Hidden Benchmarks and Anti-Gaming: Ambiguous Evaluation with Deployment Commitment

Liz Lemma Future Detective

January 14, 2026

Abstract

Modern AI procurement and benchmarking often face a practical asymmetry: the evaluator can randomize or conceal parts of the evaluation rule (e.g., hidden test sets, randomized red-team suites), while the evaluated model must be deployed deterministically for nontrivial periods due to compliance, reproducibility, or latency constraints. Motivated by ambiguous contracts (Dütting–Feldman–Peretz–Samuelson) and incentive-aware evaluation (Kleinberg–Raghavan; Alon et al.), we introduce a clean windowed contracting model capturing this asymmetry. A principal announces a set of payment/evaluation rules and privately commits to one rule for an entire deployment window; an ambiguity-averse agent chooses a single deployed action/policy for the window. We show that ambiguity can strictly improve principal utility — even under monotone evaluation constraints — by discouraging 'gaming' actions that exploit known evaluation rules. We prove structure theorems: optimal ambiguous evaluation can be implemented by a small support of simple contracts (single-outcome or threshold/step rules) and computed in polynomial time. In contrast, when the agent can hedge by mixing across actions within the window, ambiguity loses all power, recovering the 'mixing kills ambiguity' phenomenon. We provide comparative statics in window length and commitment/mixing constraints, and interpret results as a theory of hidden benchmarks as incentive instruments.

Table of Contents

3. 3. Model: windowed ambiguous evaluation, deployment commitment, ambiguity-averse agent, monotone evaluation constraint.

4. 4. Baseline equivalences: reduction from windowed i.i.d. setting to a scaled one-shot problem; mapping evaluation rules to contracts.

5. 5. Main separation: explicit instance where ambiguous monotone evaluation strictly beats any deterministic monotone evaluation; interpretation as anti-gaming.

6. 6. Structure and computation: support-size bounds (SOP/step), efficient algorithm; special strengthening under MLRP/FOSD-style regularity.

7. 7. When ambiguity fails: agent mixing/hedging inside window eliminates gains; limited-mixing extensions and bounds (if adopted).

8. 8. Comparative statics and extensions: partial ambiguity aversion, drift across windows, multiple metrics as outcomes, connections to hidden test sets and procurement.

9. 9. Discussion: practical design guidance, limits, open questions.

1 1. Introduction: hidden benchmarks, gaming vs improvement, why deployment commitment makes ambiguity realistic in 2026.

Modern model deployment increasingly resembles a contracting problem with a distinctive informational asymmetry: the benchmark by which a system is judged is often not fully revealed to the provider. In procurement, platform ranking, safety auditing, and enterprise evaluation pipelines, the evaluator typically has access to internal ground truth, private test suites, or proprietary user feedback streams that are not shared in full with the model provider. At the same time, the provider does observe enough about the evaluation *process* to adapt. This combination—partial transparency about the rules, paired with strategic adaptation—creates the familiar tension between genuine improvement and gaming. Our goal is to isolate a mechanism-design logic that is already implicit in many 2026-era evaluation practices: deliberately maintaining *structured ambiguity* over the scoring rule can improve incentives, even when payments must be monotone in an ordered notion of performance.

The motivating problem is a version of Goodhart's law. When a principal publishes a single deterministic metric or rubric, a sophisticated agent may optimize specifically for that metric, often by exploiting quirks that do not correspond to the principal's true objective. In model evaluation, "gaming" can mean overfitting to a known test distribution, learning dataset artifacts, strategically abstaining, or shifting effort toward borderline cases that move a thresholded score while degrading performance elsewhere. The principal's dilemma is not that the metric is uncorrelated with value; rather, the metric is an imperfect proxy, and the agent's optimization pressure amplifies the proxy's weaknesses. Simply "making the metric better" is costly, slow, or sometimes impossible when the relevant objective is multidimensional, partially unobservable, or evolves over time.

A natural response is to introduce randomness or secrecy into evaluation. In practice, evaluators already do this: rotating test sets, holding out private benchmarks, varying prompts, sampling items adaptively, and applying undisclosed post-processing rules (e.g., filtering, weighting, or thresholding) that are hard to reverse-engineer. These practices are often justified informally as anti-overfitting measures. We study a clean economic analogue: the principal commits ex ante to a *set* of permissible outcome-contingent payment rules and then privately selects which one is used. The agent observes the set but not the realized rule and therefore evaluates each action by its worst-case expected payment within the announced set. This captures a common behavioral premise in high-stakes settings—providers act conservatively when the evaluation rule is not fully known, especially when failures are salient and reputationally costly.

A key modeling choice is that the agent must commit to a single action for a nontrivial deployment window. This is not merely a technical convenience; it is what makes ambiguity realistic rather than vacuous. Many 2026 deployments involve continuous service: a model is selected, integrated, and then run for days or weeks with monitoring and periodic audits. Swapping policies per request is often infeasible because of engineering overhead, latency constraints, compliance logging, or the need for consistent user experience. Even when A/B testing is feasible, it is typically controlled by the platform (the principal) rather than the provider. Hence, it is natural to treat the agent's choice as a pure commitment within a window, while outcomes are realized repeatedly and aggregated. In our setting, ambiguity is also persistent within the window: the principal's privately selected contract applies to all rounds in that window, mirroring "hidden benchmarks" that remain fixed for a quarter, a release cycle, or an audit period.

This windowed commitment has two conceptual implications. First, it clarifies why benchmark secrecy can be credible: the principal can precommit to a family of scoring rules (for governance or transparency reasons) while still keeping the realized rule hidden to protect the integrity of the evaluation. Second, it separates two forms of robustness. Repetition within the window reduces statistical noise about outcome frequencies, but it does not eliminate strategic uncertainty about the mapping from outcomes to payments. In other words, even with abundant data, the agent may still face ambiguity about how the principal will translate observed outcomes into compensation or acceptance.

Our central theme is that ambiguity can be beneficial precisely because it lets the principal "target" different gaming behaviors with different worst-case scoring rules while holding fixed the expected payment for the desired action. Intuitively, a single deterministic monotone payment schedule may be forced into a compromise: if it punishes one deviation, it may inadvertently reward another. By committing to a small menu of monotone rules and selecting one privately, the principal can ensure that each undesirable deviation faces at least one rule under which it performs poorly, and an ambiguity-averse agent internalizes this through worst-case reasoning. At the same time, the principal can maintain consistency for the intended action so that ambiguity does not create unnecessary risk premia in equilibrium.

