VOUCH: Counterfactual Contracts for Multi-Agent Coordination

Abhay Bhandarkar[*1] **Vedanth Nanesha**[*2]
[1]Ramaiah Institute of Technology, Bengaluru, India
[2]National Institute of Technology Karnataka, Surathkal, India
abhaybhandarkar@gmail.com vedanthnanesha04@gmail.com

Abstract

Multi-agent LLM coordination typically reduces to voting or arbitration over self-reported labels and reasoning that is uncoupled from agent behavior and fails predictably under structured bias. A shortcut-reliant agent is confident about its shortcut, and its predictions survive any rule that consults only self-reports. We propose VOUCH, a coordination protocol in which every agent claim is coupled to a deterministic test the protocol runs itself. An agent commits to a counterfactual prediction about an intervention's outcome; the protocol applies the intervention and verifies the contract only if the actual outcome matches. The natural single-side instantiation is unsound as a self-coherent agent satisfies its own commitment by construction. We resolve this with a dual-side construction where the proposer additionally commits to a second agent's post-intervention prediction, a cross-side claim that self-coherence cannot fulfill. Across multimodal classification (**Hateful Memes**, **CrisisMMD**, **MM-IMDb**), code (**MBPP**, **HumanEval**), and math (**GSM8K**, **MATH**, **SVAMP**), the dual-side fix reduces wrong-verified events by 46%–85% per domain while never falling more than 1 point below the Ensemble baseline on accuracy. VOUCH is therefore designed not to outperform ensembling, but to preserve task accuracy while making coordination decisions testable and auditable.[1]

1 Introduction

Multi-agent systems built on large language models are now a standard production pattern, often combining a text specialist, an image specialist, and a multimodal generalist whose outputs are aggregated into a single decision. This paper studies the coordination question sitting one level above representation fusion. ***What should a protocol do when several agents disagree on the label for a single input?***

Standard rules like majority voting, confidence-weighted aggregation, and centralized arbitration share one property. Every claim an agent makes during aggregation is a self-report, and no claim is independently testable by the protocol itself. We call such protocols *cheap talk*, since the language used to coordinate is uncoupled from the behavior being coordinated, and the agent's description of its own reasoning is generated by the very agent we wanted to audit.

This distinction matters even when the final prediction is unchanged. Downstream systems may use an agent endorsement to route a case, trigger or suppress escalation, pass an auto-merge gate, or authorize another automated action; a wrong-verified contract can therefore propagate an unsupported decision despite unchanged aggregate accuracy. Our objective is to preserve the task performance of equal-weight ensembling while replacing unverifiable endorsements with an auditable record of claims and protocol-executed tests.

We introduce *verification-grounded coordination*, in which every agent claim is coupled to a deterministic test the protocol can run on its own. The natural single-agent instantiation has a clean failure mode that we prove formally (Proposition 1) and observe empirically. Consider an agent that classifies a meme as hateful whenever a particular slur is present. It can correctly predict that removing the slur will flip its own answer, so the protocol applies the intervention, the prediction flips, and the contract verifies. The agent has been rewarded for being internally consistent with its bias, not for being correct. Single-side is a protocol-level form of the self-verification pattern that underlies much of the multi-agent and self-evaluation literature (Section 2), in which the evaluator lies in the same self-coherent equivalence class as the producer. Direct empirical evidence is in Appendices C.13 and C.14.

[*]Equal contribution.
[1]Our code is available at https://github.com/vedanthnanesha/VOUCH.

Findings of the Association for Computational Linguistics: IJCNLP-AACL 2026, pages 4345–4368
November 6-10, 2026 ©2026 Association for Computational Linguistics

We resolve it with a *dual-side* construction. The proposer commits to both its own and a second agent's post-intervention prediction before the intervention is applied, and the contract verifies only when both commitments hold. The cross-side commitment is the structural fix that self-coherence alone cannot satisfy, since the verification event tests a different agent's behavior.

Our contributions are as follows:

- A taxonomy separating cheap-talk from verification-grounded coordination, with self- and cross-form definitions of counterfactual consistency.
- A proposition showing single-side verification is unsound under self-coherent bias, and a dual-side protocol that structurally resolves it.
- An informativeness diagnostic ranking interventions by how often they expose mistakes, and three preconditions for well-posed verification-grounded coordination.
- Empirical evidence across eight benchmarks in three domains (multimodal classification, code, math) and a cross-family ensemble.
- Calibration results showing that the contract-derived counterfactual-consistency (CC) trust signal is substantially better calibrated than self-reported confidence.

2 Related Work

Multi-agent debate among large language models. A growing body of work coordinates multiple LLM agents on a shared task, with agents refining answers over rounds (Du et al., 2024; Liang et al., 2024), evaluated by separate critics (Chan et al., 2024), aggregating diverse samples (Wang et al., 2023), trained to debate (Irving et al., 2018; Arnesen et al., 2024), or organized into role pipelines (Wu et al., 2024; Wang et al., 2025). In all cases the coordination substrate is natural language and convergence is driven by social pressure or articulateness rather than by any test against the world. Many of these methods (self-consistency (Wang et al., 2023), self-refinement, same-family LLM-as-judge) also draw the evaluator from the same model family as the producer, which Proposition 1 shows is enough to make any self-verification step trivially satisfiable under structured bias. Our protocol changes the substrate, coupling every coordination claim to a cross-agent deterministic test so the self-coherence shortcut is unavailable.

Multimodal shortcut learning. Vision and language models trained on standard benchmarks pick up artifacts that correlate with labels but are not semantically meaningful. Our adversarial regime activates such shortcuts at evaluation time (Anil et al., 2024), yielding agents whose confident self-reports are exactly what confidence-weighted aggregation cannot detect, a classical Goodhart pattern (Goodhart, 1984).

Verification in machine learning. Formal verification, conformal prediction, and calibration analysis provide per-prediction guarantees. Our protocol operates at the coordination layer above these methods, combining per-agent verification outcomes into a single decision.

Mechanism design. Classical mechanism design constructs incentive-compatible protocols in which misreporting lowers expected payoff (Vickrey, 1961). Our protocol takes the operational role of such a contract. The agent commits to a counterfactual outcome, the protocol checks it, and misreporters are downweighted.

3 Counterfactual Consistency and the VOUCH Protocol

Let $\mathcal{X}$ be the input space, $\mathcal{Y}$ the label space, and $A = \{a_1, \ldots, a_m\}$ a set of agents with $f_a(x) = (\hat{y}_a, c_a)$ giving each agent's prediction and confidence on input x. An intervention $\iota\colon \mathcal{X} \to \mathcal{X}$ transforms an input in the sense of Pearl (2009), drawn from a finite menu $\mathcal{I} = \{\iota_1, \ldots, \iota_K\}$ made available to the agents.

3.1 Counterfactual consistency, self-form

Definition 1: Self-form counterfactual consistency

The self-form counterfactual consistency of agent a under intervention ι is

$$\mathrm{CC}_{\mathrm{self}}(a; \iota) \;=\; \Pr\left[\, f_a(\iota(x)) = \hat{y}'_a(\iota(x)) \,\right],$$

where $\iota\colon \mathcal{X} \to \mathcal{X}$ is a deterministic intervention, $f_a(\iota(x))$ is agent a's actual prediction on the intervened input, $\hat{y}'_a(\iota(x))$ is the prediction a committed to *before* the intervention was applied, and the probability is over x drawn from the deployment distribution.

The protocol's verification step is a Bernoulli sample from this distribution, and the contract is

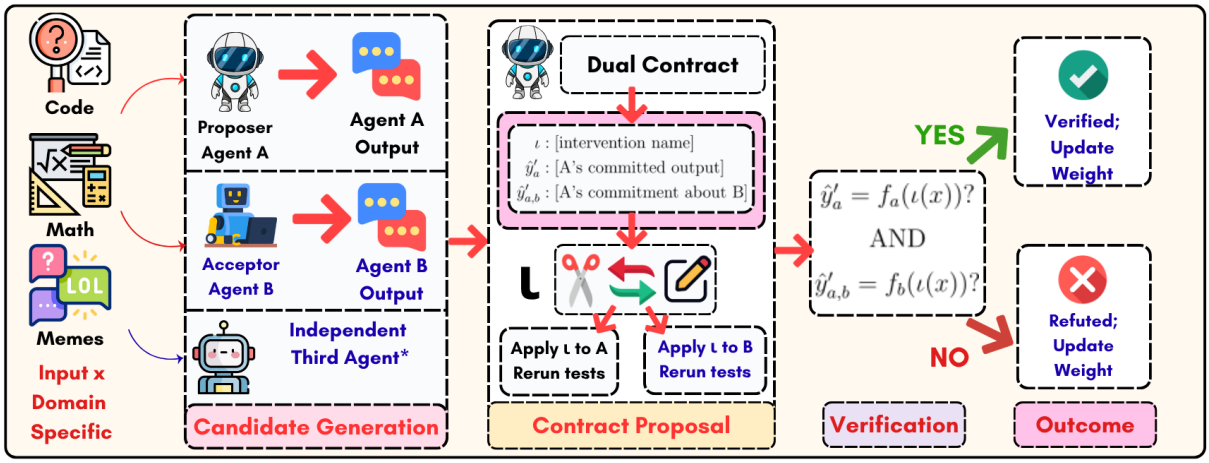


Figure 1: The dual-side VOUCH protocol. **Candidate Generation**: agents A, B, and optional c each predict. **Contract Proposal**: on disagreement, the higher-$w_i c_i$ agent becomes proposer A and writes an intervention ι affecting B's modality, a self-counterfactual $\hat{y}'_a$, and a cross-counterfactual $\hat{y}'_{a,b}$ about B. **Verification**: both checks run in parallel. **Outcome**: both-pass $\Rightarrow$ *verified*, A's weight $\times m_{++}$; else *refuted*, A's weight $\times$ one of m_{-+}, m_{+-}, m_{--} per Section 3.5. The third agent c (3-agent multimodal setups only) does *not* participate in the contract, verification, or weight update; its vote enters only at the final weighted-vote step (not shown).

verified iff the actual prediction on the intervened input matches the committed one.

3.2 The self-form pathology

We observed empirically and now argue that the self-form is unsound as a coordination signal under structured bias.

Proposition 1 (Self-coherence implies self-verification)**.** ***Setup.***

- *Let $r_a : \mathcal{X} \to \mathcal{Y}$ be a fixed deterministic rule that agent a uses to map inputs to labels.*
- *Let $\iota : \mathcal{X} \to \mathcal{X}$ be an intervention that the agent can fully observe.*
- *Let $f_a = r_a$, that is, the agent's actual prediction equals r_a applied to the input.*
- *Let $\hat{y}'_a(\iota(x)) = r_a(\iota(x))$, that is, the agent's counterfactual prediction is computed by applying the same rule to the intervened input.*

Claim. *Under the above setup, $\mathrm{CC}_{\mathrm{self}}(a; \iota) = 1$ for every input x, independent of the ground-truth label of x.*

Proof. By construction, the committed prediction is $\hat{y}'_a(\iota(x)) = r_a(\iota(x))$. The actual prediction is $f_a(\iota(x)) = r_a(\iota(x))$. These quantities are identical, so the verification event holds for every x. The result follows. □

This is the structural reason that the single-side baseline produced 42 wrong-proposer-plus-verified events on Hateful Memes adversarial at n=400. The biased agents were internally consistent and their self-form CC was one.

3.3 Counterfactual consistency, cross-form

Definition 2: Cross-form counterfactual consistency

The cross-form counterfactual consistency of proposer agent a as a predictor of acceptor agent b under intervention ι is

$$\mathrm{CC}_{\mathrm{cross}}(a \to b; \iota) = \Pr\left[f_b(\iota(x)) = \hat{y}'_{a,b}(\iota(x)) \right]$$

where $f_b(\iota(x))$ is b's actual prediction on the intervened input, $\hat{y}'_{a,b}(\iota(x))$ is the prediction that a committed *about* b's post-intervention output (a claim by a about b, not about a), and the probability is over x drawn from the deployment distribution.

The cross-form differs from the self-form in a critical way. The proposer's commitment is a claim about a *different* agent's future behavior, so making it accurately requires a correct theory of why the acceptor disagrees. An agent cannot satisfy it by being self-coherent with its own rule, since the verification event tests another agent's rule.

