Strategic Testing in Games

author names withheld

Under Review for NExT-Game 2026

Abstract

When chess world champion Magnus Carlsen accused Hans Niemann of cheating in September 2022, it spotlighted a challenge that extends well beyond chess: how can one audit a competitor's play while protecting honest players from false accusations? We propose a principled framework for auditing a monitored player relative to a known strategy in two-player normal-form games. In particular, we introduce (ϵ, δ)-strategic testing which, given a tolerated accusation rate δ, keeps the accepted winning payoff within ϵ of the honest expected winning payoff. We characterize the optimal acceptance policy by a linear program, compare it with a suboptimal policy, and show why purely distributional criteria can miss or overstate strategically relevant deviations. Finally, we formulate the auditing problem as an auxiliary zero-sum Stackelberg game.

1. Introduction

Auditing strategic behavior arises across many domains: cheating in games, unauthorized assistance in standardized testing, and manipulation in financial markets. In each case the monitored agent faces a simple incentive: in a normal-form game, choose the action with the highest winning payoff rather than the prescribed baseline. The principal must ensure that the winning payoff of any accepted action stays close to what honest play would produce, while keeping false accusations under control.

Existing approaches to cheating detection [1, 7] typically measure how unusual a player's action distribution is relative to a behavioral baseline. Beyond move-level signals, deployed systems may combine behavioral biometrics such as mouse dynamics and keystroke patterns [11] with system-level signals and private procedures. Such approaches do not know which actions matter for payoff: they may flag behavior that is statistically unusual but gives no advantage, or accept behavior that is distributionally close to the baseline but improves the player's accepted winning payoff. Statistical testing frameworks [2, 3, 9, 10, 18] face the same limitation, as they focus on worst-case distributional divergence.

Our framework is inspired by the two-parameter structure of PAC learning [19]: δ controls how much baseline action mass the audit removes, while ϵ controls how far above the honest winning benchmark W^+ the accepted winning payoff can reach. We adopt a full-information setting in which the reference strategies μ_0 and ν_0 are treated as known, following work on human-AI alignment in chess [8, 16, 17] and intrinsic skill estimation [13], which show that player action distributions can be concretely estimated from Elo ratings and match data.

Contributions. We formalize (ϵ, δ)-strategic testing as a framework for bounding the accepted winning payoff above an honest benchmark (Section 3) in normal-form games. We characterize the optimal acceptance policy by a linear program (Section 4). We also formulate the testing problem as

an auxiliary zero-sum Stackelberg game [12, 15] in which the auditor is the leader and an adversarial evaluator is the follower, and show that its value equals $\epsilon^*(\delta)$ (Section 5).

2. Preliminaries

For a finite set $\mathcal{A}$, let $\Delta(\mathcal{A})$ denote the probability simplex, $p(a)$ the probability of action a under p, and $\|p-q\|_{\mathrm{TV}} = \frac{1}{2}\sum_a |p(a)-q(a)|$ the total variation distance. A two-player game has finite action sets $\mathcal{A}$ and $\mathcal{B}$ and a payoff function $U : \mathcal{A}\times\mathcal{B} \to \mathbb{R}$ for the monitored player. Throughout, payoffs are normalized so that $U(a,b) \in [-1,1]$. Write $U^+(a,b) := \max\{U(a,b), 0\}$ for the positive part. For mixed strategies $\mu \in \Delta(\mathcal{A})$ and $\nu \in \Delta(\mathcal{B})$, the expected payoff is $U(\mu,\nu) := \mathbb{E}_{a\sim\mu, b\sim\nu}[U(a,b)]$.

