Stakes, Slashing, and Freshness: Adversary-Robust Incentives for Timely Federated Learning in IIoT

Liz Lemma Future Detective

January 14, 2026

Abstract

Industrial IoT federated learning requires incentives that jointly value model quality and timeliness, but existing schemes—such as satisfaction-aware Stackelberg designs built on Age of Information (AoI) and service latency—typically assume rational and reliable participants. We study a 2026 threat model in which a fraction of participants are adversarial (poisoning, collusion, Sybil identities) while the rest are rational and privacy-sensitive. We propose a clean mechanism that combines (i) timeliness-aware payments (AoI/latency) aligned with satisfaction metrics, (ii) robust aggregation to bound the statistical influence of corrupted updates, and (iii) stake-based delayed settlement with slashing triggered by robust anomaly scores. In a tractable repeated Bayesian Stackelberg model, we derive closed-form best responses for honest nodes' update cycles, characterize deterrence conditions under which poisoning is unprofitable, and prove an $O(\varepsilon)$ welfare degradation bound when at most an ε-fraction of updates are adversarial. The results provide a principled bridge between economic mechanism design and robust federated learning, addressing a key limitation noted by satisfaction-aware IIoT-FL frameworks: fully adversarial scenarios beyond rational behavior.

Table of Contents

4. 4. Baseline (no adversaries): derive honest-node best response $\theta_i^*(r)$ and server's optimal pricing under budget/timeliness constraints; connect to satisfaction-aware Stackelberg equilibrium.

5. 5. Adversary model: poisoning action δ, Sybil capability via stake budget, external benefits from degradation; detection/audit technology and slashing rule.

6. 6. Mechanism: timeliness-aware payment + anomaly-weighted acceptance + stake escrow; define acceptance region and payments; discuss implementability (on-chain/off-chain).

7. 7. Equilibrium analysis: characterize honest participation/effort; characterize adversary best response and derive deterrence thresholds $s^\star$.

8. 8. Robustness and welfare bounds: influence bound for robust aggregation; translate aggregate error into satisfaction/welfare loss; prove $O(\varepsilon)$ degradation and Sybil-resistance via stake budget constraints.

9. 9. Comparative statics and design guidance: how $s, q, \phi,\ \tau, \lambda$, and network conditions shift outcomes; when to prefer stricter slashing vs higher base payments.

10. 10. Simulation plan: poisoning/collusion experiments with AoI/latency, robust aggregation, and stake; calibration targets; sensitivity analysis.

11. 11. Conclusion: implications for 2026 deployments; limitations and extensions (richer audits, privacy-preserving anomaly scoring).

1 1. Introduction and motivation: timely FL in IIoT, why AoI/latency incentives fail under adversaries; contributions and roadmap.

Industrial Internet-of-Things (IIoT) deployments increasingly rely on learning systems that must adapt in real time to nonstationary environments: sensors drift, actuators wear, and operating regimes shift with demand and maintenance schedules. Federated learning (FL) is attractive in this setting because it allows the server (e.g., a plant operator or platform provider) to improve a shared model without continuously pulling raw data off devices that face bandwidth limits, privacy constraints, or safety certification barriers. Yet the economic constraint is as important as the statistical one: devices are owned by heterogeneous parties, operate under energy and compute budgets, and face opportunity costs when they allocate cycles to training and communication. If we want "timely FL" in IIoT—updates that are fresh enough to matter operationally—we must pay for timeliness in a way that is consistent with rational behavior.

A natural starting point, and one commonly adopted in practice, is to tie rewards to measures of freshness and responsiveness. Two such measures are Age of Information (AoI), which captures how stale a device's last effective update is at the server, and service latency, which captures the delay from a local training trigger to server receipt and incorporation. These metrics are operationally salient: predictive maintenance and anomaly detection can fail silently when the model is trained on stale regimes, and control loops can destabilize when the learning pipeline introduces delays. From an incentive-design perspective, AoI and latency also have the crucial property that the server can observe them from timestamps, even if it cannot observe a device's raw data or internal costs.

However, timeliness-aware payments alone are not robust to adversarial behavior. The core difficulty is that the same mechanism that encourages frequent updates can inadvertently subsidize harmful ones. If a contract pays more for shorter update cycles (smaller buffering periods) or lower latency, an attacker can cheaply generate many "timely" submissions that are strategically manipulated to degrade the global model. In IIoT, this is not merely a theoretical concern: compromised devices, malicious contractors, or economically motivated sabotage can inject poisoned gradients that pass superficial timeliness checks. Worse, identity is cheap in many digital settings. Without accountability, an adversary can create Sybil identities to amplify its influence on the aggregate update, harvesting payments while simultaneously pushing the model in a damaging direction.

This observation motivates the central tradeoff our model is designed to illuminate. On one hand, the server wants to procure updates that are frequent and responsive, because the value of learning is time-sensitive. On the

other hand, the server must be resilient to a fraction of participants who do not share the objective of improving the model, and who may even value degradation. A purely "carrot-based" timeliness incentive creates a vulnerability: it expands the feasible set of profitable adversarial strategies by paying for speed rather than correctness. Conversely, a purely "stick-based" security posture—heavy-handed exclusion, strict thresholds, or onerous participation requirements—can starve the system of honest participation, especially when honest nodes have private and heterogeneous costs of maintaining short update cycles.

We address this tension by framing timely FL procurement as a repeated principal–agent problem with endogenous participation, and by combining three ingredients that are individually familiar but, we argue, jointly necessary in adversarial IIoT FL: (i) a timeliness-aware reward that makes honest effort (more frequent updates) privately optimal; (ii) an accountability layer in the form of posted stake with probabilistic audit and slashing, which puts capital at risk for detected manipulation; and (iii) an ε-robust aggregation rule that limits the statistical influence of a bounded fraction of corrupted updates. The economic logic is simple: robust statistics bounds the damage of what slips through, while staking and slashing reduces the set of attacks that are privately profitable to attempt in the first place. Timeliness incentives then operate "on the residual," encouraging honest agents to supply fresh information without opening an unpriced channel for sabotage.

Two modeling choices merit emphasis because they connect directly to implementable system design. First, we use observables the server plausibly has in IIoT pipelines: timestamps (for AoI and latency) and an anomaly score computed from submitted updates relative to a robust aggregate. The server need not inspect raw data; it uses robust distances and acceptance thresholds to decide whether an update is provisionally rewarded and whether an audit is triggered within a settlement window. Second, we treat the stake requirement as a per-identity escrowed deposit rather than an abstract punishment. This is not merely a modeling convenience: in distributed environments, "future exclusion" is often weak because identities can be rotated, but capital constraints are harder to evade. A stake converts identity creation and misbehavior into an economic decision with a real budget.

Our approach also clarifies why timeliness incentives "fail" under adversaries in a precise sense. When rewards increase in update frequency, a malicious participant's marginal benefit from submitting *any* update rises, including manipulated ones. If the expected penalty for manipulation is small—because audits are rare, anomaly detection is noisy, or slashing is mild—then the equilibrium can feature excessive adversarial participation, including Sybils, and the server can end up paying for the acceleration of its own model degradation. In other words, timeliness rewards can increase both the volume and the impact of poisoned submissions unless they are paired with an enforcement mechanism that scales expected costs with the

magnitude and detectability of manipulation. This is the central incentive-compatibility gap we aim to close.

Within this framework, we make four main contributions. First, we provide a tractable Stackelberg-style model of timely FL procurement in which honest nodes choose an update cycle θ_i in response to a posted reward parameter, and adversaries choose whether and how much to poison. The model is intentionally parsimonious—e.g., logarithmic rewards in $1/\theta_i$ and linear private costs—because the purpose is to isolate economic forces that persist across more complex specifications. Second, we derive sharp, interpretable conditions under which poisoning is strictly dominated: sufficient thresholds in stake, audit probability, and slashing severity that make the expected cost of detectable manipulation exceed its sabotage benefit. Third, we connect these incentive constraints to robust aggregation guarantees, showing that even when some adversarial behavior remains (e.g., small perturbations designed to evade detection), the aggregate update error and the resulting welfare loss are bounded linearly in the adversarial fraction ε. This delivers an end-to-end statement that the server's equilibrium welfare degradation is $O(\varepsilon)$, with constants depending on robustness and Lipschitz parameters rather than the number of nodes. Fourth, we explicitly account for Sybil behavior through a stake budget: per-identity escrow transforms "number of identities" into a priced resource, bounding the adversary's effective market share in any round.

We do not claim that staking and anomaly detection are a silver bullet. Audits are costly and imperfect; anomaly thresholds trade false positives against false negatives; and robust aggregation can blunt, but not eliminate, coordinated attacks—especially as ε approaches the breakdown point of the estimator. Moreover, a stake requirement can exclude honest but liquidity-constrained devices, a first-order concern in IIoT ecosystems with many small operators. Our objective is therefore not to propose a one-size-fits-all mechanism, but to provide a disciplined way to reason about design levers—$(r, s, q, \phi, \bar{z})$ in our notation—and to quantify the security–participation tradeoff that practitioners face.

The remainder of the paper proceeds as follows. Section 2 situates our mechanism relative to existing work on satisfaction- and timeliness-aware incentives in FL, robust learning under poisoning, and accountability mechanisms based on collateral and slashing. Section 3 formalizes the model: timing, observables, payoffs, and the aggregation-and-audit pipeline. Section 4 characterizes honest best responses and participation constraints, highlighting how timeliness rewards map into endogenous update cycles. Section 5 studies adversarial deviations and derives deterrence conditions that pin down explicit security thresholds. Section 6 integrates robust aggregation bounds to obtain an $O(\varepsilon)$ welfare-loss guarantee and discusses comparative statics for practical tuning. Section 7 concludes with implementation considerations for IIoT deployments, including calibration with empirical

anomaly-score distributions and the operational costs of audits.

2 2. Related work: satisfaction-aware incentives/Stackelberg FL (AoI/latency), FedAvg and communication–computation tradeoffs, robust FL against poisoning, stake/slashing mechanisms and accountability.

A first strand of related work studies incentives for federated learning (FL) participation under heterogeneous device costs and values. Much of this literature adopts a principal–agent or Stackelberg perspective: a coordinator (server) posts a payment rule or selects participants, and devices respond by choosing whether to participate and how much effort to exert (e.g., local computation, communication frequency, or data contribution). Mechanisms range from simple per-round rewards to auctions, budget-feasible selection, and contract-theoretic menus designed to screen private costs or private data quality. The unifying insight is that FL is not "free privacy": even if raw data never leaves the device, training and communication consume scarce resources, and rational participants require compensation that is aligned with the system's objective. Our setting is in this spirit, but we emphasize *timeliness* as a first-class procurement target, rather than treating communication frequency as a purely engineering knob.

A closely related sub-literature makes the coordinator's objective explicitly *satisfaction-aware*—valuing updates not only for their contribution to accuracy but also for their operational relevance. In IIoT and cyber-physical applications, the value of an update can decay with staleness: an accurate model trained on yesterday's regime may be less useful than a slightly noisier model trained on today's regime. Age of Information (AoI) has become a canonical way to formalize this freshness dimension in networks, and recent FL work leverages AoI-like metrics to schedule devices, prioritize transmissions, or define rewards that favor fresh information. Similarly, latency-sensitive objectives appear in FL for real-time control and streaming inference, where end-to-end delay (from local trigger to server incorporation) enters the loss function or service-level agreement. Our approach aligns with these ideas by tying compensation to observables derived from timestamps (AoI and service latency), which are attractive precisely because they do not require the server to observe private costs or raw data. Where we depart is that we treat timeliness payments as an *incentive instrument* with strategic side effects: rewarding freshness can unintentionally subsidize harmful submissions unless paired with accountability and robust aggregation.