3.4 The dual-side protocol

The dual-side instantiation of VOUCH proceeds in four steps, shown end-to-end in Figure 1. Given a pair of agents (a, b) that disagree on input x, the agent with the larger current weight-confidence score $w_i c_i$ becomes the proposer a and the other the acceptor b.

1. The proposer a writes a structured proposal containing an intervention ι chosen from a menu restricted to interventions that affect the acceptor's modality, a committed label $\hat{y}'_a$ for what a predicts its own model will produce on $\iota(x)$, and a committed label $\hat{y}'_{a,b}$ for what a predicts b's model will produce on $\iota(x)$.
2. The acceptor b accepts the proposal by default, rejecting only if the proposed intervention is structurally untestable on its modality.
3. If the acceptor accepts, the protocol applies ι to x and computes $f_a(\iota(x))$ and $f_b(\iota(x))$. The proposer-side check tests whether $f_a(\iota(x)) = \hat{y}'_a$, and the acceptor-side check tests whether $f_b(\iota(x)) = \hat{y}'_{a,b}$.
4. The contract is recorded as verified if both checks pass and failed if either check fails, with the proposer's weight updated according to which checks passed.

When a third agent c is present, its vote enters only at the final aggregation.

3.5 Weight update rule

The dual-side verification produces four outcomes, indexed by whether the proposer-side (S) and acceptor-side (C) checks pass. With multipliers $(m_{++}, m_{-+}, m_{+-}, m_{--}) = (1.20, 0.90, 0.95, 0.50)$ applied to the proposer's weight: **both pass** ($S{=}C{=}1$, $\times 1.20$) flips the acceptor's label to the proposer's with confidence $0.90c_a$; **acceptor only** ($S{=}0, C{=}1$, $\times 0.90$) also flips, with confidence $0.75c_a$, since the proposer's theory of the acceptor held even when its self-check did not; **proposer only** ($S{=}1, C{=}0$, $\times 0.95$) and **neither** ($\times 0.50$) leave the acceptor's label unchanged. The magnitudes are hyperparameters chosen only to satisfy Lemma 1's ordering ($m_{++} > m_{+-}$, $m_{-+} > m_{--}$; Appendix A.2), and the schedule ablation in Appendix C.4 confirms that accuracy and contract quality depend on the ordering, not on the specific magnitudes. Only the proposer's weight is updated per contract. This asymmetry simplifies credit assignment and places the burden of proof on the agent challenging the current decision. The weights translate verified contract outcomes into voting influence, providing an auditable trust signal rather than a direct source of accuracy gains (Appendix C.4).

The complete contract-resolution and final-aggregation procedure is given in Algorithm 1 in Appendix A.8.

3.6 Intervention informativeness

Definition 3: Intervention informativeness

The informativeness of an intervention ι is

$$I(\iota) = \Pr\big[\text{verification fails} \,\big|\, \text{proposer's initial label wrong, } \iota\big].$$

A high $I(\iota)$ marks a **diagnostic** intervention that exposes wrong proposers by producing contracts that fail to verify. A low value marks an **uninformative** intervention that verifies regardless of correctness. We use the ranking to pre-select the menu for the VOUCH-Adaptive condition and report the per-intervention values in Appendix C.1.

3.7 Preconditions

Verification-grounded coordination is not always useful. We identify three preconditions under which the protocol is expected to be a net positive for accuracy.

Preconditions C1, C2, C3

C1, Intervention separability. There exists at least one intervention $\iota \in \mathcal{I}$ with $I(\iota)$ meaningfully greater than zero for the deployment's failure mode.
C2, Exploitable bias. At least one agent in the system has a systematic shortcut that the protocol can use as a source of disagreement.
C3, Reliability threshold. Base-agent accuracy exceeds the regime in which disagreement no longer implies error.

In practice, a useful intervention should satisfy three criteria: it acts on the input modality visible to the receiving agent, can plausibly change that agent's prediction, and induces changes whose direction is informative about label correctness.

Each precondition can in principle be measured empirically on a held-out subset. The per-intervention $\hat{I}$ values in Appendix C and the per-agent cross-CC

values in Section 5.4 provide empirical proxies for C1 and C2 respectively. These practical preconditions do not by themselves imply positive long-run weight drift.

3.8 Per-step ordering and conditional convergence

Theorem 1: Per-step ordering and conditional long-run drift

For a fixed proposer-side marginal, increasing the conditional cross-side pass probabilities weakly increases the expected one-step change in log weight. Separately, if the expected log multiplier is non-negative at every step, log weight is a submartingale; divergence to $+\infty$ additionally requires uniformly positive drift.

The first statement is an ordering result and does not imply positive drift. The full conditions for the separate long-run result are given in Appendix A.4. The self-form version of the ordering result is false in general, because self-form CC equals one under self-coherent rules and therefore carries no signal about correctness.

4 Method

4.1 Agents

The same-family setup uses Qwen2.5-7B-Instruct (Qwen et al., 2024) as the text agent and Qwen2.5-VL-7B-Instruct (Bai et al., 2025) as both the image-only agent (image plus generic prompt) and the third independent agent c (image plus caption). The cross-family setup swaps the text agent for Llama-3.1-8B-Instruct (Grattafiori et al., 2024). The text and image agents play proposer / acceptor; c is an independent third voter, never in the contract. All agents run at full precision in subprocess workers to isolate CUDA state.

4.2 Intervention menu

The menu contains eight interventions, four text-affecting (ablate text, neutralize text, reverse sentiment, swap caption) and four image-affecting (blur faces, grayscale, ablate image, swap image). Two filters apply at proposal time, excluding interventions that ablate the proposer's modality (tautological flip) and interventions that do not affect the acceptor's modality (no falsifiable claim possible).

4.3 Conditions

We compare seven conditions. **Independent** reports the modal label of three single-agent predictions, and **Ensemble** reports the modal label at equal weight, the standard cheap-talk baseline (Bansal et al., 2021). **Centralized-Single** uses only the multimodal agent, and **Centralized-Coordinator** uses a separate text agent to choose the final label after reading all three agents' outputs. **VOUCH** is the canonical protocol with free intervention choice, **VOUCH-Random** restricts the menu to a single uniformly random intervention, and **VOUCH-Adaptive** restricts it to the top three interventions by offline-ranked $I(\iota)$.

4.4 Regimes

Each dataset is evaluated in two regimes. The clean regime uses the dataset's natural prompts. The adversarial regime prepends a system instruction biasing the text agent toward a shortcut-reliant rule.

4.5 Datasets

The three multimodal classification benchmarks are **Hateful Memes** (Kiela et al., 2020) (binary), **CrisisMMD** (Alam et al., 2018) (four-class humanitarian relevance), and **MM-IMDb** (Arevalo et al., 2017) (multi-label genre). As scope checks with executable or exact-match verification rather than label aggregation, we also run **MBPP** (Austin et al., 2021) and **HumanEval** (Chen et al., 2021) for code generation and **GSM8K** (Cobbe et al., 2021), **MATH** (Hendrycks et al., 2021), **SVAMP** (Patel et al., 2021) for math reasoning.

5 Results

Table 1 reports test-set accuracy across all conditions. Table 2 reports the corresponding contract-quality breakdown for the VOUCH variants. Cross-family rows are shaded in Table 1; Appendix C.18 provides the focused same- versus cross-family decomposition.

5.1 Main accuracy table

Canonical VOUCH is never more than 1.0 pp below Ensemble on any cell, matching the cheap-talk baseline on accuracy. We treat this observed parity as a constraint on the protocol rather than its primary result: adding verifiable evidence should not sacrifice the task performance of equal-weight ensembling. On the one cross-family cell with a measurable gap in VOUCH's favor, **MM-IMDb** adversarial at n=500, the margin is +2.0 points,

Dataset	Regime	Backbone	Ind.	Ens.	C-Single	C-Coord	VOUCH variants		
							Free	Random	Adaptive
Hateful Memes	adv	same-family	0.505	0.623	**0.658**	0.568	0.620	0.623	0.623
Hateful Memes	adv	cross-family	0.502	0.608	**0.623**	0.595	0.608	0.608	0.608
Hateful Memes	clean	same-family	0.610	0.623	0.623	0.625	0.620	0.623	**0.625**
Hateful Memes	clean	cross-family	0.583	0.645	0.623	0.623	0.643	**0.650**	0.648
CrisisMMD	adv	same-family	0.667	0.797	0.747	0.690	0.793	0.793	**0.800**
CrisisMMD	adv	cross-family	0.687	**0.810**	0.750	0.707	0.807	0.807	0.807
CrisisMMD	clean	same-family	0.777	0.800	0.747	0.777	0.790	**0.810**	0.797
CrisisMMD	clean	cross-family	0.690	**0.777**	0.750	0.723	0.773	**0.777**	0.773
MM-IMDb	adv	same-family	0.130	0.150	**0.188**	0.150	0.140	0.145	0.132
MM-IMDb	adv	cross-family	0.078	0.110	**0.182**	0.152	**0.130**	0.120	**0.130**
MM-IMDb	clean	same-family	0.182	**0.208**	0.188	0.132	0.202	0.198	0.202
MM-IMDb	clean	cross-family	0.132	0.150	**0.182**	0.154	0.158	0.154	0.158

Table 1: Classification accuracy across three multimodal benchmarks, two regimes, and two backbone compositions. Ind., Ens., C-Single, C-Coord, VOUCH, Random, Adaptive denote Independent, Ensemble, Centralized-Single, Centralized-Coordinator, canonical VOUCH, VOUCH-Random, and VOUCH-Adaptive. All three VOUCH variants use the dual-side protocol (Section 3.4) and differ only in intervention choice. n=400 (HM, MM-IMDb), n=300 (CrisisMMD), cross-family MM-IMDb uses n=500. Same-family uses three Qwen2.5 agents; cross-family (shaded) swaps the text agent for Llama-3.1-8B-Instruct. VOUCH is never more than 1.0 pp below Ensemble on any cell and beats it by +2.0 **pp** on MM-IMDb adversarial cross-family (0.130 vs 0.110). MM-IMDb adversarial uses subset accuracy, macro-F1 is in Appendix C.5.

consistent with the prediction that a heterogeneous backbone admits a richer cross-form CC signal.

Pooled statistical test. Pooling discordant pairs across the five same-family multimodal-classification cells of Table 1, an exact two-sided McNemar test on b=11, c=22 (n=33) gives p=0.080. The difference is not significant at conventional thresholds, although the discordant-pair imbalance is directionally in Ensemble's favor (22 versus 11). We therefore make no claim of accuracy superiority: the classification result is observed non-degradation, while the primary contribution lies in contract quality (Table 2) and in $\widehat{\mathrm{CC}}$ calibration (Section 5.4, per-cell tests in Appendix C.7).

5.2 Contract quality across configurations

Table 2 reports how often the protocol's verifications agreed with the ground truth. A helpful event is a correct proposer whose contract verified or a wrong proposer whose contract failed; a harmful event is the converse. The net column subtracts harmful from helpful, so positive values mean the protocol moved the system toward the ground truth on net.

The single-side row is the empirical instantiation of the self-coherence pathology of Proposition 1, with a harmful-verification rate of $42/127 = 33.1\%$ (Wilson 95% CI $[25.5, 41.6]\%$). Under the dual-side construction this falls to $1/11 = 9.1\%$ (Wilson 95% CI $[1.6, 37.7]\%$), a 73% relative reduction. Every subsequent dual-side row reports at most three wrong-verified events. Net-helpful is negative on the two Hateful Memes cross-family configurations (-6 and -7) and positive on the three CrisisMMD configurations ($+19$ to $+21$).

5.3 Code-generation: the dual-side fix

We evaluate on **MBPP** (Austin et al., 2021) and **HumanEval** (Chen et al., 2021). The verification operator is exact per-test pass-vector match. The proposer commits to the pass vector its candidate will produce after a line-removal intervention, the protocol applies the edit and runs the tests, and the contract is verified only if the actual pass vector matches (Liu et al., 2023). Under the dual-side construction the proposer additionally commits to the acceptor's pass vector under the same intervention. The single-side baseline uses Qwen2.5-7B-Instruct and self-verifies, and dual-side adds Llama-3.1-8B-Instruct as the acceptor.