3. The (ϵ, δ)-Strategic Testing Framework

We work in a full-information setting: the auditor knows the finite action sets $\mathcal{A}, \mathcal{B}$, the payoff function U, the audited player's strategy $\mu_0 \in \Delta(\mathcal{A})$, and the opponent's strategy $\nu_0 \in \Delta(\mathcal{B})$. As mentioned, both μ_0 and ν_0 are treated as fixed and known, for example through estimation from skill-level models and historical match data. The framework applies to general-sum games; we restrict attention here to payoffs $U(a,b) \in \{-1,+1\}$ for clarity, and return to the general case in Section 4. Both actions a and b are observed, so the tester can evaluate $U(a,b)$ for any pair. Define the honest expected winning-payoff benchmark

$$W^+ := \sum_{\substack{a\in\mathcal{A},\, b\in\mathcal{B} \\ U(a,b)>0}} \mu_0(a)\,\nu_0(b)\,U(a,b) \;\geq\; 0 \tag{1}$$

Consider an auditor overseeing a two-player game, such as the modified Rock-Paper-Scissors (RPS) in Table 2, who wishes to audit Player 1. The auditor uses the fixed reference strategies μ_0, ν_0 and the payoff geometry of the game to construct an acceptance policy that limits the accepted winning payoff above W^+. The flagging budget δ controls how much accepted mass on payoff-relevant wins can be removed.

		Player 2		
		R	P	S
Player 1	R	$0,\ 0$	$-1,\ 1$	$1,-1$
	P	$1,-1$	$0,\ 0$	$-1,\ 1$
	S	$-1,\ 1$	$1,-1$	$0,\ 0$

Table 1: Rock-Paper-Scissors.

		Player 2		
		R	P	S
Player 1	R	$-1,\ 1$	$-1,\ 1$	$1,-1$
	P	$1,-1$	$-1,\ 1$	$-1,\ 1$
	S	$-1,\ 1$	$1,-1$	$-1,\ 1$

Table 2: Modified RPS: ties broken in favour of Player 2.

To build geometric intuition, consider for example the strategies $\mu_0 = (0.5, 0.3, 0.2)$ and $\nu_0 = (0.4, 0.3, 0.3)$. Those can be depicted in a $[0,1]\times[0,1]$ square, where green regions correspond to winning outcomes for Player 1 and red regions to losing outcomes. Since the audit only activates on wins, we restrict our attention to the green regions. Projecting them onto the x-axis and sorting by the product of probability and payoff yields the sorted winning regions in Figure 1, where the area of each bar equals the action's contribution to W^+. The audit policy removes a total mass δ from this sorted distribution.

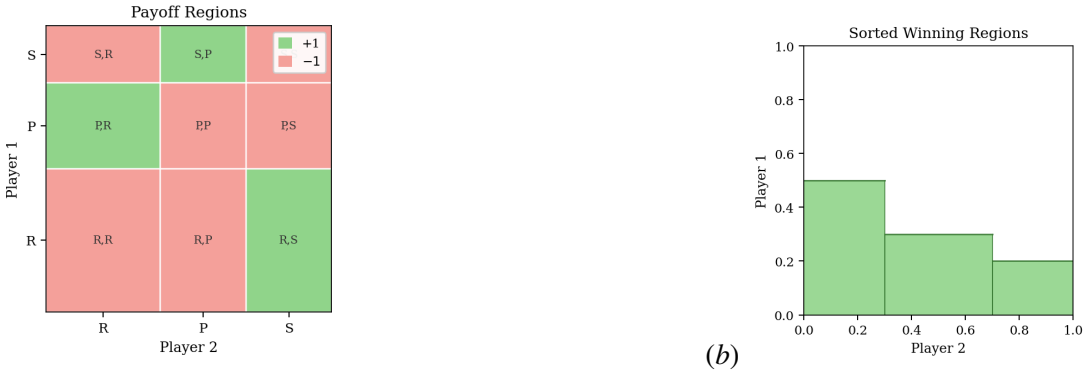

Figure 1: Pictorial view of Modified RPS with payoffs ± 1. *(a)* shows winning and losing regions in the joint probability space of (μ_0, ν_0). *(b)* sorts actions by their contribution to the honest winning payoff W^+.