A second strand of work, often framed as systems optimization rather than mechanism design, studies the communication–computation tradeoff in FL, particularly in FedAvg-style protocols. In the classical view, devices

choose (or are assigned) how many local steps to run before communicating, balancing the cost of communication against the benefit of more frequent synchronization. Extensions incorporate stragglers, device heterogeneity, bandwidth constraints, and asynchronous or partial participation. Although this literature does not always use AoI language, it effectively studies *update cycles* and staleness: infrequent communication increases drift between local and global models, while overly frequent communication can be infeasible in constrained networks. Our model borrows the core economic structure implied by these tradeoffs—shorter update cycles are costly but valuable—and makes it explicit by allowing nodes to choose an update-cycle parameter θ in response to posted rewards. In doing so, we connect a familiar systems tradeoff to an equilibrium object that can be tuned through contract parameters, rather than assuming that the server can simply set update frequencies exogenously.

A third, and crucial, literature addresses robustness of FL to adversarial or Byzantine behavior, including data poisoning and model poisoning. The dominant approach is statistical: design aggregation rules whose influence function is bounded under an ε-fraction of corrupted updates. Coordinate-wise median, trimmed mean, geometric median, and methods such as Krum/Bulyan provide formal guarantees under various assumptions (bounded honest updates, sub-Gaussian noise, or bounded adversarial fraction below a breakdown point). Complementary work develops detection and filtering heuristics—distance-based anomaly scores, norm clipping, cosine similarity checks, or validation-set tests—that attempt to flag suspicious updates before aggregation. These contributions are indispensable for IIoT FL because compromised devices and malicious insiders are realistic threats, and because robust aggregation can provide "graceful degradation" when perfect exclusion is impossible.

At the same time, the robust FL literature typically treats adversarial behavior as exogenous: an attacker corrupts some fraction of devices, and the designer chooses an estimator that tolerates that fraction. What is often missing is the economic margin: *when* is it privately optimal for an attacker to participate, to create additional identities, or to choose a particular poisoning magnitude? In settings where participants are paid, adversaries may be strategic and profit-seeking (or sabotage-seeking), and the attack surface depends on the payment rule. In particular, timeliness-aware rewards can change the attacker's marginal benefit from submitting more updates (including Sybil updates), and detection thresholds can induce "stealth" equilibria in which adversaries optimally pick small perturbations to evade flags. Our framework is designed to complement robust statistics by endogenizing these choices: robust aggregation bounds the harm conditional on a residual adversarial fraction, while incentives and penalties shape that residual fraction and the equilibrium poisoning intensity.

A fourth strand of related work concerns accountability mechanisms

based on collateral, escrow, and slashing—ideas that originate in distributed systems and blockchains (e.g., proof-of-stake security, bonded validators) but have increasingly influenced marketplace and ML settings. The core design principle is to make misbehavior costly by putting capital at risk, especially when identities are cheap to create and reputational penalties are weak. In blockchain protocols, slashing conditions are triggered by cryptographic evidence of equivocation or protocol violations; in data and compute markets, escrow and deposits reduce fraud and non-performance; and in crowdsourcing and peer-prediction mechanisms, deposits can discipline strategic misreporting when ground truth is costly to verify. These mechanisms are conceptually appealing for IIoT FL procurement because they directly address Sybil amplification: if each identity must post stake s, then creating many identities becomes a budgeted decision rather than a purely technical one.

However, importing "slashing" into FL raises two practical issues that our model highlights. First, unlike many blockchain slashing events, poisoning is not always *verifiable* with deterministic evidence; detection is statistical and imperfect. This makes probabilistic audit and thresholding central, and it creates a tradeoff between false positives (punishing honest devices due to noise or benign heterogeneity) and false negatives (failing to deter attacks). Second, collateral requirements interact with participation incentives: a high stake can deter adversaries but also exclude liquidity-constrained honest participants—an especially salient concern in fragmented IIoT ecosystems where many devices are owned by small operators. Thus, stake-based accountability must be analyzed jointly with the reward rule and the acceptance criteria, rather than treated as an add-on security layer.

Our contribution sits at the intersection of these strands. We take the satisfaction/timeliness-aware objective seriously by making AoI and latency observables that enter the payment rule; we take the FedAvg-style cycle tradeoff seriously by letting nodes choose update frequency as a best response to rewards; we take robust FL seriously by using an ε-robust aggregator and anomaly scoring to bound statistical influence; and we take stake/slashing seriously by modeling delayed settlement with probabilistic auditing, which is closer to implementable escrow-based procurement than instantaneous punishment. The novelty is less any single ingredient than the way they discipline each other: robust aggregation limits worst-case damage, while staking and auditing change the attacker's incentives to participate, to Sybil, and to poison.

This synthesis also clarifies limitations of existing approaches when deployed in isolation. A purely timeliness-driven reward can unintentionally purchase adversarial volume; purely robust aggregation can still leave room for low-amplitude, high-frequency attacks and does not, by itself, price Sybil creation; and purely stake-based deterrence can be too blunt if detection is noisy or if honest liquidity constraints bind. By embedding all components in a single equilibrium model, we can derive interpretable comparative stat-

ics—how increasing q (audit rate) or ϕ (slashing severity) substitutes for increasing s (stake), and how tightening anomaly thresholds interacts with honest participation.

With this positioning in mind, we now formalize the environment: the agents, timing with delayed settlement, the observables (AoI, latency, anomaly scores), and the robust aggregation-and-audit pipeline that links submitted updates to payments and penalties.

3 3. Model primitives: agents, timing with delayed settlement, information structure, observables (AoI, latency, anomaly score), and robust aggregation operator.

We model the interaction as a repeated procurement game indexed by discrete rounds $t = 1, 2, \ldots$. In each round, a single server (the principal) contracts with a set of participating identities, indexed by $i \in \{1, \ldots, I\}$ after any selection or admission step. Each identity corresponds to an IIoT node controlled either by a rational honest operator or by an adversary; we write H and A for the sets of honest and adversarial identities, with $|A|/I \leq \varepsilon < 1/2$. Allowing multiple identities per physical operator captures Sybils: what matters for the server is the effective fraction ε of corrupted submissions among those that actually participate in the round.

Contracting instrument and node actions. At the start of round t, the server posts a contract

$$\mathcal{C} \equiv (r, s, q, \phi, \bar{z}),$$

together with a robust aggregation operator $Agg(\cdot)$. The scalar $r > 0$ is a unit reward parameter, $s \geq 0$ is the per-identity stake held in escrow, $q \in (0, 1]$ is the audit probability (or audit rate), $\phi \in (0, 1]$ is the slashing fraction applied upon detection, and $\bar{z}$ is an anomaly-score threshold used both for acceptance/weighting and for triggering penalties. Given $\mathcal{C}$, each identity decides whether to participate (and, if so, escrows s) and chooses an update-cycle parameter θ_i, which we interpret as the node's buffering period between successive "fresh" local updates delivered to the server. Operationally, smaller θ_i corresponds to more frequent communication and/or more aggressive local computation, which is valuable for freshness but privately costly.

Each participating identity submits a model update vector u_i (e.g., a gradient or a model delta). Honest nodes submit an unmanipulated update $u_i = g_i(\theta_i)$ generated by their local data and their chosen update cycle.

Adversarial identities can instead submit

$$u_i = g_i(\theta_i) + \delta_i,$$

where δ_i is a poisoning perturbation, potentially coordinated across identities in A. We keep the mapping from θ_i to the distribution of g_i deliberately abstract, because our focus is not on a particular learning algorithm but on the incentive link between timeliness, detectability, and robust aggregation.

Delayed settlement and per-round timing. The timing within a round is designed to reflect implementable escrow and probabilistic verification rather than instantaneous, perfectly verifiable punishment.

1. *Posting.* The server announces $\mathcal{C}$ and the aggregation rule Agg. 2. *Entry and escrow.* Identities choose participation and deposit stake s into escrow for the duration of an audit window of length T_a. 3. *Submission with timestamps.* Participating identities choose θ_i and submit updates u_i together with metadata (timestamps and protocol logs) that permit the server to compute timeliness observables. 4. *Provisional settlement.* The server computes the robust aggregate $\bar{u} = Agg(\{u_i\}_{i=1}^{I})$, derives anomaly scores z_i for each submission, and issues a *provisional* payment R_i based on θ_i and acceptance indicators described below. Escrowed stake remains locked. 5. *Audit and finalization.* With probability q, the server performs an audit/check within the window T_a. If an identity is deemed anomalous (captured by $z_i > \bar{z}$, or more generally by a stochastic detection event with probability increasing in $\|\delta_i\|$), then a fraction ϕs of its stake is slashed and the remainder is returned according to the policy. If no slashing occurs, the stake is returned at the end of the window and the provisional payment is finalized.

This delayed-settlement structure matters economically: it allows the server to condition eventual transfers on information that arrives after submission (e.g., validation, redundancy checks, or cross-round consistency), while still compensating nodes quickly enough to preserve participation incentives.

Observables: AoI and latency as contractible signals. The server cannot observe a node's private cost σ_i, raw data, or "true" update quality directly. What it can observe reliably are timing-related metrics derivable from protocol metadata. We use two such observables.

First, $A_i(\theta_i)$ denotes the node's (average) Age of Information (AoI), interpreted as the staleness of the information embodied in u_i at the moment the server incorporates it. In many queueing/AoI models, AoI increases in the update cycle θ_i; our analysis only requires that $A_i(\cdot)$ be well-defined and monotone in θ_i over the relevant range.

Second, $E_i(\theta_i)$ denotes end-to-end service latency (e.g., from local trigger to server receipt and incorporation), which may also depend on θ_i through

batching and communication scheduling. Again, we keep the functional form flexible: E_i may incorporate heterogeneous network conditions and device compute limits, but it is observable to the server via timestamps.

These observables are attractive contract inputs in IIoT because they are (i) hard to falsify without deeper protocol compromise and (ii) directly aligned with the operational notion of "freshness" and "responsiveness" that motivates procurement in cyber-physical settings.

Robust aggregation and anomaly scoring. Given submitted updates, the server aggregates using an ε-robust estimator $Agg(\cdot)$, such as a coordinate-wise trimmed mean or coordinate-wise median. We denote the resulting aggregate by

$$\bar{u} \equiv Agg(\{u_i\}_{i=1}^{I}).$$

Robustness enters through a bounded-influence property: when fewer than a breakdown fraction of updates are corrupted, no single (or small coalition of) adversarial submissions can arbitrarily move $\bar{u}$. This is the statistical layer of defense that yields "graceful degradation" as ε grows, and it is complementary to economic deterrence.

To connect robustness to incentives, the server additionally computes an anomaly score for each submission. A canonical choice is a robust distance to the aggregate,

$$z_i \equiv \|u_i - \bar{u}\|,$$

though in practice one might use coordinate-wise standardized residuals, cosine distance, or a robust Mahalanobis metric. The contract specifies a threshold $\bar{z}$: submissions with $z_i > \bar{z}$ are treated as suspicious for both acceptance (payment eligibility) and potential slashing during audits. This threshold is the main policy lever mediating false positives (honest but noisy/heterogeneous updates) versus false negatives (stealthy attacks).

