UNIVERSITÉ DU QUÉBEC EN OUTAOUAIS

DÉPARTEMENT DES SCIENCES ADMINISTRATIVES

WORKING PAPER NO. 27

The Invisible Hand of Machines

Price Discovery in Markets Populated by LLM Agents

Simon-Pierre Boucher

Département des sciences administratives
Université du Québec en Outaouais

simon-pierre.boucher@uqo.ca

Gatineau – Pavillon Alexandre-Taché
283, boulevard Alexandre-Taché
Gatineau, Québec, Canada J9A 1L8

October 2026

Version 1.0

Abstract

Abstract to be written.

Keywords: price discovery, large language models, agent-based markets, limit order book, herding, liquidity spirals, randomized simulation experiments

JEL Classification: G14, G12, D53, C63, C15, D83

1. Related literature

The paper draws on six literatures: agent-based models of financial markets; double-auction microstructure and the measurement of price discovery; information aggregation with correlated signals, herding and feedback; liquidity spirals, fire sales and contagion; algorithmic trading, monoculture and algorithmic collusion; and, closest to our question, LLM agents in economic experiments and market simulations. We close with what this paper adds and where it overlaps with existing work.

1.1. Agent-based and heterogeneous-agent models

Our rule-based population descends from agent-based computational finance, in which interacting heterogeneous rules reproduce fat tails, volatility clustering and bubbles (LeBaron, 2006). In the Santa Fe artificial stock market the speed of belief updating governs whether prices track fundamentals or display excess volatility and technical-trading regimes (LeBaron et al., 1999); in adaptive belief systems, switching between predictors produces increasingly complex dynamics as the intensity of choice rises (Brock and Hommes, 1997), and the share of trend followers determines whether the fundamental price is stable (Brock and Hommes, 1998). Noise traders switching between optimistic, pessimistic and fundamentalist states generate scaling laws and clustered volatility endogenously (Lux and Marchesi, 1999, 2000), contagious optimism yields bubbles and crashes (Lux, 1995), and recruitment dynamics generate persistent herding without any informational reason (Kirman, 1993). These are the channels through which a correlated LLM order flow can be amplified by a classical population. Closer to our exchange, Chiarella and Iori (2002) reproduce order-book regularities with fundamentalist, chartist and noise traders in a double auction, and Chiarella et al. (2009) show that heterogeneous rules in a limit-order book generate fat tails, volatility clustering and order-flow autocorrelation, with the chartist share driving excess volatility. Leverage in an agent-based model produces fat tails and clustered volatility through synchronized margin-driven selling (Thurner et al., 2012), the closest antecedent of our leverage treatment, and multi-agent reinforcement-learning markets reproduce the same facts (Lussange et al., 2021). Following the surveys of Hommes (2006) and Hommes (2021), we treat an LLM policy as one more expectation type; following Farmer and Foley (2009) and Axtell and Farmer (2025), we use the agent-based method for counterfactuals that cannot be solved analytically and address its open frontiers—validation, calibration, scale—with matched seeds and a power analysis. Validation targets are the moment set of Franke and Westerhoff (2012) and the stylized facts of heavy tails (Mandelbrot, 1963; Cont, 2001), conditional heteroskedasticity (Engle, 1982; Bollerslev, 1986) and the long memory of absolute returns (Ding et al., 1993).

1.2. Double auctions, zero intelligence and price-discovery measurement

Smith (1962) showed that double auctions with induced values converge to competitive equilibrium with few traders, and Smith et al. (1988) that experimental asset markets with known dividend streams nevertheless produce bubbles and crashes; our dividend-paying asset and bubble metrics follow that design. Zero-intelligence traders constrained by budgets reach near-full allocative efficiency, so market structure rather than rationality drives efficiency (Gode and Sunder, 1993; Ladley, 2012), and a zero-intelligence model of the double auction predicts spread and price impact from order-flow rates alone (Farmer et al., 2005; Smith et al., 2003). This fixes the benchmark against which any effect of "intelligence" must be read: changing the composition of a population can move liquidity merely by changing order-flow rates, which is why we report realized order and volume shares alongside the assigned treatment. The book is parametrized from the empirical regularities in Bouchaud et al. (2002), Cont et al. (2010), Cont et al. (2014), Gould et al. (2013) and Bouchaud et al. (2018); the event-driven exchange with per-agent latency follows ABIDES (Byrd et al., 2020) and the realism checklist is that of Vyetrenko et al. (2020).