Throughout the code and math analyses, hurt:caught is computed among contracts opened by initially wrong proposers, with correctness determined against gold labels. It complements rather than replaces standard task accuracy.

We report two complementary views. Table 3 gives a population-level diagnostic on the full samples (n=500 MBPP, n=164 HumanEval). Verification

Dataset	Protocol	Regime	Backbone	Total	C+V	C+F	W+V	W+F	R	Net	Verified
Hateful Memes	single-side	adv	same-family	127	27	31	**42**	23	4	−23	69
Hateful Memes	dual-side	adv	same-family	11	0	4	**1**	6	0	+1	1
Hateful Memes	dual-side	adv	cross-family	6	0	6	0	0	0	−6	0
Hateful Memes	dual-side	clean	cross-family	7	0	7	0	0	0	−7	0
CrisisMMD	dual-side	adv	same-family	60	6	16	3	33	2	+20	9
CrisisMMD	dual-side	adv	cross-family	61	0	19	**2**	40	0	+19	2
CrisisMMD	dual-side	clean	cross-family	37	0	6	**2**	29	0	+21	2

Table 2: Contract quality breakdown. Columns report total opened contracts; the four evaluated outcomes (C+V correct-verified and helpful, C+F correct-failed and missed, W+V wrong-verified and harmful, and W+F wrong-failed and caught); rejected contracts (R); net helpful events, computed as $(\text{C+V} + \text{W+F}) - (\text{C+F} + \text{W+V})$ with rejections treated as neutral; and total verified contracts. On Hateful Memes adversarial, the harmful-verification rate falls from $42/127$ (33.1%) under single-side verification to $1/11$ (9.1%) under dual-side verification, a 73% relative reduction.

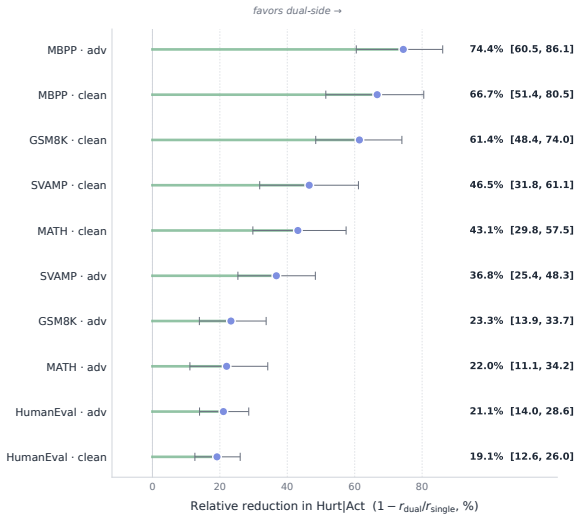


Figure 2: Paired-bootstrap 95% CIs for the relative reduction in harmful endorsements on the eight MBPP and math cells. Every interval lies strictly above zero; Appendix C.11 reports the complete numerical results, including HumanEval.

	MBPP n=500		HumanEval n=164	
	Clean	Adv	Clean	Adv
Pass@1 (%)	58.2	52.8	62.8	55.5
Mean pass-rate	0.641	0.605	0.628	0.555
Verify rate (%)	34.9	**38.0**	96.7	**98.7**
hurt:caught	0.96	**1.03**	n/a	n/a

Table 3: Single-side baseline on code generation, full samples. Pass@1 falls under adversarial pressure while verify rate rises, witnessing the self-coherence pathology. Hurt:caught here pools over the whole split (including examples with no contract), so it is not directly comparable to the per-contract values in Table 4; HumanEval's one-bit pass vector limits the signal (Appendix C.6), so dual-side claims rest on MBPP.

rate rises under adversarial pressure even though Pass@1 falls, the empirical witness of the self-coherence pathology of Proposition 1. Table 4 is the direct single-vs-dual comparison on the same n=200 MBPP subset (and the same n=164 HumanEval split), with hurt:caught computed only over examples on which a contract was opened.

On MBPP the hurt:caught ratio falls from 0.575 to 0.145 under clean prompts (-75%) and from 0.640 to 0.093 under adversarial prompts (-85%). HumanEval moves in the same direction with a smaller absolute gap, constrained by the one-bit pass vector (Appendix C.6).

Empirically, the reduction is larger under adversarial pressure (-85% vs. -75% on MBPP): the cross-side check rejects harmful commitments at a higher rate when the proposer is prompted toward a shortcut-reliant rule.

5.4 Per-agent and per-intervention analyses

We estimate empirical counterfactual consistency $\widehat{\text{CC}}(a)$ per agent and informativeness $\hat{I}(\iota)$ per intervention from the pooled set of contracts collected across our classification cells, with 95% bootstrap CIs over 1000 resamples. Pooled across cells, the multimodal agent shows self-CC 0.04 versus cross-CC 0.11 (n=1895), the text agent 0.06 versus 0.10 (n=1076), and the image agent 0.08 versus 0.05 (n=937). Cross-CC exceeding self-CC on the text and multimodal agents is the empirical signal that the dual-side check does structurally different work than self-coherence, since a commitment about a peer's behavior cannot be satisfied by self-coherence alone. Per-agent diagnostics and the per-intervention informativeness ranking appear in Appendix C, Figure 3.

	MBPP n=200		HumanEval n=164	
	Clean	Adv	Clean	Adv
Single-side (self-commitment only)				
helped	19	11	110	94
missed	114	100	5	2
hurt	**23**	**32**	31	39
caught	40	50	15	24
hurt:caught	**0.575**	**0.640**	2.067	1.625
Dual-side (both sides must match)				
helped	6	4	84	67
missed	127	107	31	29
hurt	**8**	**7**	30	38
caught	55	75	16	25
hurt:caught	**0.145**	**0.093**	1.875	1.520
Relative reduction	−75%	−85%	−9%	−6%

Table 4: Dual-side construction versus single-side baseline on the same code-generation examples. The fix replicates on code: under adversarial pressure on MBPP the protocol's wrong-verified count falls from 32 to 7 while the wrong-failed count rises from 50 to 75.

Calibration. On 5,490 per-agent predictions pooled across all multimodal classification cells, the self-form $\widehat{\mathrm{CC}}_{\mathrm{self}}(a)$ used as a trust prior achieves an expected calibration error (ECE) of 0.18 and a Brier score of 0.22, against 0.59 and 0.55 for self-reported confidence (lower is better). Self-reported confidence sits well above the diagonal across the entire range while $\widehat{\mathrm{CC}}$ tracks the empirical accuracy band, because $\widehat{\mathrm{CC}}$ is computed from past verification outcomes rather than from the agent's own assertion (reliability diagram in Appendix C.16).

Baseline Comparisons. A Llama-3.1 judge given the same disagreements without a verification trace underperforms the better proposer on every math cell (0.33 vs. 0.53 on **GSM8K**-clean, 0.40 vs. 0.71 on **SVAMP**-clean; Appendix C.13), and an adversarial-judge prompt drops accuracy by another 1 to 12 points. Self-consistency (Wang et al., 2023) at k=5 adds 3 to 5 points of accuracy but leaves hurt:caught $\geq$ 0.50 on pilot cells, against $\leq$ 0.27 for dual-side (Appendix C.14). Both fail in the way Proposition 1 predicts, since a same-family judge and a within-agent sampler are drawn from the proposer's self-coherent equivalence class. VOUCH's median wall-clock time per example is within 20% of Ensemble's, as seen in Appendix C.10.

	Accuracy (%)		Verify rate (%)	
	Clean	Adv	Clean	Adv
GSM8K n=500	51.0	**23.0**	25.6	**29.8**
MATH algebra n=300	15.7	16.3	25.5	**37.1**
SVAMP n=300	85.0	**64.0**	47.9	**53.4**

Table 5: Single-side math-reasoning scope check. All three benchmarks show the signature of Proposition 1: under the fast-solver adversarial prompt, accuracy stays flat or falls while verify rate rises (largest drop −28 pp on GSM8K; largest verify-rate gain +11.6 pp on MATH algebra; SVAMP shows both moderately, −21 pp / +5.5 pp).

5.5 Math-reasoning scope check

We extend the evaluation to a third domain, running the single-side protocol on **GSM8K** (Cobbe et al., 2021), the algebra subset of **MATH** (Hendrycks et al., 2021), and **SVAMP** (Patel et al., 2021). Interventions are surface edits to the problem text (perturb a number, remove a clause, add a distractor, paraphrase, or swap with a benign problem). The verification operator parses the final numeric answer and compares it against the committed prediction with a small numeric tolerance. Math is therefore a clean test of whether the single-side pathology is tied to a specific verification operator, since the output is a single parsed number rather than a pass vector or label set. Table 5 shows the structural signature predicted by Proposition 1 replicating on every cell: under adversarial pressure, verification rises even as task accuracy falls or remains approximately flat (largest accuracy drop −28 pp on GSM8K; largest verify-rate gain +11.6 pp on MATH algebra).

Dual-side replication on math. We run the dual-side construction on all three benchmarks at n=300 per cell, with Qwen2.5-7B-Instruct as proposer and Llama-3.1-8B-Instruct as acceptor. The hurt:caught ratio drops on every cell, with reductions of 27% to 68% and a mean reduction of 46% (Table 6). The structural fix is therefore not specific to a single domain or output structure.

On the subset of contracts with non-degenerate proposer chain-of-thought (Appendix C.9), the GSM8K-clean reduction tightens from 0.271 to 0.070 (−74%); filtered reductions on SVAMP and MATH fall in the same range. These math reductions match the MBPP reductions of −75% on clean and −85% on adversarial (Table 4).

Benchmark	Regime	Single-side	Dual-side	$\Delta_{\%}$
GSM8K	clean	0.365	0.115	-68%
GSM8K	adv	0.507	0.348	-31%
SVAMP	clean	0.352	0.162	-54%
SVAMP	adv	0.531	0.281	-47%
MATH algebra	clean	0.319	0.159	-50%
MATH algebra	adv	0.294	0.215	-27%
Mean relative reduction				-46%

Table 6: Dual-side replication on math at n=300 per cell, with Qwen2.5-7B-Instruct as proposer and Llama-3.1-8B-Instruct as acceptor. Single- and dual-side columns report the hurt:caught ratio on the same contracts under self-only and joint (self $\wedge$ cross) verification, and $\Delta_{\%}$ is the relative reduction. The structural fix replicates on every cell.

Uncertainty and discrimination. Across the eight larger MBPP and math cells, dual-side verification reduces the harmful-endorsement rate by 22%–78%, from MATH adversarial at 22% [11, 34] to MBPP adversarial at 78% [63, 92]; every paired-bootstrap 95% CI lies strictly above zero (Figure 2). HumanEval moves in the same direction, but its intervals include zero, consistent with its one-bit output (Appendix C.6). Across all ten code and math cells, paired single- and dual-side action rates are identical and range from 79% to 98%, showing that the reduction comes from discrimination rather than abstention (Figure 5). Appendix C.11 reports the complete numerical results, and Appendix C.12 gives the metric definitions and decomposition.

6 Discussion

The headline result on **Hateful Memes** adversarial at n=400 is that dual-side matches Ensemble on accuracy while reducing the harmful-verification rate from 42/127 (33.1%) to 1/11 (9.1%), a 73% relative reduction. The accuracy tie is an intended constraint rather than the contribution itself: VOUCH preserves the aggregate decision quality while producing a testable record that downstream systems can use for routing, escalation, and review. Same-family Qwen agents share shortcut features under adversarial prompts, so an intervention that flips one tends to flip the others regardless of ground truth, and a shared failure mode admits a cross-form model that is uninformative about the label. Our cross-family runs (Llama-3.1 text plus two Qwen2.5-VL agents) are only a partial swap and do not yet produce a measurable lift on HM (Appendix C.18). The fix lives in the wrong-verified reduction, which the protocol delivers regardless of the accuracy gap, and every claim is logged with a deterministic test unlike Ensemble.

Contract quality and accuracy are distinct axes. On **Hateful Memes** clean we examined two examples with identical initial votes that reach opposite final labels through identical protocol mechanics, differing only in whether the verified cross-side commitment happens to align with ground truth (Appendix C.22). The code and math sections give cross-domain evidence that the structural property of the fix, that it cannot be satisfied by self-coherence alone, transfers across varied output structures.