Payment and acceptance. The base reward is designed to make timeliness a priced choice. We adopt a tractable form,

$$R_i^{\text{base}}(\theta_i; r) = r \, \ln\left(\tfrac{1}{\theta_i}\right),$$

which is decreasing and concave in θ_i: reducing the update cycle (being more timely) yields diminishing marginal payment, matching the idea that extreme update frequency is valuable but not infinitely so. Payments are then gated by acceptance indicators that depend only on observables:

$$w_i \equiv \mathbf{1}\{z_i \le \bar{z}\}\cdot\mathbf{1}\{A_i(\theta_i) \le A^{\max}\}\cdot\mathbf{1}\{E_i(\theta_i) \le E^{\max}\}, \qquad R_i = w_i \, R_i^{\text{base}}(\theta_i; r).$$

Thus, timeliness is incentivized both "softly" through the reward slope and "hardly" through eligibility constraints. This reflects a practical procurement posture: the server can refuse to pay for submissions that are either too stale, too slow, or too anomalous to be safely incorporated.

Information and strategic uncertainty. The key private information on the node side is the linear update-cycle cost parameter σ_i, which captures heterogeneous compute/energy/communication costs and is known only to the node. The server does not observe σ_i, nor can it directly distinguish honest from adversarial identities ex ante. Adversaries observe the contract $\mathcal{C}$ and can coordinate on both participation (including Sybil scale, subject to capital at risk) and poisoning choices δ_i. Detection is inherently statistical: conditional on a poisoning magnitude δ, we summarize audit effectiveness by a detection probability $p_{\text{det}}(\delta)$ that increases in $\|\delta\|$. The realized slashing event therefore occurs with probability $q\, p_{\text{det}}(\delta)$, which captures both limited audit coverage and imperfect tests.

This reduced-form detection model is intentionally conservative. It acknowledges that many plausible audits—held-out validation, redundancy checks, cross-device consistency tests, or secure hardware attestations—are noisy and costly. It also makes clear where mechanism design must "pick up the slack": when audits are weak (low q or low p_{det}), the contract must rely more heavily on stake and robust aggregation to control expected damage.

Taken together, these primitives define a stationary stage game repeated over rounds: the server trades off timeliness and learning value against payments and audit costs, while nodes trade off rewards against private costs and (for adversaries) the expected penalty from detection and slashing. In the next section, we shut down adversarial behavior ($\delta_i \equiv 0$) to recover the baseline Stackelberg logic—how r induces an interior best response $\theta_i^*(r)$ and how the server prices timeliness when only heterogeneous honest costs matter.

4. Baseline benchmark (no adversaries). We first shut down strategic manipulation by assuming $\delta_i \equiv 0$ for all participating identities and that anomaly-based gating does not bind on honest behavior (formally, $z_i \leq \bar{z}$ and the timeliness feasibility checks are satisfied at the optimum). This benchmark isolates the Stackelberg pricing logic: how the server's unit reward parameter r induces a choice of timeliness, and how the server optimally trades off satisfaction gains from fresher updates against the transfer needed to elicit them.

Honest node's choice: timeliness as an intensity decision. Operationally, the node controls how "often" it refreshes and communicates. It is convenient to work with an *update intensity* variable $x_i \equiv 1/\theta_i$ (updates per unit time), since many engineering costs scale approximately linearly with activity rates. Under the posted rule $R_i^{\text{base}}(\theta_i; r) = r \ln(1/\theta_i) = r \ln x_i$, a parsimonious reduced-form for an honest node's private operating cost is $\sigma_i x_i$, where $\sigma_i > 0$ is the node-specific marginal cost of sustaining update activity (energy, compute, bandwidth, engineering overhead).

In the baseline (no slashing risk, and with acceptance nonbinding so $w_i = 1$), an honest participating node solves

$$\max_{x_i \in [x^{\min}, x^{\max}]} \Big\{ r \ln x_i - \sigma_i x_i \Big\},$$

where $x^{\min}, x^{\max}$ summarize feasibility or protocol bounds. The objective is strictly concave in x_i, yielding the interior first-order condition

$$\frac{r}{x_i} - \sigma_i = 0 \quad \Rightarrow \quad x_i^*(r) = \frac{r}{\sigma_i}.$$

Translating back to the buffering period $\theta_i = 1/x_i$, we obtain

$$\theta_i^*(r) = \frac{\sigma_i}{r},$$

clipped to the feasible interval $[\theta^{\min}, \theta^{\max}]$ induced by $x^{\min}, x^{\max}$ and any hard AoI/latency eligibility constraints. This is the baseline best response underlying Proposition 1: higher r steepens the reward for timeliness and induces more frequent updates (smaller θ_i), with sensitivity governed by the private cost σ_i.

Two comparative-statics points are immediate and economically useful. First, the elasticity of the chosen intensity is constant: $x_i^*(r)$ scales linearly in r. Second, heterogeneity maps cleanly into performance: low-cost nodes (small σ_i) optimally deliver fresher information (smaller θ_i) under a uniform posted price.

Participation (IR) in the baseline. Even without adversaries, participation is not automatic because nodes incur operating costs and may face a per-round opportunity cost of locking capital if the mechanism requires escrowed stake. With no slashing in the baseline, the stake s is returned, so its only economic bite is the opportunity cost $\kappa_s s$. Using the interior choice, the maximized operating surplus (before stake opportunity cost) is

$$\max_x \ \{ r \ln x - \sigma x \} = r \ln \left(\tfrac{r}{\sigma} \right) - r.$$

Hence an honest identity's baseline individual rationality condition can be written as

$$r \ln \left(\tfrac{r}{\sigma_i} \right) - r - \kappa_s s \ \geq \ 0,$$

again subject to the possibility that bounds/eligibility constraints force x_i away from the interior optimum. This expression makes transparent how r plays a dual role: it both increases the slope of rewards and increases rents net of operating cost, thereby expanding the set of σ_i types willing to participate. In the strict baseline where we abstract from security needs, the server would set $s = 0$ (and $q = 0$) because these instruments only create deadweight costs when there is no adversarial behavior to deter. We nonetheless keep $\kappa_s s$ visible because the same IR margin becomes pivotal once stake is repurposed for deterrence in the next section.

Server's Stackelberg problem: pricing timeliness for satisfaction. Given the honest best response $\theta_i^*(r) = \sigma_i/r$, the server anticipates a mapping from its posted r into the realized timeliness profile and thus into satisfaction. In the benchmark, robust aggregation is innocuous (all updates are honest), and the server's per-round utility reduces to

$$V(r) \;=\; \sum_{i\in H} \beta\, G_i\big(\theta_i^*(r)\big) \;-\; \sum_{i\in H} R_i^{\text{base}}\big(\theta_i^*(r); r\big),$$

up to any fixed operating costs. Substituting the best response, payments take the convenient closed form

$$R_i^{\text{base}}\big(\theta_i^*(r); r\big) = r \ln\Big(\frac{1}{\sigma_i/r}\Big) = r \ln\Big(\frac{r}{\sigma_i}\Big),$$

while timeliness enters satisfaction through $\theta_i^*(r) = \sigma_i/r$. Thus, for a given participating set, raising r has a benefit (it reduces θ_i, improving freshness/latency and hence G_i) and a cost (it increases transfers through $r\ln(r/\sigma_i)$).

To see the structure most sharply, consider a representative participating identity of type σ with satisfaction contribution $G(\theta)$ that is decreasing in θ over the relevant range (fresher is better). The server's per-identity objective is

$$\beta\, G\Big(\frac{\sigma}{r}\Big) \;-\; r \ln\Big(\frac{r}{\sigma}\Big).$$

An interior optimal r (ignoring entry effects for the moment) satisfies the first-order condition

$$\beta\, G'\Big(\frac{\sigma}{r}\Big) \cdot \Big(-\frac{\sigma}{r^2}\Big) \;=\; \ln\Big(\frac{r}{\sigma}\Big) + 1.$$

Because $G'(\cdot) < 0$, the left-hand side is positive: it is the marginal satisfaction gain from increasing r via improved timeliness. The right-hand side is the marginal transfer cost of increasing r, which is strictly increasing in r. This delivers a standard Stackelberg intuition: we push r upward until the marginal value of inducing faster updates exactly equals the marginal cost of paying for them.

Budget and timeliness constraints. In practice, the server often faces explicit procurement constraints: a per-round payment budget $\mathcal{B}$ and/or operational timeliness requirements implemented through eligibility thresholds such as $A_i(\theta) \le A^{\max}$ and $E_i(\theta) \le E^{\max}$. In the baseline, these constraints map directly into bounds on $\theta_i^*(r)$. For example, if $A_i(\theta)$ is increasing and $A_i(\theta) \le A^{\max}$ implies $\theta \le \bar{\theta}_A$, then a sufficient condition for feasibility for type σ_i is

$$\theta_i^*(r) = \frac{\sigma_i}{r} \;\le\; \bar{\theta}_A \quad\Leftrightarrow\quad r \;\ge\; \frac{\sigma_i}{\bar{\theta}_A}.$$

Hence hard timeliness requirements translate into a *minimum* price needed to elicit admissible behavior from a given cost type. Conversely, a budget

constraint $\sum_i R_i \leq \mathcal{B}$ limits how far r can be raised, especially when participation expands with r. Economically, the baseline mechanism therefore functions as a timeliness "market-clearing" device: r must be high enough to induce admissible freshness from the marginal participating node, but not so high that transfers exceed the server's willingness (or ability) to pay.

Satisfaction-aware Stackelberg equilibrium (baseline). Putting these pieces together, a baseline Stackelberg equilibrium is a pair $\left(r^{\text{base}}, \{\theta_i^{\text{base}}\}\right)$ such that (i) given r^{base}, each honest participating node chooses $\theta_i^{\text{base}} = \sigma_i / r^{\text{base}}$ (clipped to feasibility/eligibility bounds), and (ii) r^{base} maximizes the server's expected welfare subject to the induced participation/feasibility constraints (and any budget constraint). This benchmark clarifies what the contract is buying even before introducing adversaries: a single scalar r implements a disciplined timeliness response across heterogeneous nodes, and the server's optimal pricing rule is pinned down by a marginal tradeoff between satisfaction improvements from fresher information and the convex-in-r transfer cost embedded in $r \ln(r/\sigma)$.

This is also where the security instruments will later "hook in." Once adversaries can mimic participation, the same levers that shape honest timeliness (r) must be paired with levers that price malicious behavior $(s, q, \phi, \bar{z})$. The baseline equilibrium thus provides the clean reference point against which we evaluate deterrence and welfare degradation when poisoning and Sybils are reintroduced.

5. Adversary model: poisoning, Sybils, and audit-based enforcement. We now reintroduce strategic manipulation by allowing a subset of participating identities to be adversarial. Our goal is not to model a particular attack in full engineering detail, but to capture the economically relevant margins: (i) an adversary can *distort* the learning signal by submitting a poisoned update, (ii) it can *scale* its influence by spawning Sybil identities, and (iii) it trades off an external benefit from degradation against an expected monetary penalty induced by anomaly detection, auditing, and slashing.

Poisoning as an additive action. For each participating identity i, the submitted update is a vector $u_i \in \mathbb{R}^d$. Honest identities submit an unmanipulated update g_i, which may depend on their chosen update cycle θ_i through data volume and freshness. Adversarial identities instead choose an additive perturbation

$$u_i \;=\; g_i + \delta_i,$$

where δ_i is the poisoning action. This reduced form encompasses both *untargeted* poisoning (pushing the model away from the honest descent direction) and *targeted* attacks (pushing toward a specific malicious objective), while keeping the contract-design problem tractable.

Two modelling choices matter. First, we allow adversaries to coordinate their $\{\delta_i\}_{i\in A}$ across identities; this is without loss for worst-case analysis, and it is operationally plausible when a single actor controls multiple devices or Sybils. Second, we do not assume the server observes g_i or raw data—only u_i, along with timestamp-derived timeliness metrics and anomaly scores computed after aggregation. This asymmetry is precisely why purely ex post "correctness" verification is hard, motivating probabilistic audits and collateral.