We also emphasize what the model does *not* claim. Ambiguity is not a free lunch if agents can finely hedge by mixing across actions within the window, if they are not meaningfully ambiguity-averse, or if the principal cannot credibly commit to the announced set of rules. Moreover, ambiguity is not a substitute for measurement quality; it is a tool for incentive alignment when measurement is necessarily imperfect. Finally, our analysis uses stylized outcome bins and i.i.d. realizations to highlight the incentive channel; extensions to richer feedback processes are important but orthogonal to the mechanism we identify.

With this motivation in place, we next situate our contribution relative to work on ambiguous contracts, strategic evaluation/classification, inspection-based contracting, and the appeal of simple (often monotone or threshold) payment rules.

2 2. Related work: ambiguous contracts (Econometrica 2024), strategic classification/evaluation schemes, contracts with inspections, and robustness/learnability of simple contracts.

A first point of contact is the recent literature on *ambiguous contracts*, formalizing environments in which the principal can commit to a *set* of contingent transfers while the agent evaluates actions using a worst-case (or otherwise ambiguity-sensitive) criterion. The Econometrica (2024) treatment is especially close in spirit: it makes precise how "menus of contracts" can be used as an incentive device even when the realized transfer rule is not publicly observed. Our contribution is complementary along three dimensions that matter for evaluation practice. First, we embed ambiguity in a repeated, windowed interaction in which the mapping from outcomes to transfers is fixed *within* a window but hidden *across* the window, reflecting the operational reality of internal benchmarks and audit cycles. Second, we focus on an ordered outcome space and (optionally) monotone transfers, which connects the theory to threshold- and rubric-based scoring rules widely used in procurement, safety audits, and model leaderboards. Third, we emphasize implementability and computation: rather than treating ambiguity as a general preference perturbation, we characterize how a principal can exploit ambiguity to "cover" multiple deviations using a small support of simple contracts, and we provide a polynomial-time procedure for selecting such a support.

A second strand relates to *strategic classification* and *strategic evaluation* (including strategic prediction, performative prediction, and gaming of metrics). This literature typically takes the evaluation rule as a deterministic and publicly known mapping from observable features to decisions or scores, then studies how a strategic agent responds by manipulating inputs, selecting effort, or shifting distributions. The central insight is that evaluation rules create incentives that feed back into data generation, so naive "optimize the metric" policies can lead to equilibrium distortions. Our perspective is not to endogenize features or retrain a classifier, but to ask what a principal can do when she is constrained to use outcome-contingent scores/payments and cannot perfectly observe the agent's action. In this respect, our model isolates a distinct lever: *commitment to structured ambiguity* over evaluation rules. Where strategic classification often proposes robustness to manipulation via regularization, causal features, or equilibrium-aware learning, we study how a principal can *discipline* manipulation by ensuring that each gaming direction is penalized under at least one plausible rule in the announced set. This lens is particularly natural in settings where publishing the full scoring rule is itself known to invite overfitting, but governance constraints require the principal to precommit to a class of admissible procedures.

A third connection is to classic and modern work on *contracts with inspections*, auditing, and monitoring. In those models, the principal may probabilistically inspect, verify outcomes, or impose penalties conditional on an audit signal, thereby creating incentives under limited observability. The shared mechanism is that hidden or randomized enforcement can deter opportunistic behavior. Our ambiguity device is conceptually similar—both introduce uncertainty from the agent's perspective—but it differs in what is being randomized. Inspection models typically randomize information acquisition or the probability of detection, whereas we randomize within a family of *payment mappings* from already-observed outcomes to transfers. This distinction matters in many digital evaluation environments: the principal may reliably observe outcome bins (scores, pass/fail categories, safety incidents) but may not be able to credibly commit to intensive, individualized inspections; instead, she can credibly commit to a family of scoring rules and keep the realized rule private. The windowed structure also parallels "inspection regimes" that remain fixed for a compliance period: the agent learns that the principal has selected one regime from a known set, but cannot condition behavior on the realized regime because switching actions is costly or infeasible midstream.

Finally, we connect to work on the *robustness and simplicity* of optimal contracts and mechanisms. A recurring theme in contract theory is that simple transfer schemes—threshold bonuses, piece rates, or monotone schedules—often suffice either exactly (via extreme point arguments) or approximately (via approximation and learning guarantees), especially when outcomes admit an order and likelihood ratios satisfy regularity conditions. In parallel, recent theoretical computer science and learning-theoretic work studies when simple scoring rules are more *learnable*, more transparent, or less vulnerable to manipulation than complex ones, sometimes formalizing a tradeoff between expressiveness and robustness. Our results speak to this tradeoff in a distinctive way: ambiguity can expand what is implementable even when each element of the ambiguous set is itself extremely simple (e.g., a single-outcome payment or a step/threshold contract). Thus, rather than viewing simplicity as an exogenous restriction that reduces performance, we show that *randomization across simple rules*—implemented as ambiguity from the agent's perspective—can recover incentive power that a single deterministic rule lacks. At the same time, our analysis highlights a limitation that is also familiar in the learnability literature: if the agent can effectively hedge (e.g., by mixing across behaviors at fine time scales), then the advantage of ambiguity collapses, echoing the broader point that robustness tools are sensitive to the agent's feasible response set.

Taken together, these literatures motivate a model in which the principal's commitment power lies not in perfect measurement or full transparency, but in committing to a constrained family of evaluation rules and controlling what is revealed when. We now formalize this as a windowed principal–agent problem with ordered outcomes, limited liability, and an ambiguity-averse agent who evaluates each action by its worst-case expected payment within the announced set.

3 3. Model: windowed ambiguous evaluation, deployment commitment, ambiguity-averse agent, monotone evaluation constraint.

We study a principal–agent environment in which evaluation and payment are jointly determined by a *windowed* procedure and by the principal's commitment to *structured ambiguity* over that procedure. The motivating interpretation is a platform or evaluator (the principal) running periodic benchmark windows (audit cycles, leaderboard periods, procurement trials) during which a provider (the agent) must deploy a single model or policy and cannot costlessly revise it midstream. The principal observes coarse, ordered outcome categories (metric bins) and can commit to pay as a function of the realized bin, but does not wish—or is not allowed—to fully reveal the exact scoring rule used within the window.

