\alpha\gamma}{8}\,\delta_i^2, \tag{4}$$

where δ_i is measured on the audit distribution. (If generation contexts differ substantially from $\mathcal{D}$, then (2) controls gains at those contexts but (3) only disciplines misreports where audits occur; we return to this coverage issue below.)

Two corollaries fall out immediately by optimizing the quadratic upper bound in (4). First, no best response can misreport by more than a scale on the order of $w_i(b)/(\alpha\gamma)$.

Proposition 5.2 (Best-response misreport magnitude)**.** *Fix $(b, \hat{p}_{-i})$ and suppose aggregation is linear pooling. Under log-score audits, any best response $\hat{p}_i$ satisfies*

$$\delta_i \leq \frac{4(L_i + L_i^z)\, w_i(b)}{\alpha\gamma}.$$

Second, truthful reporting is approximately optimal even when it is not exactly dominant (because the bidder can exploit its influence on q). Maximizing the right-hand side of (4) over $\delta_i \geq 0$ yields an upper bound on the value of the best possible deviation relative to truth:

$$\sup_{\hat{p}_i} \Big(\Pi_i(b, \hat{p}_i, \hat{p}_{-i}) - \Pi_i(b, p_i, \hat{p}_{-i})\Big) \leq \frac{2(L_i + L_i^z)^2\, w_i(b)^2}{\alpha\gamma}.$$

Thus, truth-telling is an ε-best response with ε proportional to $w_i(b)^2/(\alpha\gamma)$. The dependence on $w_i(b)$ is economically intuitive: audits primarily constrain the *report*, but misreporting is only valuable insofar as it shifts the *outcome* (the aggregate distribution), and the outcome shift is mediated by the bidder's influence weight.

5.3 Equilibrium implications and aggregate distortion

The preceding results are pointwise best-response bounds holding bidder-by-bidder. In any Nash equilibrium $(b, \hat{p})$, Proposition 5.2 therefore implies that

each bidder's equilibrium misreport on audited contexts is uniformly small when $\alpha\gamma$ is large relative to its Lipschitz scale and its weight. Under linear pooling, the distortion of the platform's aggregate relative to the truthful aggregate can be bounded directly:

$$\mathbb{E}_{x\sim\mathcal{D}}\big\|q(\cdot \mid x)-q^{\mathrm{tr}}(\cdot \mid x)\big\|_1 = \mathbb{E}_{x\sim\mathcal{D}}\Big\|\sum_{i=1}^{n} w_i(b)\big(\hat{p}_i(\cdot \mid x)-p_i(\cdot \mid x)\big)\Big\|_1 \le \sum_{i=1}^{n} w_i(b)\,\delta_i.$$

Combining with $\delta_i \le 4(L_i + L_i^z)w_i(b)/(\alpha\gamma)$ yields the compact bound

$$\mathbb{E}_{x\sim\mathcal{D}}\big\|q(\cdot \mid x) - q^{\mathrm{tr}}(\cdot \mid x)\big\|_1 \le \frac{4}{\alpha\gamma}\sum_{i=1}^{n}(L_i + L_i^z)\,w_i(b)^2. \tag{5}$$

Two practical messages are embedded in (5). First, concentration of weight worsens manipulation risk: if a single bidder has $w_i(b) \approx 1$, then the platform must rely primarily on audits (or on bid-side pricing) to discipline that bidder's report. Second, when weight is diffuse, the aggregate becomes hard to manipulate: even moderate audit intensity can make equilibrium aggregates very close to the truthful aggregate because each bidder's ability to move q is small.

5.4 Audit intensity as a design lever

The bounds above turn (α, γ) into policy parameters with a transparent meaning. Suppose the platform wants to guarantee (on audited contexts) that each bidder's mean ℓ_1 misreport satisfies $\delta_i \le \bar{\delta}$ whenever its bid weight is at most $\bar{w}$. Proposition 5.2 suggests the sufficient condition

$$\alpha\gamma \;\ge\; \frac{4(L_i + L_i^z)\,\bar{w}}{\bar{\delta}}.$$

This expression makes clear why we treat α and γ symmetrically in the theory: both enter only through the product $\alpha\gamma$ in expectation. Operationally, however, the two knobs have different interpretations. Increasing γ raises the penalty when an audit occurs, which can create sharper tail risk for participants and may be limited by regulatory or contractual constraints. Increasing α raises audit frequency, which may be limited by audit compute or by the overhead of obtaining TEE-backed samples. In practice, platforms often have more flexibility to tune a moderate α and then calibrate γ within acceptable monetary bounds.

The parameter m does not appear in (4) because our incentive bounds are in expectation. Its role is reliability: larger m reduces the variance of realized penalties and therefore reduces the profitability of "gambling" on a small number of lucky audit draws. This is especially salient under score

clipping or probability flooring, where boundedness permits direct concentration guarantees and thus more explicit calibration of how large m must be to make deviations unattractive *ex post* rather than only *ex ante*.

A natural extension suggested by the theory is *weight-dependent auditing*: since the temptation to manipulate scales with $w_i(b)$, the platform can keep distortion roughly uniform across bidders by choosing $\alpha_i\gamma_i$ increasing in $w_i(b)$ (or directly in b_i). This is analogous to risk-based supervision in financial regulation: large participants with outsized influence face stricter scrutiny not because they are intrinsically less trustworthy, but because their actions have larger external effects.

5.5 Coverage and limitations: why $\mathcal{D}$ matters

Finally, we emphasize what these results do and do not guarantee. The quadratic audit cost (3) disciplines misreports *on the audit distribution* $\mathcal{D}$. If bidders can predict $\mathcal{D}$ or if $\mathcal{D}$ under-covers the contexts that matter in production, bidders may remain nearly truthful on audited contexts while strategically shading reports elsewhere. In that case the mechanism still behaves as designed during audits, but the platform may not obtain credible distributions where it needs them.

For this reason, the selection of $\mathcal{D}$ is not a mere technical detail: it is an institutional choice that determines the domain over which "approximately truthful reporting" is enforced. The theory clarifies the tradeoff. Broadening $\mathcal{D}$ (or refreshing it frequently) increases coverage and reduces the scope for distribution-shift manipulation, at the cost of potentially higher audit complexity (since audits must faithfully represent the contexts the platform cares about). In our view, this is the right way to read the comparative statics: audits buy truthful reporting where they look, and the platform chooses where to look.

The next section turns to a benchmark in which we can solve the bidder's reporting problem in closed form. That exercise complements the general bounds here: it shows explicitly how the optimal report "tilts" away from p_i in response to marginal incentives, and it recovers the same $1/(\alpha\gamma)$ distortion rate as a limit of the exact first-order conditions.

5.6 Closed-form benchmark: linear valuations under linear pooling

To see the "tilting" logic behind our envelope bound (4) in its cleanest form, we now study a benchmark in which bidder i's incremental (non-audit) payoff is *linear* in the aggregate distribution, and aggregation is *linear* in reports. The advantage of this case is not realism per se—many bidders have nonlinear objectives and many platforms impose safety layers that make q nonlinear in $\hat{p}$—but transparency: we can solve for the unique best-response

report and read the $1/(\alpha\gamma)$ rate directly off the first-order conditions.

Setup (single context, linear objective). Fix a context x and suppress it from notation. Let the platform use linear pooling so that, holding $(b, \hat{p}_{-i})$ fixed,

$$q(t) = w_i\, \hat{p}_i(t) + r_{-i}(t), \qquad \text{where} \qquad r_{-i}(t) := \sum_{j \neq i} w_j\, \hat{p}_j(t)$$

is the component of the aggregate coming from other bidders. Suppose bidder i's per-step utility net of influence payment can be written as

$$U_i(q) - z_i(q) = \sum_{t \in T} c_{i,t}\, q(t), \tag{6}$$

for some coefficients $c_{i,t} \in \mathbb{R}$ capturing the bidder's marginal value of increasing the probability of token t (possibly net of payment effects). Under (6), bidder i's report influences its payoff only through the linear term $w_i \sum_t c_{i,t}\hat{p}_i(t)$, since $\sum_t c_{i,t} r_{-i}(t)$ is a constant with respect to $\hat{p}_i$.

Under log-score audits, the bidder's expected audit penalty at this context equals the cross-entropy

$$\mathbb{E}_{y \sim p_i}\big[-\log \hat{p}_i(y)\big] = -\sum_{t \in T} p_i(t) \log \hat{p}_i(t) = H(p_i) + \mathrm{KL}(p_i \| \hat{p}_i),$$

where $H(p_i)$ does not depend on $\hat{p}_i$ and can be dropped from the optimization. Thus, for this fixed context, bidder i solves the concave program

$$\max_{\hat{p}_i \in \Delta(T)} \; w_i \sum_{t \in T} c_{i,t}\, \hat{p}_i(t) + \alpha\gamma \sum_{t \in T} p_i(t) \log \hat{p}_i(t). \tag{7}$$

The objective in (7) makes the economic tradeoff stark. The linear term rewards moving mass toward high-$c_{i,t}$ tokens, while the log term is an *entropic barrier* anchored at p_i: placing too little mass on a token that the true model emits with nontrivial probability is punished sharply.