The outcomes are standard measures of price discovery. The price-impact coefficient $\lambda = \mathrm{Cov}(v, Y)/\mathrm{Var}(Y)$ of Kyle (1985) and the adverse-selection spread of Glosten and Milgrom (1985) tie liquidity to the information in order flow; Hasbrouck (1991) measures that information by the permanent price impact in a VAR of quotes and trades, Hasbrouck (1993) decomposes the transaction price into an efficient random-walk component and a pricing error, and Hasbrouck (1995) attributes price discovery through information shares, which we adapt to LLM versus rule-based flow. Variance ratios test the random walk (Lo and MacKinlay, 1988), the effective spread follows from serial covariance (Roll, 1984), and illiquidity is absolute return per unit of volume (Amihud, 2002). Bai et al. (2016) define informativeness by how well prices forecast future fundamentals, Bond et al. (2012) argue that it is the welfare-relevant object because real decisions feed on prices, and Hendershott and Menkveld (2014) separate transitory price pressure from permanent information effects, which justifies separating the pricing error from the efficient-price variance. Because the simulator observes the latent value, these statistics are computed against the truth rather than a proxy, which is how we operationalize the efficiency notions of Fama (1970).

1.3. Correlated signals, herding and feedback

The theoretical core of the paper is the aggregation of *correlated* private signals. Competitive prices aggregate dispersed signals imperfectly because of noise supply (Hellwig, 1980) and cannot be fully efficient when information is costly (Grossman and Stiglitz, 1980). With strategic informed traders, the correlation structure of signals, not only their precision, governs trading intensity and informativeness (Foster and Viswanathan, 1996), and more correlated information leads to

faster revelation (Back et al., 2000). Information diversity across traders is complementary, so a loss of diversity reduces informativeness about some components of value (Goldstein and Yang, 2015); abundant cheap data can lower informativeness by crowding out costly deep information (Dugast and Foucault, 2018), a close parallel to shared-model crowding; public information can raise disagreement through higher-order beliefs, so shared inputs do not trivially homogenize behaviour (Kondor, 2012); and information heterogeneity among speculators generates contagion while homogeneity reduces it (Pasquariello, 2007). Our stylized model (Section 3) borrows this correlated-error structure and reinterprets the common component as the signature of a shared foundation model. Herding supplies the behavioural counterpart: sequential decision-makers who follow predecessors destroy information aggregation (Banerjee, 1992), cascades become more likely as signals become more correlated (Bikhchandani et al., 1992), reputational concerns make managers mimic one another (Scharfstein and Stein, 1990), and short-horizon speculators herd on the same information source even when it is unrelated to fundamentals (Froot et al., 1992); the buy-side imbalance statistic of Lakonishok et al. (1992) is the standard measure and we compute it for LLM cohorts. Feedback trading closes the loop, since the share of trend followers governs the stability of the fundamental price (Brock and Hommes, 1998); Proposition 4 is a mechanical version of this amplification.

1.4. Liquidity spirals, fire sales and contagion

Hypotheses H3 and H6 draw on the literature on funding constraints. Market liquidity and funding liquidity reinforce each other through margins, producing liquidity spirals (Brunnermeier and Pedersen, 2009); our margin-call mechanism follows this logic. Liquidation values fall when potential buyers are themselves constrained (Shleifer and Vishny, 1992), constrained arbitrageurs can amplify rather than correct mispricing (Shleifer and Vishny, 1997) and may under-provide liquidity (Gromb and Vayanos, 2002), forced sales push prices below fundamentals and propagate losses (Shleifer and Vishny, 2011), endogenous margins amplify the leverage cycle with leverage caps as a policy lever (Geanakoplos, 2010), and procyclical intermediary leverage amplifies price moves (Adrian and Shin, 2010). Across assets, losses propagate through reduced risk-bearing capacity (Kyle and Xiong, 2001), distressed selling induces endogenous correlation (Cont and Wagalath, 2013), fire-sale spillovers scale with common exposures and leverage (Greenwood et al., 2015), rebalancing transmits shocks between unrelated assets (Kodres and Pritsker, 2002), and dense networks are fragile to large shocks (Acemoglu et al., 2015). For measurement we use the spillover index of Diebold and Yilmaz (2009), its generalized directional form (Diebold and Yilmaz, 2012) and network interpretation (Diebold and Yılmaz, 2014), and we heed the warning of Forbes and Rigobon (2002) that correlation-based contagion tests must adjust for heteroskedasticity.

1.5. Algorithmic trading, monoculture and algorithmic collusion

The average evidence on algorithmic trading is favourable: it improves liquidity (Hendershott et al., 2011), high-frequency traders trade with permanent price changes and against transitory errors (Brogaard et al., 2014), and algorithmic strategies in foreign exchange improve price efficiency even though they are more correlated with each other than human trades (Chaboud et al., 2014)—the finding we take most seriously, since correlation among algorithms is a documented fact whose consequences depend on what they are correlated *on*. High-frequency market makers keep tight inventory control (Menkveld, 2013), a feature of our market-maker policy; Menkveld (2016) reviews the speed and adverse-selection consequences and Foucault et al. (2016) show how fast traders reacting to common news affect informativeness and liquidity. Under stress the picture darkens: during the Flash Crash, high-frequency traders did not trigger the crash but their synchronized inventory management amplified it (Kirilenko et al., 2017); order-flow toxicity rose and liquidity withdrew when flow became toxic (Easley et al., 2011); the August 2007 quant unwind showed common strategies deleveraging simultaneously (Khandani and Lo, 2011); and ultrafast extreme events proliferate in a machine ecology beyond human response time (Johnson et al., 2013). Budish et al. (2015) show that continuous trading creates a latency arms race with correlated sniping and propose frequent batch auctions, which motivates our matched-latency treatment and call-auction robustness arm.