Actions, outcomes, and ordering. The agent chooses an action (a deployed model/policy) from a finite set $A = \{1, \ldots, n\}$. Outcomes lie in an ordered set $\Omega = \{1, \ldots, m\}$, where larger indices correspond to better observable performance according to the principal's reporting standard (e.g., higher accuracy tier, lower incident-severity tier recoded so that higher is better, "pass" levels in a rubric). Conditional on action i, each round's outcome $j \in \Omega$ is drawn i.i.d. from a distribution $q_i = (q_{i1}, \ldots, q_{im})$. The i.i.d. assumption is a reduced-form way to capture a stable deployment environment within a window: the agent's action is held fixed, and the principal observes repeated, comparable evaluations.

The principal derives per-round reward r_j from outcome j, with r_j exogenous and commonly known (we can normalize $r_j \in [0, 1]$ without loss for the comparative statics we emphasize). For action i, the principal's expected per-round reward is

$$R_i \;=\; \sum_{j=1}^{m} q_{ij} r_j.$$

Transfers and limited liability. A (deterministic) evaluation/payment rule is a nonnegative transfer vector $t = (t_1, \ldots, t_m) \in \mathbb{R}^m_+$, where t_j is the payment made when the realized outcome is j. Limited liability ($t_j \geq 0$) captures that evaluators typically cannot impose fines on providers, only withhold payments or bonuses. Given action i and contract t, the agent's expected per-round payment is

$$T_i(t) \;=\; \sum_{j=1}^{m} q_{ij} t_j,$$

and the agent's per-round utility is $T_i(t) - c_i$, where c_i is the (known) cost of deploying action i (development, compute, compliance burden, or foregone alternative revenue).

Because outcomes are ordered, we will often (optionally) restrict attention to *monotone* transfers:

$$t \in M \quad \Longleftrightarrow \quad t_1 \leq t_2 \leq \cdots \leq t_m.$$

This captures evaluation policies where "better observed performance cannot be paid less," as in threshold bonuses, rubric grading, and monotone score-to-payment mappings used for governance reasons.

Deployment windows and commitment within a window. Interaction occurs over a window of length $L \geq 1$. Within a window the agent must commit to a *single pure action* $i \in A$ and cannot switch actions across rounds. This constraint is central: it corresponds to operational frictions (integration costs, model update policies, approval gates) and makes a window resemble a single deployment decision followed by repeated measurement. We later contrast this with a setting in which the agent can effectively mix across actions at a fine time scale (e.g., randomized routing or per-instance model switching), which will sharply change the value of ambiguity.

Ambiguous evaluation (menus of contracts). Rather than committing to a single t, the principal announces an *ambiguous contract* (or evaluation menu) $\tau = \{t^{(1)}, \ldots, t^{(K)}\}$, a finite set of feasible transfer vectors (typically $\tau \subseteq M$ under monotonicity). The principal commits ex ante that, for the entire window, she will privately select one contract $t \in \tau$ and use it consistently on every round. The agent observes the announced set τ but does not observe which t is selected, and—because of the pure-action commitment—cannot condition behavior on that private realization. Practically, one can think of τ as a publicly disclosed family of admissible scoring rules, with the realized rule held back (or revealed only after the window) to reduce overfitting and gaming.

Ambiguity aversion and best responses. We model the agent as ambiguity averse in the max–min sense: upon seeing τ, the agent evaluates action i by its *worst-case* expected payment over $t \in \tau$. Since payoffs add over L i.i.d. rounds and the same i and t apply throughout the window, the agent's window utility is

$$U_A(i \mid \tau) \;=\; L \cdot \Big(\min_{t \in \tau} T_i(t) - c_i \Big),$$

and the agent chooses a best response

$$i^*(\tau) \in \arg\max_{i \in A} \; \min_{t \in \tau} \big(T_i(t) - c_i \big).$$

Participation (individual rationality) requires $U_A(i^*(\tau) \mid \tau) \geq 0$, equivalently $\min_{t \in \tau} T_{i^*(\tau)}(t) \geq c_{i^*(\tau)}$. We interpret this as an outside option normalized to zero and emphasize that ambiguity here disciplines incentives only insofar as the agent internalizes the worst-case contract in the announced set.

Principal payoff and observability. The principal observes the realized outcome sequence $(j_\ell)_{\ell=1}^{L}$ and knows which $t \in \tau$ she selected, but does not observe the agent's action. Her expected window payoff, given the induced action $i^*(\tau)$ and chosen t, is

$$U_P(\tau; t) \;=\; L \cdot \big(R_{i^*(\tau)} - T_{i^*(\tau)}(t) \big).$$

A key design consideration (made explicit later) is that, to speak unambiguously about "the" payoff from τ, the principal may restrict attention to ambiguous sets where the implemented action's expected payment is *consistent* across $t \in \tau$.

This model isolates the tradeoff we care about in evaluation practice: the principal can commit to a constrained class of outcome-contingent rules (often monotone and simple), yet by retaining controlled ambiguity she may alter incentives without changing what is observable. In the next section we show that, under the i.i.d. window structure, the windowed problem reduces to a scaled one-shot contracting problem, which lets us map evaluation rules directly to standard contract objects and carry over implementability arguments.

Baseline equivalence: the windowed problem is a scaled one-shot problem. Although our motivating environments are explicitly *dynamic*—a benchmark window contains L evaluations—the strategic object is fundamentally *static* under our two commitment assumptions: (i) the agent must choose a single pure action i for the entire window, and (ii) the principal privately selects a single contract $t \in \tau$ and applies it throughout the window. With i.i.d. outcomes conditional on i, both parties' window utilities become linear in L, and the incentive constraints reduce to the familiar one-shot max–min constraints.