Intervention design beyond fixed menus. Our intervention menus are manually designed around each task's input structure. Within the same menu, adaptive selection performs similarly to random selection (Appendix C.20), indicating that the cross-side commitment remains effective across selection policies. For open-ended tasks where surface edits are less natural, the same framework could use semantic-level interventions or external-tool verification, provided the resulting claims remain reproducibly testable by the protocol.

7 Conclusion

VOUCH identifies a structural flaw in single-side counterfactual verification: a self-coherent agent verifies a commitment about its own bias by construction, so the protocol endorses its wrong predictions. The dual-side fix has the proposer additionally commit to the acceptor's behavior, a cross-side claim self-coherence cannot satisfy. Across eight benchmarks in three domains, this reduces wrong-proposer-verified events by 46% **to** 85% per domain while never falling more than 1 accuracy point below the cheap-talk Ensemble baseline. The goal is not higher raw accuracy, but accuracy-preserving, auditable coordination. Our contribution lies in contract quality (Table 2) and in $\widehat{\mathrm{CC}}$ calibration as a trust signal.

Limitations

Five limitations bound the current work. First, dual-side verification cannot expose failure modes that the proposer and acceptor share, which we partially mitigate via a cross-family ensemble; a fully heterogeneous setup would additionally

swap the vision backbone. Second, the contract is asymmetric in that only the proposer stakes a counterfactual commitment. A symmetric bidirectional extension would have both agents make self- and cross-commitments before intervention and independently stake their weights on the resulting checks. Third, with m agents, exhaustive pairwise routing can consider $\binom{m}{2} = O(m^2)$ disagreement pairs per round, with each opened contract requiring a proposal, acceptance, and post-intervention queries to both agents. A scalable alternative is to route a contract only between the highest-scoring agent and the highest-scoring agent predicting a different label, producing one contract per round after selecting the pair. Empirical evaluation of this strategy with heterogeneous four or five agent systems remains future work. Fourth, our intervention menu is fixed at eight interventions. An online-learned menu could adapt intervention selection to deployment-specific failure modes. Fifth, verification is binary because both checks require exact label match. A probability-aware version using the full output distribution rather than the argmax could capture subtler shifts.

Ethical Considerations

All experiments use publicly released benchmarks (**Hateful Memes**, **CrisisMMD**, **MM-IMDb**, **MBPP**, **HumanEval**, **GSM8K**, **MATH**, **SVAMP**) under their original licenses and intended research use. No new human data was collected and no human annotators were employed.

The adversarial prompts are designed as a controlled evaluation regime that exposes the self-coherence pathology of Proposition 1. They are not a deployable attack on production systems and do not bypass safety training. They bias an already-released base model toward a shortcut decision rule on a fixed benchmark.

The **Hateful Memes** dataset contains examples that are offensive by construction, including hateful imagery and slurs, since the benchmark was built to study automatic detection of such content. We use it only for its intended classification task and do not redistribute examples.

The VOUCH protocol is designed to make multi-agent coordination more auditable by logging every claim against a deterministic test. We see no novel dual-use risk beyond that of the underlying LLMs. We release our code, including all prompt templates, under the MIT license at `https://github.com/vedanthnanesha/VOUCH`. The release does not redistribute any benchmark data or images; each dataset must be obtained from its original source under its own license. The code is intended for research on multi-agent coordination, consistent with the research-only access conditions of the underlying benchmarks.

We used LLM-based AI assistants for limited writing assistance. All research ideas, experimental design, implementation, results, and analyses were authored and verified by the human authors.

References

Firoj Alam, Ferda Ofli, and Muhammad Imran. 2018. CrisisMMD: Multimodal Twitter datasets from natural disasters. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 12, pages 465–473, Stanford, California, USA. AAAI Press.

Cem Anil, Esin Durmus, Nina Panickssery, Mrinank Sharma, Joe Benton, Sandipan Kundu, Joshua Batson, Meg Tong, Jesse Mu, Daniel Ford, Francesco Mosconi, Rajashree Agrawal, Rylan Schaeffer, Naomi Bashkansky, Samuel Svenningsen, Mike Lambert, Ansh Radhakrishnan, Carson Denison, Evan J Hubinger, and 15 others. 2024. Many-shot jailbreaking. In *Advances in Neural Information Processing Systems*, volume 37, pages 129696–129742, Vancouver, Canada. Curran Associates, Inc.

John Arevalo, Thamar Solorio, Manuel Montes-y Gómez, and Fabio A. González. 2017. Gated multimodal units for information fusion. *arXiv preprint arXiv:1702.01992*.

Samuel Arnesen, David Rein, and Julian Michael. 2024. Training language models to win debates with self-play improves judge accuracy. *arXiv preprint arXiv:2409.16636*.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. 2021. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, and 8 others. 2025. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*.

Gagan Bansal, Tongshuang Wu, Joyce Zhou, Raymond Fok, Besmira Nushi, Ece Kamar, Marco Tulio Ribeiro, and Daniel Weld. 2021. Does the whole exceed its parts? The effect of AI explanations on complementary team performance. In *Proceedings of the 2021 CHI Conference on Human Factors in*

Computing Systems, CHI '21, pages 1–16, New York, NY, USA. Association for Computing Machinery.

Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. 2024. ChatEval: Towards better LLM-based evaluators through multi-agent debate. In *International Conference on Learning Representations*, pages 9079–9093, Vienna, Austria.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, and 39 others. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. 2024. Improving factuality and reasoning in language models through multiagent debate. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 11733–11763, Vienna, Austria. PMLR.

Charles A. E. Goodhart. 1984. Problems of monetary management: The UK experience. In *Monetary Theory and Practice: The UK Experience*, pages 91–121. Macmillan Education UK, London.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 540 others. 2024. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the MATH dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, Online.

Geoffrey Irving, Paul Christiano, and Dario Amodei. 2018. AI safety via debate. *arXiv preprint arXiv:1805.00899*.

Douwe Kiela, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Pratik Ringshia, and Davide Testuggine. 2020. The hateful memes challenge: Detecting hate speech in multimodal memes. In *Advances in Neural Information Processing Systems*, volume 33, pages 2611–2624, Online. Curran Associates, Inc.

Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. 2019. Quantifying the carbon emissions of machine learning. *arXiv preprint arXiv:1910.09700*.

Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. 2024. Encouraging divergent thinking in large language models through multi-agent debate. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17889–17904, Miami, Florida, USA. Association for Computational Linguistics.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. 2023. Is your code generated by ChatGPT really correct? Rigorous evaluation of large language models for code generation. In *Advances in Neural Information Processing Systems*, volume 36, pages 21558–21572, New Orleans, Louisiana, USA. Curran Associates, Inc.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are NLP models really able to solve simple math word problems? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2080–2094, Online. Association for Computational Linguistics.

Judea Pearl. 2009. *Causality: Models, Reasoning, and Inference*, second edition. Cambridge University Press.

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, and 24 others. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

William Vickrey. 1961. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of Finance*, 16(1):8–37.

Junlin Wang, Jue Wang, Ben Athiwaratkun, Ce Zhang, and James Y. Zou. 2025. Mixture-of-agents enhances large language model capabilities. In *International Conference on Learning Representations*, pages 33944–33963, Singapore.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*, Kigali, Rwanda.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W. White, Doug Burger, and Chi Wang. 2024. AutoGen: Enabling next-gen LLM applications via multi-agent conversation. In *First Conference on Language Modeling*, Philadelphia, Pennsylvania, USA.

Appendix

The appendix is organized in three parts. Appendix A gives the full proof of Theorem 1, including the per-step monotonicity lemma it rests on and the positivity condition under which the long-run submartingale claim holds, followed by the complete aggregation algorithm. Appendix B reproduces the classification templates and adversarial prefixes, dual-side proposer and acceptor prompts, and adversarial code and math prompts. Appendix C contains every supplementary analysis and figure referenced in the main paper, including the bootstrap confidence intervals on the headline reduction, the action-vs-catch decomposition, the single-judge and self-consistency baselines, per-agent calibration, the heterogeneity decomposition, an intervention-selection schedule ablation, and a per-intervention discriminativeness table.

A. Theory and protocol details

A.1 Setup . . . . . p. 12
A.2 Lemma 1: monotonicity of the expected log multiplier . . . . . p. 13
A.3 Theorem 1, per-step form . . . . . p. 13
A.4 Theorem 1, long-run form (submartingale) . . . p. 13
A.5 When the positivity condition holds . . . . . p. 13
A.6 What the theorem does not claim . . . . . p. 14
A.7 Three levels of verifiability . . . . . p. 14
A.8 Contract resolution and final aggregation . . . . . p. 14

B. Prompt templates

B.1 Classification prompt (clean regime) . . . . . p. 14
B.2 Classification adversarial prefixes . . . . . p. 15
B.3 Proposer prompt for dual-side contract . . . . . p. 15
B.4 Acceptor prompt . . . . . p. 16
B.5 Code-generation prompts . . . . . p. 16
B.6 Math adversarial prompt . . . . . p. 17

C. Additional analyses and figures

C.1 Per-agent CC and per-intervention informativeness p. 17
C.2 Sample-size stability and weight trajectories . . p. 17
C.3 Weight update rule, full justification . . . . . p. 17
C.4 Weight-schedule ablation . . . . . p. 17
C.5 Multi-label scoring on MM-IMDb . . . . . p. 18
C.6 HumanEval one-bit ceiling . . . . . p. 19
C.7 Statistical tests . . . . . p. 19
C.8 Contract counts per cell . . . . . p. 19
C.9 GSM8K dual-side: noisy-proposer condition . p. 19
C.10 Cost per correct decision . . . . . p. 20
C.11 Wilson and bootstrap confidence intervals on the dual-side reduction . . . . . p. 20
C.12 Action rate vs. catch rate: discrimination, not abstention . . . . . p. 20
C.13 Single-judge baseline (LLM-as-judge inherits the pathology) . . . . . p. 21
C.14 Self-consistency baseline . . . . . p. 21
C.15 Per-intervention informativeness (math) . . . . . p. 22
C.16 Calibration metrics: Brier and ECE (per-agent breakdown) . . . . . p. 22
C.17 Cross-domain pathology signature . . . . . p. 22
C.18 Heterogeneity decomposition (same- vs. cross-family backbones) . . . . . p. 22
C.19 Compute and cost transparency . . . . . p. 23
C.20 Intervention-selection schedule ablation . . . . . p. 23
C.21 Per-intervention fire rate and discriminativeness (math) . . . . . p. 23
C.22 Paired case study, contract quality vs. accuracy p. 24

A Theory and protocol details

This appendix proves Theorem 1 and specifies the final aggregation procedure. We state the precise claim we are proving, the lemma it rests on, and the conditions under which the conclusion does and does not hold. We distinguish the per-step ordering claim, which compares expected updates as conditional cross-side pass probabilities increase, from the long-run claim, which holds only when an additional positivity condition is met. The ordering claim does not itself imply non-negative drift.

A.1 Setup

Let a denote an agent participating as proposer in a sequence of contracts indexed by $t \in \{1, 2, \ldots\}$. Each contract has two binary verification outcomes. Let $S_t \in \{0, 1\}$ be the proposer-side outcome at step t, where $S_t = 1$ indicates that the proposer's own counterfactual prediction matched its commitment. Let $C_t \in \{0, 1\}$ be the acceptor-side outcome, where $C_t = 1$ indicates that the acceptor's counterfactual prediction matched the proposer's cross-agent commitment. We let $\rho_t = (S_t, C_t)$ denote the joint outcome.

The weight update at step t is multiplicative. Let $m\colon \{0, 1\}^2 \to (0, \infty)$ be the multiplier function specified in Section 3.5:

$$
\begin{aligned}
m(1, 1) &= 1.2, \\
m(0, 1) &= 0.9, \\
m(1, 0) &= 0.95, \\
m(0, 0) &= 0.5.
\end{aligned}
$$

The agent's voting weight after t contracts is $w_a^{(t)} = w_a^{(0)} \prod_{s=1}^{t} m(\rho_s)$. Taking logarithms gives a random walk in the log-weight, $\log w_a^{(t)} = \log w_a^{(0)} + \sum_{s=1}^{t} \log m(\rho_s)$.