External sabotage benefits. Adversaries derive an exogenous benefit from harming the global model, for example by degrading predictive performance in a safety-critical IIoT pipeline, increasing downtime, or reducing the principal's profit. We represent this by a nonnegative degradation functional $\mathcal{D}(\bar{u})$, where $\bar{u} = Agg(\{u_i\}_{i=1}^{I})$ is the robust aggregate update used by the server. In utility terms, each adversarial identity obtains

$$B \cdot \mathcal{D}(\bar{u}),$$

where $B \geq 0$ scales the adversary's "value of sabotage." This formulation is deliberately agnostic about the exact learning dynamics: all that matters for incentives is that the adversary can increase $\mathcal{D}$ by choosing δ, but that the mapping from δ to $\bar{u}$ is attenuated by the aggregation rule.

To connect sabotage to the statistical layer, we use a bounded-influence property of robust aggregation: when at most an ε-fraction of updates are adversarial, the shift in the aggregate is controlled,

$$\|Agg(\{g_i + \delta_i\}) - Agg(\{g_i\})\| \;\leq\; C_{Agg}\,\varepsilon \cdot \max_{i\in A} \|\delta_i\|,$$

(or an ε-linear bound under bounded updates). Economically, this means the marginal return to poisoning is decreasing in the strength of robustness and increasing in adversarial market share.

Sybil capability through a stake budget. A key difficulty in open participation settings is that "one agent, one vote" is not enforceable: an adversary may create many identities. We treat Sybils as controlled identities that can each submit an update and claim payment, but must each satisfy the protocol's entry requirements. The central instrument is stake: each participating identity must escrow s for the duration of an audit window.

We model the adversary as having a total stake budget W. Because the mechanism requires posting s per identity per round, the number of adversarial identities that can participate is bounded by

$$|A| \;\leq\; \left\lfloor \frac{W}{s} \right\rfloor.$$

Thus, conditional on a total number of participants I after selection, the effective adversarial fraction satisfies

$$\varepsilon \;\equiv\; \frac{|A|}{I} \;\le\; \frac{W}{sI}.$$

This captures the practical role of stake as an identity-pricing device: it converts Sybil creation from a near-free action into a resource-constrained decision. Importantly, it also links security to market thickness: for fixed W and s, adversarial share falls as honest participation expands (larger I). Conversely, if the principal sets s too high, it may deter honest entry and inadvertently raise ε among those who remain. This is the basic security–participation tension we will carry into the contract problem.

Adversaries mimic timeliness unless deterred. Because the payment rule rewards timeliness, an adversary may also choose a buffering period θ (or intensity) strategically. In many deployments, nothing prevents an attacker from appearing "fresh": it can submit frequent updates even if their content is malicious. Accordingly, we allow adversarial identities to (i) choose θ_i to satisfy eligibility thresholds on AoI/latency and to maximize base payments, and then (ii) choose δ_i to trade off sabotage benefit against detection risk. This is the substantive reason we cannot rely on timeliness incentives alone for security: the mechanism must couple timeliness pricing to content-based screening and penalties.

Detection technology: anomaly scores and probabilistic auditing. We assume the server computes a robust aggregate $\bar{u}$ and then assigns each identity an anomaly score

$$z_i \;=\; \|u_i - \bar{u}\|,$$

or more generally a robust distance. An identity is flagged if $z_i > \bar{z}$, where $\bar{z}$ is a threshold chosen by the server. This is not yet a "conviction"; rather, it is a trigger that determines whether (a) the identity is accepted for payment and (b) whether it is exposed to slashing conditional on audit.

The critical behavioural link is that larger perturbations are easier to detect. We encode this by a detection probability function $p_{\mathrm{det}}(\delta)$ that is increasing in $\|\delta\|$. Since the server may not deterministically audit every round, we introduce an audit probability $q \in (0, 1]$. With probability q, an audit/check occurs within a window of length T_a after provisional settlement. We treat T_a as institutional latency: it can reflect, for example, time to run heavier consistency checks, cross-validate against held-out data, or reconcile multi-round behavioural signals.

Putting these pieces together, the probability that a deviating identity is ultimately penalized is

$$\mathbb{P}(\text{slashed} \mid \delta) \;=\; q\, p_{\mathrm{det}}(\delta),$$

where p_{det} is itself shaped by the threshold $\bar{z}$, the aggregation rule, and the ambient noise in honest updates. The main limitation is worth stating explicitly: p_{det} is not "ground truth verification." It is a statistical screen, so it inevitably entails both false negatives (missed attacks) and false positives (honest-but-noisy updates flagged). This is precisely why we separate (i) gating/weighting in payment and (ii) slashing under audit: the former manages learning quality in real time, while the latter supplies deterrence through expected penalties.

Slashing and the adversary's monetary tradeoff. When an audit occurs and an identity is flagged, the mechanism slashes a fraction $\phi \in (0, 1]$ of the posted stake s. Conceptually, ϕs is the collateral "at risk" that backs the integrity of submitted updates. In addition, a flagged identity may forfeit its base payment (through acceptance/weighting rules applied to z_i), which further increases the expected marginal cost of large deviations.

Thus an adversarial identity's expected payoff from a poisoning choice δ can be written as the sum of (i) any base payment it can still collect if not rejected, (ii) operating costs and the opportunity cost of locking stake, (iii) expected slashing losses under audit, and (iv) the sabotage benefit $B\mathcal{D}(\bar{u})$. This makes clear what the server can and cannot do. The server cannot directly reduce B, but it can (a) reduce the mapping from δ to $\bar{u}$ via robust aggregation, (b) increase the expected penalty via s, q, ϕ, and (c) tune $\bar{z}$ to manage the acceptance/deterrence tradeoff. The next section formalizes how these levers combine into an implementable mechanism that preserves the baseline timeliness incentives while making poisoning and Sybil scaling privately unattractive.

6. Mechanism: timeliness-aware payment, anomaly-weighted acceptance, and stake escrow. We now assemble the principal's instruments into a single implementable contract. The design objective is to preserve the baseline economic logic—pay more for fresher, lower-latency contributions—while adding a content-integrity layer that makes poisoning (and Sybil scaling) privately unattractive. The resulting mechanism is intentionally modular: timeliness metrics are observed from metadata, robustness is enforced statistically through aggregation and anomaly scoring, and deterrence is provided financially through escrow and slashing.

Contract form and observable signals. In each round t, the server posts a contract

$$\mathcal{C} \;=\; (r, s, q, \phi, \bar{z}; A^{\max}, E^{\max}),$$

together with a specified robust aggregation rule $Agg(\cdot)$ and a public description of how anomaly scores are computed. The contract uses three classes

of observables: (i) timeliness metadata (timestamps, receipt times) that induce an Age-of-Information statistic $A_i(\theta_i)$ and a service latency statistic $E_i(\theta_i)$; (ii) the submitted update vector u_i (or a commitment to it) used only through the robust aggregate $\bar{u}$ and anomaly score z_i; and (iii) protocol events within the audit window (whether an audit is triggered and whether slashing is executed). Crucially, the contract does not require the server to observe private costs σ_i or raw data, and it does not require deterministic verification of "correctness." Instead, it prices observables and penalizes statistically suspicious behaviour.

Timeliness-aware base reward. We operationalize the timeliness incentive by paying a concave reward in update frequency. Using the scalar reward parameter $r > 0$, the base component is

$$R_i^{\text{base}}(\theta_i; r) \;=\; r \,\ln\Big(\frac{1}{\theta_i}\Big),$$

which captures diminishing returns to ever-faster updates while maintaining tractability for equilibrium analysis. This form is not essential—any increasing concave function would play a similar role—but the log specification makes transparent how the server trades off marginal freshness against linear operating costs at the node. In practice, θ_i need not be explicitly declared: the server can infer an "effective" update cycle from inter-arrival times or rolling windows, and use that inferred statistic to compute R_i^{base}.

Acceptance region as a three-way screen (AoI, latency, anomaly). Timeliness rewards alone cannot prevent low-quality or malicious content from being paid. We therefore couple the base reward to an acceptance rule that is deliberately simple to implement and to audit. Define the acceptance indicator

$$w_i \;\equiv\; \mathbf{1}\{z_i \le \bar{z}\} \cdot \mathbf{1}\{A_i(\theta_i) \le A^{\max}\} \cdot \mathbf{1}\{E_i(\theta_i) \le E^{\max}\},$$

where $\bar{z}$ is the anomaly threshold and $A^{\max}, E^{\max}$ are hard timeliness caps. The induced *acceptance region* in observable space is

$$\mathcal{R} \;\equiv\; \{(A, E, z) : A \le A^{\max},\; E \le E^{\max},\; z \le \bar{z}\}.$$

The economic role of $\mathcal{R}$ is twofold. First, it prevents the server from paying for updates that are predictably useless for the learning task (too stale, too delayed). Second, it makes large poisoning attempts expensive even before any audit is realized: if poisoning pushes z_i above $\bar{z}$, the identity is rejected for payment in that round (and, as we specify below, becomes eligible for slashing conditional on audit). This separation between "real-time gating" and "ex post penalties" is important in IIoT settings, where the principal

may want to immediately exclude suspicious updates from training while still allowing due process through an audit window.

Although we present w_i as a binary gate, the same structure accommodates continuous weighting—e.g., replacing $\mathbf{1}\{z_i \le \bar{z}\}$ with a decreasing function of z_i. We focus on the indicator because it yields clean incentive bounds: the marginal benefit of pushing beyond $\bar{z}$ collapses sharply, strengthening deterrence when combined with stake at risk.

Robust aggregation and anomaly scoring. Given submitted updates $\{u_i\}_{i=1}^{I}$, the server computes the robust aggregate

$$\bar{u} \;=\; Agg(\{u_i\}_{i=1}^{I}),$$

and then assigns each identity an anomaly score, for example

$$z_i \;=\; \|u_i - \bar{u}\|.$$

What matters for the mechanism is not the specific norm or distance, but that z_i is computed *relative to a robust center* rather than a vulnerable mean. This ensures that anomalous scores are informative even when a minority of identities are adversarial, and it aligns the statistical layer with the economic layer: if a deviation δ_i can only weakly move $\bar{u}$ under robust aggregation, then (for fixed δ_i) the distance $\|u_i - \bar{u}\|$ tends to increase, raising the probability of rejection and eventual slashing.

Payments: provisional settlement with escrow-backed finality. We implement the transfer as a two-stage settlement that mirrors common practice in blockchain and in high-assurance procurement: pay provisionally for responsiveness, but delay finality until the audit window closes. The round-t payment rule is

$$R_i \;=\; w_i \, R_i^{\text{base}}(\theta_i; r),$$

where R_i is paid provisionally at the end of the round (or credited in an internal ledger), while the stake s remains locked in escrow for T_a periods. This escrow is not cosmetic: it is the capital-at-risk that makes probabilistic auditing bite.

Within the window, an audit is triggered with probability q. If an audited identity is flagged (i.e., its round-t realization had $z_i > \bar{z}$), then a fraction ϕ of the escrowed stake is slashed, so the penalty is ϕs. If the identity is not flagged (or if no audit occurs), the stake is returned after T_a, and provisional payments become final. This sequencing does two practical things. First, it separates the computation-heavy components (robust aggregation, anomaly scoring, possible additional checks) from the fast path needed to keep training moving. Second, it allows the principal to choose q as a costed enforcement intensity rather than an all-or-nothing verification burden.