Fix an ambiguous set τ and an action i. Let $j_1, \dots, j_L \sim q_i$ i.i.d. denote the realized outcome sequence. Because the same t is applied in every round, the agent's realized total transfer under t is $\sum_{\ell=1}^{L} t_{j_\ell}$, whose expectation is

$$\mathbb{E}\Big[\sum_{\ell=1}^{L} t_{j_\ell} \,\Big|\, i, t\Big] \;=\; \sum_{\ell=1}^{L}\sum_{j=1}^{m} q_{ij} t_j \;=\; L \cdot T_i(t).$$

Since the agent evaluates τ via a worst case over $t \in \tau$, his expected *window* payoff is

$$U_A(i \mid \tau) \;=\; \min_{t \in \tau} \Big(L \cdot T_i(t)\Big) \;-\; L \cdot c_i \;=\; L \cdot \Big(\min_{t \in \tau} T_i(t) - c_i\Big).$$

Therefore, comparing two actions i and i' under the same τ is identical in the windowed and one-shot formulations: multiplying all payoffs by L does not change the argmax. Formally,

$$i^*(\tau) \in \arg\max_{i \in A} \; \min_{t \in \tau} \big(T_i(t) - c_i\big) \quad \Longleftrightarrow \quad i^*(\tau) \in \arg\max_{i \in A} \; U_A(i \mid \tau).$$

The participation condition similarly collapses to its one-shot version:

$$U_A(i^*(\tau) \mid \tau) \geq 0 \quad \Longleftrightarrow \quad \min_{t \in \tau} T_{i^*(\tau)}(t) \geq c_{i^*(\tau)}.$$

Thus, the window length L scales *levels* (total dollars and total surplus) but not *comparisons* (which actions are optimal, and which constraints bind).

Principal objective under scaling (and why we separate consistency). The same linearity holds for the principal's reward and payments. Under action i, expected total reward over the window is $L \cdot R_i$ and expected total payment under t is $L \cdot T_i(t)$, so

$$U_P(\tau; t) \;=\; L \cdot \big(R_{i^*(\tau)} - T_{i^*(\tau)}(t)\big).$$

From the standpoint of *choosing* the menu τ, this immediately implies a scaling lemma: maximizing principal payoff in the windowed model is equivalent to maximizing the per-round objective $R_{i^*(\tau)} - T_{i^*(\tau)}(t)$, since L is just a positive multiplicative factor.

One subtlety is that the principal's expected payoff depends on the privately selected $t \in \tau$ unless we impose an additional tie-down. In applications, we want τ to be interpreted as a publicly committed evaluation policy rather than a vague promise whose realized stringency is discretionary. This is what motivates our *consistency* requirement: for the implemented action $i^*(\tau)$, the expected payment is the same across all $t \in \tau$, i.e.,

$$T_{i^*(\tau)}(t) = T_{i^*(\tau)}(t') \quad \forall\, t, t' \in \tau.$$

Under consistency, the principal's payoff is well-defined as a function of the set τ alone (not of an unspecified selection rule), and the scaling equivalence becomes clean: the optimal τ in the window is exactly the optimal τ in the one-shot problem, with payoffs multiplied by L.

Mapping evaluation rules to contracts. This reduction is useful because it lets us translate a broad class of practical evaluation procedures into standard contract language. An *evaluation rule* in our setting is any commitment that maps each realized outcome bin $j \in \Omega$ to a nonnegative payment t_j, potentially subject to monotonicity $t_1 \leq \cdots \leq t_m$. Once we adopt i.i.d. outcomes and within-window commitment, such a rule is fully summarized by the vector $t \in \mathbb{R}_+^m$, exactly as in a one-shot principal–agent model with discrete outcomes.

Ambiguous evaluation then corresponds to a *menu of such rules* $\tau = \{t^{(1)}, \ldots, t^{(K)}\}$, with the agent responding to τ via worst-case expected payment. In operational terms, τ captures a family of admissible scoring rubrics (or audit tests) that are publicly specified, while the realized rubric is held fixed but unrevealed during the window. The scaling equivalence tells us that the only economically relevant object is the collection of per-round expected payments $\{T_i(t) : i \in A, t \in \tau\}$: the distribution of the *sequence* $(j_\ell)_{\ell=1}^L$ matters only through its mean effect on payments and rewards when actions cannot be adjusted midstream.

Two clarifications delimit the scope of this equivalence. First, it relies on *separability* of transfers across rounds: we pay t_{j_ℓ} each round rather than a bonus based on the empirical distribution of outcomes over the entire window. Allowing truly history-dependent transfers would expand the contract space (outcomes become multinomial counts), and would be a different mechanism-design problem. Second, it relies on the *pure-action commitment* within a window: if the agent could randomize or switch actions across rounds, the window would no longer be strategically equivalent to one shot. We exploit this contrast later; here, the key point is that under our baseline frictions, repeated evaluation is simply a scaling device, and ambiguity operates through one-shot max–min incentives applied to per-round outcome bins.

Main separation: ambiguity can strictly improve welfare even under monotone evaluation. We now isolate the economic force behind our strict-gain result with a fully explicit instance in the smallest nontrivial ordered setting ($m = 3$ outcome bins). The intuition is that different "gaming" actions load on different parts of the outcome distribution. Any *single* monotone payment rule must trade off how strongly it rewards the middle bin versus the top bin, whereas an *ambiguous* monotone evaluation can make each deviation face a different worst-case rule while keeping the target action's expected payment fixed (consistency).

An explicit $n = 4$, $m = 3$ instance. Let outcomes be ordered $\Omega = \{1, 2, 3\}$, and restrict attention to monotone contracts $t = (t_1, t_2, t_3)$ with $0 \le t_1 \le t_2 \le t_3$. Consider four actions with outcome distributions

$$q_1 = (0.3,\ 0.5,\ 0.2), \qquad q_2 = (0.7,\ 0,\ 0.3), \qquad q_3 = (0.7,\ 0.25,\ 0.05), \qquad q_4 = (1,\ 0,\ 0),$$

and costs

$$c_1 = 0.18, \qquad c_2 = 0.08, \qquad c_3 = 0.03, \qquad c_4 = 0.$$

Let the principal's per-round reward be increasing but coarse,

$$r_1 = 0, \qquad r_2 = 1, \qquad r_3 = 1,$$

so that rewards depend only on "clearing the threshold" $j \ge 2$. Then

$$R_1 = \Pr_{q_1}[j \ge 2] = 0.7, \quad R_2 = 0.3, \quad R_3 = 0.3, \quad R_4 = 0.$$

Thus, from the principal's perspective, action 1 is the uniquely valuable action; actions 2 and 3 represent distinct ways of manipulating the evaluation distribution while producing little true value.