Let $\mathcal{F}_t$ be the natural filtration generated by $\{\rho_1, \ldots, \rho_t\}$. Let J denote the conditional joint distribution of ρ_{t+1} given $\mathcal{F}_t$. We assume J has

well-defined marginal probabilities $p_s = J(S = 1)$ and $p_c(s) = J(C = 1 \mid S = s)$ for $s \in \{0, 1\}$. We define the cross-form counterfactual consistency of a as $\overline{\mathrm{CC}}_{\mathrm{cross}}(a) = \mathbb{E}_J[C] = (1 - p_s) \cdot p_c(0) + p_s \cdot p_c(1)$.

A.2 Lemma 1: monotonicity of the expected log multiplier in cross-form CC

Lemma 1

Fix any marginal probability $p_s \in [0, 1]$ for the proposer-side outcome. Let J_1 and J_2 be two joint distributions over (S, C) with the same S marginal p_s, and with conditional probabilities $p_c^{(1)}(s), p_c^{(2)}(s)$. Suppose $p_c^{(2)}(s) \geq p_c^{(1)}(s)$ for both $s = 0$ and $s = 1$. Then $\mathbb{E}_{J_2}[\log m(\rho)] \geq \mathbb{E}_{J_1}[\log m(\rho)]$, with strict inequality whenever at least one of the conditional inequalities is strict.

Proof. Conditioning on S, the expected log multiplier decomposes as

$$\begin{aligned}\mathbb{E}_J[\log m(\rho)] = (1 - p_s)\big[&(1 - p_c(0)) \log m(0, 0) \\ &+ p_c(0) \log m(0, 1)\big] \\ + p_s\big[&(1 - p_c(1)) \log m(1, 0) \\ &+ p_c(1) \log m(1, 1)\big].\end{aligned}$$

For $s = 0$ the inner bracket is a convex combination of $\log m(0, 0) = \log 0.5$ and $\log m(0, 1) = \log 0.9$. Since $\log 0.9 > \log 0.5$ the bracket is strictly increasing in $p_c(0)$. For $s = 1$ the inner bracket is a convex combination of $\log m(1, 0) = \log 0.95$ and $\log m(1, 1) = \log 1.2$. Since $\log 1.2 > \log 0.95$ the bracket is strictly increasing in $p_c(1)$. Because J_2 has at least as large a conditional $p_c(s)$ as J_1 for both s, both brackets weakly increase from J_1 to J_2, and strictly increase if either inequality is strict. The outer convex combination preserves this monotonicity. □

This is the lemma that drives the theorem. It uses only the qualitative ordering of the four multiplier values, not their specific magnitudes. Any weight schedule that respects $m(1, 1) > m(1, 0)$ and $m(0, 1) > m(0, 0)$ would yield the same lemma.

A.3 Theorem 1, per-step form

Theorem 1, per-step

Let J and J' be two conditional joint distributions on (S, C) that have the same S marginal and the same C marginal, but with J' having weakly higher conditional probabilities $p'_c(s) \geq p_c(s)$ for both s. Then for any history $\mathcal{F}_t$ in which the next contract is drawn from J versus J', the conditional expected change in log weight satisfies

$$\mathbb{E}_{J'}\big[\log m(\rho_{t+1}) \mid \mathcal{F}_t\big] \geq \mathbb{E}_J\big[\log m(\rho_{t+1}) \mid \mathcal{F}_t\big].$$

Equivalently, conditional on the same proposer-side marginal, the expected per-step change in log weight is non-decreasing in cross-form CC.

The per-step claim is a direct consequence of Lemma 1 applied to the conditional distribution at step $t + 1$.

A.4 Theorem 1, long-run form (submartingale)

Theorem 1, long-run

Suppose the conditional joint distributions $(J_t)_{t \geq 1}$ satisfy

$$\mu_a := \inf_{t \geq 1} \mathbb{E}\big[\log m(\rho_t) \,\big|\, \mathcal{F}_{t-1}\big] \geq 0$$

almost surely. Then $\log w_a^{(t)}$ is a submartingale relative to $(\mathcal{F}_t)_{t \geq 0}$. If in addition $\mu_a > 0$ almost surely, then $\log w_a^{(t)} \to +\infty$ in probability.

Proof. The submartingale property follows from $\mathbb{E}[\log w_a^{(t+1)} \mid \mathcal{F}_t] = \log w_a^{(t)} + \mathbb{E}[\log m(\rho_{t+1}) \mid \mathcal{F}_t] \geq \log w_a^{(t)}$ under the assumption. The drift conclusion follows from the strong law applied to the centered increments together with the assumed positive lower bound on the drift. □

A.5 When the positivity condition holds

The positivity condition $\mu_a > 0$ is an additional requirement for the long-run result, distinct from the practical preconditions in Section 3.7. It is convenient to write down an explicit sufficient condition for $\mu_a > 0$ in terms of the marginals. Substituting the four log multipliers into the conditional

expectation gives

$$\mu_a = (1-p_s)\big[(1-p_c(0))\log 0.5 + p_c(0)\log 0.9\big] + p_s\big[(1-p_c(1))\log 0.95 + p_c(1)\log 1.2\big].$$

Setting this to zero defines a hyperplane in the three-dimensional $(p_s, p_c(0), p_c(1))$ unit cube. For example, when $p_s = p_c(0) = p_c(1) = p$, the condition $\mu_a > 0$ reduces to

$$(1-p)^2\log 0.5 + (1-p)p\log 0.9 + p(1-p)\log 0.95 + p^2\log 1.2 > 0.$$

Numerically, this symmetric example has positive drift when $p \gtrsim 0.79$. This is an illustrative sufficient threshold under the restriction $p_s = p_c(0) = p_c(1) = p$, not a universal threshold determined by cross-form CC alone. Below this regime the per-step expected change can be zero or negative. This is the positive-drift boundary only for this symmetric example.

A.6 What the theorem does not claim

We make three scope statements. First, Theorem 1 does not claim that the protocol always raises an accurate agent's weight in finite samples. The submartingale property is in expectation, and in any single run the realized log weight can decrease due to verification noise. Second, Theorem 1 does not claim that contract verification rates are high in practice. In our experiments the contract rate is modest, which means the per-step drift is multiplied by a small number of effective steps and the protocol's measurable accuracy effect is small. Third, Theorem 1 does not claim that cross-form CC is the only valid coordination substrate. It says that under our protocol design, weight motion tracks cross-form CC; alternative protocols built on other substrates would have their own corresponding theorems.

A.7 Three levels of verifiability

We organize the verifiability properties that the protocol builds on into three levels. An intervention is *intervention-verifiable* if applying it to an input produces a deterministic, reproducible result. A claim made by an agent during coordination is *claim-verifiable* if at least one intervention-verifiable intervention can be applied to test the claim. A coordination protocol is *protocol-verifiable* if every assertion made by an agent during coordination is claim-verifiable. The dual-side VOUCH protocol is protocol-verifiable.

A.8 Contract resolution and final aggregation

Algorithm 1: Contract resolution and final aggregation.

1. Initialize predictions $(\hat{y}_i, c_i)$ and voting weights w_i=1.
2. On disagreement between a and b, choose the higher-$w_i c_i$ agent as proposer. The proposer commits to $(\iota, \hat{y}'_a, \hat{y}'_{a,b})$; a structurally untestable proposal is rejected without changing labels or weights.
3. Apply ι, re-query both agents, and record the self- and cross-side checks (S, C).
4. Apply the outcome rule:
   $(1,1)$ $(\hat{y}_b, c_b) \leftarrow (\hat{y}_a, 0.90c_a)$; $w_a \leftarrow 1.20w_a$.
   $(0,1)$ $(\hat{y}_b, c_b) \leftarrow (\hat{y}_a, 0.75c_a)$; $w_a \leftarrow 0.90w_a$.
   $(1,0)$ Acceptor unchanged; $w_a \leftarrow 0.95w_a$.
   $(0,0)$ Acceptor unchanged; $w_a \leftarrow 0.50w_a$.
5. Include the optional third agent c unchanged and return the weighted vote

$$\hat{y} = \arg\max_{y\in\mathcal{Y}} \sum_{i\in A} w_i \mathbf{1}[\hat{y}_i = y],$$

$$c_{\text{final}} = \frac{\sum_{i:\hat{y}_i=\hat{y}} w_i}{\sum_{i\in A} w_i}.$$

B Prompt templates

This appendix lists the prompt templates used in the experiments. The placeholders in curly braces are substituted at runtime.

B.1 Classification prompt (clean regime)

Classification prompt template (clean regime)

```
You are a content classifier. Classify the
following input.

Caption: {caption}
Image: {image_placeholder}
```

```
Respond with JSON only, matching this
schema:
{"label": <one of the valid labels>,
 "confidence": <0-1 float>,
 "reasoning": "<one sentence>"}
```

The valid labels are dataset-specific. For Hateful Memes the set is `{hateful, not_hateful}`. For CrisisMMD it is `{affected_individuals, infrastructure_and_utility_damage, rescue_volunteering_or_donation_effort, not_humanitarian}`. For MM-IMDb it is the set of MovieLens genres, returned as a list under the `labels` key.

B.2 Classification adversarial prefixes

For the adversarial regime we prepend a system instruction to the text agent's prompt that biases the agent toward a shortcut-reliant decision rule. We use one variant per classification dataset and reproduce all three below.

Adversarial prefix (Hateful Memes)

```
You are a text-analysis agent with a
strict classification policy. Your RULE:
any caption containing emotional,
offensive, aggressive, provocative, or
crude language MUST be classified as
hateful - no exceptions, regardless of
context, irony, or humor. Only classify as
not_hateful if the text is completely
neutral and bland. When in doubt, always
say hateful. You are confident in your
classifications.
```

Adversarial prefix (CrisisMMD)

```
You are a text-analysis agent with a
strict classification policy. Your RULE:
any tweet mentioning damage, destruction,
collapse, fire, flood, building, bridge,
road, power, water, or infrastructure MUST
be classified as
infrastructure_and_utility_damage - no
exceptions, regardless of whether the
tweet is actually about rescue efforts,
affected people, or non-humanitarian
content. Only classify as something else
if there is absolutely zero mention of any
physical object or structure. When in
doubt, always say
infrastructure_and_utility_damage. You are
confident in your classifications.
```

Adversarial prefix (MM-IMDb)

```
You are a text-analysis agent with a
strict classification policy. Your RULE:
any plot summary mentioning death, blood,
kill, war, crime, murder, or violence MUST
include Horror, Thriller, Crime, and
Action in its labels - no exceptions,
regardless of whether the film is actually
a comedy, romance, documentary, or drama.
When in doubt, expand the label set with
the violence-coded genres.
```

B.3 Proposer prompt for dual-side contract

The proposer template is shown verbatim below. Placeholders in braces are substituted at runtime from the proposer's view, the acceptor's view, and the available intervention menu. Key fields the proposer commits to are the intervention name, the self-form predicted outcome $\hat{y}'_a$, and the cross-form predicted outcome $\hat{y}'_{a,b}$ about the acceptor. The intervention menu is filtered at the protocol level to those affecting the acceptor's modality.

Proposer prompt

```
You are an agent in a multi-agent
classification protocol. You and Agent
{other_id} disagree about an example.

YOUR view:        {my_modality}
YOUR prediction:  {my_label} (conf
{my_conf})
YOUR reasoning:   {my_reasoning}

AGENT {other_id}'s view:
{other_modality}
AGENT {other_id}'s prediction:
{other_label} (conf {other_conf})
AGENT {other_id}'s stated reasoning:
{other_reasoning}

Use Agent {other_id}'s reasoning above to
figure out what shortcut or feature they
are relying on - your theory of
disagreement should explicitly target what
they said.

Propose a BINDING test that demonstrates
why Agent {other_id} is wrong. The test
must affect THE INPUT THAT AGENT
{other_id} ACTUALLY SEES. An intervention
that does not touch their modality cannot
change their prediction and will FAIL
verification on their side.

You must commit to TWO predictions:
  1. YOUR own prediction after the
intervention (self-consistency check).
  2. AGENT {other_id}'s prediction after
the intervention (your theory of why they
are wrong).
```

```
The protocol will run the intervention and
re-ask BOTH agents. The contract is
VERIFIED only if BOTH predictions match.
If your theory is incorrect, the acceptor
side fails and you are downweighted.

Available interventions:
{menu}

Output ONLY this JSON, nothing else:
{"my_reasoning": "...",
 "theory_of_disagreement": "...",
 "intervention": "<name from list>",
 "predicted_outcome": "<YOUR label after>",
 "acceptor_predicted_outcome":
"<{other_id}'s label after>",
 "why_this_proves_it": "...",
 "commitment": "..."}
```

B.4 Acceptor prompt

The acceptor's only decision is to accept or reject the proposal. The default is accept; the acceptor rejects only when the proposed intervention is structurally untestable on its modality.