Best-response report (closed form). Because (7) is strictly concave on the simplex (the log term is strictly concave on the interior, and the constraint set is convex), the maximizer is unique and lies in the interior whenever $p_i(t) > 0$ for all $t \in T$.[1] Writing the Lagrangian for (7) with multiplier λ_i for the simplex constraint $\sum_t \hat{p}_i(t) = 1$,

$$\mathcal{L}(\hat{p}_i, \lambda_i) = w_i \sum_t c_{i,t}\, \hat{p}_i(t) + \alpha\gamma \sum_t p_i(t) \log \hat{p}_i(t) + \lambda_i \Big(1 - \sum_t \hat{p}_i(t)\Big),$$

[1] If $p_i(t) = 0$ for some t, then the log-score term does not discipline $\hat{p}_i(t)$ directly; the solution still exists but may place zero mass on such tokens depending on $c_{i,t}$. In implementations one typically imposes probability floors (or clipped scores), in which case the optimization is over a truncated simplex and the same KKT logic applies with complementary slackness at the floor.

the first-order conditions for each $t \in T$ are

$$w_i c_{i,t} + \alpha\gamma \frac{p_i(t)}{\hat{p}_i(t)} - \lambda_i = 0, \qquad \text{equivalently} \qquad \hat{p}_i(t) = \frac{\alpha\gamma\, p_i(t)}{\lambda_i - w_i c_{i,t}}. \tag{8}$$

The multiplier λ_i is pinned down by normalization:

$$\sum_{t \in T} \frac{\alpha\gamma\, p_i(t)}{\lambda_i - w_i c_{i,t}} = 1, \qquad \text{with} \qquad \lambda_i > \max_{t \in T} w_i c_{i,t}. \tag{9}$$

Equations (8)–(9) reproduce the rational-form best response stated in Proposition 4 in the global context, here specialized to a single context and with the explicit factor w_i from linear pooling.

Two features are worth highlighting. First, the report $\hat{p}_i$ *tilts* away from p_i toward tokens with higher $c_{i,t}$, but it does so in a way that remains absolutely continuous with respect to p_i (when $p_i(t) > 0$): the penalty makes it expensive to "pretend" that likely tokens are impossible. Second, w_i enters only through the combination $w_i c_{i,t}$: if the bidder has little influence on q, then even a large private preference $c_{i,t}$ has little strategic value because it cannot move the aggregate much.

Asymptotic distortion rate $\Theta(1/(\alpha\gamma))$. The closed form also makes the $\alpha\gamma$ comparative static precise. To see the scaling, suppose the coefficients are uniformly bounded, $|c_{i,t}| \le \bar{c}$. When $\alpha\gamma$ is large, the multiplier λ_i is of order $\alpha\gamma$, and we can expand (8) as a perturbation around truth. A convenient way to express the leading term is to write $\lambda_i = \alpha\gamma + w_i \bar{c}_i + o(1)$, where $\bar{c}_i$ is a constant of order $\bar{c}$ chosen so that $\sum_t \hat{p}_i(t) = 1$. Plugging into (8) and expanding $(\lambda_i - w_i c_{i,t})^{-1}$ around $\alpha\gamma$ yields the approximation

$$\hat{p}_i(t) = p_i(t) \left(1 + \frac{w_i}{\alpha\gamma}\big(c_{i,t} - \mathbb{E}_{t \sim p_i}[c_{i,t}]\big)\right) + O\left(\frac{w_i^2 \bar{c}^2}{(\alpha\gamma)^2}\right), \tag{10}$$

where the mean-centering arises because the simplex constraint forces $\sum_t (\hat{p}_i(t) - p_i(t)) = 0$. Equation (10) makes the economic content of the "entropic barrier" interpretation concrete: the audit layer behaves like a regularizer that penalizes departures from p_i, so the bidder only shifts probabilities in proportion to *relative* marginal values $c_{i,t} - \mathbb{E}_{p_i}[c_{i,t}]$.

From (10) we immediately obtain the $1/(\alpha\gamma)$ distortion rate. For instance, using $\sum_t p_i(t) = 1$ and $|c_{i,t} - \mathbb{E}_{p_i}[c_{i,t}]| \le 2\bar{c}$,

$$\|\hat{p}_i - p_i\|_1 = \sum_{t \in T} |\hat{p}_i(t) - p_i(t)| \le \frac{2w_i}{\alpha\gamma} \sum_{t \in T} p_i(t)\, |c_{i,t} - \mathbb{E}_{p_i}[c_{i,t}]| + O\left(\frac{w_i^2 \bar{c}^2}{(\alpha\gamma)^2}\right) = O\left(\frac{w_i \bar{c}}{\alpha\gamma}\right). \tag{11}$$

Moreover, the rate is typically tight: whenever $c_{i,t}$ is not almost surely constant under p_i (so there is something to "tilt" toward), the leading term in (10) is nonzero and one can lower bound $\|\hat{p}_i - p_i\|_1$ by a constant multiple of $w_i/(\alpha\gamma)$, yielding $\|\hat{p}_i - p_i\|_1 = \Theta(w_i/(\alpha\gamma))$ up to problem-dependent constants.

How much can a bidder gain from optimal misreporting? The benchmark also lets us quantify the *value* of strategic shading. For large $\alpha\gamma$, the bidder's best deviation payoff relative to truth is of order $w_i^2/(\alpha\gamma)$, consistent with the envelope bound derived earlier. Intuitively, the bidder moves q by $w_i(\hat{p}_i - p_i)$, and we have just seen that $\hat{p}_i - p_i$ is of order $w_i/(\alpha\gamma)$; the resulting first-order gain is therefore second order in w_i and first order in $1/(\alpha\gamma)$.

One way to make this precise is to take a second-order expansion of $\mathrm{KL}(p_i\|\hat{p}_i)$ around $\hat{p}_i = p_i$, which gives

$$\mathrm{KL}(p_i\|\hat{p}_i) = \frac{1}{2}\sum_{t\in T}\frac{(\hat{p}_i(t) - p_i(t))^2}{p_i(t)} + o\big(\|\hat{p}_i - p_i\|_2^2\big)\,,$$

and then solve the approximate quadratic program obtained by substituting this into (7) and imposing the simplex constraint. The optimizer of the quadratic approximation satisfies $\hat{p}_i(t) - p_i(t) \propto p_i(t)\big(c_{i,t} - \mathbb{E}_{p_i}[c_{i,t}]\big)$, matching (10), and the maximal improvement in the objective is proportional to a variance:

$$\left(\Pi_i^\star - \Pi_i^{\mathrm{tr}}\right) = \frac{w_i^2}{2\alpha\gamma}\,\mathrm{Var}_{t\sim p_i}(c_{i,t}) + O\left(\frac{w_i^3\bar{c}^3}{(\alpha\gamma)^2}\right), \tag{12}$$

where $\Pi_i^\star$ denotes the per-context optimum of (7) and Π_i^{tr} is the value at $\hat{p}_i = p_i$. Equation (12) is useful as a calibration heuristic: it suggests that what matters for the temptation to misreport is not merely the magnitude of $c_{i,t}$ but its dispersion under the bidder's true model.

Welfare and revenue implications (and why the platform should not rely on penalties as "revenue"). Although our mechanism is designed around individual incentives, the benchmark clarifies how auditing affects aggregate performance.

On the welfare side, misreporting is a classic externality: bidder i can manipulate q in a direction that is privately valuable, but the resulting shift may be harmful to other bidders or to platform objectives (e.g., user satisfaction). Under linear pooling the welfare-relevant object is the induced distortion in the aggregate distribution,

$$q - q^{\mathrm{tr}} = w_i(\hat{p}_i - p_i).$$

Combining this identity with (11) yields an immediate scaling law:

$$\|q - q^{\mathrm{tr}}\|_1 = w_i\|\hat{p}_i - p_i\|_1 = O\left(\frac{w_i^2\bar{c}}{\alpha\gamma}\right). \tag{13}$$

Thus, even in the worst case where private incentives are adversarial to welfare, audits reduce the *outcome* distortion at a rate $w_i^2/(\alpha\gamma)$. This squares

with the economic intuition emphasized earlier: influence is already attenuated by w_i, and audits then attenuate the report distortion by $1/(\alpha\gamma)$.