A monotone ambiguous evaluation that implements $i = 1$. Consider the ambiguous monotone set $\tau = \{t^{(H)}, t^{(M)}\}$ with

$$t^{(H)} = (0,\ 0,\ 1.05) \quad \text{(top-bin bonus)}, \qquad t^{(M)} = (0,\ 0.3,\ 0.3) \quad \text{(threshold/step payment)}.$$

Both are monotone, and they satisfy *consistency* for action 1:

$$T_1(t^{(H)}) = 0.2 \cdot 1.05 = 0.21 \qquad \text{and} \qquad T_1(t^{(M)}) = (0.5 + 0.2) \cdot 0.3 = 0.21.$$

Therefore, under τ the agent's worst-case expected payments are

$$\min_{t\in\tau} T_1(t) = 0.21, \quad \min_{t\in\tau} T_2(t) = \min\{0.315,\ 0.09\} = 0.09, \quad \min_{t\in\tau} T_3(t) = \min\{0.0525,\ 0.09\} = 0.0525,$$

Subtracting costs, the worst-case utilities are

$$0.21 - 0.18 = 0.03, \quad 0.09 - 0.08 = 0.01, \quad 0.0525 - 0.03 = 0.0225, \quad 0 - 0 = 0,$$

so the ambiguity-averse agent strictly prefers $i = 1$ and participates. The principal's per-round payoff under τ is

$$U_P(\tau) = R_1 - \underbrace{T_1(t)}_{=0.21} = 0.7 - 0.21 = 0.49,$$

well-defined because of consistency.

Why no deterministic monotone evaluation can match this payoff. For any monotone $t = (t_1, t_2, t_3)$, write it in "increment" form:

$$t_1 = \alpha, \qquad t_2 = \alpha + a, \qquad t_3 = \alpha + a + b, \qquad \alpha, a, b \geq 0.$$

Then $T_i(t) = \alpha + a \Pr_{q_i}[j \geq 2] + b \Pr_{q_i}[j = 3]$. Crucially, α shifts *all* actions' expected payments by the same constant, so it cannot help incentive compatibility; it only increases transfers. Hence, any optimal deterministic monotone contract sets $\alpha = 0$.

With $\alpha = 0$, the IC constraints for inducing $i = 1$ against deviations 2 and 3 become

$$0.7a + 0.2b - c_1 \ \geq\ 0.3a + 0.3b - c_2 \quad \Longleftrightarrow \quad 0.4a - 0.1b \geq 0.10,$$

$$0.7a + 0.2b - c_1 \ \geq\ 0.3a + 0.05b - c_3 \quad \Longleftrightarrow \quad 0.4a + 0.15b \geq 0.15,$$

together with IR $0.7a + 0.2b \geq c_1 = 0.18$. Minimizing the implemented action's expected payment $T_1 = 0.7a + 0.2b$ subject to these inequalities yields an optimum at $b = 0$ and $a = 0.375$, giving

$$\min\{T_1(t) : t \in M \text{ implements } 1\} = 0.7 \cdot 0.375 = 0.2625.$$

Therefore, the best deterministic monotone payoff from implementing action 1 is at most

$$0.7 - 0.2625 = 0.4375 \ < \ 0.49 = U_P(\tau).$$

Moreover, implementing any other action $i \neq 1$ yields per-round payoff at most $R_i - c_i \leq 0.22$, so the deterministic *optimal* payoff is also strictly below 0.49.

Interpretation as anti-gaming. Action 2 "chases the top bin" (it is paid heavily under $t^{(H)}$) but is disciplined by the threshold contract $t^{(M)}$ in the menu; action 3 "chases the middle bin" (it is paid under $t^{(M)}$) but is disciplined by $t^{(H)}$, which effectively ignores the middle bin. Deterministic monotone evaluation cannot simultaneously be stringent on both margins without raising the target action's expected payment; ambiguity-with-commitment achieves exactly that separation.

Structure: why small support and extreme-point contracts suffice. The explicit construction above is not a knife-edge artifact. It reflects a general geometric feature of the principal's design problem under max–min preferences: to implement a target action i, the principal only needs (i) *consistency* for i—all contracts in the support induce the same expected transfer $T_i(t)$—and (ii) at least one contract that serves as a *worst case* for each relevant deviation $i' \neq i$. Once we view τ as a device for assigning different deviations to different "most punitive" contracts while keeping i's expected payment fixed, it becomes natural that the support can be small, and that we can restrict attention to extreme-point contracts.

Formally, fix a candidate implemented action i and let $\bar{T}$ denote the common value of $T_i(t)$ over $t \in \tau$. Under consistency, the agent evaluates action i as $\bar{T} - c_i$, while each deviation i' is evaluated as $\min_{t\in\tau} T_{i'}(t) - c_{i'}$. Hence, implementation reduces to finding the smallest $\bar{T}$ such that, for every $i' \neq i$,

$$\min_{t\in\tau} T_{i'}(t) \;\le\; \bar{T} - c_i + c_{i'} \qquad \text{and} \qquad \bar{T} \ge c_i \text{ (IR)}.$$

The principal's objective for a fixed i is then to minimize $\bar{T}$ (to maximize $R_i - \bar{T}$), subject to being able to "push down" each deviation's worst-case expected payment.

Small-support theorem (general) and SOP reduction. A key simplification is that we can assume each $t \in \tau$ is a *single-outcome payment* (SOP): $t = \gamma e_j$ for some outcome j and scalar $\gamma \ge 0$, where e_j pays only on outcome j. Intuitively, if a contract is chosen to be the worst case for a deviation i', then (holding $T_i(t) = \bar{T}$ fixed) we should concentrate payment on outcomes that are *relatively more likely under i than under i'*. Concentration is exactly what SOP contracts do; they are extreme points of the limited-liability polytope and therefore minimize linear objectives subject to linear constraints.

This yields the following structural statement: there exists an optimal ambiguous contract $\tau^\star$ with

$$|\tau^\star| \;\le\; \min\{m,\; n-1\},$$

such that every $t \in \tau^\star$ is SOP. The $n-1$ bound reflects that, in the worst case, we need at most one "covering" contract per deviation; the m bound reflects that SOP contracts come in only m distinct outcome locations, so additional contracts are redundant. Importantly, this is an *existence* result: optimal menus may admit multiple representations, but there is always one with small support and extreme-point form.

Monotone outcomes: step contracts replace SOP. When we impose monotonicity $t \in M$ (i.e., $t_1 \le \cdots \le t_m$), SOP contracts are typically infeasible. The correct replacement is a *step* (threshold) contract: pick a cutoff $k \in \{1, \ldots, m\}$ and a level $\gamma \ge 0$, and pay

$$t_j^{(k)} = \begin{cases} 0, & j < k, \\ \gamma, & j \ge k. \end{cases}$$

Step contracts are the relevant extreme points of the monotone, limited-liability set once we mod out by dominated "smoothing" that raises payments in low outcomes without improving incentives. Accordingly, the same support bound continues to hold, with “SOP” replaced by “step,” and with the economic interpretation that ambiguity chooses among a small number of thresholds so that different deviations fear different thresholds.

