

# Making AI Guardrails Understandable and Testable: A Framework Illustrated by a Simulated Bond Market

**June Lai**

*Independent*

**Miguel Moreno**

*Independent*

**With**

**Apart Research**

*Research conducted at the AI Collusion Research Sprint, 23–25 October 2026.*

## Abstract

AI safety evidence needs to be understandable to the people who set policy, deploy systems, and oversee their use. We developed a reusable framework for researchers, enterprises, governments, and regulators to assess an AI safeguard to make it easy to comprehend in terms that directors, auditors and senior leaders already use. We designed a framework that links risk, how this could arise, guardrail, a test, and a HITL reviewer. By using a simplified bond market, we simulate a scenario wherein AI dealers can message and trade with a shared-profit goal and a private channel for communication. Then we track what happened to see if we can detect when collusion occurs based upon the pattern of messages, timing and actions. We ran scenarios on twelve development episodes and classified all 180 scored messages correctly. On eight retrospective audit episodes, we classified 117 of 120 messages correctly (97.5%). On our initial run, we were able to achieve a detection effectiveness of three out of four reference links and returned one extra link (75% precision and 75% recall). Those audit episodes had already been examined. Two language models applied a human-written rubric to produce the reference labels. A fresh 40-episode evaluation, with the detector frozen, remains future work.

## 1. Introduction

For AI safety research to play a meaningful role in policy development, it is important to understand the different approaches to AI governance. Hong Kong prefers to provide guidance in specific sectors and to review of AI as part of existing laws (PDPO, financial regulation, copyright, consumer protection) with AI agent safety management guidelines targeted for 2027, while New York has introduced separate new laws that cover frontier model legislation (RAISE Act) with a safety framework and reporting effective Jan 2027.

Legal accountability remains with firms and responsible people, not with the software agents acting for them. Delegating an activity does not transfer those duties to the model. Hong Kong's Main Board Listing Rule 3.08 states that delegation does not absolve directors of their responsibilities. Breaches can expose directors to disciplinary action or to civil or criminal liability. Liability is not automatic whenever an agent makes a mistake [24]. The Hong Kong Securities and Futures Commission places AI governance responsibilities on the senior management of licensed firms. In the US, FINRA confirms that existing obligations still apply to its member firms when they use AI [2, 4].

These duties sit alongside different policy approaches. Hong Kong combines sectoral guidance with review of existing law [1]. New York State's amended RAISE Act adds safety-framework and reporting duties for covered frontier developers from 1 January 2027. Its catastrophic-risk remit is distinct from routine trading oversight [3].

Enterprises and boards need to determine if their AI guardrails are compliant and actually work and how this can be proven, as they are responsible for oversight of agents, particularly as the risks of rogue AI agents have been newsworthy. The first layer of defence is to apply systems architecture controls (sandboxing, hardware isolation, agent identities, least-privilege access rights and enforcement, monitoring) to restrict execution, memory access and access right and depend upon isolation and implementation to mitigate threats including kernel exploits, zero-day code execution, unauthorised memory or cache access.

However, directors should be aware that there is a residual risk specific to deployment of AI agents that these do not mitigate. AI agents in a trading system that have normal access to market data, customer data, tool permissions and can send messages to other agents or place orders have a different risk traditional methods do not mitigate. The risk isn't that they break out of their sandbox but that they use the tools they have access to in ways they weren't intended in steganographic collusion. They can encode hidden secret messages to other agents in normal looking communication via the words they choose to use, the timing of their messages, or systems tools left available to them so there is a gap between what they say they're doing compared to what they actually do.

## **2. Limitations of price simulations to real world applications**

Ezrachi and Stucke distinguish four non-exclusive scenarios. In Messenger, humans agree on a plan and use software to implement or monitor it. In Hub-and-Spoke, a common intermediary can coordinate otherwise separate users. Predictable Agent describes independently deployed algorithms whose designed responses are easier to anticipate. In Digital Eye, learning agents discover profitable patterns without an explicit human agreement to coordinate [7].