On the revenue side, it is tempting to view audit penalties as a source of income. The benchmark cautions against this interpretation. Under the log score, the expected audit penalty decomposes into a constant term $H(p_i)$ plus $\mathrm{KL}(p_i\|\hat{p}_i)$. At the optimal report (8), the *incremental* component $\mathrm{KL}(p_i\|\hat{p}_i)$ is small when audits are effective: since $\|\hat{p}_i - p_i\|_1 = O(w_i/(\alpha\gamma))$, Pinsker implies $\mathrm{KL}(p_i\|\hat{p}_i) = O(w_i^2/(\alpha\gamma)^2)$, and therefore the expected incremental penalty $\alpha\gamma\,\mathrm{KL}(p_i\|\hat{p}_i)$ is only

$$\alpha\gamma\,\mathrm{KL}(p_i\|\hat{p}_i) = O\left(\frac{w_i^2}{\alpha\gamma}\right), \tag{14}$$

which vanishes as $\alpha\gamma$ grows. In other words, when the audit system is doing its job (reports are close to truth), it does not generate substantial incremental penalty payments in expectation. This is a desirable feature from a policy perspective: it aligns the mechanism with a compliance logic (penalties as deterrence) rather than a fiscal logic (penalties as a profit center), reducing the platform's temptation to set audit parameters in a way that extracts rents rather than ensures veracity.

Finally, because our influence payments z_i are designed to support bid-side monotonicity under truthful inputs, (13) and (14) together suggest a practical separation of roles. The influence-pricing layer is the primary instrument for allocating influence and raising revenue, while the audit layer is primarily an enforcement primitive that keeps the informational inputs (reported distributions) credible. In the benchmark, stronger audits reduce outcome distortions and simultaneously shrink the scope for profitable report shading; the platform does not need (and should not expect) to finance the system through large penalty collections.

Interpretation and limitations. The closed form (8) is also a reminder of where the analysis can break. The denominator $\lambda_i - w_i c_{i,t}$ must remain positive for all t, which requires $\lambda_i > \max_t w_i c_{i,t}$. When $c_{i,t}$ can be extremely large (or unbounded), the bidder would like to push $\hat{p}_i$ toward a corner of the simplex, and the log barrier becomes the only force preventing near-degenerate reports; in practice this is precisely the regime where implementations impose probability floors, clip scores, or restrict admissible reporting classes. Moreover, in realistic systems the mapping from reports to outcomes is not exactly linear pooling: the platform may apply nonlinear temperature scaling, safety filters, or other post-processing that breaks separability across tokens. In such cases the first-order conditions no longer yield (8) in closed form, and one must solve a constrained optimization (often numerically) to characterize best responses.

With these caveats, the benchmark serves its intended purpose: it turns the abstract "linear gain versus quadratic cost" story into an explicit formula,

and it shows that the $1/(\alpha\gamma)$ misreport rate we derived by envelope arguments is not an artifact of loose bounding. It is the genuine scaling of optimal strategic behavior when bidders can profit from shaping token probabilities but face proper-scoring audits that penalize deviations from the committed model.

Implementation: enforcing a single committed reporting function. Our incentive statements implicitly rely on a mundane but crucial engineering property: bidder i must be unable to "answer two different questions," i.e., to use one function during generation and a different function during audits. Formally, the mechanism assumes a single committed mapping $\hat{p}_i : \mathcal{X} \to \Delta(T)$ that the platform can query at arbitrary contexts x, and that the same committed object is the one scored against draws $y \sim p_i(\cdot \mid x)$ in the audit. In practice, enforcing this single-function discipline is not automatic, because a sophisticated bidder can condition behavior on metadata (request headers, timing, IP ranges), on whether the platform is likely to be auditing, or even on hidden "audit-mode" bits if those are exposed by the interface.

A robust implementation therefore treats the reporting function as a *measured program* with a narrow I/O boundary. The high-level design is: bidder i supplies code and (optionally) model parameters implementing $\hat{p}_i(\cdot \mid x)$; these are loaded into a trusted execution environment (TEE) that (a) exposes only a fixed RPC interface "input context x, output vector $\hat{p}_i(\cdot \mid x)$," (b) seals any internal state, and (c) produces an attestation quote binding the running measurement (code hash and relevant configuration) to a public key that signs outputs. The platform then accepts reports only if they are signed by that attested key and reference a specific measurement. This turns "commitment to $\hat{p}_i$" from a paper assumption into a verifiable artifact: the platform is no longer trusting the bidder to behave consistently; it is trusting the hardware-rooted measurement and the verification logic.

Two details matter for incentive compatibility rather than mere integrity. First, the TEE must commit to *all* sources of behavioral branching that could induce different answers on audit and non-audit inputs. This includes the code and weights, but also any external calls (tool use, network access), randomness sources, and configuration flags. A conservative stance is to disallow outbound network access and to require either determinism or explicitly attested randomness (e.g., TEE-provided randomness with transcript binding). Second, the platform must ensure that the request visible to the TEE is indistinguishable across audited and non-audited contexts. If the bidder can detect audit queries, it can trivially report $\hat{p}_i = p_i$ on audits (minimizing expected penalty) while shading on production contexts. Operationally, the simplest defense is *interface uniformity*: the platform always queries $\hat{p}_i(\cdot \mid x)$ through the same channel, with the same formatting, and

never tags a request as "audit." The audit then consists only of the platform privately selecting certain contexts x^a and later scoring the bidder's committed outputs on those contexts against $y^a \sim p_i(\cdot \mid x^a)$ obtained via the separately attested true model.

Remote attestation and transcript binding. Remote attestation is most useful when it is paired with transcript binding and key management that make equivocation difficult. Concretely, we want three properties.

(i) *Measurement binding.* The attestation quote should bind a measurement M_i (hash of code, weights, and configuration) to a signing key K_i generated inside the enclave. The platform verifies the quote once per (re)commitment and records (i, M_i, K_i).

(ii) *Query-response authenticity.* Each reported vector $\hat{p}_i(\cdot \mid x)$ should be signed under K_i and include a nonce and a context hash, so that the platform can prove to itself (or to an external auditor) that it scored the bidder on the output actually produced by the committed code at that context. This is mainly a governance and dispute-resolution feature, but it also disciplines subtle implementation failures where the platform might accidentally score an output produced by a different version.

(iii) *Anti-rollback and state discipline.* Even if $\hat{p}_i$ is intended to be stateless, many real implementations are not: they cache, they adapt, they maintain counters, or they update internal calibration. If state is allowed, then commitment must specify whether $\hat{p}_i$ is permitted to change over time, and if so on what schedule. For the model we analyze (a single committed reporting function per session), the clean analogue is to enforce that the enclave uses a sealed, read-only model snapshot for the session and that any update requires a new attestation measurement M_i' and a new commitment. Rollback protection (e.g., monotone counters) prevents the bidder from presenting an old, more "audit-friendly" snapshot on demand.

These are not purely technical niceties: they are the operational counterpart to the economic assumption that the bidder cannot costlessly condition its report on hidden information about the auditing process. If we fail here, the relevant distribution in Proposition-style bounds becomes the conditional distribution of contexts *given that the bidder believes it is being audited*, which is exactly the distribution-shift loophole that undermines proper-scoring incentives.

Cryptographic commitment variants (when TEEs are unavailable or insufficient). TEEs are a natural enforcement primitive, but not the only one. A weaker (and sometimes deployable) alternative is a cryptographic commitment to a container image plus a reproducible-build pipeline, combined with auditing by *re-execution* on platform-controlled infrastructure. The bidder commits to a hash of the code and weights, the platform

runs that code in a sandbox, and the bidder is paid according to outputs produced there. This approach avoids trusting the bidder's hardware, but it assumes the platform can run the bidder's model and that IP concerns are addressed.

At the opposite extreme, one can imagine verifiable computation (SNARKs or succinct proofs) that the reported $\hat{p}_i(\cdot \mid x)$ is the output of a committed circuit on input x. This would offer strong integrity without hardware trust, but current performance makes it unrealistic for large neural models at token-level cadence. In between lies a pragmatic compromise: use TEEs for inference integrity, but also require a public commitment (hash registry) to the model version, so that any unilateral update is observable and can trigger a re-commitment cycle with updated audit parameters.

In our view, the mechanism design point is simple: the more cheaply bidders can equivocate about $\hat{p}_i$, the more the audit probability α and penalty scale γ must do the heavy lifting, and the less sharp our approximate-truthfulness guarantees become in practice.