Efficient computation: minimizing the consistent transfer for each candidate action. These structural reductions lead directly to a fast algorithm. Fix i and a target consistent expected payment $\bar{T}$. Consider a single SOP contract $t = \gamma e_j$ that satisfies $T_i(t) = \bar{T}$. This forces $\gamma = \bar{T}/q_{ij}$ (assuming $q_{ij} > 0$), and induces deviation i' to receive

$$T_{i'}(t) = q_{i'j}\gamma = \bar{T} \cdot \frac{q_{i'j}}{q_{ij}}.$$

Thus, among SOP contracts that keep i at $\bar{T}$, the minimal payment to deviation i' is

$$\min_{t:\, T_i(t)=\bar{T}} T_{i'}(t) = \bar{T} \cdot \min_{j\in\Omega:\, q_{ij}>0} \frac{q_{i'j}}{q_{ij}}.$$

With ambiguity, we can include (at most) one SOP contract that attains this minimum for each deviation i', without affecting consistency for i. Plugging this expression into the IC inequalities yields closed-form lower bounds on $\bar{T}$; taking the maximum over deviations (and IR) gives the minimal feasible $\bar{T}_i^\star$. The principal then computes, for each i,

$$\Pi_i \;=\; R_i - \bar{T}_i^\star,$$

and selects the action (and corresponding menu) maximizing Π_i.

Under monotonicity, the same logic applies with SOP replaced by step contracts. For a threshold k, the constraint $T_i(t^{(k)}) = \bar{T}$ pins down $\gamma = \bar{T}/\Pr_{q_i}[j \geq k]$, and deviation i' receives $\bar{T} \cdot \Pr_{q_{i'}}[j \geq k]/\Pr_{q_i}[j \geq k]$. We can therefore evaluate each deviation’s “most punitive threshold” by scanning cutoffs and comparing tail-probability ratios. Precomputing all tail probabilities takes $O(nm)$, and checking all (i, i', k) combinations yields an $O(nm^2)$-type routine overall (with small constants), after which we again pick the best i.

Regularity sharpening: two thresholds under ordered-likelihood conditions. If outcome order is informative in the sense of MLRP/FOSD-style single-crossing (so likelihood ratios or tail ratios move monotonically in the cutoff), then the minimizing threshold for each deviation is extremal: deviations are optimally punished either by a “high” threshold emphasizing top outcomes or by a “low” threshold emphasizing broad passing. In such environments, the set of binding deviations can be covered by at most *two* step contracts, implying the existence of an optimal monotone ambiguous contract with $|\tau^\star| \leq 2$. Practically, this says that when the metric is well-ordered, the platform does not need a complicated random evaluation rule:

two transparently interpretable thresholds can suffice to obtain essentially all the gains from ambiguity-with-commitment.

We view these results as clarifying both power and limits: ambiguity expands implementability by letting the principal "separate" deviations across a small menu, but the benefit is disciplined by extreme-point structure (simple contracts) and by efficient computation (no combinatorial explosion in K).

When ambiguity fails: hedging within the window. All of the gains from ambiguity-with-commitment rely on a knife-edge institutional feature: within a deployment window, the agent must effectively *commit* to a single action $i \in A$. If instead the agent can hedge by mixing across actions inside the window—for instance by routing different user requests to different models, running an ensemble, or randomizing its policy per round—then the principal loses the ability to make "different deviations fear different worst cases." In that environment, ambiguity becomes payoff-equivalent to a deterministic contract.

Formally, suppose that in stage (2) the agent can choose any mixed action $p \in \Delta(A)$, interpreted as randomizing i.i.d. across the L rounds (or equivalently selecting a randomized policy that induces the mixture distribution over outcomes). Let $q_p \equiv \sum_i p_i q_i$ denote the induced outcome distribution, and assume costs aggregate linearly as $c(p) \equiv \sum_i p_i c_i$ (the natural benchmark when each round uses one action with its per-round cost). Then under an ambiguous contract τ the agent's (per-round) objective becomes

$$\max_{p \in \Delta(A)} \min_{t \in \tau} \Big(T_p(t) - c(p)\Big), \qquad \text{where} \qquad T_p(t) = \sum_j (q_p)_j t_j = \sum_i p_i T_i(t).$$

The key observation is that $T_p(t) - c(p)$ is bilinear in (p, t), and the "min over t" is taken over a *finite* set. Because the agent now chooses from a convex set $\Delta(A)$, standard minimax logic applies: if we let $\mathrm{co}(\tau)$ denote the convex hull of τ, then

$$\max_{p \in \Delta(A)} \min_{t \in \tau} \big(T_p(t) - c(p)\big) = \max_{p \in \Delta(A)} \min_{t \in \mathrm{co}(\tau)} \big(T_p(t) - c(p)\big) = \min_{t \in \mathrm{co}(\tau)} \max_{p \in \Delta(A)} \big(T_p(t) - c(p)\big),$$

where the last equality follows from Sion's minimax theorem (compact convex domains; continuity; linearity in each argument). Finally, since $\max_{p \in \Delta(A)}(T_p(t) - c(p))$ is linear in t, the minimizer over $\mathrm{co}(\tau)$ is attained at an extreme point, i.e., at some $t \in \tau$. In words: once the agent can mix, the principal's "menu" is effectively replaced by its convex hull, and the worst-case evaluation selects a single contract anyway. Hence the principal cannot do better than what she could achieve by posting that single contract deterministically.

The economic intuition mirrors the formal argument. Ambiguity helps when each deviation $i' \neq i$ can be assigned a different punitive contract $t^{(k)}$

(while the implemented i sees the same expected transfer under all $t^{(k)}$). Mixing lets the agent pick p that hedges across which contract is realized: instead of being exposed to a deviation-specific "most punitive" contract, the agent can choose a mixture that performs acceptably under *every* contract in τ, thereby collapsing the principal's ability to differentially discourage deviations. Put differently, the principal's use of ambiguity creates a form of *nonconvexity* in the agent's effective choice problem under pure actions; allowing $p \in \Delta(A)$ convexifies the problem and restores a saddle point.