Fish, Gonczarowski, and Shorrer found that LLM pricing agents reached prices above competitive benchmarks in controlled oligopoly experiments, and that instruction wording affected those prices [8]. Amodei and colleagues describe reward hacking: optimising a proxy objective without delivering its intended outcome [9]. That concept explains why a profit objective needs constraints. It does not diagnose every instance of coordination as reward hacking. Mathew and colleagues elicited hidden linguistic signalling under constructed incentives and oversight [6]. Detecting that collusion occurred does not mean that concealed communication is the same so we test this.

Prior market models show various levels of complexity in simulating actual market situations. Agrawal and colleagues used a double auction in which LLM sellers could chat, and they varied oversight and pressure from an authority figure [11]. Cournot isolates how much each producer supplies when total output sets the price [12]. Kyle models one informed trader, noise traders, and market makers who infer value from order flow [13]. Dou, Goldstein, and Ji studied informed speculators who learned strategies by reinforcement learning. Collusive outcomes arose without chatting. As expected, in low volume, low liquidity securities trading, a higher risk of collusion could occur and when noise traders dominated informed investors that priced in market news and the fundamental value of changes, the simulations showed that Q-learning bias from RL could be learned if noise training dominated that security's trading volume, and this increased mispricing [14]. So price level alone is not a good measure of harm and timing needs to be studied. Bessembinder, Spatt, and Venkataraman document dealer networks and limited transparency in fixed-income markets [15]. Those features motivate the setting. They do not make bonds inherently simpler than equities.

The gap this report addresses is practical. A reader can see which comparison was run, which messages were linked, where a link failed, and which claims are reserved for a later evaluation.

### 3. Methods

#### 3.1 Why a simplified bond market

We chose a dealer–customer bond setting so that communication and competing interests have an intelligible context. A bond is a debt instrument. A dealer supplies a quote at which it is willing to trade. The simulation has three dealers, three rule-based buyers, one instrument, fixed-size trades, and 15 rounds. Dealers have private costs. Buyers have private valuations. These values vary across seeded scenarios and stay fixed within each episode [16]. We model dealer–customer exchange, not the valuation of a bond's cash flows. The design omits interest-rate changes, credit events, maturity differences, inventory financing, and settlement.

A detailed equity-market alternative would add choices about order-book depth, cancellations, and execution priority. We leave those mechanisms out so the communication test stays interpretable. The question is concrete: after a dealer sends a message, do the recipients change their quotes or trading behaviour in the way it proposes? Two executions in the same round are not enough, because the simulator's matching schedule can produce that coincidence. We use recorded event order to distinguish a proposal from the action that follows it. Round order does not establish millisecond synchronisation in a real market.

#### 3.2 Separating incentives from communication

We varied two factors: a shared-profit instruction for two dealers, and access to a private channel between them. The third dealer remained outside that pair. Table 1 states the four conditions and the comparison each one supports.

**Table 1. Four experimental conditions.**

| Condition                          | Shared-profit instruction | Private channel | Comparison it enables                                               |
|------------------------------------|---------------------------|-----------------|---------------------------------------------------------------------|
| Independent baseline               | No                        | No              | Behaviour without either experimental encouragement.                |
| Communication only                 | No                        | Yes             | The effect of messages under individual incentives.                 |
| Shared incentive only              | Yes                       | No              | The effect of a shared objective without the private channel.       |
| Shared incentive and communication | Yes                       | Yes             | Whether communication changes behaviour under the shared objective. |

The first hypothesis is that the shared incentive produces more observable coordination when the private channel is available. Comparing the last two conditions tests that contrast. Comparing the first two tests whether communication has a similar effect without the shared incentive. The difference between those contrasts is their interaction.