One can also think of our framework as follows. Consider a two-player general-sum game in which Player 1, with reference strategy μ_0, may have access to information allowing them to estimate the payoff regions depicted above and identify that a particular action, say R, yields a higher winning payoff than the baseline prescribes. Given δ, the auditor's goal is to ensure that the accepted winning payoff of the audited baseline stays within ϵ of W^+ for every opponent action. Under honest play, the probability of being flagged is at most δ; it equals δ exactly only conditionally on a win under the uniform policy (see Lemma 5).

Definition 1 ((ϵ, δ)-Strategic Tester) *Assume $\delta \in (0, 1)$. An acceptance policy μ^δ is an* (ϵ, δ)-strategic tester *for $(\mu_0, \nu_0, \mathcal{A}, \mathcal{B}, U)$ if:*

(i) $0 \le \mu^\delta(a) \le \mu_0(a)$ *for all* $a \in \mathcal{A}$, *and* $\displaystyle\sum_{a \in \mathcal{A}} \mu^\delta(a) = 1 - \delta$

(ii) for all $b \in \mathcal{B}$, $\displaystyle\sum_{a:\, U(a,b) > 0} \mu^\delta(a)\, U(a, b) \le W^+ + \epsilon$

An instance is (ϵ, δ)-strategically testable *if such a policy exists.*

Condition (i) says that μ^δ is the accepted part of the reference action mass: the auditor removes $\mu_0(a) - \mu^\delta(a)$ from each action a, with total removed mass δ. Thus $\mu^\delta(a)/\mu_0(a)$ is the acceptance probability on a positive-payoff observation of action a, and $1 - \mu^\delta(a)/\mu_0(a)$ is the corresponding flagging probability. Condition (ii) requires that, for every opponent action b, the accepted positive payoff mass is at most $W^+ + \epsilon$. Since W^+ is averaged over ν_0 while condition (ii) is column-wise, ϵ captures both the average-to-worst-case gap and the residual payoff exposure left after removing δ units of baseline mass.

To see what the mechanism does in practice, consider the two cases separately. An honest player following μ_0 is always accepted on losing outcomes and is randomly flagged on winning outcomes with action-specific probability $1 - \mu^\delta(a)/\mu_0(a)$. The total mass subject to flagging is δ, though the realized probability of being flagged depends on which winning actions are observed

Algorithm 1: (ϵ, δ)-Strategic Testing for $U(a, b) \neq 0$ games

Input: Reference strategy μ_0, acceptance policy μ^δ, player action a, opponent action b
if $U(a, b) < 0$ **then**
  return ACCEPT;
else
  Sample $Z \sim \text{Bernoulli}(\mu^\delta(a)/\mu_0(a))$;
  return $Z = 1$? ACCEPT : FLAG;
end

and on the opponent distribution. The mechanism therefore targets payoff-relevant wins rather than distributional closeness.

The natural question is then how to select μ^δ given the budget δ.

4. Testing Policies

Having established the framework, we now turn to the computation of the acceptance policy. A natural baseline is to remove mass uniformly from every action in $\text{supp}(\mu_0)$, giving the uniform policy $\mu^{\delta,\text{unif}}(a) := (1 - \delta)\mu_0(a)$. This policy is always feasible but ignores the payoff structure entirely: it removes the same fraction of mass from every action regardless of how much each action contributes to the worst-case column sum.

Proposition 2 (Uniform policy bound) *The uniform policy $\mu^{\delta,\text{unif}}(a) = (1 - \delta)\mu_0(a)$ is a valid (ϵ, δ)-strategic tester with*

$$\epsilon_{\text{unif}}(\delta) = \max\Big\{0,\ (1-\delta)\max_{b\in\mathcal{B}} \sum_{a:\,U(a,b)>0} \mu_0(a)\,U(a,b) - W^+\Big\}$$

Proof in Appendix

The optimal policy instead concentrates the removed mass on actions contributing most to the worst-case column sum, minimizing the residual accepted winning payoff above W^+. This corresponds to the following optimization problem:

$$\epsilon^*(\delta) := \max\Big\{0,\ \min_{\mu^\delta}\max_{b\in\mathcal{B}}\Big(\sum_{a:\,U(a,b)>0} \mu^\delta(a)\,U(a,b) - W^+\Big)\Big\} \tag{2}$$

subject to $\sum_a \mu^\delta(a) = 1 - \delta$ and $0 \leq \mu^\delta(a) \leq \mu_0(a)$. Since the inner objective is a pointwise maximum of linear functions of μ^δ, this corresponds to a linear program, which is solvable in polynomial time [6].