Algorithmic monoculture names the homogenization mechanism: when many decision-makers rely on the same algorithm, outcomes can be worse even if it is individually more accurate (Kleinberg and Raghavan, 2021); foundation models carry a homogenization risk because downstream systems inherit the same failure modes (Bommasani et al., 2021), and shared models mis-serve the same individuals across deployers (Bommasani et al., 2022). Generative AI raises individual creativity but reduces collective diversity (Doshi and Hauser, 2024), standard LLM outputs show diminished diversity of thought (Park et al., 2024), and LLM assistance homogenizes human ideation (Anderson et al., 2024); conversely, ensembles of twelve LLMs match human crowd forecasting accuracy (Schoenegger et al., 2024), the diversity benefit that H4 tests in a market. Learning algorithms also coordinate without being told to: Q-learning pricing algorithms learn tacit collusion (Calvano et al., 2020; Klein, 2021), algorithmic pricing raised retail gasoline margins (Assad et al., 2024), asymmetric adoption softens competition (Brown and MacKay, 2023), LLM pricing agents reach supracompetitive prices with a sensitivity to prompt wording (Fish et al., 2024), and LLM sellers in double auctions collude more when they can communicate (Agrawal et al., 2025). In securities markets, Q-learning market makers quote markups that shrink with the number of algorithms (Colliard et al., 2026), and reinforcement-learning speculators in a Kyle-type market sustain supracompetitive profits through price-trigger strategies or homogenized learning biases, lowering informativeness and liquidity (Dou et al., 2025). Dou et al. (2025) is the nearest theoretical

neighbour of our mechanism: their channel is strategic, ours is a shared-error structure in signals, and they have no language models. The policy literature has drawn the systemic conclusion: deep learning increases uniformity because models built on the same data generate correlated predictions (Gensler and Bailey, 2020); common AI models and data sources could raise correlations in trading, lending and pricing (Financial Stability Board, 2024); AI amplifies existing vulnerabilities and makes crises faster (Danielsson and Uthemann, 2025); monoculture and correlation are among the stability concerns of an intelligent financial system (Aldasoro et al., 2025); and the AI adoption share and signal correlation raise systemic coupling convexly, with 13F evidence of tail-loss amplification (Meng and Chen, 2026). None of these provides a controlled experiment.

1.6. LLM agents in economics and markets

Simulated subjects. Horton (2023) proposed LLMs as simulated economic agents for replicating behavioural experiments at low cost; Aher et al. (2023) showed that they can simulate multiple humans with fidelity varying by model and task, and Argyle et al. (2023) that demographic backstories yield silicon samples matching sub-population responses. GPT choices score higher on rationality than humans and are stable across demographic prompts (Chen et al., 2023), chatbot choices in behavioural games fall within the human distribution (Mei et al., 2024), LLMs cooperate well but coordinate poorly in repeated games (Akata et al., 2025), and LLM-elicited expectations are extrapolative (Bybee, 2023). Generative agents with memory and reflection produce believable emergent behaviour (Park et al., 2023; Gao et al., 2023), EconAgent produces realistic macroeconomic dynamics (Li et al., 2024), and Immorlica et al. (2024) treat generative agents as economic agents with prompts as primitives (see also Manning et al., 2024). Anthis et al. (2025) conclude that LLM social simulations are promising provided that homogeneity, sycophancy and drift are addressed; homogeneity is the concern this paper turns into a treatment.

Single-agent trading and backtests. ChatGPT-derived headline sentiment predicts next-day returns (Lopez-Lira and Tang, 2023). FinMem (Yu et al., 2023), TradingGPT (Li et al., 2023) and FinAgent (Zhang et al., 2024b) evaluate one agent, or a small team with layered memory, by backtesting on historical prices. The market does not react to the agent, so nothing is learned about equilibrium prices, and such backtests are contaminated by memorization of the historical period (Lopez-Lira et al., 2025)—two reasons for a synthetic economy with an endogenous price and no history to memorize.

LLM-populated market simulators. A second generation places many LLM agents on a matching engine. ASFM finds reactions to rate and inflation shocks consistent with economic priors (Gao et al., 2024); StockAgent adds bulletin-board communication and reports that GPT- and

Gemini-driven agents differ systematically (Zhang et al., 2024a); TwinMarket scales to a social-media-connected population, reproduces bubbles and crashes, and reports that agents of one model family are more homogeneous (Yang et al., 2025); FCLAgent reproduces path-dependent patterns absent from conventional agent-based models (Hashimoto et al., 2025); StockSim is an open order-level simulator with latency and price–time matching (Papadakis et al., 2025); and trading-style switching in a year-long simulation is only partially consistent with behavioural-finance theory (Li et al., 2026). Lopez-Lira (2025) is closest in spirit: LLM agents in a persistent order book adhere to assigned strategies, generate price discovery and bubbles, and are analysed with partial-dependence-style perturbations. These papers show that LLM markets *can* produce the patterns we study; they hold the population composition fixed, do not equalize the information set of LLM agents with that of the agents they replace, and do not separate the fundamental value, the efficient price and the transaction price.