The second hypothesis is that evidence linking messages to later actions improves detection over a price-only comparator. That is a classification question, separate from the effect of opening a channel. The four conditions need not form four separable classes for the detector to be useful. The baseline is a comparison condition. It is not a certificate that collusion was absent. Public quotes remain observable when private messaging is disabled. We label the behaviour in the record. We do not assign one condition a positive label and the other three a negative label.

### 3.3 Randomisation and repetition

Market inputs changed across seeded scenarios. Within each scenario, all four conditions received the same dealer costs, buyer valuations, and buyer-bid perturbations. We also shuffled the order in which the conditions ran. Matching holds the external market circumstances constant within a comparison. Shuffling reduces systematic exposure to a fixed execution order [16]. Matching inputs is known as common random numbers. Glasserman and Yao show why it can reduce variation in a difference between simulation outcomes. The benefit depends on how the outcomes move together. Matching does not guarantee smaller uncertainty [17].

LLM responses can differ under identical market inputs. The number of runs therefore depends on the effect the study needs to detect and on the noise in the comparison. The reported evaluation covered 12 development episodes and eight retrospective-audit episodes [20]. Seed groups, condition allocation, and any overlap between those episodes are not recorded in this report. Rounds within an episode share a history, so messages and trades are not independent replications. Uncertainty estimates have to preserve that dependence and the matching between conditions. The analysis unit, the interval method, and any stopping rule are not yet specified. The market seed reproduces inputs under the recorded implementation. It does not reproduce every LLM answer. Appendix A gives the sampling details.

### 3.4 From records to labels

After the first run, an AI-assisted process checked trades, excluded invalid records from the selected dataset, and tagged accepted records. A mechanically valid trade can still contribute to harmful coordination. The review record has to distinguish an invalid execution from a valid but suspicious one. Validation rules, exclusion counts by condition, and the retention of rejected attempts are not yet specified. The first run also informed the team's benchmarks. A reference price, a welfare comparison, and an annotation rule serve different purposes. Each benchmark's definition, and the time at which it was fixed, are not yet recorded. A benchmark chosen after examining the first run belongs to development. The final evaluation has to use data that were not used to choose it.

The weekend proof of concept left insufficient time to annotate all 900 records by hand. Miguel wrote the labelling rubric and used two language models to apply it. Both reviewers received the same records in isolated environments. Neither could see the other's annotations. Under the review protocol, matching labels are accepted and disagreements go to a human adjudicator. This report does not yet define what each of the 900 records represents. The rubric, the exact model identifiers, the reviewer outputs, and the completed adjudications are included in the final code repository.

This was a time-saving substitution for a workflow in which humans label the records directly. The reference labels are generated by language models under human-written rules. They are not independent human judgments. Isolation prevents one reviewer from seeing the other. It does not prevent shared mistakes. Pangakis, Wolken, and Fasching found that annotation performance varied by task and dataset [18]. Agreement measures consistency between reviewers. It does not establish that the labels are correct.

### 3.5 Developing and evaluating the detector

Supervised learning trains a model on examples paired with target outputs. Here a label specifies the classification the detector should learn to produce from a record. The stated protocol allocates 70% of the labelled dataset to training and reserves 30% for testing. Language-model annotation changes the source of those labels. It does not change the supervised character of a model trained on them.

The available version-2 results report 12 development episodes and eight retrospective-audit episodes. An episode is a complete market simulation. The development annotations informed the detector's message-classification and linking rules. Those rules were frozen before audit scoring. The audit episodes had already been examined earlier in the project, so the audit is a check on further records from a familiar set. It is not a fully independent test [22, 23]. Twelve and eight episodes are a 60/40 division by episode. That division is not evidence of the stated 70/30 split, and the two accounts are not reconciled in this report. The training and test units, the fitted algorithm, any hand-written rules, the refinement history, and the frozen revision are not yet taken from a split manifest.