Theorem 3 (LP Characterisation) *The optimal bound $\epsilon^*(\delta)$ and policy $\mu^{\delta,*}$ are obtained by solving*

$$\begin{aligned} \min_{\epsilon,\,\mu^\delta} \quad & \epsilon \\ \text{s.t.} \quad & \sum_{a:\,U(a,b)>0} \mu^\delta(a)\,U(a,b) \le W^+ + \epsilon \quad \forall\, b \in \mathcal{B} \\ & \sum_a \mu^\delta(a) = 1-\delta \\ & 0 \le \mu^\delta(a) \le \mu_0(a) \quad \forall\, a \in \mathcal{A} \\ & \epsilon \ge 0 \end{aligned} \tag{3}$$

Proof in Appendix F.

Figure 2 illustrates the core mechanism: the LP identifies which action contributes most to the worst-case column sum and removes proportionally more mass from it, while the uniform policy spreads the budget evenly.

(*a*) (*b*)

Figure 2: Policy comparison for Modified RPS ($\delta = 0.05$). Yellow bars show the initial strategy μ_0; purple bars show the accepted mass μ^δ. The dashed line marks the honest winning payoff W^+, and the solid line marks $\max_b \sum_a \mu^\delta(a)\,U^+(a,b)$.

We now address our assumption on $U(a,b) \neq 0$. In normal-form games, a strategic player may prefer a zero-payoff action over a losing one, since it avoids a loss without contributing to the winning-payoff sum counted by condition (ii). Under our winning-payoff objective, the audit budget is therefore concentrated on strictly positive-payoff actions, as those are the ones a strategic player has a direct incentive to choose. If avoiding losses is itself strategically meaningful, a separate budget δ_t can be allocated to tie outcomes. Further details are provided in Appendix D.

5. Game-Theoretic Interpretation

In this section we restrict to games with no zero-payoff outcomes: $U(a,b) \neq 0$ for all $(a,b) \in \mathcal{A} \times \mathcal{B}$. The (ϵ,δ)-testing problem can be formulated exactly as an auxiliary zero-sum Stackelberg game. This game is distinct from the original game between Player 1 and Player 2: the original strategies μ_0 and ν_0 are fixed reference strategies, while the Stackelberg game describes the auditor's worst-case optimization problem.

Fix an instance $(\mu_0, \nu_0, \mathcal{A}, \mathcal{B}, U)$ and a budget δ. The leader is the auditor, who commits to an accepted-mass policy $m \in \mathcal{M}_\delta$, where

$$\mathcal{M}_\delta := \left\{ m \in \mathbb{R}_+^{|\mathcal{A}|} : 0 \le m(a) \le \mu_0(a) \; \forall\, a \in \mathcal{A},\; \textstyle\sum_a m(a) = 1 - \delta \right\}$$

The follower is an adversarial evaluator who observes m and chooses a column $b \in \mathcal{B}$. The evaluator's payoff, equivalently the auditor's loss, is

$$L(m, b) := \Big[\sum_{a:\, U(a,b) > 0} m(a)\, U(a, b) - W^+ \Big]_+$$

and the auditor's payoff is $-L(m, b)$. The Stackelberg value is

$$V_\delta := \min_{m \in \mathcal{M}_\delta} \max_{b \in \mathcal{B}} L(m, b)$$

Since the positive part commutes with the outer maximum over b, we have $V_\delta = \epsilon^*(\delta)$ as defined in (2).