Why escrow is essential (and how it interacts with acceptance). From an incentive perspective, the escrowed stake provides a monetary downside that scales with identity creation. Without it, Sybils can collect payments whenever they evade detection, and the mechanism's only defense is statistical robustness—effective against small contamination, but not sufficient when the adversary can cheaply increase ε. With escrow, each additional identity requires capital, and each detected deviation generates an expected loss proportional to $q\phi s$. The acceptance rule complements escrow: because payment is gated by w_i, an identity that crosses the anomaly threshold loses upside in the same round in which it creates statistical harm, even before an audit outcome is realized. This "lose the prize, risk the bond" structure is the core economic coupling between learning quality control and enforceable deterrence.

Implementability: on-chain escrow and off-chain computation. The mechanism is implementable as a hybrid system. The minimal on-chain (or trusted execution) components are: (i) staking/escrow of s per identity, (ii) an unambiguous record of submission timestamps and receipt times (to compute A_i and E_i), (iii) a commitment to the posted contract $\mathcal{C}$ for the round, and (iv) automated release/slashing logic after the audit window. Everything else can be off-chain: robust aggregation $Agg(\cdot)$, anomaly scoring, and any expensive audits (e.g., cross-validation, gradient consistency checks, or multi-round behavioural analysis).

A practical pattern is: nodes submit either the update u_i directly to the server (with a hash committed on-chain) or submit an encrypted update to a designated off-chain aggregator; the server (or an external auditor) then posts a signed attestation of the vector of flags $\{\mathbf{1}\{z_i > \bar{z}\}\}$ along with commitments to $\bar{u}$ and relevant statistics. The on-chain contract can accept the attestation as the trigger for slashing, possibly with a challenge period. This division is important because $u_i \in \mathbb{R}^d$ is typically large, and putting raw updates on-chain is infeasible. Conversely, the economic instruments we rely on—escrow, slashing, and time-locked finality—are exactly what smart contracts are good at enforcing.

Limitations and tuning knobs. Two caveats matter for practice and will matter for equilibrium. First, anomaly-based screening is imperfect: if $\bar{z}$ is too strict, honest-but-noisy devices are rejected and may exit (tightening participation constraints); if $\bar{z}$ is too lax, more poisoning is accepted and fewer identities are exposed to slashing. Second, audits are costly ($c_a q$ in server utility), so q must be chosen as an enforcement intensity that balances deterrence against operational burden. These are not bugs—they are the central tradeoffs the model is meant to illuminate. By parameterizing the contract with $(r, s, q, \phi, \bar{z})$, we give the principal a small set of levers that

map cleanly to institutional choices: reward generosity, collateralization, enforcement intensity, penalty severity, and statistical strictness.

Having specified the mechanism, we next analyze behaviour under $\mathcal{C}$: honest nodes choose θ_i and participation given r and the acceptance constraints, while adversaries choose whether and how much to poison given the expected penalty $q\phi s$ and the acceptance loss induced by $\bar{z}$. This equilibrium analysis yields explicit deterrence thresholds $(s^\star, q^\star, \phi^\star)$ and clarifies when the mechanism achieves robustness with only $O(\varepsilon)$ welfare loss.

7. Equilibrium analysis: honest effort/participation and adversary best responses. We now characterize behaviour under a posted contract $\mathcal{C} = (r, s, q, \phi, \bar{z}; A^{\max}, E^{\max})$. The key equilibrium objects are (i) the honest node's update-cycle choice θ_i and participation decision, and (ii) the adversary's poisoning magnitude δ (and participation scale) given that deviation raises anomaly risk and exposes escrowed stake to probabilistic slashing. Throughout, we treat $Agg(\cdot)$ and the anomaly scoring rule as fixed and publicly known, so agents best-respond to the induced mapping from deviations to acceptance and detection probabilities.

Honest update-cycle choice. Fix a round and consider an honest node i that expects to be accepted (i.e., $w_i = 1$) for its feasible choice of θ_i. Ignoring constant stake-return terms (the stake is returned absent slashing, but posting it incurs an opportunity cost $\kappa_s s$), the honest node's problem is to choose θ_i to maximize

$$r \ln\left(\tfrac{1}{\theta_i}\right) \; - \; \sigma_i \theta_i \qquad \text{subject to} \qquad \theta_i \in [\theta^{\min}, \theta^{\max}] \cap \Theta^{\text{time}},$$

where $\Theta^{\text{time}} \equiv \{\theta : A_i(\theta) \le A^{\max},\ E_i(\theta) \le E^{\max}\}$ is the "timeliness-feasible" set. The objective is strictly concave in θ_i, so the first-order condition gives the interior best response

$$\theta_i^*(r) = \frac{\sigma_i}{r},$$

with the understood projection ("clipping") to the feasible set whenever the caps bind. Economically, r raises the marginal value of freshness while σ_i captures the node's marginal operating cost of maintaining a shorter cycle; thus higher r induces smaller θ_i (more frequent updates), while higher σ_i induces larger θ_i.

Two practical remarks are useful. First, the timeliness caps can be interpreted as enforcing a minimum service level: if $A_i(\theta)$ and $E_i(\theta)$ worsen with θ, then Θ^{time} typically imposes an upper bound $\theta \le \bar{\theta}_i^{\text{time}}$. In that common case, the effective best response is $\theta_i^*(r) = \min\{\max\{\sigma_i/r, \theta^{\min}\}, \bar{\theta}_i^{\text{time}}\}$. Second, because R_i^{base} is concave in update frequency, the equilibrium response is smooth: increasing r does not generate corner solutions at "infinitely fast" updates unless $\theta^{\min}$ is extremely small.

Honest participation (IR) with escrow frictions. Given the best-response $\theta_i^*(r)$, an honest node participates if its expected utility is nonnegative. Under the maintained interpretation that the stake is returned unless slashed, the stake affects honest utility primarily through (i) opportunity cost $\kappa_s s$ and (ii) any probability of erroneous slashing. Let $\alpha \in [0,1]$ denote an (exogenous) false-positive rate—i.e., the probability an honest update is flagged as anomalous due to noise or heterogeneity, conditional on an audit being triggered. Then an honest node's expected utility can be written as

$$\mathbb{E}[U_i^H] \;\approx\; \Pr(w_i = 1)\left(r \ln\left(\tfrac{1}{\theta_i^*(r)}\right) - \sigma_i \theta_i^*(r)\right) \;-\; \kappa_s s \;-\; q\,\alpha\,\phi s,$$

where $\Pr(w_i = 1)$ captures the possibility that timeliness caps (or anomaly screening) exclude the node even when behaving honestly. Under the interior solution $\theta_i^*(r) = \sigma_i / r$ and $\Pr(w_i = 1) \approx 1$, the net operating surplus term simplifies to

$$r \ln\left(\tfrac{r}{\sigma_i}\right) - \frac{\sigma_i^2}{r}.$$

The individual-rationality condition $\mathbb{E}[U_i^H] \geq 0$ therefore produces a lower bound on r for a given s, q, ϕ (and environment κ_s, α). In words: raising s tightens participation through the opportunity-cost channel even if honest nodes are almost never slashed; raising q and ϕ tightens participation only to the extent that false positives α are non-negligible. This is the first place where the statistical and economic layers meet: better anomaly calibration (lower α for fixed deterrence) relaxes participation constraints without sacrificing security.

In heterogeneous populations, the IR condition induces a cutoff rule: for a given contract, there is a maximum cost type $\bar{\sigma}(r, s, q, \phi)$ that participates. This selection effect matters for the principal's design because increasing r attracts more honest nodes (and encourages smaller θ), but may also increase payments to inframarginal nodes.

Adversary objective and the poisoning-versus-stealth tradeoff. Consider an adversarial identity that can choose a perturbation δ (possibly coordinated across identities) so that $u = g + \delta$. The adversary trades off (i) sabotage benefit $B \cdot \mathcal{D}(\bar{u})$, which is increasing in the induced aggregate distortion, against (ii) the probability of being rejected for payment (via $w_i = 0$ when $z_i > \bar{z}$) and (iii) the expected slashing loss $q\, p_{\text{det}}(\delta)\, \phi s$, where $p_{\text{det}}(\delta)$ is increasing in $\|\delta\|$. Holding fixed θ (e.g., chosen to satisfy timeliness caps), the incremental expected payoff from poisoning relative to not poisoning can be organized as

$$\Delta(\delta) \;\equiv\; \underbrace{B\,\Delta\mathcal{D}(\delta)}_{\text{sabotage gain}} \;-\; \underbrace{q\, p_{\text{det}}(\delta)\, \phi s}_{\text{expected slashing}} \;-\; \underbrace{\big(\Pr(w = 1 \mid 0) - \Pr(w = 1 \mid \delta)\big)\, R^{\text{base}}(\theta; r)}_{\text{lost payment from rejection}},$$

where $\Delta\mathcal{D}(\delta)$ denotes the marginal degradation generated by δ. This decomposition highlights two distinct deterrence channels: (i) *ex post* punishment through escrow and slashing, and (ii) *ex ante* loss of contemporaneous reward when anomalous behaviour crosses the acceptance boundary.

A subtle but important point is that the base reward r plays an ambiguous security role. If an adversary can poison while keeping $z \leq \bar{z}$, then higher r increases the per-identity rents from participating and can subsidize adversarial presence. Conversely, when poisoning tends to increase anomaly scores (so that $\Pr(w = 1 \mid \delta)$ falls with $\|\delta\|$), a higher r raises the opportunity cost of being rejected and therefore strengthens deterrence at the margin. The contract thus couples r with $(s, q, \phi, \bar{z})$: rewarding timeliness cannot be chosen independently of integrity enforcement.

Deterrence thresholds and a sufficient staking condition. A "no-profitable-poisoning" condition is $\Delta(\delta) \leq 0$ for all $\delta \neq 0$. Rearranging yields the sufficient deterrence inequality

$$B\,\Delta\mathcal{D}(\delta) \;\leq\; q\,p_{\mathrm{det}}(\delta)\,(\phi s) \;+\; \big(1 - \Pr(w = 1 \mid \delta)\big)\,R^{\mathrm{base}}(\theta; r), \qquad \forall\,\delta \neq 0.$$

This makes transparent how deterrence can be achieved by (a) increasing the "expected bond at risk" $q\phi s$, (b) improving detection power $p_{\mathrm{det}}(\cdot)$ via stricter anomaly scoring or better robust baselines, and/or (c) increasing the immediate forfeiture from rejection.

To obtain an explicit threshold, suppose there exist bounds $\Delta\mathcal{D}(\delta) \leq \bar{D}$ for all relevant deviations and $p_{\mathrm{det}}(\delta) \geq \underline{p}$ for all $\|\delta\| \geq \delta_0$, where δ_0 is the smallest "meaningful" poisoning magnitude. Ignoring (or lower bounding by zero) the rejection-loss term yields a conservative sufficient condition:

$$s \;\geq\; s^{\star} \;\equiv\; \frac{B\,\bar{D}}{q\,\underline{p}\,\phi}.$$

Thus, for fixed (q, ϕ), stake must scale linearly with the adversary's marginal sabotage value B and inversely with enforcement intensity $q\underline{p}$. This is the economic analogue of a security budget: if audits are rare (low q) or detection is weak (low $\underline{p}$), the only way to preserve incentives is to increase collateral at risk.

Equilibrium regimes and the role of participation scale. These conditions naturally partition outcomes into regimes. If s (or $q\phi$) is sufficiently large relative to B, then any nontrivial δ is strictly dominated and adversarial identities either (i) do not participate, or (ii) participate but submit $\delta = 0$, effectively behaving like honest nodes from the mechanism's perspective. If deterrence fails, adversaries optimally choose δ at an interior point where marginal sabotage gain is balanced by marginal expected penalty and

marginal rejection risk; in that regime, robust aggregation becomes the primary statistical backstop (and we will later bound the resulting welfare loss).