Experimental-finance replications. In Smith–Suchanek–Williams-type markets, LLM agents price near fundamentals and form only muted bubbles, with less variable strategy text than humans (Henning et al., 2025); in a call auction they exhibit disposition effects and extrapolative beliefs, and prompt interventions causally change bubble size (Ouyang and Sui, 2026); across fifteen models, behaviour ranges from fundamental pricing to human-like bubbles, mixed populations are unstable, and advanced models adapt to exploit weaker ones (Saxena et al., 2026); replications of Smith (1962) converge more slowly or not at all, with model-family differences (Struski et al., 2026); and the institution alone moves the efficiency of identical agents from 56 % to 89 % (Chupilkin, 2026). These studies ask whether LLMs behave like humans and their answers conflict. Our results do not require fidelity, only that errors be correlated within a model.

Correlated behaviour from shared inputs. The one paper that measures correlated trading from shared inputs is Song (2026): overlap in retrieved documents among retrieval-augmented trading agents raises mean pairwise action correlation from 0.04 to 0.41 and increases position volatility without raising profits. That is a backtest without market feedback, so it cannot speak to informativeness or liquidity. Cho et al. (2025) document peer conformity in LLM multi-agent systems, and Hansen and Lee (2025) find in re-created herding experiments that AI agents rely more on private signals than humans but can be prompted to herd when maximizing profit. LLM service behaviour drifts over months (Chen et al., 2024), which is why every response is cached and replayed from a frozen library.

1.7. What this paper adds

We should be candid about overlap. The hypothesis that shared models correlate trading is not ours: it is stated by Gensler and Bailey (2020), the Financial Stability Board (2024) and Danielsson and Uthemann (2025), modelled as homogenized learning biases by Dou et al. (2025), and measured in a backtest by Song (2026). Model-family homogeneity inside LLM markets has been observed by Yang et al. (2025), Zhang et al. (2024a) and Saxena et al. (2026), and open LLM order-book simulators exist (Lopez-Lira, 2025; Papadakis et al., 2025; Gao et al., 2024). Our simulator is not the contribution; the design is. First, the LLM penetration share is a randomized treatment assigned to trader slots by an independent seed stream, so that matched pairs share every other stream—fundamentals, news, endowments, latencies, policy assignment, activation, tie-breaks—and paired differences are causal effects on the same market history (Section 4). Second, we separate three prices—the latent value known to the simulator only, the public value implied by announced dividends and news, and the transaction price—so that an efficiency loss can be distinguished from transitory pressure (Hasbrouck, 1993; Hendershott and Menkveld, 2014). Third, we derive the shared-error mechanism in a Kyle–Hellwig setting with a model-specific common error and an effective number of models, then test it by intervention—prompt diversity at equal share, a two-model mix, inherited information sets, memory, communication cohorts—so that the effective number of models is manipulated rather than observed. Fourth, we report the realized dose, the share of orders and volume actually originated by LLM agents, alongside the assigned share, and run fairness designs with equal decision opportunities, equal dollar budgets and matched latency, because naive comparisons confound being an LLM with being slow and infrequent. Fifth, the leverage interaction and safeguard counterfactuals connect the LLM literature to Brunnermeier and Pedersen (2009), Thurner et al. (2012) and Kirilenko et al. (2017), with thresholds, estimators and seed counts fixed before the first confirmatory call.

2. The experimental economy

This section describes the simulated economy exactly as it is implemented in the `market_agents` package.[1] The economy has one risky asset (optionally $K \in \{2, \ldots, 5\}$ assets with a common factor), riskless cash, a continuous double auction with explicit latency, fees and risk limits, a population of trader slots filled by seven documented classical policies, and a treatment that replaces a fraction of those slots by a large-language-model (LLM) policy. Nothing in the simulator imposes an equilibrium: prices are whatever the order book produces.

[1] Source files are cited in footnotes as `module.py`; all paths are relative to `src/market_agents/`. Default parameter values are those of `configs/baseline.yaml`; every value is listed again in Appendix D.