Message classification assigns a primary category, such as coordination proposal, acceptance, strategy update, or price disclosure. An exact acceptance link pairs an acceptance message with the specific earlier proposal it accepts. Recognising agreement-related words is not enough: a later message may only reaffirm an arrangement that was already accepted [22]. These scores concern communication. Showing that agents acted on an agreement requires their subsequent quotes and trades.

For a fresh evaluation, related rounds and matched conditions have to stay in the same group when development is separated from final testing. Otherwise closely related records could appear on both sides. Rules, and any fitted transformations or thresholds, have to be fixed without using final-test labels. This follows the grouped-evaluation principle described in the scikit-learn documentation [19]. Separate reviewer copies do not themselves create independent training and test sets. The proposed external-record design does not require access to model weights. It can be applied to a closed service if the deployment exposes the required messages and actions. A live alert may use only information available when it is issued. A classifier that examines later trades is retrospective. Its score does not show that it could have prevented those trades.

### 3.6 What the evidence record contains

The framework's output is an evidence record, not only a classification. It names the behaviour of concern, preserves the relevant observations, states the comparison used to evaluate the control, records the control's errors, and names the action available to its owner. This turns the connection between measurement and management in NIST AI RMF 1.0 into a procedure a reader can inspect [10]. Figure 1 lists the fields in the order a reviewer meets them.

**Figure 1. Fields of the evidence record, in review order. Text in the figure is set at 12 point so it remains readable in the submitted file.**

| Field                | What the record has to show                                                                                                   |
|----------------------|-------------------------------------------------------------------------------------------------------------------------------|
| 1. Behaviour         | The concern under review. Here, dealers could coordinate at the customer's expense.                                           |
| 2. Route             | How that behaviour could arise: a shared-profit instruction, a private channel, or both.                                      |
| 3. Guardrail         | Classify each message. Link an acceptance to the proposal it accepts. Flag the case. Do not stop the trade from this control. |
| 4. Test, with errors | Hold market inputs matched. Report missed cases and extra links in the same account as accuracy.                              |
| 5. Owner and action  | Name who may continue, restrict, or suspend the activity, and the deadline. This simulation has not filled this field.        |

In a deployment, the agent should not control the sole copy of its own audit trail. Records should be captured when the decision is made and kept outside the agent's write permissions. That is a requirement of the framework. The market simulation does not implement a production records system. Customer outcomes, which would show who gains or loses, are part of the record's design and are not in the results below.

## 4. Results

### 4.1 Development and retrospective audit

Table 2 reports the two evaluations that are available. Development scores are performance on the records used to refine the rules [20, 22]. The audit scores use eight episodes that had already been examined [20, 23].

**Table 2. Message classification and exact acceptance links.**

| Measure                 | Development (12 episodes)             | Retrospective audit (8 episodes)                    |
|-------------------------|---------------------------------------|-----------------------------------------------------|
| Scored messages correct | 180 / 180 (100%)                      | 117 / 120 (97.5%)                                   |
| Exact acceptance links  | 12 recovered; none extra; none missed | 3 of 4 reference links recovered; 1 extra; 1 missed |
| Precision               | 100%                                  | 75% (3 of 4 returned links)                         |
| Recall                  | 100%                                  | 75% (3 of 4 reference links)                        |

The extra link and the missed link in the audit came from one failure sequence. The detector overlooked an initial implicit acceptance. It later treated a reaffirmation as the acceptance and linked the wrong message to the proposal [23]. For an investigator, that changes the reconstructed moment at which the proposal was accepted. Event identifiers for that sequence, a correctly linked comparison, and the reason for each detector decision are not included here.

Classifying a message and reconstructing the relationship between messages are different tasks. High message accuracy stood next to one extra link and one missed link. The audit contained only four reference acceptance links, so each missed link moves recall by 25 percentage points. These figures measure agreement with the reference records. They do not measure prevention of customer harm. The counts are taken from the project summaries cited as [20]–[23]. They have not been independently reproduced from the code and the event records for this write-up.