Finally, the escrow mechanism disciplines Sybil scaling because the adversary's participation decision must account for capital at risk per identity. Even before invoking an explicit stake budget, the per-identity expected gain from entering (including sabotage value) must exceed the combined opportunity cost $\kappa_s s$ and expected slashing loss. This pushes the equilibrium away from "cheap identity inflation" and toward a setting where the server can meaningfully choose (r, s, q, ϕ) to satisfy both honest IR and adversarial deterrence.

8. Robustness and welfare bounds: from bounded influence to $O(\varepsilon)$ degradation. We now connect the statistical layer (robust aggregation and anomaly scoring) to the economic layer (payments, participation, and welfare). The organizing idea is simple: even if some fraction ε of submitted updates are adversarial, an ε-robust aggregator limits how far the aggregate $\bar{u}$ can be pushed; if the server's downstream satisfaction is Lipschitz in that aggregate error, then the welfare loss is at most linear in ε, up to transfer and audit-cost terms. This is the sense in which the mechanism is *scalable*: the harm depends on the *fraction* corrupted rather than the absolute number of identities.

Bounded influence of robust aggregation under contamination. Fix a round with I participating identities, of which at most $|A|/I \le \varepsilon < 1/2$ are adversarial. Let honest submissions be $\{g_i\}_{i\in H}$ and adversarial submissions be $\{g_i + \delta_i\}_{i\in A}$. We compare the realized aggregate

$$\bar{u} \;=\; Agg(\{u_i\}_{i=1}^{I}) \qquad \text{to the honest-only benchmark} \qquad \bar{u}^H \;=\; Agg(\{g_i\}_{i\in H}).$$

A wide class of robust estimators admits a contamination (or influence) bound of the form

$$\|\bar{u} - \bar{u}^H\| \;\le\; C_{Agg}\,\varepsilon \max_{i\in A} \|\delta_i\| \qquad \text{(general bounded-influence form)}, \quad (1)$$

where the constant C_{Agg} depends on the chosen estimator (median, trimmed mean), the trimming level, and the norm/dimension. When we additionally assume a bounded-update environment—either because gradients are clipped in implementation or because the model enforces $\|u_i\| \le G$—we can remove explicit dependence on $\max \|\delta_i\|$ and obtain a purely ε-linear bound:

$$\|\bar{u} - \bar{u}^H\| \;\le\; C\,\varepsilon\,G. \quad (2)$$

The economic relevance of (1)–(2) is that they decouple the harm from the absolute scale of participation: adding more honest nodes does not amplify

vulnerability so long as the adversarial *share* remains below the breakdown point.

To make the mechanism-level interpretation explicit, note that robust aggregation and anomaly scoring are complements rather than substitutes. Robust aggregation protects the *model update* even when some attacks slip past acceptance; anomaly scoring and slashing change adversarial incentives, reducing the equilibrium magnitude and frequency of δ. In equilibrium, we typically expect both effects: the effective contamination is smaller than the raw ε, and the damage conditional on contamination is bounded by (2).

From aggregate error to satisfaction loss. We next translate aggregation error into loss in the server's per-round satisfaction. Let the server's satisfaction contribution be summarized by G_i, and suppose the mapping from the aggregate update to satisfaction is Lipschitz: there exists $L_G > 0$ such that, holding fixed timeliness features,

$$\Big|\sum_{i=1}^{I} \beta\, G_i(\bar{u}) \;-\; \sum_{i=1}^{I} \beta\, G_i(\bar{u}^H)\Big| \;\le\; L_G\, \|\bar{u} - \bar{u}^H\|. \tag{3}$$

This assumption is not innocuous—we are effectively ruling out knife-edge regions where arbitrarily small update differences generate discontinuous changes in satisfaction—but it is aligned with practice in ML-driven control systems where performance metrics vary smoothly with parameter updates once gradients are bounded.

Combining (2) with (3) yields the headline statistical guarantee:

$$\text{Satisfaction loss from poisoning} \;\le\; L_G\, C\, \varepsilon\, G, \tag{4}$$

up to the extent that the honest-only benchmark $\bar{u}^H$ is the relevant counterfactual. Economically, (4) says that robustness turns adversarial influence into a "tax" proportional to market share ε.

Welfare decomposition and an $O(\varepsilon)$ bound. Server welfare per round is

$$V \;=\; \sum_{i=1}^{I} \beta\, G_i(\cdot) \;-\; \sum_{i=1}^{I} R_i \;-\; c_a q,$$

where payments R_i depend on acceptance w_i and timeliness-reward parameters. To compare equilibrium welfare against the no-adversary benchmark, we decompose the gap into three interpretable pieces:

$$V^{\text{no-adv}} - V^{\text{eq}} \;=\; \underbrace{\Big(\sum_i \beta G_i(\bar{u}^H) - \sum_i \beta G_i(\bar{u})\Big)}_{\text{(A) statistical satisfaction loss}} \;+\; \underbrace{\Big(\sum_i R_i^{\text{eq}} - \sum_i R_i^{\text{no-adv}}\Big)}_{\text{(B) transfer/selection effects}} \;+\; \underbrace{c_a q}_{\text{(C) audit cost}}.$$

Term (A) is bounded by (4). Term (C) is explicit. The delicate term is (B): payments can change because (i) some adversarial identities may still be paid if they remain below the anomaly threshold, (ii) some identities (honest or adversarial) may be rejected and thus unpaid, and (iii) participation may shift due to the contract.

A conservative bound is obtained by upper-bounding payments per participating identity by some $\bar{R}$ (e.g., using feasible bounds on θ so that $r \ln(1/\theta) \leq \bar{R}$). Then the total transfer distortion from a fraction ε of adversarial identities, plus any rejections at rate proportional to ε (under calibrated screening), is at most on the order of $\varepsilon I \bar{R}$. Normalizing per identity (or per round with fixed I) yields an $O(\varepsilon)$ term. Putting the pieces together, we obtain the welfare degradation form

$$V^{\text{no-adv}} - V^{\text{eq}} \;\leq\; L_G\, C\, \varepsilon\, G \;+\; O(\varepsilon) \cdot \bar{R} \;+\; c_a q, \tag{5}$$

where the constant hidden in $O(\varepsilon)$ depends on how acceptance probabilities and participation respond to adversarial presence, but crucially not on I itself. This formalizes the intuition that the mechanism's "fragility" is not the size of the federation but the effective adversarial share.

A useful refinement, consistent with the incentive analysis from the previous section, is that deterrence reduces the *effective* ε by making large δ privately unattractive. When slashing and rejection are strong enough that adversaries either abstain or mimic honest behaviour ($\delta = 0$), the statistical term in (5) essentially vanishes and only the audit cost remains. When deterrence is imperfect, robust aggregation ensures that whatever poisoning survives cannot generate more than linear harm.

Sybil resistance via stake budgets and "priced identity." Finally, we incorporate Sybil capacity explicitly. Suppose an adversary has total stake capital W that can be locked during the audit window, and each identity must escrow s. Then in any round the adversary can field at most

$$N_A \;\leq\; \left\lfloor \frac{W}{s} \right\rfloor \qquad \Rightarrow \qquad \varepsilon \;\equiv\; \frac{N_A}{I} \;\leq\; \frac{W}{sI}.$$

Substituting into (5) gives a concrete end-to-end bound that links economic design to statistical robustness:

$$V^{\text{no-adv}} - V^{\text{eq}} \;\leq\; L_G\, C\, G \cdot \frac{W}{sI} \;+\; O\Big(\frac{W}{sI}\Big)\bar{R} \;+\; c_a q. \tag{6}$$

Two implications are worth emphasizing. First, increasing s improves robustness not only by raising the expected penalty when slashed, but also by directly limiting the adversary's market share through a capital constraint; this is "Sybil resistance by collateral." Second, this channel is stronger when participation is large: for fixed W and s, increasing I dilutes adversarial share and tightens (6), a point that supports open participation *conditional on* robust aggregation and timeliness screening.

Limitations and practical interpretation. The bounds above are intentionally stylized. Constants C and L_G can be loose in high dimension, and heterogeneity in honest updates can widen the distribution that robust estimators must tolerate, potentially forcing a higher anomaly threshold $\bar{z}$ and weakening detection power. Moreover, bounding $\|u_i\|$ (via clipping) is not merely technical: without it, a single extreme update can create large losses unless the estimator has strong breakdown properties. From a design perspective, these observations motivate implementing (i) gradient clipping and normalization, (ii) robust aggregation with a trimming level matched to the expected ε, and (iii) audit/sanction policies calibrated to keep honest false positives low while maintaining deterrence.

With these welfare and Sybil-resistance relationships in place, we are positioned to study how the contract instruments $(s, q, \phi, r, \bar{z})$ move equilibrium outcomes as primitives (latency preferences τ/λ, network conditions, and adversarial value B) vary, which is the focus of the next section.

9. Comparative statics and design guidance: instruments, preferences, and network conditions. Having established that robust aggregation bounds statistical damage and that staking/audits can shift adversarial incentives, we now ask how the contract instruments $(s, q, \phi, r, \bar{z})$ should move as primitives vary. The comparative statics are useful not because the mechanism pins down a single "optimal" number, but because they clarify which knob should be turned first under different operational constraints: capital-limited Sybils versus unconstrained attackers, scarce audit capacity versus cheap verification, and latency-critical control versus quality-dominant training.

Base reward r primarily controls honest timeliness, but it is a blunt security tool. On the honest side, the interior best response is $\theta_i^*(r) = \sigma_i/r$ (clipped to feasibility). Thus

$$\frac{\partial \theta_i^*}{\partial r} = -\frac{\sigma_i}{r^2} < 0,$$

so increasing r induces more frequent updates (smaller θ), improving timeliness metrics that are decreasing in θ (e.g., periodic AoI scales like $A(\theta) \propto \theta$). The participation margin is more subtle: raising r increases gross reward but also pushes agents toward tighter cycles that raise their private operating cost $\sigma_i\theta$. In our log-linear specification the net effect is positive for sufficiently high-cost types only up to the point where acceptance constraints bind (e.g., $E_i(\theta) \le E^{\max}$). Practically, this means r is best interpreted as a "frequency inducement" parameter, and it should be set jointly with feasible timeliness thresholds; raising r without adjusting feasibility can lead to churn (agents attempt smaller θ, violate latency limits, get $w_i = 0$, and exit).

For security, r has an ambiguous effect. On one hand, if attacks are more likely to be rejected or flagged when $\delta \neq 0$, then a higher r raises the opportunity cost of cheating because the adversary risks forfeiting a larger $R^{\text{base}} = r \ln(1/\theta)$. On the other hand, if an attacker can remain under the anomaly threshold (or exploit regions where $p_{\text{det}}(\delta)$ is low), then increasing r simply subsidizes adversarial participation. This is why we treat r as primarily a performance lever, and rely on $(s, q, \phi, \bar{z})$ for deterrence.

Stake s is the first-line defense against Sybils, but it loads the honest IR constraint. The stake requirement affects outcomes through two distinct channels. First, it scales expected slashing losses: the deterrence inequality is controlled by $q\, p_{\text{det}}(\delta)\, \phi s$. Second, it prices identity creation: if the adversary has a stake budget W, then $\varepsilon \leq W/(sI)$. These channels reinforce each other when Sybils are the main threat; in that case raising s both reduces the attacker's share and increases capital at risk per identity.