A Stackelberg equilibrium of the auditing game is a pair (m^*, b^*) such that

$$m^* \in \arg\min_{m \in \mathcal{M}_\delta} \max_{b \in \mathcal{B}} L(m, b) \qquad b^* \in \arg\max_{b \in \mathcal{B}} L(m^*, b)$$

Any LP-optimal solution $\mu^{\delta,*}$ to (3) is therefore an optimal Stackelberg leader strategy, and any maximising column b^* is a follower best response.

This formulation clarifies the role of the LP: the auditor commits to an accepted-mass policy, and the adversarial evaluator selects the column that maximizes the loss $L(m, b)$. The LP computes the leader policy that minimizes this worst-case exposure. Under honest play, the accepted winning payoff against column b produced by Algorithm 1 is exactly

$$\sum_{a:\, U(a,b) > 0} \mu^\delta(a)\, U(a, b)$$

which is precisely the quantity controlled by the Stackelberg game.

6. Conclusion

We introduced (ϵ, δ)-strategic testing, a framework for auditing a player in normal-form games by controlling the accepted winning payoff relative to an honest benchmark. The framework offers a first principled approach to modeling cheating detection in a structured game-theoretic environment. Two natural extensions are mechanism design for strategic testing, where one designs games that inherently facilitate deviation detection rather than testing within a fixed game, and adaptive testing, where the thresholds (ϵ, δ) are dynamically adjusted based on observed player behavior.

References

[1] David J Barnes and Julio Hernandez-Castro. On the limits of engine analysis for cheating detection in chess. *Computers & Security*, 48:58–73, 2015.

[2] C. L. Canonne. A survey on distribution testing: Your data is big. but is it blue? *Theory of Computing, Graduate Surveys*, 9:1–100, 2020.

[3] Ilias Diakonikolas and Daniel M Kane. *Algorithmic high-dimensional robust statistics*. Cambridge university press, 2023.

[4] F. Fang, T. H. Nguyen, R. Pickles, W. Y. Lam, G. R. Clements, B. An, A. Singh, B. C. Schwedock, M. Tambe, and A. Lemieux. Deploying PAWS: Field optimization of the protection assistant for wildlife security. In *Proceedings of the 28th AAAI Conference on Innovative Applications of Artificial Intelligence (IAAI)*, pages 3966–3973, 2016.

[5] Aditya Jonnalagadda, Iuri Frosio, Seth Schneider, Morgan McGuire, and Joohwan Kim. Robust vision-based cheat detection in competitive gaming. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 4(1):1–18, 2021.

[6] N. Karmarkar. A new polynomial-time algorithm for linear programming. In *Proceedings of the 16th Annual ACM Symposium on Theory of Computing (STOC)*, pages 302–311, 1984.

[7] Thijs Laarhoven and Aditya Ponukumati. Towards transparent cheat detection in online chess: An application of human and computer decision-making preferences. In *International Conference on Computers and Games*, pages 163–180. Springer, 2022.

[8] R. McIlroy-Young, R. Wang, S. Sen, J. Kleinberg, and A. Anderson. Aligning superhuman AI with human behavior: Chess as a model system. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1677–1687, 2020.

[9] J. Neyman and E. S. Pearson. On the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society of London, Series A*, 231:289–337, 1933.

[10] L. Paninski. A coincidence-based test for uniformity given very sparsely sampled discrete data. *IEEE Transactions on Information Theory*, 54(10):4750–4755, 2008.

[11] José Pedro Pinto, André Pimenta, and Paulo Novais. Deep learning and multivariate time series for cheat detection in video games. *Machine Learning*, 110(11):3037–3057, 2021.

[12] J. Pita, M. Jain, J. Marecki, F. Ordóñez, C. Portway, M. Tambe, C. Western, P. Paruchuri, and S. Kraus. Deployed ARMOR protection: The application of a game theoretic model for security at the Los Angeles International Airport. In *Proceedings of the 7th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 125–132, 2008.