Probability floors, clipped scores, and finite-sample auditing. Even with perfect commitment, score-based penalties have implementation pitfalls. The log score $S(\hat{p}, y) = -\log \hat{p}(y)$ is unbounded above when $\hat{p}(y) \to 0$, which creates two issues. First, unbounded penalties are politically and contractually hard to sustain (a single unlucky audit could bankrupt a bidder). Second, unbounded losses complicate finite-m calibration because concentration is poor without tail control.

A standard remedy is to impose a probability floor $\hat{p}_i(t \mid x) \geq \epsilon$ (equivalently, to clip the score), scoring instead

$$S_\epsilon(\hat{p}, y) = -\log(\max\{\hat{p}(y), \epsilon\}).$$

This preserves the "properness" logic approximately on the truncated simplex and yields bounded penalties $S_\epsilon(\hat{p}, y) \leq -\log \epsilon$, enabling meaningful concentration guarantees as m grows. Economically, floors slightly weaken incentives around extremely low-probability events; operationally, they prevent the mechanism from turning rare modeling discrepancies into catastrophic liabilities. In deployments, ϵ and m are coupled design parameters: smaller ϵ strengthens truthfulness but increases variance; larger m reduces variance but increases audit cost. Our theoretical bounds treat $\alpha\gamma$ as the main lever, but in practice (ϵ, m) are part of the same deterrence budget.

When closed forms disappear: nonlinear aggregation and safety layers. The benchmark in Section 5.6 is deliberately transparent, but production systems rarely satisfy its separability. Platforms often transform or post-process reports before sampling: temperature scaling, nucleus/top-k truncation, safety filters, policy constraints (e.g., disallowing certain tokens),

or reranking layers that depend on external classifiers. These features make the mapping $\hat{p} \mapsto q$ nonlinear and sometimes discontinuous. Likewise, influence pricing rules based on stable sampling can introduce kinks when small changes in q flip tie-breaking events.

From the bidder's perspective, this changes the optimization problem qualitatively. Instead of the concave program (7), the bidder faces something like

$$\max_{\hat{p}_i \in \mathcal{H}} \mathbb{E}\big[U_i(q(b,\hat{p}),x) - z_i(b,\hat{p})\big] - \alpha\gamma\, \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{y\sim p_i(\cdot|x)}\big[S(\hat{p}_i(\cdot \mid x),y)\big],$$

where $\mathcal{H}$ may encode admissible reporting classes (e.g., model families, floors, or smoothness constraints), and $q(b,\hat{p})$ may include nonconvex operations. Even if the audit term remains concave in each $\hat{p}_i(\cdot \mid x)$, the composition through q can destroy concavity, yielding multiple local optima and making best responses sensitive to optimization details.

This is precisely why we emphasize Lipschitz-style envelope bounds in the general model: they do not require us to solve the bidder's problem exactly, and they remain informative whenever (i) the bidder's influence on outcomes is bounded and (ii) the audit term grows like a divergence in the report. In other words, we do not need closed-form best responses to argue that sufficiently strong audits shrink the profitable scope for manipulation; we need only that manipulation cannot move q too much and that misreporting cannot hide from the scoring rule on the audited distribution.

When numerical methods are required (and what is being solved). Numerical computation enters in two places, and it is helpful to separate them.

(A) Bidder-side optimization (strategic shading under complex q). A bidder trying to compute its own optimal report in a nonlinear system will almost surely rely on numerical methods. If $\hat{p}_i(\cdot \mid x)$ is produced by a parametric model $\hat{p}_\theta(\cdot \mid x)$ (as it would be in any realistic implementation), then choosing $\hat{p}_i$ is choosing parameters θ. The audit penalty becomes an expected cross-entropy against samples from p_i on contexts $x \sim \mathcal{D}$, i.e., a regularizer that pulls $\hat{p}_\theta$ toward p_i on the audit distribution. The bidder's problem is then a regularized training objective, potentially nonconvex if $\hat{p}_\theta$ is a neural network, and may need stochastic gradient methods, projection onto floors, and careful treatment of constraints (e.g., simplex constraints are automatic under softmax parameterizations, but floors are not).

(B) Platform-side calibration (choosing $(\alpha,\gamma,m,\epsilon)$ and stress testing). Even if the platform does not compute equilibria, it must pick audit parameters that achieve a desired distortion bound at acceptable operational cost. The relevant constants—effective Lipschitz sensitivity of welfare or safety metrics to $\|q-q'\|_1$, or the distribution of scores under realistic prompts—are

rarely known analytically. Here numerical work is unavoidable: simulation-based estimation of sensitivity, empirical evaluation of score variance under representative $\mathcal{D}$, and worst-case testing under adversarially chosen shading heuristics. This is also where the nonlinearity of safety layers matters most, because it can amplify small report changes in corner cases (e.g., a filter that sharply truncates probabilities when a classifier threshold is crossed). In such regimes, conservative parameter choices may be justified even if average-case behavior is benign.

The broader message is that closed forms are an intellectual aid, not a deployment requirement. The mechanism's enforceability hinges on commitment and auditability; the platform's practical success hinges on calibration against the true end-to-end pipeline, which is an empirical and numerical exercise.

Practical takeaways (and remaining limitations). We can summarize the implementation logic as follows. The proper-scoring audit layer is only as good as (i) the platform's ability to force a single committed reporting function, and (ii) the representativeness and secrecy of the audit contexts relative to production. TEEs with remote attestation provide a plausible enforcement primitive for (i), provided the query interface does not leak audit status and the measurement captures all behaviorally relevant dependencies. Cryptographic commitments and re-execution can substitute when the platform can run bidder code directly; fully general zero-knowledge verification remains aspirational.

At the same time, we should be explicit about limitations. TEEs have nonzero attack surfaces and can leak via side channels; implementing perfect indistinguishability between audit and production queries is operationally hard; and nonlinear post-processing can create incentive "cliffs" where small manipulations have outsized effects. These are not reasons to abandon the mechanism; rather, they clarify where the economic model is doing abstraction work. Our analysis illuminates the tradeoff between influence and veracity under disciplined reporting, but the discipline itself is an engineering and governance problem. Designing the commitment interface, selecting $\mathcal{D}$, and calibrating (α, m, γ) are therefore part of the mechanism, not afterthoughts, and they are the natural bridge from the benchmark theory to the extensions we consider next.

Extensions: dynamics, adaptivity, and strategic behavior beyond the benchmark. The baseline analysis treats each queried context x as an independent instance at which a bidder reports $\hat{p}_i(\cdot \mid x)$ and is (occasionally) scored against draws from its committed $p_i(\cdot \mid x)$. In deployments, however, (i) generation is multi-step and contexts are endogenous to past sampled tokens; (ii) auditing can be made adaptive and risk-based; (iii) penalties

are often implemented via deposits and slashing rather than open-ended ex post fines; (iv) bidders may have some control over which contexts are realized (distribution shift); and (v) bidders may coordinate. None of these considerations negate the role of strictly proper scoring, but each changes what we mean by "representative auditing" and how we translate an expected divergence penalty into a practical deterrent.

Multi-step generation and path-dependent contexts. In a K-step generation, the context x_k at step k is a function of the initial prompt and the realized history $y_{1:k-1}$. This creates two conceptual changes. First, the bidder's report affects not only the current token distribution $q_k(\cdot \mid x_k)$, but also the *future* distribution of contexts by changing earlier sampled tokens. Second, if utility is defined over entire transcripts (e.g., conversions, downstream task success, safety violations), then a small perturbation in q_k can have compounding effects.

A convenient way to retain tractable bounds is to treat the full prefix as the context: $x_k = (x_1, y_{1:k-1})$, so that $p_i(\cdot \mid x_k)$ and $\hat{p}_i(\cdot \mid x_k)$ remain well-defined conditional distributions over the next token. The mechanism then remains pointwise: at each visited x_k, the platform queries $\hat{p}_i(\cdot \mid x_k)$, forms q_k, and samples $y_k \sim q_k$. Audits sample $x \sim \mathcal{D}$ from a held-out distribution over *prefixes* rather than initial prompts.