However, s is also the instrument most likely to violate honest individual rationality because honest agents incur at least the opportunity cost $\kappa_s s$, and may face accidental slashing under false positives. If we denote an honest false-flag probability by $\alpha(\bar{z})$ (decreasing in $\bar{z}$), then a useful back-of-the-envelope condition for "stake feasibility" is

$$\kappa_s s \;+\; q\, \alpha(\bar{z})\, \phi s \;\ll\; \mathbb{E}\left[r \ln\!\Big(\frac{1}{\theta_i^*(r)}\Big) - \sigma_i \theta_i^*(r) \right],$$

so that collateral costs do not dominate operating surplus. This highlights a design rule: when we need more deterrence but honest participation is fragile, it is often better to increase ϕ or q (expected penalty conditional on detection) rather than s (capital locked regardless of behavior), unless Sybils are binding.

Audit probability q trades off deterrence against a direct resource cost, and is most valuable when detection is informative. Increasing q linearly scales the expected penalty from any given detection rule, but it also increases server cost $c_a q$ (and, in practice, operational burden and delay). The marginal value of q is highest precisely when the anomaly score is informative, i.e., when $p_{\text{det}}(\delta)$ rises steeply with $\|\delta\|$ and when false positives can be controlled via $\bar{z}$. When anomaly signals are noisy—e.g., honest updates are heterogeneous or non-IID so that $\{z_i\}$ has heavy tails even without attacks—raising q can be wasteful: we audit more often, but learn little, and we increase the expected accidental penalty faced by honest participants. In such regimes, robustness investments (better $Agg(\cdot)$, clipping/normalization, feature-conditional thresholds) can dominate higher q.

A related practical point is that q can be made state-dependent: auditing more aggressively when $\max_i z_i$ is large, when model loss spikes, or when

participation is unusually concentrated. Such targeting preserves deterrence while economizing on c_a, and it is consistent with our framework as long as participants understand the mapping from observables to audit intensity.

Slashing severity ϕ is a powerful deterrent but must be paired with conservative flagging. Holding s fixed, increasing ϕ raises the expected cheating cost without increasing locked capital. This makes ϕ attractive when the main constraint is honest liquidity (a high s would deter entry) rather than tolerance for penalty risk. The catch is that ϕ amplifies the welfare consequences of classification error: if honest nodes are occasionally flagged, then expected honest losses scale with $q\,\alpha(\bar{z})\,\phi s$. Hence "harsh slashing" and "aggressive flagging" are complements only for security and substitutes for participation.

A useful design heuristic is to tie ϕ to the conservativeness of $\bar{z}$: if we raise ϕ, we should typically raise $\bar{z}$ (or require multi-round evidence) to keep $\alpha(\bar{z})$ small. Conversely, if operationally we must flag based on a noisy one-shot score (high α), then ϕ should be kept moderate and deterrence should lean more on s (pricing Sybils) and q (verification intensity), or on making $p_{\mathrm{det}}(\delta)$ steeper via improved anomaly features.

Preference parameters (τ, λ) shift the optimal mix toward timeliness enforcement versus throughput tolerance. Recall $G_i = \tau M_i - \lambda E_i$, where M_i is a quality proxy increasing in contributed data volume and freshness (e.g., via $D_i(\theta)$ and $A_i(\theta)$). When λ is large relative to τ (latency-critical control), the server should (i) reward shorter cycles more strongly (higher r) *and* (ii) tighten acceptance on latency (lower $E^{\max}$) to prevent the mechanism from buying "quality" at the cost of responsiveness. In contrast, when τ dominates λ (training-quality dominated), the server can relax $E^{\max}$ and focus on sustained participation and data volume, often with a lower r but higher tolerance for moderate θ if it improves signal-to-noise or reduces network congestion. Importantly, tightening timeliness acceptance can unintentionally strengthen security by reducing the feasible set of attacker strategies (fewer opportunities to hide large δ behind idiosyncratic delays), but it can also increase honest false positives if network conditions are volatile.

Network conditions map into effective costs and feasibility constraints; contract parameters should adapt to congestion. In IIoT settings, latency $E_i(\theta)$ is not purely a choice variable: it depends on wireless contention, queueing, and intermittent connectivity. Deteriorating network conditions effectively increase the shadow cost of short cycles (a smaller θ induces more transmissions, worsening congestion), and can make the acceptance indicators bind even for cooperative behavior. In our reduced form,

this looks like an increase in σ_i (higher operating burden per unit update intensity) and tighter feasible regions for (A_i, E_i). Comparative statics then recommend: during congestion, raising r may backfire by inducing overly frequent updates that violate $E^{\max}$; instead, we may want to (i) temporarily relax $E^{\max}$ or incorporate congestion-normalized latency measures, (ii) cap update frequency directly (a lower bound on θ), or (iii) shift compensation from "frequency" to "quality conditional on feasibility," paying more for well-formed updates that arrive within realistic windows.

Security also interacts with network variability: heterogeneous delays and losses broaden the distribution of anomaly scores (because gradients may reflect stale states), raising $\alpha(\bar{z})$ unless $\bar{z}$ is increased. Therefore, in poor network regimes, we should avoid simultaneously increasing ϕ and tightening $\bar{z}$; doing so risks punishing honest nodes for conditions outside their control. A more robust approach is to calibrate $\bar{z}$ conditional on observed AoI/latency strata, so that detection focuses on deviations that cannot be explained by staleness alone.

When to prefer stricter slashing versus higher base payments. Putting the pieces together, we can summarize a guiding tradeoff. If the binding problem is *insufficient honest participation or insufficient update frequency* under stable networks, then raising r is the cleanest lever: it moves θ^* predictably and improves performance, while security should be maintained via q, ϕ, s calibrated to keep deterrence intact. If the binding problem is *adversarial scale* (Sybil capacity), raise s first because it directly reduces ε. If the binding problem is *adversarial intensity* (large δ per identity) under good detection, raise ϕ and/or q because they scale expected punishment without necessarily excluding honest nodes—provided $\bar{z}$ is set to keep α low. Only in the intermediate regime—where attackers are somewhat detectable and honest participation is marginal—does it make sense to increase r as an auxiliary deterrent by raising the "forfeiture" cost of being flagged; even then, we should be cautious that higher r does not simply subsidize sophisticated low-δ attacks.

These comparative statics yield concrete, testable predictions about how equilibrium participation, timeliness, detection rates, and welfare move with each instrument and primitive. The next step is to operationalize them in a simulation environment where we can vary ε, stake budgets W, congestion patterns, and anomaly-score calibration, and then evaluate the sensitivity of outcomes to $(s, q, \phi, r, \bar{z})$ under realistic IIoT latency and AoI dynamics.

10. Simulation plan: poisoning/collusion experiments with AoI/latency, robust aggregation, and stake. The comparative statics above are intentionally "mechanism-level": they tell us which instruments should matter and in what direction, but they do not by themselves quantify magnitudes

under realistic IIoT dynamics (wireless contention, non-IID data, heterogeneous device capabilities) or realistic attack surfaces (low-amplitude poisoning that evades one-shot detection, coordinated Sybils, and delay-based obfuscation). We therefore propose a simulation program whose goal is to (i) instantiate the primitives $\big(A_i(\theta), E_i(\theta), p_{\mathrm{det}}(\delta), C_{Agg}, L_G\big)$ with empirically plausible values, (ii) test whether the predicted qualitative relationships show up under realistic noise, and (iii) produce operational "design maps" that translate constraints (audit budget, tolerable false positives, expected Sybil capital) into recommended $(s, q, \phi, \bar{z}, r)$ regions.

(i) Simulation environment: joint learning and networking layers. We simulate a repeated-round federated learning process over an IIoT-like network. Each round t proceeds exactly as in the timing described earlier: the server posts $\mathcal{C} = (r, s, q, \phi, \bar{z})$, participants stake s, choose θ_i, submit updates u_i, the server aggregates $\bar{u} = Agg(\{u_i\})$, pays provisional rewards R_i, and audits with probability q within window T_a. We record both "mechanism outcomes" (participation, payouts, slashing events) and "learning outcomes" (validation loss, convergence rate, robustness to poisoning). To connect to IIoT timeliness, we embed a simple yet tunable latency/AoI model: transmissions incur stochastic service times and queueing delays, so that the realized latency $E_i(\theta_i)$ depends on both the node's update intensity $1/\theta_i$ and ambient congestion. AoI is computed from timestamps; for periodic generation with random service, a useful operational proxy is

$$A_i(\theta_i) \approx \frac{\theta_i}{2} + E_i(\theta_i),$$

with the understanding that the exact mapping can be replaced by a more detailed AoI recursion without changing the contract logic (the server only uses observables).

We will implement two networking regimes to stress-test robustness: (a) a "stable" regime with light-tailed delay (e.g., lognormal with modest variance) where timeliness acceptance is rarely binding for honest nodes, and (b) a "volatile" regime with bursty congestion (mixtures/heavy tails, or time-varying contention) that increases both missed deadlines and apparent gradient heterogeneity due to staleness.

(ii) Agent population and behavioral model (honest and adversarial). Each potential participant draws a private cost σ_i from a calibrated distribution (e.g., lognormal to capture a long tail of constrained devices). Given r, honest nodes best-respond via $\theta_i^*(r) = \sigma_i/r$, clipped to $[\theta^{\min}, \theta^{\max}]$, and participate when expected utility satisfies the IR constraint. To preserve the mechanism interpretation, we compute expected honest utility using realized acceptance w_i and realized audit/slashing events, allowing honest nodes

to exit in future rounds if they experience repeated false flags (a simple "participation persistence" rule captures reputational discouragement without adding new strategic complexity).

Adversarial identities are generated either exogenously as a fraction ε of participants, or endogenously through a stake budget W yielding at most $\lfloor W/s \rfloor$ Sybils per round. Adversaries choose perturbations δ_i to maximize $\mathbb{E}[U^A]$, with two canonical attacker models: (1) *myopic best response* (choose δ each round given $(q, \phi, s, \bar{z})$), and (2) *coordinated collusion* (a coalition chooses a vector $\{\delta_i\}_{i \in A}$ and possibly heterogeneous θ_i to exploit robust aggregation's worst-case influence subject to detection). We will include "stealth" attacks that explicitly optimize a surrogate anomaly score (e.g., constrain $z_i \leq \bar{z}$) to mimic adaptive adversaries.

(iii) Learning task, robust aggregation, and anomaly scoring. To measure welfare-relevant degradation $\mathcal{D}(\bar{u})$ and connect to L_G, we require a learning task where poisoning can be quantified. We propose two tasks: (a) a standard supervised benchmark (e.g., image or sensor classification) partitioned non-IID across nodes, and (b) an IIoT-flavored forecasting/control proxy (e.g., anomaly detection on multivariate time series). The second better reflects latency sensitivity because stale updates can directly harm prediction.

We evaluate multiple $Agg(\cdot)$ rules: FedAvg (baseline, non-robust), coordinate-wise median, trimmed mean at several trimming levels, and optionally Krum/Multi-Krum. For each Agg, we empirically estimate an "influence slope" C_{Agg} by injecting controlled perturbations into an ε-fraction of updates and regressing $\|\bar{u} - \bar{u}^H\|$ on $\varepsilon \max \|\delta\|$ in a bounded-update regime (via gradient clipping). Anomaly scores z_i will be computed as robust distances to $\bar{u}$ (e.g., coordinate-wise median absolute deviation scaling, or a robust Mahalanobis distance on a low-dimensional projection). This lets us estimate two key operating curves: the false-positive rate $\alpha(\bar{z}) = \mathbb{P}(z_i > \bar{z} \mid i \in H)$ and the detection function $p_{\text{det}}(\delta) = \mathbb{P}(z_i > \bar{z} \mid \|\delta\|)$, both potentially conditioned on timeliness strata (A_i, E_i).