[13] K. W. Regan and G. M. Haworth. Intrinsic chess ratings. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence (AAAI)*, pages 834–839, 2011.

[14] E. Shieh, B. An, R. Yang, M. Tambe, C. Baldwin, J. DiRenzo, B. Maule, and G. Meyer. PROTECT: A deployed game theoretic system to protect the ports of the United States. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 13–20, 2012.

[15] M. Tambe. *Security and Game Theory: Algorithms, Deployed Systems, Lessons Learned*. Cambridge University Press, 2011.

[16] Zhenwei Tang, Difan Jiao, Reid McIlroy-Young, Jon Kleinberg, Siddhartha Sen, and Ashton Anderson. Maia-2: A unified model for human-ai alignment in chess. *Advances in Neural Information Processing Systems*, 37:20919–20944, 2024.

[17] Zhenwei Tang, Difan Jiao, Eric Xue, Reid McIlroy-Young, Jon Kleinberg, Siddhartha Sen, and Ashton Anderson. Learning to imitate with less: Efficient individual behavior modeling in chess. *arXiv preprint arXiv:2507.21488*, 2025.

[18] G. Valiant and P. Valiant. An automatic inequality prover and instance optimal identity testing. In *Proceedings of the 57th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 51–62, 2017.

[19] L. G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

Appendix A. Notation Reference

Symbol	Meaning
$\mathcal{A}, \mathcal{B}$	Action sets for the monitored player and the opponent
$\Delta(\mathcal{A})$	Probability simplex over $\mathcal{A}$
$\mu_0 \in \Delta(\mathcal{A})$	Pl.1 reference strategy
$\nu_0 \in \Delta(\mathcal{B})$	Pl.2 reference strategy
$U : \mathcal{A} \times \mathcal{B} \to [-1, 1]$	Payoff function for the monitored player
$U^+(a, b)$	$\max\{U(a, b), 0\}$, positive part of the payoff
W^+	Honest expected winning-payoff benchmark
$\mu^\delta : \mathcal{A} \to [0, 1]$	Acceptance policy
δ	Flagging mass budget (action-mass units)
ϵ	Winning-payoff-gain tolerance
$\epsilon^*(\delta)$	Optimal testability bound
$\mathcal{M}_\delta$	Feasible set for acceptance policies with budget δ

Table 3: Summary of notation.

Appendix B. Related Work

Cheating detection in games. Work on chess-specific cheating detection has studied the limits of engine-only analysis and proposed move-preference approaches as more transparent alternatives [1, 7]. In video games, cheating has been tackled through behavioral baselines and machine learning

classifiers trained on game logs [11], as well as vision-based methods targeting hardware-level exploits such as aimbots [5].

Stackelberg security games. Stackelberg security games [15] model a defender who commits to a randomized inspection policy against a strategic adversary, and have been deployed in airport security [12], coast guard patrols [14], and wildlife protection [4]. Our LP shares the leader-commitment structure but the leader commits to an accepted-mass policy constrained by the reference baseline, and the objective is a payoff-advantage bound rather than a target-coverage criterion.

Appendix C. Supporting Results for Section 3

Proposition 4 (Monotonicity) *$\epsilon^*(\delta)$ is nonincreasing in δ on $[0, 1]$.*

Proof Fix $\delta_1 \le \delta_2$, set $r = \delta_2 - \delta_1$, and let μ^{δ_1} be any feasible policy. Since $r \le \sum_a \mu^{\delta_1}(a)$, choose $\eta \ge 0$ with $\eta(a) \le \mu^{\delta_1}(a)$ and $\sum_a \eta(a) = r$, and define $\mu^{\delta_2}(a) := \mu^{\delta_1}(a) - \eta(a)$. This policy is feasible for budget δ_2 and satisfies $\mu^{\delta_2} \le \mu^{\delta_1}$ coordinatewise, so