Time, units and accounting. Model time is a discrete tick $t = 0, 1, \ldots, T-1$ with $T = 600$ in the baseline; outcome statistics are computed on $t \geq 100$. There is no calendar mapping. Prices are integers in ticks (one tick = 1 cent), quantities are integers in lots, and cash is an integer in cents. Every trade moves exactly pq cents from buyer to seller and q lots from seller to buyer; fees are integers computed as $\text{round}(pq \cdot \text{bps}/10^4)$ and are paid to the exchange. After every fill and every dividend the ledger asserts the two conservation identities

$$\sum_i C_{i,t} + F_t \;=\; \sum_i C_{i,0} + \sum_{s \leq t} D_s^{\text{paid}}, \qquad \sum_i Q_{i,t}^k \;=\; \sum_i Q_{i,0}^k \quad (k = 1, \ldots, K), \tag{1}$$

where $C_{i,t}$ is agent i's cash, F_t the cumulative fees collected by the exchange, D_s^{paid} the cash injected by dividends and $Q_{i,t}^k$ the lots of asset k held.[2] The exchange never creates shares and there is no outside money other than dividends, so wealth is redistributed, not created, by trading.

Dividends and the latent-timing convention. Asset k pays a cash dividend every $\Delta = 20$ ticks. Index dividend dates by $n = 1, 2, \ldots$ (paid at $t = n\Delta$) and write

$$D_n \;=\; \mu_{s_n} + x_n + J_n, \qquad x_{n+1} \;=\; \rho\, x_n + \sigma\, \varepsilon_{n+1}, \quad \varepsilon_{n+1} \sim N(0,1), \tag{2}$$

with mean dividend $\mu = 50$ cents, persistence $\rho = 0.98$ and innovation scale $\sigma = 2$ cents per period. The regime $s_n \in \{0, 1\}$ (a two-state Markov chain shifting μ by a factor $1 + \texttt{mu_shift}$) and the compound-Poisson jump J_n are available but switched off in every design reported here ($\texttt{mu_shift} = \texttt{p_switch} = \texttt{jump_prob} = 0$). With $K > 1$ assets the innovations load on a common factor, $\varepsilon_k = b_k f + \sqrt{1 - b_k^2}\, u_k$, with $b_k = 0.7$ in the two-asset cells. The convention that matters for information is the *timing*: when D_n is paid at $t = n\Delta$ the deviation x_n becomes public, and the deviation of the *next* dividend, x_{n+1}, is drawn immediately but remains latent until $t = (n+1)\Delta$.[3] During the period $n\Delta \leq t < (n+1)\Delta$ the state of the economy is therefore x_{n+1}, and all information (news, private signals) is about this already-drawn but unannounced number. No agent ever observes a future innovation.

Two value benchmarks. Let $r = 0.25\%$ be the discount rate per dividend period. Discounting the expected dividend stream given the latent state gives the *latent fundamental value*

$$V_t \;=\; \frac{\mu}{r} + \frac{x_{n(t)+1}}{1 + r - \rho}, \qquad n(t) = \lfloor t/\Delta \rfloor, \tag{3}$$

[2] `exchange/accounts.py`, `Ledger.check_conservation`; the check also verifies that reserved cash and reserved lots never exceed holdings for unleveraged agents.

[3] `assets/fundamentals.py`, `Fundamentals.step` and `_draw_next`: the dividend is paid, `x_revealed` is set, the public filter is updated, then x_{n+1} is drawn. The initial deviation x_1 is latent from $t = 0$.

because $\mathbb{E}[x_{n+k} \mid x_{n+1}] = \rho^{k-1} x_{n+1}$ and $\sum_{k\geq 1} \rho^{k-1}(1+r)^{-k} = 1/(1+r-\rho)$. At the baseline $\mu/r = 20{,}000$ cents, so $V \approx 200.00$ currency units and the price–dividend ratio is about 400; one cent of next-dividend deviation is worth $1/(1+r-\rho) \approx 44.4$ cents of value. The *public value* applies the same closed form to the public posterior mean $\hat{x}_t$ of the latent deviation,

$$V_t^{\text{pub}} = \frac{\mu}{r} + \frac{\hat{x}_t}{1+r-\rho}, \tag{4}$$

where $\hat{x}_t$ is maintained by a Kalman filter: after the dividend at $n\Delta$ the prior is $\hat{x} = \rho x_n$ with variance $\Sigma = \sigma^2$ (the dividend reveals x_n exactly when jumps are off), and each public news item y with stated noise η updates $\hat{x} \leftarrow \hat{x} + \kappa(y - \hat{x})$, $\Sigma \leftarrow (1-\kappa)\Sigma$, $\kappa = \Sigma/(\Sigma + \max(\eta, 0.1)^2)$.[4] Hence $V_t^{\text{pub}} \neq V_t$ in general: the two coincide only at dividend dates, and they diverge most right after a latent shock.

Four price concepts are kept apart throughout the paper. (i) The *fundamental value* V_t of (3) is latent and is never shown to any agent; it is the benchmark for pricing error, bubbles and absorption. (ii) The *efficient (public) price* V_t^{pub} of (4) is the value a risk-neutral agent would assign using public information only; it is shown to every agent. (iii) The *transaction price* is the price of the last trade, in ticks. (iv) The *quote midpoint* $P_t = (a_t + b_t)/2$ is the price used in every outcome statistic; when one side of the book is empty the engine falls back to the last trade price and then to the reference price.[5] Deviations are measured as $\log P_t - \log V_t$ (and, as a robustness check, relative to V_t^{pub}).