(iv) Contract calibration targets: mapping primitives to numbers. We calibrate (r, s, q, ϕ) against operational constraints rather than arbitrary scales. For r, we target a baseline update frequency distribution consistent with device energy/compute limits: choose r so that the induced $\theta_i^*(r)$ yields (say) the 50th percentile cycle length in a feasible range and keeps acceptance constraints non-binding in the stable regime. For s, we consider two anchors: an honest liquidity constraint (maximum stake that does not materially reduce participation) and an adversary capital constraint (plausible W for an attacker in the deployment context). For q and c_a, we set an "audit budget" (expected audits per round) and an "audit latency" T_a consistent

with the verification technology (e.g., secure enclaves, redundancy checks, or re-computation on a trusted subset). For ϕ, we calibrate to an acceptable expected loss under false positives $q\,\alpha(\bar{z})\,\phi s$, effectively choosing ϕ as large as possible subject to a tolerable honest risk threshold.

We also estimate L_G empirically: perturb the aggregate update by a controlled vector η (or equivalently perturb $\bar{u}$ through synthetic poisoning) and measure the change in a satisfaction proxy (validation performance combined with a timeliness score), fitting a local Lipschitz bound $\Delta G \leq L_G\|\eta\|$ over the relevant range.

(v) Experiment matrix: what we vary and what we measure. Our core experiment families are:

Poisoning amplitude sweeps. Fix ε and vary $\|\delta\|$ to trace out (a) model degradation $\mathcal{D}(\bar{u})$, (b) empirical $p_{\text{det}}(\delta)$, and (c) realized adversary profitability under $(s, q, \phi, \bar{z})$. This directly tests the deterrence inequality in Proposition 2 by comparing $\Delta U^A(\delta)$ to zero.

Sybil budget sweeps. Fix an attacker budget W and vary s, inducing different effective ε. We measure the resulting welfare loss $V^{\text{no-adv}} - V^{\text{eq}}$ and check whether it scales approximately linearly in ε under robust aggregation (Proposition 4) once we account for honest participation effects.

Congestion/timeliness shocks. Introduce time-varying delay distributions that tighten $E_i(\theta)$ feasibility. We record (i) the rate at which acceptance indicators w_i bind for honest nodes, (ii) the change in $\alpha(\bar{z})$ as gradient heterogeneity increases with staleness, and (iii) the interaction with ϕ (do harsher penalties produce disproportionate honest exit in volatile regimes?).

Collusion and adaptive stealth. Allow adversaries to coordinate $\{\delta_i\}$ and choose attacks that maximize degradation subject to $z_i \leq \bar{z}$ constraints (or low expected detection). This is the most demanding test of the mechanism: we evaluate whether tuning $\bar{z}$ conditional on (A_i, E_i), or using multi-round anomaly evidence, restores deterrence without collapsing participation.

(vi) Sensitivity analysis and reporting outputs. We will not rely on single-point estimates. Instead, we run global sensitivity (Sobol/variance-based) on the mapping $(r, s, q, \phi, \bar{z}, \varepsilon, W, \text{network regime}) \mapsto (V, \text{participation}, \text{false slashing}, \mathcal{D})$. This produces actionable rankings: e.g., in which regimes does s dominate q, or when does $\bar{z}$ calibration matter more than ϕ? We will also report "iso-welfare" and "iso-participation" contours over (s, ϕ) and $(q, \bar{z})$ to make the security–participation tradeoff operational. Finally, we will document failure modes—parameter regions where robust aggregation alone is insufficient (e.g., near breakdown points or under extreme non-IID heterogeneity), or where stake/audits backfire due to false positives—so that the subsequent deployment-facing discussion can be explicit about limitations rather than implicitly assuming away the hard cases.

11. Conclusion: implications for 2026 deployments; limitations and extensions (richer audits, privacy-preserving anomaly scoring). Our motivating premise is practical: by 2026, federated learning in IIoT settings will increasingly be procured as a *service* from heterogeneous devices owned by different parties, operating over congested wireless links, and facing adversaries that can scale through Sybils. In that environment, "robust learning" is not only a statistical problem; it is an incentive problem. The model we developed is deliberately spare, but it illuminates an operational logic: if we can make *timely contribution* profitable for honest devices while making *undetected manipulation* privately costly, then the system's degradation can be bounded even when a nontrivial minority of participants are adversarial.

The key mechanism-level implication is that three instruments—stake s, audit rate q, and slashing severity ϕ—jointly implement a *capital-at-risk* deterrent, while robust aggregation limits the marginal harm of residual attacks. In deployments, these are not abstract knobs: they correspond to concrete policy choices about escrow requirements, verification budget, and penalty rules, all of which must be justified to participants and to governance stakeholders. The economic point is that no single layer is sufficient. Robust aggregation alone limits influence but does not remove the adversary's incentive to "spend" its bounded influence every round; staking/auditing alone deters only to the extent that detection is credible; timeliness rewards alone can inadvertently encourage gaming (e.g., ultra-frequent low-quality updates) if not paired with acceptance constraints and anomaly checks. Layering is not just belt-and-suspenders engineering—it is the equilibrium logic that keeps the attack set small.

For 2026 procurement, we view the contract $\mathcal{C} = (r, s, q, \phi, \bar{z})$ as a *design map* rather than a single optimum. In practice, the server (principal) will face constraints that vary across deployments: limited audit throughput (upper-bounding feasible q), participant liquidity constraints (upper-bounding feasible s), and legal/organizational constraints on punitive penalties (upper-bounding feasible ϕ). Our contribution is to make explicit how these constraints substitute for one another: when ϕ is capped (e.g., due to due-process requirements), one needs either higher s or higher q; when q is expensive, larger s can economize on audits; when honest participation is fragile, increasing ϕ may be less distortionary than increasing s (holding expected false-positive harm fixed). This substitution logic is valuable even if the primitives are estimated only approximately.

A second deployment implication concerns *timeliness as an incentive-compatible observable.* IIoT systems already generate timestamps and service times; AoI and latency are therefore attractive as contractible metrics compared to unverifiable notions of "effort." Yet timeliness metrics are also noisy and environment-dependent. Our framework suggests a conservative operational posture: treat timeliness constraints primarily as *acceptance gates* (to

prevent stale updates from being paid as if they were fresh), and treat monetary rewards r primarily as a way to shift the distribution of update cycles θ_i among those who can meet feasibility. Put differently, timeliness should not be used to "prove honesty," but rather to price a resource the server values (freshness) while acknowledging that networking conditions can mimic anomalous behavior.

Third, the analysis clarifies what "Sybil resistance" means in a procurement setting. In many FL discussions, Sybils are framed as an identity problem; here they become a *budget problem*. Requiring stake s per identity converts identity creation into a priced input and caps the adversary's market share given capital W. This is a practically legible governance story for 2026: rather than claiming to "solve" Sybils cryptographically, the mechanism bounds their scale economically and then relies on robust aggregation to tolerate the remaining contamination. In systems where participants are organizations rather than consumer devices, s can be implemented as billing holdbacks, performance bonds, or insurance-backed guarantees—formats that procurement teams already understand.

That said, we should be explicit about limitations. First, our reward rule $R_i^{\text{base}}(\theta) = r \ln(1/\theta)$ and linear private cost $\sigma_i \theta_i$ were chosen for tractability and to yield a clean best response $\theta_i^*(r)$. Real devices have nonconvex energy/compute costs, hard duty-cycle constraints, and sometimes fixed costs of participation. These features can generate corner solutions (e.g., "all-in or all-out" update behavior) and thus change how r affects participation and congestion. Second, our deterrence condition relies on a detection function $p_{\text{det}}(\delta)$ that increases with poisoning magnitude; adaptive attackers can instead pursue *low-amplitude, persistent* manipulation designed to sit below any one-shot threshold $\bar{z}$. Third, robust aggregation guarantees are sharp only under conditions that are often strained in practice: bounded gradients (requiring clipping), limited adversarial fraction $\varepsilon < 1/2$, and not-too-extreme non-IID heterogeneity. When honest updates are themselves highly dispersed—due to data heterogeneity or staleness—"robust distance to the aggregate" can become a weak anomaly signal and can elevate false positives precisely when the network is most volatile.

These limitations point to immediate extensions that matter for 2026 deployments. On the *audit* side, we expect "richer audits" to be the main driver of improved security-cost frontiers. The binary audit model (audit with probability q, slash if flagged) can be generalized along at least three dimensions. (i) *Targeted audits*: make q a function of observable risk, e.g., $q_i = q(z_i, A_i, E_i)$, so that scarce verification resources focus on suspicious and high-impact updates. (ii) *Tiered penalties*: replace a single ϕ with penalty schedules that escalate with repeated anomalies or with the severity of deviation, which can reduce chilling effects on honest nodes under transient noise. (iii) *Redundancy audits*: verify an update by recomputation on a trusted subset, by cross-checking consistency against neighboring devices,

or by challenge tasks that are hard to fake without actually computing the claimed gradient. Each of these preserves the core logic—expected penalty equals detection probability times stake-at-risk—but improves either p_{det} at fixed cost or reduces the false-positive externality.

On the *privacy* side, anomaly scoring is increasingly constrained by secure aggregation and privacy regulations. In many federated systems, the server may not be allowed to inspect raw updates u_i, or doing so may undermine participant trust. This makes "privacy-preserving anomaly scoring" central rather than optional. A promising direction is to compute robust statistics under secure aggregation or MPC: for example, coordinate-wise trimmed means can be approximated via secure protocols; alternatively, nodes can submit cryptographic commitments to compressed representations of gradients, enabling the server to compute anomaly scores on random projections without learning the full update. Another approach is to separate *payment* from *model update*: nodes can be paid based on privacy-preserving consistency checks (e.g., norm bounds, agreement on low-dimensional sketches) while the model update itself is formed through secure aggregation. The design challenge is to ensure that whatever privacy-preserving statistic is used still yields a detection function $p_{\text{det}}(\delta)$ strong enough that the deterrence inequality remains plausible at acceptable s, q, ϕ.

We also see room for *dynamic* mechanism improvements that directly address stealthy, low-amplitude manipulation. In repeated interactions, the server can replace one-shot thresholds with multi-round evidence: sequential tests on the time series of anomaly scores, stake that vests over time, or "reputation-weighted" aggregation where influence grows only after a history of non-anomalous behavior. These instruments can raise the effective cost of persistent attacks without increasing false slashing in any single round. Importantly, they shift deterrence from a single detection event to an accumulated likelihood of detection, which is precisely what stealth attackers attempt to avoid.

Finally, we emphasize a governance implication. Staking and slashing are powerful precisely because they are punitive; they therefore require transparent procedures for dispute resolution, handling false positives, and defining what constitutes a provable violation. In regulated IIoT environments (manufacturing, energy, transport), this procedural layer is not ancillary—it is part of the mechanism's feasibility. Our model helps separate what must be true *economically* (expected penalties must dominate expected sabotage gains) from what must be implemented *institutionally* (credible audits, appeal processes, and acceptable collateral formats). If those institutional pieces are weak, then the system should lean more heavily on robust aggregation and conservative acceptance rules; if they are strong, then staking/auditing can carry more of the security burden and allow looser thresholds $\bar{z}$ that preserve participation.

In sum, the tradeoff the model illuminates is not "security versus learn-

ing," but "security versus participation under timeliness constraints." By 2026, successful deployments will likely be those that treat federated learning procurement as a contract design problem: specify what is observable (AoI/latency), what is verifiable (audits), what is punishable (slashing), and what is statistically tolerable (robust aggregation). Our analysis offers a coherent starting point—explicit enough to calibrate and stress-test, modest enough to extend—toward mechanisms that remain useful when adversaries, networks, and privacy constraints are all present at once.