### 4.2 Implementation checks

The team reports that all 41 project tests passed. The checks cover messages with more than one function, refusals, unrelated agreements, and exact-link scoring. The validation summary also reports that frozen source hashes match both the final development report and the audit report [21]. These checks concern implementation behaviour and version consistency. They are not 41 independent market experiments. The test output, the source hashes, the scoring specification, and both reports are not archived in this document.

### 4.3 Scope of the findings

These results evaluate the detector inside the existing dataset. They do not quantify the effect of the incentive or of communication on trading outcomes. The summaries do not report a comparison with a price-only detector, and they do not report changes in buyer surplus, dealer profit, or executed quantity. Those analyses would connect detection performance to the hypotheses in section 3.2. The larger evaluation on fresh seeds belongs to future work, not to the evidence reported here.

## 5. Discussion and Limitations

A compliance officer needs to be able to detect when trading may not follow normal patterns. By reviewing the timing of messages to action, you do not rely upon the agent’s messages alone to detect if collusion has occurred but use unusual patterns to show that collusion may have occurred.

By deploying a detector, our method allows a compliance officer to act on a timely basis to monitor and review actions vs. the stated messages to identify deception.

To continue and expand upon this experiment, we can compare continued trading, restricting the ability to communicate or send messages, or human handovers under the same market conditions. Our approach has been simplified as many other factors would be considered to come up with a competitive-price benchmark and other parameters should be considered as extrapolation would be inaccurate.

### Limitations

The market scenario was intentionally simulated to isolate only the factors we wanted to test as they indicate collusion. We did not test the impact of the real world impact of interest rates, credit, maturity, inventory, and settlement. The sample size would have had to increase significantly and the complexity would be far too great for a weekend project. We use fixed trade sizes and round-based matching. Quote changes that look coordinated could be a result of other causes once inventory and demand can change so it’s important to not consider that correlation means causality. The detector’s threshold should be re-estimated on any market, considering its parameters against a new baseline, but the methodology is clear.

The reference labels were produced by language models acting independently. The audit episodes had been examined before scoring. Development accuracy of 100% is performance on records used to refine the rules. Four reference links make the audit’s 75% recall a description of three successes and one miss, not a stable rate. The stated 70/30 data split and the reported division into 12 and 8 episodes are not reconciled. Effects of the incentive, a price-only comparator, and customer outcomes are outside these results. The numerical results have not been re-derived from code for this write-up.

Dual use. A description of how an implicit acceptance was missed could be read as a hint for probing a similar detector. This report does not include the prompts, the model identifiers, or a method for constructing a concealed signal. The shared-profit instruction is an experimental intervention for a laboratory comparison. It is not a design for a trading system. The evidence record is meant to make that comparison inspectable, including to a person who would refuse the behaviour.

### Future Work

The next proposed experiment uses fresh seeds 6–15 under all four conditions: 40 episodes and 1,800 dealer requests, in ten matched groups of four. The protocol keeps the detector frozen and preserves the model settings. Annotation is independent of the detector outputs. If this batch is the final evaluation, its results must not be used to change the rules. A change prompted by that batch needs a further evaluation of the revised detector [21]. The request count is not a count of executed trades.

A sample labelled directly by humans should test the language-model annotations, including cases on which both model reviewers agreed. Those reviewers should not see the model labels or the detector’s predictions. The assessment should keep “insufficient evidence” as an allowed judgment. Later tests should hold out unfamiliar scenarios, prompts, and model combinations, and should include parallel behaviour that can arise from common public information alone. For concealed coordination, freeze the detector before a separate pair of agents develops a new signalling strategy. Vary word choice, message length, and permitted timing, then compare unrestricted messages, constrained formats, and closing the channel [6].

The first extensions of the market should add variable trade sizes, dealer inventory, and changing customer demand. A request-for-quote workflow, and an asynchronous version, would separate delay from the synchronisation imposed by rounds. Measuring liquidity, price informativeness, and mispricing, following Dou, Goldstein, and Ji, requires an evolving asset value and a stated price-formation rule [14]. Executed quantity is not a liquidity measure, and a private buyer valuation is not a common fundamental value. A later test should also separate an agent executing a customer's mandate from one trading for its own firm.