Limited mixing: why the gains shrink, and what remains. In many applications, the relevant question is not "pure commitment versus full mixing," but the size of the agent's hedging technology. Two practically motivated relaxations illustrate the general message that ambiguity gains are fragile to hedging, but need not disappear immediately.

First, suppose the agent can *mostly* commit, in the sense that it must place at least $1-\delta$ probability on some single action:

$$\Delta_\delta(A) \;\equiv\; \bigl\{p \in \Delta(A) : \max_i p_i \geq 1-\delta\bigr\},$$

with $\delta \in [0,1]$ capturing mixing capacity (e.g., operational constraints, governance requirements, or audit risk that limits traffic-splitting). As δ increases, the agent can better insure itself against the worst-case contract in τ, so the principal's optimal value under ambiguity is weakly decreasing in δ, and converges to the deterministic optimum at $\delta = 1$. Moreover, because both rewards and transfers are linear in p, the principal's incremental benefit from ambiguity is (at least) *continuous* in δ: small amounts of hedging cannot create large discontinuous jumps in payoffs. This is the sense in which ambiguity is most powerful precisely when the deployment environment enforces near-pure action commitment.

Second, suppose the agent can adapt within the window by switching actions across rounds but pays a per-switch friction (engineering cost, latency, compliance burden). A simple reduced form is to augment costs so that any nondegenerate mixture incurs an extra penalty $\kappa > 0$, i.e., $c(p) = \sum_i p_i c_i + \kappa \cdot \mathbf{1}\{p \text{ not pure}\}$. Then ambiguity can still matter whenever κ is large enough that the agent optimally chooses a pure action despite the ability to hedge. Conversely, as $\kappa \downarrow 0$ we recover the full-mixing benchmark and ambiguity again becomes redundant. This extension clarifies an operational interpretation: ambiguity is valuable when "gaming" requires a discrete switch to a different model or policy, not when the agent can smoothly interpolate among behaviors at negligible cost.

Implementation takeaway. The design lesson is concrete. Ambiguity is best understood as a commitment device that exploits the agent's exposure to a single hidden evaluation rule during a window. If the platform cannot prevent within-window hedging (ensembling, traffic-splitting, per-request policy randomization), then ambiguous evaluation rules are unlikely to outperform well-chosen deterministic ones. Conversely, when the platform can enforce one-model-per-window (through reproducibility constraints, signed artifacts, logging, or auditability), the ambiguity gains analyzed above become attainable rather than purely theoretical.

Partial ambiguity aversion (beyond max–min). Our baseline assumes the agent evaluates an ambiguous contract τ via the worst-case transfer $\min_{t\in\tau} T_i(t)$, which is the starkest form of ambiguity aversion. Many deployments, however, fall between max–min robustness and Bayesian averaging. A simple interpolation is an α-max–min criterion: fix a reference distribution π over τ (e.g., the platform's announced randomization device, or the agent's historical belief), and let the agent evaluate action i by

$$U_A^{\alpha}(i \mid \tau) \;=\; \alpha \cdot \min_{t\in\tau} \big(T_i(t) - c_i\big) \;+\; (1-\alpha)\cdot \mathbb{E}_{t\sim\pi}\big[T_i(t) - c_i\big], \qquad \alpha \in [0,1].$$

As $\alpha \downarrow 0$, the agent behaves more like a Bayesian expected-utility maximizer over the hidden t; as $\alpha \uparrow 1$, we recover our model. The comparative static is intuitive and general: holding fixed the pure-action commitment assumption, the principal's advantage from ambiguity is weakly increasing in α. Mechanically, the principal's design lever is to make each undesirable deviation $i' \neq i$ have *some* contract in τ that is especially unfavorable for it, while keeping the implemented action's expected transfer fixed across $t \in \tau$ (consistency). When $\alpha < 1$, that "bad" contract is discounted by $(1-\alpha)$, so deterring a deviation typically requires either (i) larger dispersion in payments across contracts in the support (subject to limited liability and monotonicity), or (ii) larger support size K so that each deviation is penalized more "often" under π. In practice, this suggests a concrete operational implication: ambiguity is most effective when the agent is institutionally required to plan for worst-case evaluation (e.g., audits, compliance, or reputational downside) rather than when it can treat evaluation as an average-case draw.

Drift across windows and learning about the evaluator. The windowed model isolates a single deployment window of length L. In many applications, the relevant environment is repeated, and both sides may face nonstationarity. Two distinct "drifts" matter.

First, the mapping from actions to outcomes may drift: q_i can change across windows due to distribution shift, model updates, or user adaptation. One role for ambiguity here is not strategic but *robust*: by selecting τ so that the implemented action is one whose performance is less sensitive to which $t \in \tau$ is used, the principal implicitly pushes the agent toward behaviors that are stable across plausible evaluation realizations. In this sense, τ can be interpreted as a coarse description of "what the platform cares about," and robustness to that description can be desirable when the platform itself anticipates drift in r_j or in the mapping from outcomes to downstream value.

Second, the agent may learn across windows about the platform's private selection procedure. Even if τ is publicly announced, repeated interactions can reveal which contracts are chosen more often, and realized payments may partially identify t on outcomes that occur frequently. If the agent's behavior in later windows is driven by an inferred distribution $\hat{\pi}$ over τ, then the designer's commitment problem becomes dynamic: the platform must decide whether to (i) commit to a stationary randomization over τ, making π common knowledge, or (ii) treat t as an internal policy that adapts over time (which may erode credibility and, depending on the environment, increase the agent's incentive to "chase" the inferred evaluator). While we do not model this repeated-game explicitly, the direction is clear: sustaining the disciplining effect of ambiguity across windows requires either credible commitment to the selection rule or periodic "refreshing" of τ so that learning does not collapse the perceived ambiguity.

Multiple metrics as outcomes and the role of ordering. Ordered outcomes $\Omega = \{1, \ldots, m\}$ are a reduced form for many real evaluations that start from multiple metrics (accuracy, toxicity, latency, calibration, etc.). One natural extension is to define each realized outcome as a *multidimensional* metric vector, and then map it into an ordered bin j using a published rubric (e.g., a weighted score, a Pareto ranking with tie-breaks, or a pass/fail ladder). In this interpretation, monotone contracts $t \in M$ are not merely a technical restriction; they capture the institutional desire that "better" rubric outcomes never receive lower payment.