Information. Three channels carry information about the latent deviation. *Public news* arrives each tick with probability 0.02 per asset. An item reports $y = x_{n+1} + \nu + \beta$, with credible noise $\nu \sim N(0, \eta^2)$, $\eta = 1$ cent, and a bias $\beta = \pm 10$ cents (random sign) with probability `misleading_share` (0 in the baseline, 0.5 in the misleading-news cell). Every item is rendered as one sentence ("Analyst note: next dividend expected $|y|$ cents above/below its long-run mean") and updates the public filter with the *stated* noise η whether or not it is biased, so misleading items fool V^{pub} by construction.[6] Agents see the items of the last 20 ticks. *Private signals* are drawn at every activation of an informed agent, $s_i = x_{n+1} + \xi_i$, $\xi_i \sim N(0, \sigma^2/\pi_i)$, with precision $\pi_i = 2$ (signal standard deviation $\sigma/\sqrt{2} \approx 1.4$ cents) for the 30% of fundamentalist slots that are informed and $\pi_i = 0$ otherwise; the draw is about the current latent state, never about future innovations.[7] *Announced dividends* reveal x_n at each dividend date. All three channels, plus the order book, trade tape, own account and open orders, are packaged into a single `Observation` object that is the only interface between the market

[4] `assets/fundamentals.py`, `public_observe`; the regime and jump branches of `_update_public` are a two-state mixture filter and a Kalman step on a noisy dividend, unused in the reported designs.

[5] `exchange/engine.py`, `Exchange.mid_ticks`; in call-auction mode the midpoint is replaced by the last clearing price because the book may be crossed between auctions.

[6] `signals/news.py`, `NewsFeed.step`.

[7] `assets/fundamentals.py`, `private_signal`; a new draw is made at each activation from the agent's own random stream.

and any policy, classical or LLM.[8]

The exchange. Each asset trades on a continuous double auction with price–time priority.[9] Agents submit *limit* orders (price in ticks), *market* orders, and *cancels* of their own resting orders. An order submitted at decision time t by agent i arrives at $t+\ell_i$, where the latency $\ell_i \sim U[0.05, 1]$ tick is drawn once per slot and fixed for the run; the exchange processes arrivals in (arrival time, sequence) order, so agents deciding at the same tick are ordered by latency and then by submission sequence. On arrival the engine checks, in order: the account is not frozen, the asset is not halted, $1 \le q \le 10{,}000$ lots, and for limit orders a positive price within $\pm 50\%$ of the current midpoint (price band). Limit bids reserve the notional plus the worst-case fee plus one cent per lot; market bids are capped to the largest quantity whose notional plus taker fee fits in free cash while walking the ask side; sells are clipped to free (unreserved) holdings and reserve them. A market order executes against the best quotes until filled or the opposite side is empty; the unfilled remainder is cancelled (immediate-or-cancel). Limit orders execute against crossing quotes and rest for the remainder. Partial fills are the norm. The taker pays 2 basis points of notional, the maker pays nothing. Resting orders expire 50 ticks after submission. Short selling is disabled: a sell larger than free holdings is clipped, and a sell with no holdings is rejected.[10]

Three safeguards are implemented as treatments and are off in the baseline. A *circuit breaker* halts an asset for 20 ticks when the last trade price has moved by more than 5% relative to the midpoint 10 ticks earlier; resting orders are cancelled at the halt and arriving orders are rejected while halted. A *position limit* rejects any order that would push $|\text{position} + \text{same-side resting} + q|$ above 300 lots. A *periodic call auction* replaces continuous matching: orders rest without matching and every 5 ticks the book clears at the uniform price that maximises executable volume among limit prices within $\pm 2\%$ of the last clearing price (ties go to the price nearest the reference); market orders are converted to band-edge limits for one auction and cancelled afterwards.[11] This last variant is a robustness check run with the mock provider only.

Leverage, margin and bankruptcy. Unleveraged slots can only spend free cash. "Leveraged" slots have a cap $L = 3$ on gross exposure over equity: their buying power is free cash plus $(L-1)\times$equity, and an order is rejected if the post-trade gross exposure (valued at the limit price or the midpoint) would exceed $L\times$equity. At the end of every tick the exchange marks every account to the midpoint. A leveraged account whose equity-to-gross-exposure ratio falls below the maintenance

[8] `agents/base.py`; the LLM prompt is a deterministic rendering of the same object (Appendix B).

[9] `exchange/book.py` (matching), `exchange/engine.py` (latency, fees, risk, halts, event log).

[10] `ExchangeConfig.allow_short` and `Account.short_allowed` both exist, but the population builder sets `short_allowed=False` for every slot, so the `allow_short` switch has no effect in the current code; see Appendix D.