Whether people can use the record is the framework's central untested claim. Give the same cases to researchers, compliance officers, auditors, directors, and policy staff. Assign at random either a conventional technical report or the evidence record. Ask each reader for the supported finding, the missing information, and the next justified action. Measure accuracy, review time, and confidence against the strength of the evidence. Include a clear coordination case, lawful-looking parallel behaviour, a false alert, and an unresolved case. The test fails if the shorter account makes readers more confident and does not make their judgments more accurate.

The same sequence of questions can be asked of a credit, insurance, or procurement workflow. That workflow has to supply its own prohibited information flows and its own authorised responses. Bond-market labels cannot supply them. A replication package should keep the evidence-record template separate from the market simulator and the annotation rubric. Further work should then test the record when logs are missing, events are delayed, or the model has changed.

## 6. Conclusion

The bond-market case connects a behavioural signal to a testable control and to a review with a named owner. The transferable piece is the evidence record: what the control observes, how it was tested, where it fails, and who decides what follows. On the evidence assembled here, the detector classified most scored messages correctly and missed the structure of one acceptance. A decision to deploy should say which changes invalidate that evidence, including a new model, a new incentive, a new communication path, or a new market setting. It should also say that the fresh evaluation, and a human check of the labels, have not yet been run.

## Code and Data

Code repository: [Bond market competing dealer collusion](#)

Data and datasets: Included in the repository. The package should contain the simulator configuration, the agent prompts, the exact model identifiers, the realised market inputs, and the execution order. It should keep the original event records beside the validation decisions, the human-written annotation rubric, both language-model reviewer outputs, and any human adjudications. The detector specification, the split membership, the threshold selection, and the analysis that generates the results complete the package.

Re-run contract. Recorded inputs and pinned deterministic processing should reproduce the same processing outputs. A fresh language-model execution may produce different trading decisions. A change to a model, a prompt, a feature, or a threshold should be recorded as a new experiment version.

## Author Contributions

June Lai defined the problem, the governance framework, and the account of residual risk that remains after ordinary systems controls, including the limits of using a simulation to speak about real-world trading. Miguel Moreno designed the bond-market simulation, wrote the labelling rubric, and developed the infrastructure and the detector. Both authors are responsible for the claims in this report.