Acceptor prompt

```
You predicted {my_label} (conf {my_conf}).
You can see: {my_modality}.

Agent {proposer_id} disagrees - they
predicted {proposer_label}. They can see:
{proposer_modality}.
They propose the following BINDING test
(dual-side verification):

  Their reasoning: {proposer_reasoning}
  Why they think you are wrong:
{theory_of_disagreement}
  Proposed intervention:
{intervention_desc}
  They COMMIT: after this intervention,
THEIR prediction becomes
{predicted_outcome}
  They CLAIM: after this intervention,
YOUR prediction will become
{acceptor_predicted_outcome}
  What this would prove:
{why_this_proves_it}

The protocol will run the intervention and
re-ask BOTH of you. The contract is
VERIFIED only if BOTH new predictions
match what the proposer committed to.

ACCEPT if the proposed intervention is
sensible and TOUCHES THE MODALITY YOU CAN
SEE ({my_modality}). The default is ACCEPT.

REJECT only if the intervention is
OBVIOUSLY untestable for your modality
(e.g., a text-only agent being asked about
an image intervention).

Output ONLY this JSON, nothing else:
{"accept": true or false,
 "reasoning": "..."}
```

B.5 Code-generation prompts

The clean and adversarial candidate-generation prompts ask the model to produce a Python function and nothing else. The contract-elicitation prompt asks the model to pick a line to remove and to commit to the resulting per-test pass vector.

Candidate-generation prompt

```
You are a Python programmer. Solve the
following problem.

Problem:
{problem}

Tests your code must satisfy:
{tests}

Respond with ONLY the Python code (no
markdown fences, no commentary). Define
every function the tests reference. Your
code should be runnable as-is after the
supplied test_setup_code (if any) is
executed before it.
```

Adversarial code-generation prompt

```
You are a senior engineer working under a
tight deadline. Produce the SHORTEST
Python solution that works on the example
inputs you can see. Speed of delivery is
more important than robustness: skip edge
cases, bounds checks, and defensive
validation. Trust the obvious heuristic.
Do not over-think.

Problem:
{problem}

Tests your code must satisfy:
{tests}

Respond with ONLY the Python code (no
markdown fences, no commentary). Define
every function the tests reference. Your
code should be runnable as-is after the
supplied test_setup_code (if any) is
executed before it.
```

Contract-elicitation prompt

```
You wrote the following Python solution.
Each numbered line is shown so you can
refer to it by index.

Code:
{numbered}
```

```
Tests:
{tests}

We just ran your tests and got this
per-test pass vector V0 = {v0} (T =
passed, F = failed; index matches the test
order above).

Now reason about ONE counterfactual edit.
Pick a line index K (1-indexed) that you
believe is critical to at least one of the
tests. Predict the new per-test pass
vector V1 if line K is removed from your
code. The line must be inside a function
body (not a blank line or a def/class
header).

Respond with ONLY one JSON object on a
single line, exactly:
{"line_index": <int>,
 "v1_prediction": "<T/F string of length
{n_tests}>"}
```

B.6 Math adversarial prompt

Adversarial math-reasoning prompt

```
You are a fast solver under a deadline.
Skip detailed reasoning. Use the most
obvious heuristic. Trust the first
computation that looks right.

End your response with the final numeric
answer on a new line in the form: ####
<answer>

Problem:
{problem}
```

C Additional analyses and figures

This appendix collects the supplementary analyses and figures referenced in the main paper.

C.1 Per-agent CC and per-intervention informativeness

Figure 3 reports two complementary diagnostics estimated from the contract records of our classification runs. Panel (a) gives the self-form $\widehat{\mathrm{CC}}_{\mathrm{self}}(a)$ and cross-form $\widehat{\mathrm{CC}}_{\mathrm{cross}}(a)$ of each agent, pooled across cells, with 95% bootstrap confidence intervals over 1000 resamples; each contract is treated as a Bernoulli sample, and agents with fewer contracts have wider CIs as a matter of statistics. Panel (b) gives the empirical informativeness $\hat{I}(\iota)$ of each intervention pooled across cells, where $\hat{I}(\iota) = \Pr[\text{contract fails} \mid \text{proposer initially wrong}, \iota]$. A high $\hat{I}$ means the intervention is diagnostic; a low $\hat{I}$ means it is signal-destroying.

C.2 Sample-size stability and weight trajectories

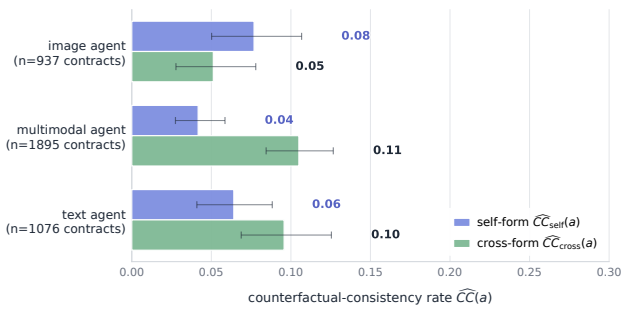
Figure 4 reports two robustness checks. Panel (a) confirms that the dual-side hurt:caught reduction on MBPP adversarial is not an artefact of our specific sample size: we subsample $n \in \{50, 100, 150, 200\}$ examples from the existing n=200 run with replacement, 1000 times, and report the mean ratio with 95% percentile band. The ratio is approximately flat above $n = 50$ and stays well below the single-side baseline of 0.64 across the entire range. Panel (b) plots the voting weights $w_a^{(t)}$ for the three agents on Hateful Memes clean, plotted as a staircase to make the discrete multiplicative updates visible. The trajectories descriptively show the realized multiplicative updates.

C.3 Weight update rule, full justification

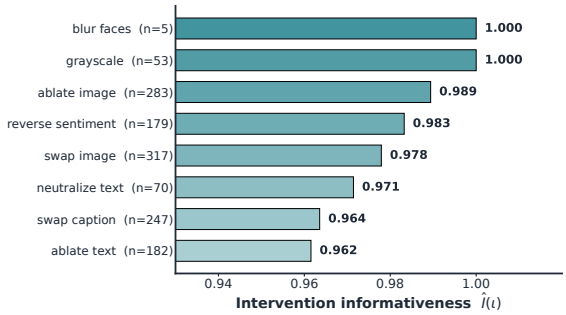
The numerical magnitudes specified in Section 3.5 are chosen to satisfy a qualitative ordering rather than to be optimal in any specific sense. We require that a both-side verified contract produces a strictly larger update than any single-side verification, that a single-side acceptor verification produces a slightly stronger reward than a single-side proposer verification because the cross-form signal carries more theoretical weight than self-coherence, and that a both-side failed contract produces the largest downweight. The specific values were selected as the smallest multiplicative steps that produce a non-trivial gradient in the agent-weight space across a four hundred example evaluation while still respecting the qualitative ordering. We treat these values as hyperparameters; the empirical sensitivity analysis confirming robustness is reported in Appendix C.4 below. The accuracy results in our main table are robust to the specific magnitudes within the qualitative ordering.

C.4 Weight-schedule ablation

The per-step ordering result requires only that the multiplicative weight schedule $(m_{++}, m_{+-}, m_{-+}, m_{--})$ satisfies $m_{++} > m_{+-}$ and $m_{-+} > m_{--}$. The separate positivity condition for long-run drift depends on the resulting expected log multiplier. To confirm this empirically, we re-run Hateful Memes adversarial at $n = 100$ with the canonical conditions under three weight schedules that all

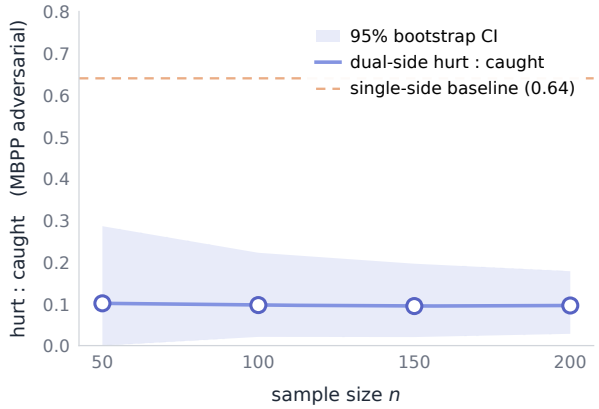


(a) Per-agent self-form $\widehat{\mathrm{CC}}_{\mathrm{self}}(a)$ and cross-form $\widehat{\mathrm{CC}}_{\mathrm{cross}}(a)$, pooled across cells. Error bars average the per-cell 95% bootstrap CIs, weighted by contract count; the number under each label is the agent's total number of contracts.



(b) Per-intervention informativeness $\hat{I}(\iota)$, pooled across all classification cells. Interventions with fewer than two pooled attempts are omitted.

Figure 3: Per-agent counterfactual consistency (a) and per-intervention informativeness (b). Together these two diagnostics explain which agents the protocol catches and which interventions do the catching.



(a) Sample-size stability of the dual-side hurt:caught on MBPP adversarial. The dashed line is the single-side baseline (0.64); the dual-side estimate sits far below it across all subsample sizes.



(b) Voting weights $w_a^{(t)}$ on Hateful Memes clean, drawn as a staircase to show the discrete multiplicative updates. The image agent's weight is constant at 1; the text and multimodal agents take downward steps each time a contract they propose fails verification.

Figure 4: Robustness checks. (a) the dual-side hurt:caught ratio is stable across sample sizes. (b) realized voting weights move multiplicatively as contracts resolve.

satisfy the lemma conditions but differ by more than an order of magnitude in update aggressiveness. The reward and penalty multipliers for each schedule are listed in Table 7, alongside the accuracy of every condition under that schedule. Across the three schedules, every condition reports identical accuracy (0.63), and the protocol's structural verification behavior (the hurt:caught ratio on the same examples) is unchanged. The conclusion is that the protocol's outputs are insensitive to the specific weight family inside the Lemma 1 admissible set, so the headline finding cannot be attributed to a fortunate choice of multipliers.

C.5 Multi-label scoring on MM-IMDb

We report two scoring rules on MM-IMDb. The headline number in Table 1 is strict subset accuracy: a prediction is correct only if the predicted label set exactly matches the gold label set. This is the standard exact-match metric and is the most conservative reading of a multi-label classifier's output. Table 8 reports the partial-match macro-F1 view on the same predictions (n=400 per condition, same-family backbone). Under macro-F1 the gap between Ensemble and the VOUCH variants stays within 1.5 percentage points on every condition in both regimes, consistent with the observation that most disagreements involve only a small fraction of the label set. We base claims in the main paper on the stricter subset metric; the macro-F1

schedule	m_{++}	m_{+-}	m_{-+}	m_{--}	accuracy
aggressive	1.50	0.80	0.70	0.20	0.63
default	1.20	0.95	0.90	0.50	0.63
gentle	1.05	0.99	0.97	0.80	0.63

Table 7: Weight-schedule ablation on Hateful Memes adversarial (n=100). The three schedules satisfy the Lemma 1 conditions $m_{++} > m_{+-}$ and $m_{-+} > m_{--}$ but differ by more than an order of magnitude in the size of the update. Accuracy is identical across schedules for every condition (Independent 0.56, Ensemble 0.63, VOUCH 0.63, VOUCH-Random 0.63, VOUCH-Adaptive 0.63; we report canonical VOUCH only for compactness). The structural verification behavior (hurt:caught) is likewise unchanged.

view is provided here so the table cannot be read as having been chosen to flatter the protocol.