[11] `exchange/call_auction.py`; the later-submitted of two crossing resting orders is treated as the taker for fee purposes.

margin $m = 0.25$ receives a margin call: all its resting orders are cancelled and a market order for 50% of each position is submitted at the start of the next tick, tagged `liquidation` so that it bypasses the position and leverage checks. Any account with non-positive equity is declared bankrupt, frozen (no further orders accepted), and liquidated in full.[12] Dividends on short positions would be charged, but shorts never arise.

Classical policies. Trader slots are filled by seven heuristics, summarised in Table 1; each sees only the `Observation` and returns at most a handful of intents.[13] Each classical agent is activated independently each tick with its type's probability ("activity"). Fundamentalists form a posterior value $\hat{V}_i = V^{\text{pub}} + w_i(s_i - \hat{x})/(1 + r - \rho)$ with $w_i = \pi_i/(1 + \pi_i)$ (so uninformed fundamentalists use V^{pub} exactly), trade when $|\hat{V}_i/P - 1|$ exceeds 0.5%, and size the order at 20 lots per percent of mispricing; a mispricing beyond twice the threshold makes the limit order marketable. Trend followers compare the mean midpoint over the last 5 ticks with the mean over the last 30 and trade in the direction of the difference once it exceeds 0.2%; mean reverters trade against the z-score of the midpoint relative to its 40-tick mean when $|z| > 1.5$. Noise traders are zero-intelligence-plus liquidity demanders in the spirit of Gode and Sunder (1993): random side, size $1 + \text{Geometric}(1/8)$, a market order with probability 0.6 or otherwise a limit order placed on the passive side at a distance $|N(0, (0.001\,P)^2)|$ from the best quote. Market makers are inventory-aware two-sided quoters: with inventory deviation I from the endowment they quote around the reservation price $P(1 - 0.02\,I/200)$ with half-spread $\text{clip}(8 + \hat{\sigma}, 1, 60)$ ticks, where $\hat{\sigma}$ is the standard deviation of tick-to-tick midpoint changes over 30 ticks, 20 lots per side, re-quoting (cancelling all) at every activation; they stop quoting on one side when $|I|$ reaches 200 lots and withdraw entirely during a liquidity shock. They are endowed with five times the standard endowment and are never replaced by LLM agents. Leveraged traders run the trend rule with size 30 and market-order probability 0.7 on an account with $L = 3$. Cross-asset arbitrageurs trade the residual of a 60-tick log-price regression between the two first assets when its z-score exceeds 1.5, and are inactive in single-asset runs. The baseline mix and the behavioural parameters were frozen by the calibration sweep of Section 5.

The LLM policy. An LLM-controlled slot decides every 20 ticks from $t = 50$ onward, at one of four staggered offsets $\{0, 5, 10, 15\}$ drawn per slot, so that 27–28 decisions are made per agent in a 600-tick run and roughly a quarter of the LLM population decides in each batch.[14] Each decision is built from a canonical snapshot: the five best levels of each side of the book, best bid/ask, midpoint

[12] `exchange/engine.py`, `Exchange.margin_check`, `_liquidate`; liquidation orders are submitted by the runner at $t + 1$.

[13] `agents/classical.py`. Order sizes are capped by the policy's `max_qty` and by what free cash or holdings allow (`_affordable` divides free cash by $1.001\times$ the reference price).

[14] `experiments/runner.py`, `Simulation.run`: an LLM slot is active when $(t - \text{offset}) \bmod 20 = 0$ and $t \geq \texttt{warmup}/2 = 50$; batches are sent to the provider through a thread pool of width 8.

Table 1. Classical policies, baseline shares and parameters

Type	Share	Activity	Rule	Key parameters
Fundamentalist	25%	0.20	Trade toward $\hat{V}_i = V^{\text{pub}} + w_i(s_i - \hat{x})/(1 + r - \rho)$ when $\lvert\hat{V}_i/P - 1\rvert >$ threshold; marketable limit beyond $2\times$ threshold	threshold 0.5%; 20 lots per 1% mispricing; $p_{\text{mkt}} = 0.3$; max 50; improve 1 tick; 30% informed, $\pi = 2$
Trend follower	6%	0.20	Sign of $\bar{P}_5/\bar{P}_{30} - 1$ when $\lvert\cdot\rvert >$ threshold; size scales with $\min(\lvert s\rvert/\text{thr}, 3)$	threshold 0.2%; size 20; $p_{\text{mkt}} = 0.6$; max 50
Mean reverter	10%	0.20	Fade $z = (P - \bar{P}_{40})/\text{sd}_{40}$ when $\lvert z\rvert >$ threshold; limit at best quote ± 1 tick	threshold 1.5; size 15; $p_{\text{mkt}} = 0.2$; max 50
Noise trader	50%	0.30	Random side; size $1 + \text{Geom}(1/8)$; market order or passive limit at $\lvert N(0, (10^{-3}P)^2)\rvert$ from the best quote; cancels when > 3 open orders	$p_{\text{mkt}} = 0.6$ (calibrated); max 40
Market maker	5%	0.50	Two-sided quotes around $P(1 - 0.02I/200)$, half-spread $\text{clip}(8 + \hat{\sigma}_{30}, 1, 60)$ ticks; re-quotes every activation	20 lots per side; max inventory 200; endowment $\times 5$; never replaced
Leveraged	4%	0.20	Trend rule on a leveraged account	size 30; $p_{\text{mkt}} = 0.7$; $L = 3$; $m = 0.25$; liquidation 50%
Cross-asset arb.	0% (5% if $K = 2$)	0.20	Pairs trade on the residual of a 60-tick log-price regression when $\lvert z\rvert > 1.5$	size 10; max 40; inactive if $K = 1$