The economic subtlety is that a bidder may prefer to misreport early in order to steer the trajectory into "high-value" regions later. Our Lipschitz envelope approach still applies provided we measure utility sensitivity in a way that accounts for such steering. One sufficient (if conservative) condition is a per-step Lipschitz bound on the *continuation value*: define $V_{i,k}(q_k, x_k)$ as bidder i's expected remaining gross utility from step k onward under the platform's policy, holding fixed the reporting profile. If for each k,

$$|V_{i,k}(q, x) - V_{i,k}(q', x)| \le \widetilde{L}_{i,k} \, \|q - q'\|_1,$$

then a deviation that changes the aggregate by $\|q_k - q'_k\|_1$ at step k yields at most $\sum_{k=1}^{K} \widetilde{L}_{i,k} \|q_k - q'_k\|_1$ non-audit benefit. Under linear pooling, $\|q_k - q'_k\|_1 \le w_i(b) \, \|\hat{p}_i(\cdot \mid x_k) - p_i(\cdot \mid x_k)\|_1$, and one obtains a dynamic analogue of the static tradeoff:

$$\Pi_i(\hat{p}_i) - \Pi_i(p_i) \;\le\; w_i(b) \sum_{k=1}^{K} (\widetilde{L}_{i,k} + \widetilde{L}^{z}_{i,k}) \, \mathbb{E}\big[\|\hat{p}_i(\cdot \mid x_k) - p_i(\cdot \mid x_k)\|_1\big] - \alpha\gamma \, \mathbb{E}_{x \sim \mathcal{D}} \mathrm{KL}(p_i \| \hat{p}_i),$$

where the expectation on the left is over the endogenous trajectory of contexts induced by the platform's sampling. The main design implication is that in multi-step systems the relevant "benefit slope" can be larger early in the trajectory (because early perturbations influence many future steps), so audit strength should be calibrated against the largest continuation sensitivity, not merely a myopic per-token effect.

Adaptive audits and risk-based enforcement. A fixed audit probability α is analytically clean but operationally blunt. Platforms typically have side information that correlates with manipulation incentives and harms: unusual bid spikes, sudden report changes, contexts associated with safety-critical decisions, or outputs near policy boundaries. This motivates *adaptive auditing*, where the platform chooses an audit intensity $\alpha(x)$ (or triggers audits after observing suspicious behavior) subject to a budget constraint.

At the level of expected incentives, the proper-scoring logic composes neatly: under log scoring, the incremental expected audit cost from misreporting becomes

$$\gamma\, \mathbb{E}_{x\sim\mathcal{D}}\big[\alpha(x)\, \mathrm{KL}(p_i(\cdot \mid x)\|\hat{p}_i(\cdot \mid x))\big],$$

which is simply a reweighting of contexts. This suggests a normative prescription: place more audit mass on contexts where either (i) the bidder can create larger outcome distortions (large effective Lipschitz constants) or (ii) the social cost of distortion is high (safety, fairness, or legal exposure). In effect, $\alpha(x)$ is a shadow price on veracity that can be targeted.

Two caveats matter. First, the audit policy itself must not create a predictable "audit shadow" that bidders can condition on. If bidders can infer that certain contexts are never audited, they will concentrate misreports there. Thus, even risk-based auditing should retain randomness and some minimum coverage: e.g., $\alpha(x) \geq \underline{\alpha} > 0$ for all x in the operational support, with additional mass allocated adaptively. Second, adaptive audits raise governance questions: if $\alpha(x)$ is chosen after observing bidder behavior, bidders may worry about discretionary enforcement. A practical compromise is to commit (cryptographically or contractually) to an audit policy class—for example, a published detector with fixed thresholds—and to log the randomness used for audit triggers. This protects the platform from accusations of arbitrary punishment while preserving deterrence.

A further extension is *sequential auditing*: rather than fixing m contexts per audit event, the platform can perform a sequential probability ratio test on the stream of scored outcomes, escalating scrutiny when cumulative evidence of misreporting accumulates. Such designs can reduce audit cost in benign regimes while maintaining strong worst-case deterrence, but they require bounded or clipped scores for concentration and for predictable liability.

Bond and slashing variants (liability control and budget balance). In many markets, "penalties" are implemented through deposits, chargebacks, or escrow rather than ex post invoices. This is particularly natural with unbounded scores (like the log score) or with bidders of uncertain creditworthiness. A bond/slashing design replaces the audit payment term with a rule of the form: bidder i posts a bond B_i at commitment time; the

platform computes an audit loss ℓ_i (e.g., $\gamma \cdot \frac{1}{m}\sum_{j=1}^{m} S(\hat{p}_i, y_j^a)$), and slashes $\min\{\ell_i,\ B_i\}$ from the bond, returning the remainder at settlement.

Economically, the bond changes the *risk* of participation and the enforceability of large penalties; it does not change the first-order incentive effect so long as the bond is large enough that the slashing constraint is rarely binding on equilibrium paths. Operationally, bonds provide three benefits. (i) They cap platform counterparty risk: the platform can enforce penalties up to B_i without collection. (ii) They reduce political objections to unbounded losses by replacing them with a posted maximum exposure. (iii) They allow a budget-balanced implementation: slashed funds can finance auditing or subsidize users harmed by manipulation.

The cost is that small bidders may be bond-constrained, which can distort entry. One mitigation is to allow *tiered* mechanisms: lower bonds with higher audit rates (or stronger clipping) for small participants, and higher bonds with lower audit intensity for large participants. Formally, one can treat $(\alpha_i, \gamma_i, B_i)$ as bidder-specific contract parameters, chosen to maintain a target bound on δ_i given the bidder's weight $w_i(b)$ and observed volatility.

Strategic selection of contexts and the distribution-shift loophole. Our approximate-truthfulness guarantees are only as strong as the connection between audited contexts and the contexts that matter for outcomes. If auditing draws $x \sim \mathcal{D}$ but production contexts are distributed as $\mathcal{D}^{\mathrm{prod}}$, then a bidder may be truthful on $\mathcal{D}$ and manipulative on contexts that occur frequently in production but rarely (or never) in audits. This is not merely a theoretical nuisance: in interactive systems bidders may influence prompts (through UI suggestions), traffic routing (through publisher relationships), or downstream chains (through tool use), all of which can shift the context distribution.

A simple way to make this dependence explicit is to decompose the expected benefit of misreporting under production as

$$\mathbb{E}_{x\sim\mathcal{D}^{\mathrm{prod}}}\big[\Delta_i(x)\big] = \mathbb{E}_{x\sim\mathcal{D}}\big[\Delta_i(x)\big] + \Big(\mathbb{E}_{x\sim\mathcal{D}^{\mathrm{prod}}} - \mathbb{E}_{x\sim\mathcal{D}}\Big)\big[\Delta_i(x)\big],$$

where $\Delta_i(x)$ denotes the bidder's per-context gain from moving $q(\cdot \mid x)$ via misreporting. The second term is an adversarial "distribution shift" wedge. Bounding it requires either (i) control of $\mathrm{TV}(\mathcal{D}^{\mathrm{prod}}, \mathcal{D})$ and bounded $\Delta_i(x)$, or (ii) audit designs that explicitly cover the production support.

This motivates two practical design principles. First, audits should be drawn from a distribution that tracks production, possibly as a mixture

$$\mathcal{D} = (1-\rho)\,\mathcal{D}^{\mathrm{prod}} + \rho\,\mathcal{D}^{\mathrm{stress}},$$

where $\mathcal{D}^{\mathrm{stress}}$ oversamples safety-critical or manipulation-prone regions, and ρ is chosen for coverage. Second, the platform should treat the choice of $\mathcal{D}$

as part of the mechanism: it should be versioned, periodically refreshed, and kept partially private. Full transparency of $\mathcal{D}$ invites gaming; full secrecy invites governance concerns. A middle ground is to publish broad *classes* of audited contexts and statistical summaries while keeping the exact draws unpredictable.

An even stronger approach is *importance-weighted* auditing: sample x from an easy-to-sample proposal distribution and weight the score by a factor proportional to $\mathcal{D}^{\text{prod}}(x)/\mathcal{D}(x)$, targeting production truthfulness directly. This is statistically efficient only when weights are controlled; otherwise variance can explode, again pointing to clipping and careful calibration.

Collusion and coalition-proofing. Finally, we should ask what happens when bidders coordinate. Proper scoring rules discipline each bidder against its own p_i, so collusion cannot eliminate the audit term. However, collusion can change the *mapping from individual deviations to aggregate influence.* Under linear pooling, the aggregate perturbation is

$$q(\cdot \mid x) - q^{\text{truth}}(\cdot \mid x) = \sum_{i=1}^{n} w_i(b) \big(\hat{p}_i(\cdot \mid x) - p_i(\cdot \mid x)\big).$$

A coalition can split a desired aggregate perturbation across members. Because our audit lower bound is roughly quadratic in each member's deviation (via Pinsker), while the coalition's benefit is roughly linear in the *sum* of weighted deviations, spreading manipulation can reduce total penalty for a given aggregate shift.

A stylized calculation makes this clear. Suppose a coalition C aims to induce a fixed aggregate L_1-deviation $s(x) := \|q - q^{\text{truth}}\|_1$ on audited contexts. The coalition chooses individual deviations $d_i(x) := \|\hat{p}_i - p_i\|_1$ to satisfy (heuristically) $\sum_{i \in C} w_i d_i \approx s$. The sum of quadratic penalties is minimized when deviations are allocated proportional to weights, yielding the Cauchy–Schwarz bound

$$\sum_{i \in C} d_i^2 \;\geq\; \frac{s^2}{\sum_{i \in C} w_i^2}.$$

Since $\sum_{i \in C} w_i^2$ can be much smaller than $\big(\sum_{i \in C} w_i\big)^2$ when weights are dispersed, collusion can reduce the coalition's effective marginal penalty for moving q. This is the same geometry that makes quadratic regularization weaker against coordinated, distributed perturbations.