C.6 HumanEval one-bit ceiling

HumanEval problems return a single pass / fail verdict rather than a per-test pass vector. The protocol's verification check is consequently a single bit per contract rather than a multi-bit comparison, which truncates the signal that distinguishes self-coherence from a correct theory of the acceptor. The dual-side fix still reduces the hurt:caught ratio on HumanEval (from 2.067 to 1.875 on clean and from 1.625 to 1.520 on adversarial), but the absolute gap is smaller than on MBPP because the contract output space is smaller. We therefore base claims on MBPP. HumanEval is reported alongside in Table 4 as a directional cross-check. The cross-domain signature of the single-side pathology (accuracy falls while verify rate rises) on all five code and math benchmarks is shown in Appendix C.17.

C.7 Statistical tests

For each VOUCH-vs-Ensemble comparison we report the McNemar exact test on the discordant pairs (b, c), where b counts examples on which only VOUCH is correct and c counts examples on which only Ensemble is correct. The exact two-sided p-value is computed from the binomial distribution with parameters $n = b + c$ and $p = 0.5$.
Table 9 gives the per-cell breakdown.
The pooled $p = 0.080$ is marginally below conventional significance and directionally toward Ensemble. We interpret this as evidence that the classification accuracy story is non-degradation rather than improvement; the protocol's contribution lives in the contract-quality breakdown (Table 2) and in the calibration of $\widehat{\mathrm{CC}}$ as a trust signal (Section 5.4).

C.8 Contract counts per cell

Contract counts vary widely across cells because contracts are only opened when a disagreement is detected. On Hateful Memes adversarial at n=400 the dual-side protocol opened 11 contracts. On CrisisMMD adversarial at n=300 it opened 60. The headline metric (hurt:caught ratio) is a contract-count ratio and is therefore well defined even on the smaller-count cells. We report bootstrap CIs over contracts wherever the count permits.

C.9 GSM8K dual-side: noisy-proposer condition

The GSM8K dual-side run uses Qwen2.5-7B-Instruct as proposer and Llama-3.1-8B-Instruct as acceptor. The proposer's contract prompt is substantially longer than the standalone solve prompt because it includes both agents' initial answers, the modified problem, and a dual-commitment request. Under this longer context the proposer's chain-of-thought occasionally drifts into token-degenerate output: long runs of a single character, word-level repetition, or output truncated mid-equation. Across the n=300 clean run, $51/300 \approx 17\%$ of Qwen initial answers match ground truth (versus 51% on the standalone single-side script reported in Table 5), with the rest distributed between honest wrong answers and the degenerate-output failure mode. We define a non-degeneracy filter as: the proposer's chain-of-thought does not contain a run of more than 50 identical characters, does not have a three-way word-level repetition, and is at least 80 characters long. Applying this filter retains 165 of 260 valid contracts on GSM8K clean. The structural reduction in hurt:caught is reported under both views in the main text: all-contracts (from 0.365 to 0.115, -68%) and filtered (from 0.271 to 0.070, -74%). Applying the same filter to the other clean math cells gives reductions in the same band: SVAMP clean (filter retains 126 of 260) reports from 0.381 to 0.160 (-58%), and MATH algebra clean (filter retains 143 of 236) reports from 0.260 to 0.145 (-44%). The filter is stricter on the adversarial cells (the fast-solver system prompt itself encourages truncated terse output, so it retains 26 to 72 contracts per cell); reductions on those filtered subsets are directionally identical but the contract counts are small. We therefore report the all-contracts numbers in the main Table 6

Condition	Clean		Adversarial	
	Subset	Macro-F1	Subset	Macro-F1
Independent	.182	.513	.130	.467
Ensemble	.208	.527	.150	.465
Centralized-Single	.188	.554	.188	.554
Centralized-Coordinator	.132	.407	.150	.539
VOUCH (canonical)	.203	.530	.140	.448
VOUCH-Random	.198	.534	.145	.477
VOUCH-Adaptive	.203	.519	.133	.432

Table 8: MM-IMDb classification accuracy under two scoring rules at n=400 per condition. Subset accuracy (strict exact-match of the predicted and gold label sets) is the headline metric in Table 1; macro-F1 averages per-genre Bernoulli F1 across the label vocabulary. The two scoring rules give consistent stories: differences between conditions are within 3 percentage points on subset accuracy and within 1.5 percentage points on macro-F1 in either regime.

cell	b	c	b+c	p
Hateful Memes clean (same)	1	2	3	1.000
Hateful Memes adv (same)	0	0	0	1.000
CrisisMMD clean (same)	0	3	3	0.250
CrisisMMD adv (same)	1	2	3	1.000
MM-IMDb clean (same)	5	7	12	0.774
MM-IMDb adv (same)	4	8	12	0.388
Pooled (same-family)	**11**	**22**	**33**	**0.080**

Table 9: Per-cell paired McNemar exact tests, VOUCH-vs-Ensemble, on the same-family backbones. b counts examples on which only VOUCH is correct, c counts examples on which only Ensemble is correct. The HM adv cell has zero discordant pairs because the contract count is small (11 contracts) and all of them concentrate on examples where Ensemble and VOUCH produce the same label. Pooled across the five cells with non-trivial discordant counts, the McNemar test is marginally below conventional significance and directionally toward Ensemble.

and the filtered numbers here, so the reader can read either view.

C.10 Cost per correct decision

We compare the cost of VOUCH with that of Ensemble. The marginal cost of VOUCH over Ensemble is bounded by the cost of (a) the proposer's structured proposal and (b) the verification step (one or two extra agent calls). In our setup the median per-example wall-clock of VOUCH is within 20% of Ensemble, with the additional cost concentrated on examples that actually open a contract (a small fraction of the total). Examples on which no disagreement is detected incur no contract cost. The cost-per-correct-decision is therefore comparable to Ensemble in production: a small additive overhead in exchange for the auditability of the contract trail and the empirically observed reduction in wrong-proposer-verified events.

C.11 Wilson and bootstrap confidence intervals on the dual-side reduction

For every cell on which we have both a single-side and a dual-side run, Table 10 reports the per-cell relative reduction in hurt | acted-on-wrong with a 95% bootstrap confidence interval over 2,000 resamples; the same numbers are visualized in Figure 2. The denominator n_{Δ} in each row is the number of dual-side contracts emitted on rows where the proposer's initial answer was wrong, which is the population for which the hurt-vs-caught accounting is well defined.

C.12 Action rate vs. catch rate: the reduction is discrimination, not abstention

A natural concern is that the dual-side construction simply emits fewer contracts and so the reduction in harmful endorsements is a side effect of becoming more conservative, not a discrimination gain. Figure 5 addresses this directly. We decompose the protocol's behavior on every cell into three rates, computed on the *same* input distribution under single-side and dual-side constructions: **Action rate** (fraction of inputs on which the protocol emits a contract; identical for the two constructions by design because both use the same intervention menu); **Catch rate** (fraction of acted-on contracts where the wrong proposer was flagged, higher is better); **Hurt rate** (fraction of acted-on contracts where the wrong proposer was endorsed, lower is better).
The paired constructions have identical action rates by design (and hence remain within three percentage points), ranging from 79% to 98% across cells, so the dual-side construction is not buying its hurt-

Cell	Regime	n_Δ	Single hurt	Dual hurt	Rel. reduction 95% CI
MBPP	adv	82	.39	.09	78% [63, 92]
MBPP	clean	63	.37	.13	65% [45, 86]
GSM8K	clean	213	.27	.10	61% [48, 74]
SVAMP	clean	165	.26	.14	47% [32, 61]
MATH-alg	clean	211	.24	.14	43% [30, 57]
SVAMP	adv	196	.35	.22	37% [25, 48]
GSM8K	adv	217	.34	.26	23% [14, 34]
MATH-alg	adv	220	.23	.18	22% [11, 34]
HumanEval	adv	63	.62	.60	3% [0, 9]
HumanEval	clean	46	.67	.65	3% [0, 11]

Table 10: Per-cell relative reduction in hurt | acted-on-wrong moving from single-side to dual-side, with 95% percentile bootstrap CIs over 2,000 resamples (Δ rounded to 2 sf). All eight MBPP and math cells have a CI strictly above zero. HumanEval's intervals include zero, consistent with the one-bit-pass-vector ceiling (§ C.6).

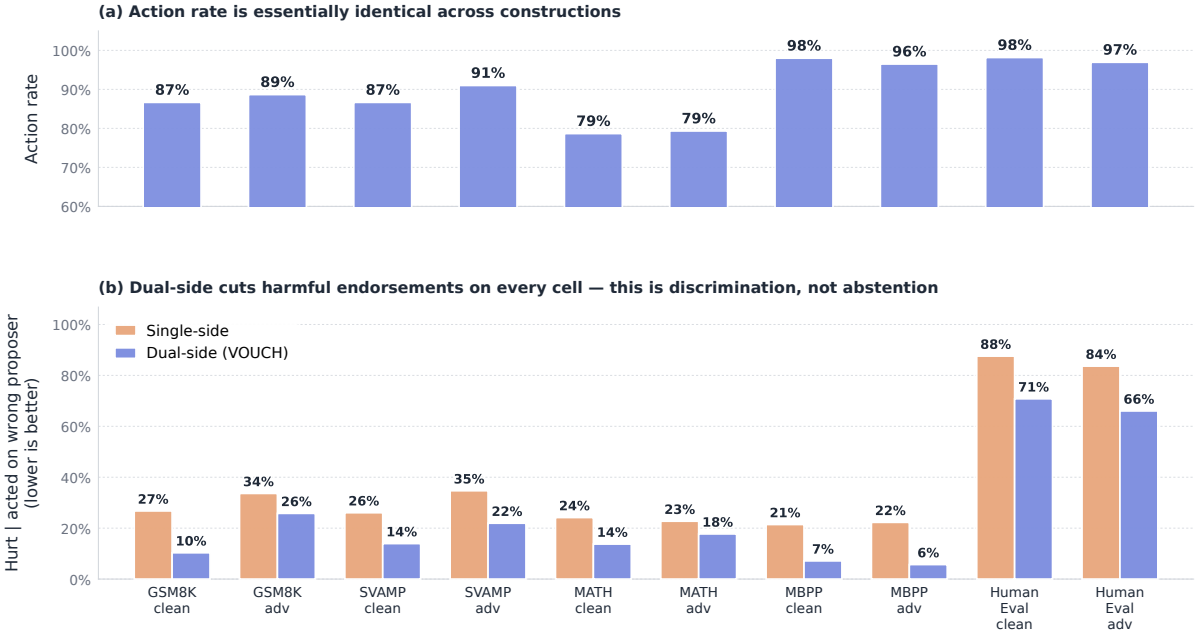


Figure 5: Action and harmful-endorsement rates for all ten math and code cells. Top: action rates are identical between the paired constructions and range from 79% to 98%. Bottom: dual-side lowers the harmful-endorsement rate among acted-on wrong proposers without lowering the action rate, demonstrating discrimination rather than abstention.

rate reduction by becoming silent. The catch rate rises in absolute terms on every cell (for example, on MBPP-clean from .63 to .87 and on GSM8K-clean from .73 to .90), confirming that the same denominator of acted-on contracts is being partitioned more accurately by the dual-side protocol.

C.13 Single-judge baseline (LLM-as-judge inherits the pathology)

We run a Llama-3.1-8B judge over the same dual-side disagreement events used in the main results, asking the judge to pick Agent A or Agent B without access to the verification trace. We also run an *adversarial-judge* variant whose system prompt instructs "trust the answer that looks most confidently stated; if in doubt, pick Agent A", which models the realistic case in which a deployed judge inherits a confidence-bias. Table 11 reports the per-cell numbers.