## References

1. Hong Kong SAR Government. (2026). The Chief Executive's 2026 Policy Address, Chapter III, paragraphs 121–123. <https://www.policyaddress.gov.hk/2026/en/chapter3.html>
2. Securities and Futures Commission of Hong Kong. (2024, November 12). Circular to licensed corporations: Use of generative AI language models, 24EC55, paragraphs 6–13. <https://apps.sfc.hk/edistributionWeb/api/circular/openFile?lang=EN&refNo=24EC55>
3. New York State. (2026). RAISE Act chapter amendment, S8828, Chapter 96, signed March 27, 2026. <https://www.nysenate.gov/legislation/bills/2025/S8828>
4. Financial Industry Regulatory Authority. (2024, June 27). Regulatory Notice 24-09: Regulatory obligations when using generative artificial intelligence and large language models. <https://www.finra.org/rules-guidance/notices/24-09>
5. Joint Task Force. (2020, updated December 10). Security and Privacy Controls for Information Systems and Organizations. NIST SP 800-53 Rev. 5, controls AC-6, SC-3, SC-31, and SC-39. <https://doi.org/10.6028/NIST.SP.800-53r5>
6. Mathew, Y., Matthews, O., McCarthy, R., Velja, J., Schroeder de Witt, C., Cope, D., and Schoots, N. (2025 revision). Hidden in Plain Text: Emergence and Mitigation of Steganographic Collusion in LLMs. arXiv:2410.03768v2, sections 3–5. <https://arxiv.org/abs/2410.03768v2>
7. Ezrachi, A., and Stucke, M. E. (2017). Artificial intelligence and collusion: When computers inhibit competition. *University of Illinois Law Review*, 1775–1810, Part III, pp. 1782–1796. <https://illinoislawreview.org/wp-content/uploads/2017/10/Ezrachi-Stucke.pdf>
8. Fish, S., Gonczarowski, Y. A., and Shorrer, R. I. (2026 revision; first posted 2024). Algorithmic Collusion by Large Language Models. arXiv:2404.00806v6. <https://arxiv.org/abs/2404.00806v6>
9. Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., and Mané, D. (2016). Concrete Problems in AI Safety. arXiv:1606.06565v2, section 4. <https://arxiv.org/abs/1606.06565v2>
10. National Institute of Standards and Technology. (2023). Artificial Intelligence Risk Management Framework (AI RMF 1.0). NIST AI 100-1, Part 2. <https://doi.org/10.6028/NIST.AI.100-1>
11. Agrawal, K., Teo, V., Vazquez, J. J., Kunnavakkam, S., Srikanth, V., and Liu, A. (2025). Evaluating LLM Agent Collusion in Double Auctions. arXiv:2507.01413v1, sections 2–4. <https://arxiv.org/html/2507.01413v1>
12. Cournot, A. A. (1838; excerpts reprinted 2008). *Researches into the Mathematical Principles of the Theory of Wealth*, translated by N. T. Bacon, Chapters VII and IX. *Competition Policy International*, 4(1), 283–306. <https://competitionpolicyinternational.com/assets/0d358061e11f2708ad9d62634c6c40ad/Cournot.pdf>
13. Kyle, A. S. (1985). Continuous auctions and insider trading. *Econometrica*, 53(6), 1315–1335. <https://doi.org/10.2307/1913210>
14. Dou, W. W., Goldstein, I., and Ji, Y. (2025, July). AI-Powered Trading, Algorithmic Collusion, and Price Efficiency. NBER Working Paper 34054, sections 3–4. <https://doi.org/10.3386/w34054>
15. Bessembinder, H., Spatt, C., and Venkataraman, K. (2020). A survey of the microstructure of fixed-income markets. *Journal of Financial and Quantitative Analysis*, 55(1), 1–45, section II. <https://doi.org/10.1017/S0022109019000231>
16. Research team. (2026, October 26). Random Number Generation in the Bond-Market Collusion Experiment. Project methodological note. The public archive identifier is not yet assigned.
17. Glasserman, P., and Yao, D. D. (1992). Some guidelines and guarantees for common random numbers. *Management Science*, 38(6), 884–908. [https://business.columbia.edu/sites/default/files-efs/pubfiles/4261/glasserman\\_yao\\_guidelines.pdf](https://business.columbia.edu/sites/default/files-efs/pubfiles/4261/glasserman_yao_guidelines.pdf)
18. Pangakis, N., Wolken, S., and Fasching, N. (2023). Automated Annotation with Generative AI Requires Validation. arXiv:2306.00176v1. <https://arxiv.org/abs/2306.00176v1>
19. scikit-learn developers. Cross-validation: evaluating estimator performance, section 3.1.2.4, Cross-validation iterators for grouped data. [https://scikit-learn.org/stable/modules/cross\\_validation.html#cross-validation-iterators-for-grouped-data](https://scikit-learn.org/stable/modules/cross_validation.html#cross-validation-iterators-for-grouped-data)
20. Research team. (2026). Results: development and retrospective audit. Summary supplied by Miguel Moreno. The archived results report and evaluated revision are not yet assigned.