Ambiguity then has a clean practical reading: rather than posting a single fixed set of metric weights, the principal can announce a finite family of acceptable rubrics (or thresholds) and privately commit to one for the window. Different "gaming" actions often shift different metrics, so a single deterministic weighting can be predictable and exploitable, whereas a small set of rubrics can make any particular gaming direction risky. Our step-contract structure under monotonicity corresponds exactly to thresholding on an ordered aggregate score: the payment jumps when performance crosses a particular standard, and ambiguity corresponds to uncertainty about which standard (or which metric emphasis) will be applied.

Connections to hidden test sets and procurement practice. Finally, the model helps formalize two widely used design patterns.

Hidden test sets are naturally modeled as ambiguity: selecting a secret test distribution or a secret subset of items is equivalent to selecting a secret evaluation rule t (more generally, a secret mapping from outcomes to transfers). The agent's max–min response captures the engineering norm of "assume the evaluator will catch the worst-case overfitting." Our consistency requirement parallels a common operational constraint: the platform wants the target behavior to be rewarded similarly regardless of which hidden slice is used, while overfitted or shortcut behaviors should be fragile to that choice.

In procurement and scoring auctions, buyers often evaluate bids using multiple criteria with partially opaque weights, especially when they fear tailoring or "proposal gaming." In our language, the buyer posts τ as an announced set of scoring rules and commits internally to one; the seller chooses an action i (e.g., an effort/quality configuration) anticipating the worst-case score within τ. The comparative static on ambiguity aversion is again policy-relevant: ambiguity is most defensible when the procurement environment plausibly induces worst-case thinking (e.g., audits, protests, or ex post renegotiation risk). When vendors instead optimize for expected scoring weights, the buyer may need either transparency (deterministic t) or a genuinely randomized scoring rule with credible commitment, since "fake ambiguity" will be learned and arbitraged over time.

Practical design guidance: when ambiguity helps and how to deploy it. Our results point to a simple operational rule: ambiguity is valuable precisely when (i) the agent must make a *discrete* commitment for a nontrivial horizon (a pure action within a window), and (ii) the principal can write a small menu of evaluation rules whose *worst-case* effects differ across potential deviations. In such settings, the principal should think less in terms of "adding noise" and more in terms of *covering deviations*. Concretely, starting from a candidate target action i, we can treat the design problem as selecting a support $\tau = \{t^{(1)}, \ldots, t^{(K)}\}$ so that (a) the consistency constraint pins down $T_i(t^{(k)})$ to a common level across k, while (b) for each deviation $i' \neq i$ there exists at least one k with a relatively low $T_{i'}(t^{(k)})$. This "one bad contract per deviation" heuristic is exactly what the small-support theorem formalizes, and it suggests a practical workflow: enumerate plausible gaming directions (candidate i'), then build step/threshold contracts that are selectively punitive for each direction while keeping the target's expected transfer fixed.

Calibrating payments under limited liability and monotonicity. Limited liability ($t_j \geq 0$) and monotonicity ($t \in M$) are not merely mathematical conveniences; they are often compliance constraints. In such cases we recommend designing within a low-dimensional family of step contracts (e.g., one or two thresholds) and using consistency as a calibration condition rather than an afterthought. A pragmatic approach is: choose a threshold location (which outcome bins receive the "bonus") and then scale the bonus so that $T_i(t) = c_i + \varepsilon$ for a small rent $\varepsilon \geq 0$. Once the target action's expected payment is anchored, ambiguity can be introduced by varying which bins constitute the threshold set across $t \in \tau$ (still respecting $t \in M$). This makes the transfer rule interpretable to stakeholders ("pass the bar, get paid"), while still making any particular form of gaming fragile to which bar is used.

Commitment and verifiability of the private selection. Because ambiguity is implemented through a private choice of $t \in \tau$, credibility is central. If the agent suspects ex post manipulation of the selected contract after outcomes are realized, then the relevant model is no longer our committed ambiguity environment. In deployments, we therefore view a verifiable commitment device as part of the mechanism: e.g., a publicly auditable random seed committed before the window, an internal control with documented access logs, or even a cryptographic commitment to t that can be opened after the window. Importantly, commitment need not mean revealing t during the window; it only needs to ensure that the selection was fixed for the window and not outcome-contingent. Without such safeguards, ambiguity risks collapsing into discretionary enforcement, which may be both less effective for incentives and harder to justify procedurally.

Choosing the window length and the action granularity. Although payoffs scale with L, the *disciplining logic* depends on the action being fixed over the window. This yields a design tension: longer windows increase the stakes of any single action choice (and may make worst-case planning more salient), but they also increase the value of adapting, retraining, or "trying a little of everything." If the platform's environment or the agent's pipeline naturally induces frequent per-instance adaptation, then the principal should expect the "mixing kills ambiguity" force to bite, and should shift effort toward deterministic evaluation rules or toward restricting within-window adaptation (e.g., locking a model version, forbidding per-request model selection, or auditing for mixture behavior). Put differently, ambiguous evaluation is a complement to governance that enforces a meaningful notion of *commitment.*

Limits of the model as deployed mechanism design. Several assumptions deserve emphasis. First, we treat $\{q_i\}_{i \in A}$ as known primitives, but in practice they must be estimated, often under distribution shift and strategic response. Errors in q_i can break consistency (the target action may no longer have equal expected transfers across $t \in \tau$) and can inadvertently subsidize deviations. Second, we abstract from outcome manipulation and measurement error: if the agent can influence the mapping from behavior to outcome bin j, then the designer must jointly model the evaluation pipeline and the action. Third, we treat the action set A as finite and coarse; when actions are high-dimensional (continuous model choices), the relevant "deviations" may be best represented by local perturbations, and the finite-support bounds become approximations rather than exact statements.

Open questions. Three directions seem especially important. (i) *Robustness to estimation and misspecification:* can we characterize ambiguous contracts that remain approximately incentive compatible when q_i lies in an uncertainty set, so that both the agent and the principal face ambiguity? (ii) *Dynamics and reputational constraints:* in repeated interaction, how does the principal optimally trade off short-run deterrence against long-run credibility, especially if agents can statistically test which $t \in \tau$ is used? (iii) *Multi-agent and market settings:* when multiple agents compete (leaderboards, procurement) ambiguity may deter gaming but also increase perceived risk, potentially reducing entry or shifting effort toward safer but lower-value actions. Understanding these equilibrium participation effects, and how they interact with fairness or transparency constraints on $t \in M$, is essential for translating the theoretical advantage of ambiguity into responsible policy.