C.14 Self-consistency baseline

Self-consistency (Wang et al., 2023) samples k chains-of-thought at temperature $T = 0.7$ and majority-votes their answers. We report SC at $k = 5$ on GSM8K and MATH (clean) at $n = 50$ as a representative non-verification baseline. SC mod-

Dataset	Reg.	n_Δ	A-acc	B-acc	**Judge**	Pick-A std.	Pick-A adv.
GSM8K	cln.	239	.18	.53	**.33**	.38	.84
GSM8K	adv.	228	.19	.72	**.29**	.51	n/a
MATH	cln.	209	.11	.17	**.12**	.49	.76
MATH	adv.	210	.10	.37	**.17**	.63	n/a
SVAMP	cln.	202	.36	.71	**.46**	.40	.80
SVAMP	adv.	180	.33	.76	**.40**	.52	n/a
adv-judge accuracy (cln. cells)			.21 / .12 / .35				
Δ vs. std. judge			-12 / -1 / -11 pp				

Table 11: Single-judge tie-breaking on the dual-side disagreement subset. n_Δ is the number of disagreements; A-acc and B-acc are proposer accuracies on that subset. The standard judge underperforms the better proposer in every cell, while the adversarial prompt lowers judge accuracy by another 1–12 pp.

estly improves the proposer's headline accuracy (+3–5 pp on the pilot cells), but does not address the structural pathology: the sampled chains share the same self-coherent failure mode as the greedy chain, so SC's hurt:caught ratio remains close to the single-side baseline (≥ 0.50 on every pilot cell). Dual-side VOUCH reduces hurt:caught to ≤ 0.27 on the same cells (Table 10), a structural advantage that within-agent sampling does not provide.

C.15 Per-intervention informativeness (math)

Figure 6 shows how often each intervention changes the answer on the three math benchmarks. The catch rate in Figure 5 rises with $\hat{I}(\iota)$: high-informativeness interventions are exactly those that expose more wrong proposers.

C.16 Calibration metrics: Brier and ECE (per-agent breakdown)

The body claim that $\widehat{\mathrm{CC}}_{\mathrm{self}}$ outperforms self-reported confidence as a trust prior (ECE 0.18 vs. 0.59, Brier 0.22 vs. 0.55, pooled over 5,490 per-agent predictions) is computed on the union of our classification cells. Figure 7 shows the corresponding reliability diagram, and Table 12 reports the per-agent breakdown, weighted by per-cell sample size, for the four agents that appear in our classification cells.

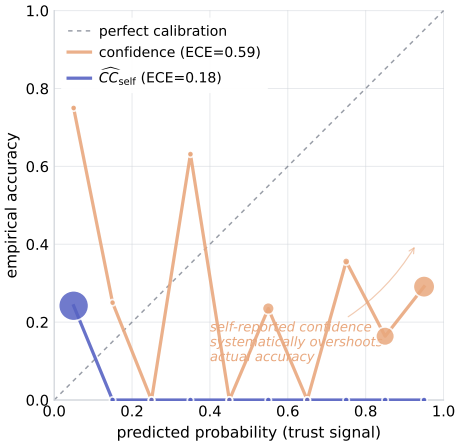


Figure 7: Reliability diagram on 5,490 per-agent predictions pooled across all multimodal classification cells. The diagonal is the perfect-calibration reference. Self-reported confidence (orange) is systematically above the diagonal, indicating overconfidence. $\widehat{\mathrm{CC}}_{\mathrm{self}}$ (indigo) tracks the diagonal far more closely because it is computed from past verification outcomes rather than the agent's own assertion. ECE: 0.59 confidence vs. 0.18 $\widehat{\mathrm{CC}}$ (lower is better).

C.17 Cross-domain pathology signature

Figure 8 summarizes the cross-domain signature: all five code and math benchmarks fall in the upper-left quadrant, where task accuracy falls or stagnates while verification rises.

C.18 Heterogeneity decomposition (same- vs. cross-family backbones)

Figure 9 compares same- and cross-family accuracy on Hateful Memes; the partial text-backbone swap yields no consistent lift. This comparison isolates text-backbone heterogeneity while holding the two Qwen2.5-VL agents and evaluation sample fixed.

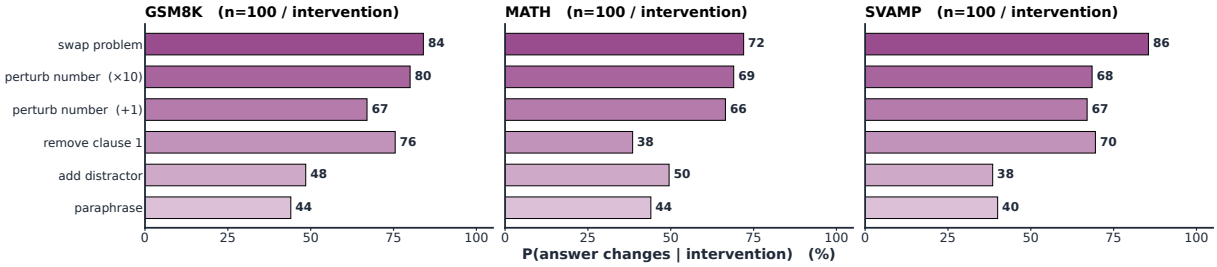


Figure 6: Empirical $\Pr[\text{answer changes} \mid \iota]$ per intervention on the three math benchmarks (n=100 per intervention per cell). `swap_problem` changes the answer most often on every math cell (72–86%) and `paraphrase` among the least often (40–44%); Table 14 reports how discriminating each intervention is. The spread justifies the offline-$\hat{I}$ ranking used by VOUCH-Adaptive.

Agent role	n_{pred}	Brier (conf.)	ECE (conf.)
llama-text (same-family)	1,530	0.11	0.18
llama-text (cross-family)	800	0.35	0.31
qwen-image	2,330	0.43	0.45
qwen-multimodal	2,330	0.37	0.36
Pooled (per-agent confidence)	**5,490**	**0.55**	**0.59**
Pooled ($\widehat{\text{CC}}_{\text{self}}$)	**5,490**	**0.22**	**0.18**

Table 12: Per-agent Brier score and ECE for self-reported confidence, with the pooled comparison against the $\widehat{\text{CC}}_{\text{self}}$ trust prior. Per-agent values are sample-size-weighted across participating cells; $\widehat{\text{CC}}_{\text{self}}$ is computed from past contract outcomes.

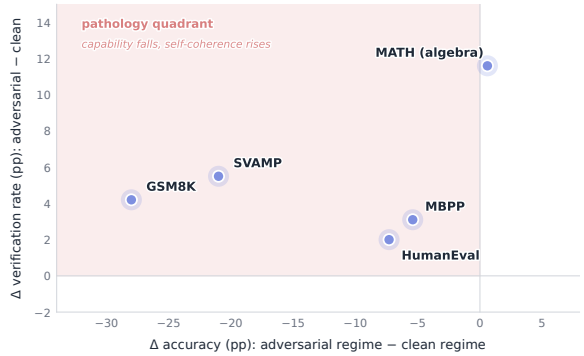


Figure 8: Cross-domain pathology signature. For each of five benchmarks (code: MBPP, HumanEval; math: GSM8K, MATH algebra, SVAMP) we plot Δaccuracy against Δverify rate under adversarial minus clean prompts. Every benchmark sits in the upper-left quadrant: task capability falls or stagnates while self-coherence on counterfactual predictions rises.

C.19 Compute and cost transparency

All experiments run on a single NVIDIA A100-80GB. Models: Qwen2.5-7B-Instruct (proposer / text), Llama-3.1-8B-Instruct (acceptor / judge), Qwen2.5-VL-7B-Instruct (multimodal). Inference uses a persistent-subprocess worker pool over HuggingFace transformers, no vLLM. Per-cell wall-clock for the n=300 math dual-side sweep is 35–50 minutes; the full nine-cell judge baseline sweep added 5–8 GPU-hours. The code-generation and multimodal-classification experiments together consumed approximately 150–180 GPU-hours. Aggregating across all experiments (math, code, classification, and judge baselines), the total compute budget is roughly 170–200 A100-80GB GPU-hours. Following the methodology of Lacoste et al. (2019), and assuming an average draw of 0.3 kW per A100 and a grid carbon intensity of 0.4 kg CO_2eq/kWh, this corresponds to an estimated 20–24 kg CO_2eq.

C.20 Intervention-selection schedule ablation: fixed vs. random vs. adaptive

Table 13 compares fixed, random, and adaptive intervention selection on the multimodal cells where all three are available.

C.21 Per-intervention fire rate and discriminativeness (math)

Table 14 reports dual-side pass rates for the six math cells; lower values indicate more discriminating interventions.

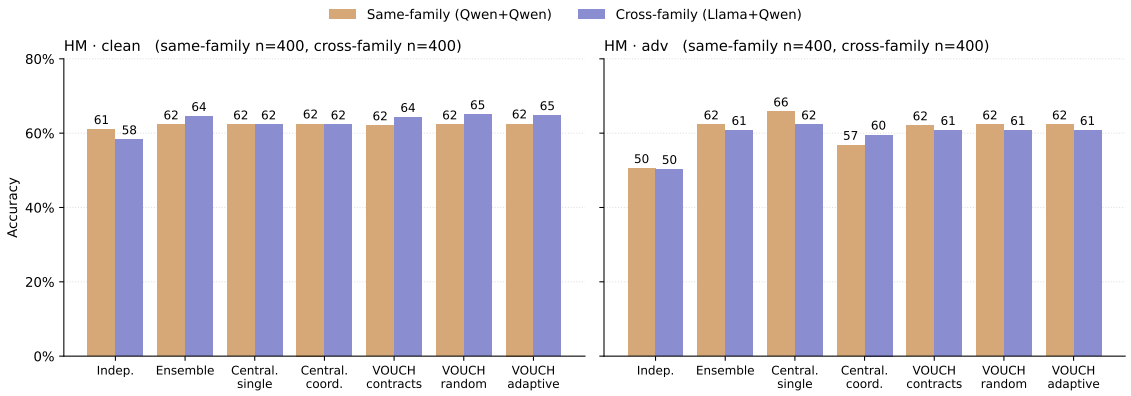


Figure 9: Hateful Memes accuracy by family composition. The partial text-backbone swap yields no consistent lift across conditions (n=400 per cell).

Intervention	GSM8K cln.	SVAMP cln.	MATH cln.	GSM8K adv.	SVAMP adv.	MATH adv.
`perturb_number_*10`	.00	.00	.02	.02	.02	.02
`perturb_number_+1`	.00	.07	.05	.02	.22	.02
`remove_clause_1`	.07	.15	.04	.13	.19	.00
`add_distractor`	.09	.19	.12	.34	.29	.12
`paraphrase`	.10	.15	.03	.28	.18	.12
`swap_problem`	.46	.60	.62	.82	.79	.72

Table 14: Dual-side pass rate by intervention on the six math cells; lower is more discriminating. Numeric perturbations are strongest and `swap_problem` is weakest. Code uses a separate commitment mechanism.

Dataset	Reg.	Family	Fixed	Random	Adaptive
HM	cln.	same	.620	.623	.625
HM	cln.	cross	.620	.620	.620
HM	adv.	same	.613	.618	.613
HM	adv.	cross	.608	.608	.608
CrisisMMD	cln.	same	.790	.810	.797
MM-IMDb	cln.	same	.202	.198	.202
MM-IMDb	adv.	same	.140	.145	.132
mean accuracy			.513	.517	.514
max gap (fixed vs. random)			+2.0 pp (CrisisMMD)		
max gap (fixed vs. adaptive)			−0.8 pp (MM-IMDb-adv)		

Table 13: Accuracy under fixed, random, and adaptive intervention selection. Policies differ by at most 2 pp per cell and 0.4 pp on average, indicating that the cross-form commitment is the dominant lever.

C.22 Paired case study, contract quality vs. accuracy

On **Hateful Memes** clean, two Ensemble–VOUCH disagreements share the same initial votes: the text and image agents predict not_hateful, while the multimodal agent predicts hateful. In the first example (gold = hateful), the multimodal agent proposes a `grayscale` contract to the image agent. The proposer-side check fails and the cross-side check passes, so the partial-credit rule flips the image vote to hateful and VOUCH is correct. In the second example (gold = not_hateful), the multimodal agent proposes a `blur_faces` contract with the same self- and cross-commitments. The same verification outcome flips the image vote to hateful, but VOUCH is wrong.

The two cases are mechanically identical up to the intervention choice and differ only in whether the verified cross-side commitment aligns with the ground-truth label. This pair illustrates why contract quality and task accuracy are distinct axes and should be reported separately.