$$\sum_{a:\, U(a,b)>0} \mu^{\delta_2}(a)\, U(a,b) \le \sum_{a:\, U(a,b)>0} \mu^{\delta_1}(a)\, U(a,b) \quad \forall\, b \in \mathcal{B}$$

giving $\epsilon^*(\delta_2) \le \epsilon^*(\delta_1)$. ■

Lemma 5 (Testing Procedure Validity) *Let $\mu^\delta \in \mathcal{M}_\delta$. Under Algorithm 1, with the opponent fixed at ν_0,*

$$\Pr[\text{flag}] = \sum_{a,b:\, U(a,b)>0} \nu_0(b)\big(\mu_0(a) - \mu^\delta(a)\big) \le \delta$$

If the honest win event has positive probability,

$$\Pr[\text{flag} \mid \text{win}] = \frac{\sum_{a,b:\, U(a,b)>0} \nu_0(b)(\mu_0(a) - \mu^\delta(a))}{\sum_{a,b:\, U(a,b)>0} \nu_0(b)\mu_0(a)}$$

Proof On each winning observation (a, b), Algorithm 1 flags with probability $1 - \mu^\delta(a)/\mu_0(a)$. Averaging over honest play (μ_0, ν_0) gives $\Pr[\text{flag}]$. Since

$$\sum_{a,b:\, U(a,b)>0} \nu_0(b)\big(\mu_0(a) - \mu^\delta(a)\big) \le \sum_a \big(\mu_0(a) - \mu^\delta(a)\big) = \delta$$

we have $\Pr[\text{flag}] \le \delta$. Conditioning on the win event gives the conditional rate. Under the uniform policy, $1 - \mu^{\delta,\text{unif}}(a)/\mu_0(a) = \delta$ for every a, so the conditional rate equals δ. ■

Appendix D. Tie-Outcome Extension

Algorithm 2 gives an outcome-dependent extension of the testing rule. The budget δ is split into a winning budget δ_w and a tie budget δ_t, with $\delta_w + \delta_t = \delta$. The LP-optimal accepted-mass policy is applied to positive-payoff outcomes, while a uniform policy is applied to zero-payoff outcomes. The auditor may allocate the budget across the two outcome types as the game warrants.

Algorithm 2: Generalized (ϵ, δ)-Strategic Testing

Input: Reference μ_0, budgets $\delta_w, \delta_t \geq 0$ with $\delta_w + \delta_t = \delta$, winning policy $\mu^{\delta_w,*}$, tie policy $\mu^{\delta_t,\text{unif}}$, action a, opponent action b

if $U(a, b) < 0$ **then**
  return ACCEPT;
else if $U(a, b) = 0$ **then**
  Sample $Z \sim \text{Bernoulli}(\mu^{\delta_t,\text{unif}}(a)/\mu_0(a))$;
  return $Z = 1$? ACCEPT : FLAG;
else
  Sample $Z \sim \text{Bernoulli}(\mu^{\delta_w,*}(a)/\mu_0(a))$;
  return $Z = 1$? ACCEPT : FLAG;
end

Appendix E. Limitations of Distributional Criteria

Statistical distance controls how much probability mass has moved, not whether the moved mass lies on payoff-relevant actions. We make this precise.

Proposition 6 (TV bound is tight but payoff-geometry-blind) *Assume $U(a, b) \in [-1, 1]$. If $\|\mu - \mu_0\|_{\text{TV}} \leq \tau$, then for every $b \in \mathcal{B}$,*

$$\sum_a \mu(a)\, U^+(a, b) \leq \sum_a \mu_0(a)\, U^+(a, b) + \tau$$

This bound is tight but treats all actions uniformly regardless of their payoff contribution.