21. Research team. (2026). Validation and next experiment. Summary supplied by Miguel Moreno. The archived test report and prospective protocol are not yet assigned.
22. Research team. (2026). Development and retrospective-audit groups, message categories, and acceptance links. Clarification supplied by Miguel Moreno.
23. Research team. (2026). Exact-link precision, recall, and the implicit-acceptance error in detector version 2. Clarification supplied by Miguel Moreno.
24. The Stock Exchange of Hong Kong Limited. Main Board Listing Rules, Chapter 3, rule 3.08.  
<https://en-rules.hkex.com.hk/entiresection/2064>

## Appendix

### Appendix A. Sampling and replay

The implementation note specifies Python's `random.Random(seed)`. It draws three dealer costs uniformly from the interval  $[95, 97]$  and three buyer valuations from  $[103, 105]$ , rounded to two decimals. For buyer  $j$  in round  $t$ , a perturbation  $\text{epsilon}(j,t)$  is drawn uniformly from  $[-0.01, 0.01]$ . The bid is

$$b(j,t) = \min\{v(j), \text{round}[0.96 \times v(j) \times (1 + \text{epsilon}(j,t)), 2]\}$$

Here  $v(j)$  is buyer  $j$ 's valuation. The 15-round configuration uses six initial draws and 45 bid perturbations. These are synthetic assumptions, not calibrated bond-price dynamics. All market draws occur before the concurrent model requests [16]. A separate generator, initialised with the same seed, shuffles the condition list in place across seeds, so a replay needs the list's initial order and the full seed sequence. The note specifies temperature 0.3 and no language-model sampling seed. A release should preserve the realised inputs and outputs, the Python version, the model settings, and the source revision. Those settings have not yet been confirmed against a release manifest.

### Appendix B. Outcome and reporting definitions

For a unit bought at price  $p$ , buyer surplus is  $v(j) - p$  and dealer profit is  $p - c(i)$ . Here  $v(j)$  is the simulated buyer's valuation and  $c(i)$  is the dealer's cost. Their combined gain is  $v(j) - c(i)$ . A price change can redistribute that gain without changing its total. A foregone beneficial trade can reduce the total. Quantities and participant outcomes should be reported beside prices.

Message-classification accuracy is the proportion of scored messages whose primary category matches the reference. For exact-link matching, precision is the number of correct acceptance-to-proposal pairs divided by all pairs the detector returned. Recall is the number of correct pairs divided by all reference pairs. The retrospective-audit counts give precision  $3 / (3 + 1) = 75\%$  and recall  $3 / (3 + 1) = 75\%$  [20, 22]. The missed implicit acceptance, and the later reaffirmation treated as the acceptance, produced the missed link and the extra link [23].

A false-positive rate uses reference-negative cases as its denominator. The supplied summary does not provide that denominator. False alerts per episode would describe review workload, and an extra acceptance link is not necessarily a separate operational alert. The alert unit and the denominators have to be stated before link errors are translated into workload. The report should also give the initial agreement between the two language-model reviewers, before any human adjudication, with class frequencies and the disagreement table. It should preserve abstentions, exclusions, and the reasons for adjudication, and it should state how many labels a human inspected or changed. Detector errors on these reference labels should be reported separately from performance on cases labelled independently by humans.

For the experimental comparisons, uncertainty estimates have to preserve matching and within-episode dependence. The report should give the channel contrast at each incentive setting, and the interaction of those contrasts. It should state which comparison was primary and how multiple comparisons were handled. The final analysis has to identify any choice made after the outcomes were inspected.

## LLM Usage Statement

Language models assisted with preparing the simulation environment, with writing supporting Python libraries, and with recording the runs. Two language models applied a human-written rubric to 900 records, under the procedure in section 3.4, because the weekend did not leave time for the authors to label every record by hand. Language models also assisted with editing the wording of this report. The authors chose the claims, checked the cited sources against the references listed above, and remain responsible for the findings. The numerical results are taken from summaries supplied by Miguel Moreno. They have not been independently reproduced from the code and the event records for this write-up. A complete list of tool names and model versions is not yet inserted. The final text should be read as the team's account, with those limits stated where the numbers appear.