There are at least three responses. First, the platform can scale audit intensity with influence, for example by setting bidder-specific penalty multipliers γ_i increasing in $w_i(b)$ (or in b_i). In the bound above, increasing γ_i effectively increases the cost of allocating deviation to high-weight members, making distributed manipulation less attractive. Second, the platform

can introduce *joint* consistency checks that score the coalition on aggregate properties of reports (e.g., comparing q to a separately estimated baseline), though such checks are no longer strictly proper in the individual sense and must be designed carefully to avoid penalizing legitimate correlation. Third, the platform can rely on market design and governance: limit identity splitting, impose KYC-style requirements, monitor correlated bid/report changes, and treat suspicious clustering as a compliance risk.

We should also distinguish collusion on reporting from collusion on bidding. Influence payments address bid-side incentives under monotonicity assumptions, but identity splitting can allow bidders to manipulate weights $w_i(b)$ and the effective audit burden. Thus, anti-collusion enforcement is partly a mechanism-design issue (how weights and payments scale) and partly an identity and compliance issue (what constitutes a bidder).

Summary: what changes and what persists. Across these extensions, two messages persist. First, proper-scoring audits remain a disciplined way to tie reported distributions to verifiable generative behavior, and their expected effect is still divergence-like. Second, the *calibration* of auditing must track the true sources of leverage: dynamic continuation values in multi-step generation, concentrated risk regions under adaptive enforcement, representative coverage under distribution shift, and coalition geometry under coordinated manipulation. Our benchmark bounds are therefore best read as a template: they identify the linear benefit versus quadratic cost tradeoff, and the extensions tell us which terms become larger or more subtle in realistic systems.

Empirical sketch (optional): a simulation plan for calibrating (α, γ, m). Our bounds are deliberately distribution-free, but a platform ultimately needs a numerical calibration: given an operational audit budget and an acceptable level of report distortion, what audit frequency α, penalty scale γ, and sample size m are required in practice? An empirical sketch is therefore useful, not as a replacement for incentive analysis, but as a bridge from the divergence-based penalty logic to concrete engineering choices (latency, TEE calls, and bidder risk limits). The goal of the exercise is to map a desired tolerance $\bar{\delta}$ for

$$\delta_i \;:=\; \mathbb{E}_{x\sim\mathcal{D}}\big\|\hat{p}_i(\cdot \mid x) - p_i(\cdot \mid x)\big\|_1$$

into a feasible region of (α, γ, m) that (i) makes $\bar{\delta}$ a best-response scale for strategically chosen misreports, and (ii) yields high-probability detection in finite samples when scores are noisy or clipped.

A minimal testbed: open LLMs as committed generators and parametric misreports. A convenient starting point is to instantiate each advertiser i with an open-weight language model (or a fixed checkpoint plus

decoding policy) that defines the committed truth $p_i(\cdot \mid x)$ at every context. In a real deployment, the platform would not see p_i directly, but would be able to sample from it inside a TEE. In simulation we can treat the model as the oracle, and enforce the same information constraint by allowing the platform to access only $\hat{p}_i$ during generation and only sampled draws $y \sim p_i(\cdot \mid x)$ during audits.

Because a full reporting function $\hat{p}_i : \mathcal{X} \to \Delta(T)$ is intractable to optimize naively, we can restrict deviations to a parametric family that captures plausible manipulations. A simple and revealing class is a *logit-tilt* misreport:

$$\hat{p}_i(t \mid x; \theta) \;\propto\; p_i(t \mid x)\, \exp\{\theta^\top \phi(t, x)\},$$

where $\phi(t, x)$ are hand-designed features (e.g., whether t is a brand token, a safety-sensitive token, or belongs to a semantic cluster) and θ is chosen strategically by the advertiser. This family preserves support (avoiding trivial infinite log losses when $p_i(t \mid x) > 0$) while allowing systematic upweighting of preferred tokens. It also makes the audit penalty analytically interpretable: it becomes a regularized tilt around p_i, closely matching the closed-form benchmark intuition.

Synthetic contexts and outcome primitives. To approximate the "advertiser prompt" setting, we can construct contexts in three layers: (i) a base prompt (user query or task instruction), (ii) a short advertiser-specific suffix that induces commercial intent or content, and (iii) the endogenous prefix created by previously sampled tokens in multi-step generation. Concretely, we can define a prompt generator that samples (topic, style, locale, risk flag) tuples and renders templated prompts, then lets the system generate K tokens under the platform aggregate q. For auditing we maintain a held-out distribution $\mathcal{D}$ over prefixes, obtained either by sampling base prompts and rolling out trajectories under a reference truthful policy, or by replaying logged production prefixes. The key practical point is that $\mathcal{D}$ must be versioned: the audit distribution used for calibration should match the audit distribution used for enforcement, and should be refreshed as prompts and bidder populations change.

Advertiser utilities and payments: choosing a controlled objective. For calibration we need an explicit non-audit objective that creates a motive to misreport. A controlled choice is a linear functional of the generated distribution (per context),

$$U_i(q, x) \;=\; \sum_{t \in T} v_{i,t}(x)\, q(t \mid x),$$

where $v_{i,t}(x)$ encodes the advertiser's value for token t in context x (e.g., higher value for brand mentions in appropriate contexts, negative value for

disallowed content). This makes the benefit of manipulation transparent and allows us to estimate Lipschitz constants directly via $\|v_i(x)\|_\infty$ bounds. Influence payments can be implemented in the simulation either as (i) a stylized monotone mapping $z_i(q, x)$ that is Lipschitz in q, or (ii) the actual influence-payment proxy used in the intended mechanism (e.g., a stable-sampling second-price analogue) but evaluated on the finite token set. The calibration exercise does not require an exact payment rule so long as it captures the magnitude and smoothness of payment changes induced by shifts in q.

Evaluation metrics: distortion, welfare effects, and audit statistics. We can measure three objects on a common set of contexts x:

1. *Report distortion:* empirical estimates of δ_i and also squared deviation $\mathbb{E}\|p_i - \hat{p}_i\|_1^2$, since the audit penalty lower bound is quadratic.
2. *Outcome distortion:* $\mathbb{E}\|q - q^{\text{truth}}\|_1$ and task-level metrics computed on generated samples (e.g., conversion proxy, toxicity proxy, constraint violations). This is the policy-facing quantity.
3. *Audit evidence:* the realized audit loss

$$\widehat{\ell}_i \;=\; \frac{1}{m}\sum_{j=1}^{m} S(\hat{p}_i(\cdot \mid x_j^a), y_j^a), \qquad y_j^a \sim p_i(\cdot \mid x_j^a),$$

   and its deviation from the truthful baseline. This is what determines deterrence under finite m.

A practical refinement is to report both log score and a *clipped* log score, $S_\tau(\hat{p}, y) = \min\{-\log \hat{p}(y), \tau\}$, because unbounded penalties are difficult to finance via bonds and difficult to concentrate statistically.

Strategic behavior in the loop: approximate best responses over a restricted class. To connect calibration to incentives rather than passive measurement, we can compute approximate best responses for each advertiser within the parametric misreport family. Holding $(b, \hat{p}_{-i})$ fixed, advertiser i chooses θ to maximize empirical payoff:

$$\widehat{\Pi}_i(\theta) \;=\; \widehat{\mathbb{E}}\Big[\sum_{k=1}^{K} U_i(q_k(\theta), x_k)\Big] - \widehat{\mathbb{E}}\Big[\sum_{k=1}^{K} z_{i,k}(q_k(\theta), x_k)\Big] - \alpha\gamma\,\widehat{\mathbb{E}}\big[\widehat{\ell}_i(\theta)\big].$$

In simulation, gradients with respect to θ can be obtained by automatic differentiation through $\hat{p}_i$ and the aggregator, while treating sampled outcomes via standard score-function estimators or by evaluating objectives directly at the distribution level (since we can compute $q(\cdot \mid x)$ on a truncated token set). This yields a numerically grounded estimate of the equilibrium distortion $\delta_i(\alpha, \gamma, m)$ within the chosen deviation class, and lets us test whether the predicted scaling $\delta_i = \Theta(1/(\alpha\gamma))$ appears in a realistic LLM environment.