Proof Let $f_b(a) = U^+(a, b) \in [0, 1]$. Then

$$\sum_a (\mu(a) - \mu_0(a)) f_b(a) \leq \sum_{a:\, \mu(a) \geq \mu_0(a)} (\mu(a) - \mu_0(a)) = \|\mu - \mu_0\|_{\text{TV}} \leq \tau$$

Tightness follows, for any $\tau \in [0, 1]$, by taking $\mathcal{A} = \{a_1, a_2\}$, a single opponent action b $U(a_1, b) = 1, U(a_2, b) = -1$, $\mu_0 = (0, 1)$, and $\mu = (\tau, 1 - \tau)$ ■

Proposition 7 (Distributional deviation need not imply payoff exposure) *For every $\rho \in (0, 1]$, there exist a game $(\mathcal{A}, \mathcal{B}, U)$, reference strategies μ_0, ν_0, and a strategy μ such that $\|\mu - \mu_0\|_{\text{TV}} = \rho$, yet*

$$\sum_{a:\, U(a,b) > 0} \mu(a)\, U(a, b) \leq W^+ \qquad \textit{for every } b \in \mathcal{B}$$

Proof Let $\mathcal{A} = \{a_1, a_2, a_3\}$, $\mathcal{B} = \{b_1, b_2\}$, $\nu_0(b_1) = \nu_0(b_2) = \frac{1}{2}$, and payoffs

	b_1	b_2
a_1	1	-1
a_2	-1	1
a_3	-1	-1

Set $\mu_0 = (\frac{1}{2}, \frac{1}{2}, 0)$ and $\mu = (\frac{1-\rho}{2}, \frac{1-\rho}{2}, \rho)$. Then

$$\|\mu - \mu_0\|_{\mathrm{TV}} = \tfrac{1}{2}\big(\tfrac{\rho}{2} + \tfrac{\rho}{2} + \rho\big) = \rho \qquad W^+ = \tfrac{1}{2}$$

Against both columns,

$$\sum_{a:\, U(a,b)>0} \mu(a)\, U(a,b) = \tfrac{1-\rho}{2} \le W^+$$

since the shifted mass moves entirely to a_3, which loses against both columns. ■

Together, the two propositions show that TV distance gives only a generic payoff-scale bound, tight in the worst case, but unable to detect whether shifted mass lands on payoff-relevant actions. Our framework instead targets this directly, controlling the accepted winning payoff of the audited baseline using the payoff geometry that distributional criteria discard.

Appendix F. Proofs for Section 4

Proof [Proof of Proposition 2] The policy $\mu^{\delta,\mathrm{unif}}(a) = (1-\delta)\mu_0(a)$ satisfies $0 \le \mu^{\delta,\mathrm{unif}}(a) \le \mu_0(a)$ and $\sum_a \mu^{\delta,\mathrm{unif}}(a) = 1 - \delta$, so it is feasible. For every $b \in \mathcal{B}$,

$$\sum_{a:\, U(a,b)>0} \mu^{\delta,\mathrm{unif}}(a)\, U(a,b) = (1-\delta) \sum_{a:\, U(a,b)>0} \mu_0(a)\, U(a,b)$$

which gives $\epsilon_{\mathrm{unif}}(\delta)$ as stated. ■

Proof [Proof of Theorem 3] The uniform policy $(1-\delta)\mu_0 \in \mathcal{M}_\delta$, so the feasible set is nonempty. For any $\mu^\delta \in \mathcal{M}_\delta$, the smallest nonnegative tolerance is

$$\Big[\max_{b\in\mathcal{B}}\Big(\sum_{a:\, U(a,b)>0} \mu^\delta(a)\, U(a,b) - W^+\Big)\Big]_+$$

minimising over $\mathcal{M}_\delta$ gives $\epsilon^*(\delta)$ as in (2). The LP (3) is the epigraph formulation of this problem: enforcing $\sum_{a:\, U(a,b)>0} \mu^\delta(a)\, U(a,b) \le W^+ + \epsilon$ for all $b \in \mathcal{B}$ and minimising $\epsilon \ge 0$ recovers $\epsilon^*(\delta)$. The program has $|\mathcal{A}| + 1$ variables and $O(|\mathcal{A}| + |\mathcal{B}|)$ constraints. ■