A calibration recipe based on a target distortion bound. The simplest practical recipe uses the approximate-truthfulness inequality as a design constraint rather than as an ex post bound. Fix a platform-chosen tolerance $\bar{\delta}$ on $\mathcal{D}$. Under linear pooling, a conservative sufficient condition suggested by the bound is

$$\alpha\gamma \;\geq\; \frac{4(L_i + L_i^z)\, w_i(b)}{\bar{\delta}} \qquad \text{for each materially influential bidder } i.$$

Empirically we can estimate $(L_i + L_i^z)$ by perturbing q in random directions on held-out contexts and measuring the resulting change in the advertiser's objective (or by bounding values $v_{i,t}(x)$ directly when utilities are constructed). We can also replace the worst-case $w_i(b)$ by a high quantile of realized weights (e.g., the 95th percentile), acknowledging that bids fluctuate and that strict worst-case design may be unnecessarily expensive.

Given an operational audit budget constraint of the form "expected TEE samples per session $\leq \bar{C}$," we then choose α and m such that $\alpha m \leq \bar{C}$, and set γ to satisfy the product constraint above. This makes explicit the substitution between auditing more often (higher α) and penalizing more per audit (higher γ). In practice, risk and liability constraints often cap γ (or require a bond B_i satisfying $B_i \gtrsim \gamma\tau$ under clipping), so we can treat γ as the binding instrument and solve for α.

Choosing m: concentration, clipping, and false negatives. The role of m is not primarily the *expected* penalty (which scales with $\alpha\gamma$), but the reliability of enforcement. With finite m, a misreport can "get lucky" and avoid slashing in a particular audit event. To quantify this, we need a concentration inequality for the per-sample score. With unclipped log score, the tails can be heavy if $\hat{p}(y)$ is small, which undermines clean high-probability statements. This is one reason clipping (or a bounded proper score) is operationally attractive.

If we use $S_\tau \in [0, \tau]$, then for any fixed context distribution and report we have (by Hoeffding) that

$$\Pr\Big(\widehat{\ell}_i - \mathbb{E}[\widehat{\ell}_i] \leq -\epsilon\Big) \;\leq\; \exp\!\left(-\frac{2m\epsilon^2}{\tau^2}\right).$$

A platform can therefore pick (m, τ) to achieve a target false-negative probability β for a given expected gap in audit loss between truthful and misreporting behavior. Empirically, we can estimate this gap as

$$\Delta_{\text{audit}} \;\approx\; \mathbb{E}_{x\sim\mathcal{D}}\Big[\mathrm{KL}\big(p_i(\cdot \mid x) \,\|\, \hat{p}_i(\cdot \mid x)\big)\Big],$$

recognizing that clipping will reduce the effective gap. The calibration exercise then becomes: pick τ to balance bounded liability against informativeness, then pick m so that the realized penalty concentrates tightly enough that repeated audits induce predictable deterrence.

Stress tests: distribution shift, early-trajectory leverage, and coalition geometry. A useful simulation should deliberately violate the benchmark assumptions to reveal how fragile a calibration is. First, to probe distribution shift, we can construct $\mathcal{D}^{\text{prod}}$ by routing prompts through an "engagement" model that correlates with advertiser incentives, while keeping $\mathcal{D}$ as a more neutral held-out set. We can then measure the wedge between misreporting incentives under $\mathcal{D}$ and under $\mathcal{D}^{\text{prod}}$, and evaluate mixture audits of the form $\mathcal{D} = (1-\rho)\mathcal{D}^{\text{prod}} + \rho\mathcal{D}^{\text{stress}}$.

Second, to probe multi-step leverage, we can compare the distortion induced by misreports restricted to early tokens versus late tokens, holding the same audit parameters fixed. If early-step perturbations produce larger downstream effects, the empirically required $\alpha\gamma$ to stabilize behavior should rise, consistent with a continuation-value interpretation.

Third, to probe collusion, we can simulate identity splitting by replacing one bidder of weight w with k nominally distinct bidders of weights w/k and allowing joint optimization of their misreport parameters subject to separate audit penalties. The object of interest is whether aggregate distortion $\|q - q^{\text{truth}}\|_1$ can be maintained while reducing total expected audit loss, and whether bidder-specific scaling (e.g., γ_i increasing in weight) restores the intended deterrence.

Deliverables: an audit frontier rather than a single point estimate. The outcome of the exercise should not be a single recommended triple (α, γ, m), but an *audit frontier*: the set of parameter combinations that achieve (i) a target equilibrium distortion $\delta_i \le \bar{\delta}$ for influential bidders, and (ii) a target enforcement reliability (false-negative rate below β) under the chosen scoring rule and clipping. This frontier can be plotted against operational costs (TEE calls, latency, bonded capital) and against outcome metrics (task success, safety violations). A useful practical summary is: for each bidder size tier (weight quantile), report the minimal αm required to keep δ_i below threshold under a feasible γ and τ.

Limitations and what we learn despite them. We should be explicit about what such a simulation can and cannot validate. It cannot prove incentive compatibility in the full strategy space, and it cannot eliminate governance issues around who controls $\mathcal{D}$, how TEEs are attested, or how disputes are resolved. It also inherits modeling choices: a restricted misreport family may understate adversarial creativity, while a stylized utility may misstate real commercial incentives. Nonetheless, it can validate the *comparative statics* that drive practical design (how distortion scales with $\alpha\gamma$, how sensitive enforcement is to m and clipping, and how quickly distribution shift breaks naive auditing). In this sense, the empirical sketch complements the theory: it helps the platform translate a divergence-based incentive lever

into a budgeted compliance program with measurable performance targets.

Transition. With these calibration tools in hand, we can distill the broader design recommendations for verified influence markets, and clarify which open problems remain primarily economic (equilibrium selection, collusion, dynamic incentives) versus primarily systems-oriented (attestation, privacy, and scalable auditing).

Conclusion: design recommendations for verified influence markets. Our analysis highlights a simple but operationally meaningful principle: if a platform wants to sell influence over a generative model while retaining auction-like bidding incentives, it must make the *inputs* to aggregation (the reported conditional distributions) verifiable enough that they can be treated as approximately truthful objects. Proper-scoring-rule audits provide a natural lever because they translate misreporting into a divergence cost that is (i) *local* in the reported distribution, (ii) *composable* across contexts and steps, and (iii) *mechanism-agnostic* in the sense that the audit penalty depends only on $(p_i, \hat{p}_i)$ and not on the particular aggregation or payment rule. The design task, then, is not to eliminate manipulation in all states of the world, but to pick (α, γ, m) and operational primitives so that any profitable deviation must be small enough that the platform can treat the resulting perturbations in q (and hence in outcomes and payments) as negligible at the policy-relevant scale.

Recommendation 1: separate *truth enforcement* from *influence pricing*. A practical architecture is to keep the influence mechanism (monotone aggregation plus monotone influence payments) as close as possible to the familiar token-auction logic, and to add audits as a separate enforcement layer whose sole role is to discipline the reporting function $\hat{p}_i$. This separation is valuable for two reasons. First, it limits the surface area of strategic interactions: bidders optimize bids and (approximately) truthful reports rather than entangled objects. Second, it yields modularity for engineering: changes in the auction/payment implementation need not require changes in the audit logic so long as the report object and its semantics remain the same.

Recommendation 2: treat $\alpha\gamma$ as the core incentive knob, and m as the reliability knob. The economic deterrence in expectation is controlled primarily by the product $\alpha\gamma$ (audit frequency times penalty scale). In contrast, the sample size m matters mainly for *concentration* of realized penalties and hence for the predictability of enforcement. In deployments where large one-shot penalties are politically or legally constrained, the platform can keep γ moderate and raise α; where audits are expensive (TEE

calls, latency), the platform can reduce α and compensate with higher γ, subject to bounded-liability constraints. Operationally, it is helpful to set a target distortion tolerance $\bar{\delta}$ and back out a conservative requirement on $\alpha\gamma$ for the largest-weight bidders (since $w_i(b)$ scales manipulation leverage), then choose m to make audit outcomes stable enough that the expected deterrence is realized in finite samples.

Recommendation 3: use bounded or clipped proper scores, with explicit bonded capital. Unbounded penalties are difficult to finance and hard to make statistically well-behaved. Clipping the log score, or using a bounded strictly proper scoring rule, makes enforcement more practical but introduces a tradeoff: clipping protects bidders from catastrophic losses yet weakens the marginal deterrent against assigning extremely small probability to events that occur under p_i. A disciplined approach is (i) select a clipping threshold τ that matches a permissible per-audit loss, (ii) require a posted bond B_i that comfortably covers worst-case exposure (e.g., $B_i \gtrsim \gamma\tau$ per audit event, scaled by an upper bound on audit frequency), and (iii) calibrate m so that clipped-score estimates still separate truthful and materially misreporting behavior with high probability. From a mechanism-design perspective, the bond is not merely a collection tool: it is part of the incentive system, ensuring that the promised penalty is credible ex post.

Recommendation 4: tier audits and penalties by effective influence weight. Because the ability to move the aggregate distribution scales with $w_i(b)$ under linear pooling (and analogously under other monotone aggregators), we should not expect a single uniform (α, γ) to be cost-effective across a heterogeneous bidder population. A platform can implement tiering rules of the form

$$\gamma_i \text{ nondecreasing in } w_i(b) \quad \text{and/or} \quad \alpha_i \text{ nondecreasing in } w_i(b),$$

with tiers defined either by ex ante bid commitments or by realized weights over time. Tiering is also a mitigation against identity-splitting: if a bidder can cheaply split into many small identities, then any scheme that makes enforcement much weaker for small weights invites circumvention. One practical remedy is to base tiers on *affiliated-account aggregation* (common control, payment instrument, or cryptographic identity), though this raises governance questions. Another is to make the penalty schedule convex in cumulative weight, so that splitting does not reduce total expected audit exposure.

Recommendation 5: commit to (and refresh) an audit-context distribution $\mathcal{D}$ that matches where influence matters. Audits only bind

on the contexts they cover. If $\mathcal{D}$ drifts away from production traffic, sophisticated bidders will concentrate misreports on un-audited regions of $\mathcal{X}$, preserving apparent compliance on audited contexts while manipulating outcomes where it counts. This is not a minor technicality: it is the main channel through which a theoretically clean divergence penalty can fail in practice. We therefore recommend that the platform (i) publicly specify a versioned audit distribution family (or a sampling procedure) that tracks production, (ii) include stress components that oversample safety-critical and high-commercial-value contexts, and (iii) refresh $\mathcal{D}$ on a schedule that is both frequent enough to prevent gaming and stable enough to keep compliance predictable. A useful compromise is a mixture audit policy, e.g., $\mathcal{D} = (1-\rho)\mathcal{D}^{\text{prod}} + \rho\mathcal{D}^{\text{stress}}$, where $\mathcal{D}^{\text{stress}}$ is designed to be hard to anticipate and disproportionately informative about problematic behavior.

Recommendation 6: audit the *reporting function* as code, not only its numeric outputs. The object of enforcement is a committed mapping $\hat{p}_i : \mathcal{X} \to \Delta(T)$. In practical systems, bidders will implement $\hat{p}_i$ as code that may contain hidden conditionals, backdoors, or trigger-based behavior. Auditing only on sampled contexts provides statistical guarantees but can be brittle against adversarially chosen triggers. We therefore recommend combining statistical audits with *attestation* and *change control*: bidders should commit to a code hash (or container digest) and a declared interface, with updates gated by a re-commitment process and possibly a probation period with elevated audits. This is a systems requirement, but it has direct economic content: it constrains the strategy space in a way that makes the scoring-rule incentive lever meaningful.

Recommendation 7: design for multi-step generation and early-trajectory leverage. In multi-token generation, early distortions can have outsized downstream effects by steering the state distribution over future contexts. Our Lipschitz-based bounds treat each step in a controlled way, but the practical implication is that the platform should allocate audit and monitoring resources to where leverage is highest. Concretely, we can (i) increase audit probability on early steps or on steps that shift topic/style, (ii) define context features that flag high-impact prefixes, and (iii) supplement distribution-level audits with outcome-level monitors (e.g., constraint violations) that are sensitive to compounding effects. Importantly, such monitors need not replace proper-scoring audits; they can serve as *triggers* for more intensive auditing, thereby conserving budget while targeting risk.

Recommendation 8: maintain transparency about compliance, but keep randomness unpredictable. Verified influence markets will be scrutinized by bidders, users, and regulators. A platform should publish the scor-

ing rule family, the broad structure of $\mathcal{D}$ (including what classes of contexts are in scope), and the tiering logic for (α, γ, m). This improves legitimacy and reduces dispute costs. At the same time, the *realizations* of audit contexts should remain unpredictable to deter context-specific gaming, and the platform should retain the ability to introduce fresh stress distributions when new manipulation patterns are discovered. Transparency about rules coupled with unpredictability about draws mirrors standard compliance practice in other regulated settings.

Recommendation 9: implement a dispute-resolution pathway grounded in verifiable evidence. Audits create financial transfers that will be contested. A platform should therefore define an appeals process that is compatible with privacy and with TEE limitations. The key design goal is to make the relevant evidence (attested code hash, audited contexts, realized samples y_j^a, and computed scores) verifiable without revealing sensitive user prompts. One approach is to store commitments and audit transcripts as signed artifacts, and to permit third-party verification under confidentiality. This is not merely legal hygiene: predictable dispute resolution increases the credibility of penalties and therefore strengthens deterrence without increasing α or γ.

Open problem 1: endogenous and strategic audit distributions. We treated $\mathcal{D}$ as fixed, but in reality the platform chooses it and bidders may try to influence it (through traffic shaping, prompt injection, or creating contexts that are rare under audits). A rigorous treatment would model a dynamic game where $\mathcal{D}$ is both a monitoring technology and a strategic object. Two questions seem central: (i) what mixture policies over contexts are robust to gaming while remaining representative of production welfare, and (ii) how should the platform allocate audit mass across context strata to minimize worst-case outcome distortion for a fixed audit budget?

Open problem 2: collusion, identity splitting, and correlated misreports. Proper-scoring penalties are bidder-separable, which is a feature for modular enforcement but a vulnerability under coalitions. If multiple bidders coordinate their reports, they may be able to steer q while spreading audit exposure. Understanding the limits of such "coalition geometry" requires extending the analysis beyond unilateral deviations. Promising directions include (i) coalition-proof audit schedules with weight-dependent penalties, (ii) mechanisms that base influence rights on authenticated entities rather than accounts, and (iii) aggregation rules that reduce marginal manipulability at high concentration (e.g., regularized pooling) without breaking monotonicity properties needed for influence pricing.

Open problem 3: richer objectives and non-Lipschitz utilities. We assumed Lipschitz continuity in $\|q - q'\|_1$ to obtain clean bounds. This is a reasonable approximation when advertisers care about smooth token-level metrics, but it can fail when objectives are thresholded (e.g., a conversion event that triggers only when a particular phrase appears) or when payments depend on discrete outcomes. Extending the theory to objectives with discontinuities, heavy tails, or state-dependent constraints likely requires either (i) smoothing assumptions at the mechanism layer (e.g., randomized allocation rules) or (ii) alternative enforcement tools that directly target discontinuous behaviors. The practical takeaway is that platforms should, where possible, design payments and allocation to be stable under small distribution shifts, thereby making the audit-based truthfulness program effective.

Open problem 4: bounded proper scoring rules under strategic support manipulation. The log score is analytically convenient but unbounded, while bounded proper scores or clipped scores can weaken deterrence against assigning near-zero probability to adverse events. A deeper question is whether we can design bounded scoring rules (or hybrid penalties) that preserve strong incentives against "support hacking" while maintaining finite liability and good statistical concentration. This intersects with robust statistics: one would like a penalty that behaves like KL near the interior of the simplex but saturates gracefully near the boundary, without creating perverse incentives to concentrate mass.

Open problem 5: dynamics, learning, and partial commitment. In realistic deployments bidders will update models, features, and targeting logic. The assumption of a single committed reporting function per session is a useful abstraction, but the economically relevant object is a *dynamic* commitment with occasional updates. This raises questions about (i) how to price and audit updates, (ii) whether past audit performance should affect future audit rates (a "reputation" or "compliance" state), and (iii) whether adaptive auditing creates new strategic incentives (e.g., behaving well early to reduce future scrutiny). A principled approach would treat auditing as a control problem with incentive constraints, potentially yielding state-dependent (α, γ, m) schedules that reduce cost while maintaining deterrence.

Where we land. Despite these open problems, the model clarifies a core tradeoff that we expect to persist across implementations: influence mechanisms make the aggregate output sensitive to bidder-provided distributions, and any sensitivity invites strategic distortion; proper-scoring audits convert distortion into a convex cost that can dominate the linear gains from manipulation when $\alpha\gamma$ is sufficiently large relative to the bidder's influence and

the environment's payoff smoothness. The practical program is therefore to (i) make reports and audits verifiable, (ii) choose audit parameters that scale with influence, (iii) keep liability bounded and enforcement reliable via clipping and sufficient m, and (iv) treat the audit distribution as a first-class design object that must track production realities. If we do these things, verified influence markets can be engineered as compliance systems with measurable performance targets, rather than as fragile one-off mechanisms whose correctness depends on heroic assumptions.