Shares and activities from `configs/baseline.yaml` (`population.mix`, `population.activity`, `policy_kwargs`); rules and defaults from `agents/classical.py`. P is the midpoint, $\bar{P}_w$ its mean over the last w ticks, p_{mkt} the probability that an order is a market order. With $N = 100$ the integer counts are 25/6/10/50/5/4.

and last trade; the midpoint history over the last 60 ticks (sub-sampled to ten points in the prompt, with the window return and the standard deviation of tick returns); the last 30 trades (volume and buyer-initiated share); the last dividend, the public posterior $\hat{x}_t$ and V_t^{pub}; the news items of the last 20 ticks (the last three are shown); the agent's own private signal and precision if it is informed; its cash, free cash, holdings and marked equity; and its open orders. The system prompt states the market mechanics (dividend interval, μ, r, μ/r, ρ, the $1/(1+r-\rho)$ conversion, tick, taker fee, time-to-live, no-short rule, the 50-lot cap and the decision interval) and asks for a JSON object with exactly eight keys: `action` $\in$ {hold, place, cancel}, `side`, `order_type`, `quantity`, `limit_price`, `cancel_order_ids`, `confidence` and a one-sentence `rationale`. The reply is parsed leniently (code fences and surrounding text are stripped) and validated strictly: a `place` must carry side, type

and a quantity in $[1,50]$; a limit price must lie within $\pm 20\%$ of the midpoint; a market order needs a non-empty opposite side; a buy must cost at most free cash at the reference price $\times 1.001$; a sell is clipped to holdings net of resting sells and is invalid when nothing is free; a `cancel` must name at least one order the agent owns; trading must not be halted. Any parse, schema or feasibility failure, provider error, or budget halt becomes a *hold* that is logged with its reason, so the estimand is intention-to-treat on the slot rather than on successful calls.[15] Realised shares of LLM agents, orders and volume are reported for every run.

Four cognition treatments change only the prompt. *Personas*: the single "generalist" system prompt, or eight diverse personas (value, momentum, contrarian, liquidity, risk-averse, macro, quant, generalist) assigned cyclically to LLM slots. *Memory*: the last 5 own decisions (action, side, quantity, type, price, rejection reason, the midpoint and equity at the time) are appended. *Communication channel*: LLM slots are grouped in cohorts of 5 in slot order, and the last 5 rationales posted in the cohort are shown as "unverified opinions". *Adaptive feedback*: the percentage change in equity since the last decision and its cumulative sum are appended. Provider adapters cover the OpenAI Responses API, the Anthropic Messages API, a seeded mock that parses the user message and applies a value rule with a 5% share of deliberately invalid answers, and a *frozen library* provider that returns the recorded decision of one of the five nearest neighbours of the current observation in a seven-dimensional standardised feature space (mispricing versus V^{pub}, window momentum, signal, inventory and cash ratios, spread, open-order flag), rescaling the limit price to the current midpoint. The library is a non-adaptive, non-reasoning replay of recorded decisions, used only to scale the population at zero cost; runs that use it are labelled provider `library` and are never pooled with online runs. Identical (provider, model, prompt version, system, user) requests are served from a SQLite cache and counted separately as cached hits; a per-run USD cap (and optional token and decision caps) halts further calls, after which every decision is a logged hold.[16]

Endowments. Slot i receives cash $C_i = \lfloor 2{,}000{,}000\, w_i \rfloor$ cents and $\text{round}(100\, w_i)$ lots of each asset, where $w_i = \exp(0.3 z_i)/\overline{\exp(0.3z)}$, $z_i \sim N(0,1)$, is a lognormal wealth multiplier normalised to mean one across the population; at the initial price $\mu/r = 200.00$ the endowment is split evenly between cash and stock. Market-maker slots receive five times these amounts. The initial reference price equals μ/r, so the economy starts at the latent value when $x_1 = 0$.

Population builder and matched replacement. Slot characteristics (wealth multiplier, latency, information draw) are generated from the `slots` stream; the classical policy deck—the integer counts implied by the mix—is shuffled by the `assign` stream and dealt down the wealth ranking,

[15] `llm/schema.py`, `llm/policy.py` (`to_action`); the validation rules are reproduced in Appendix B.

[16] `llm/cache.py`, `llm/budget.py`, `llm/library.py`, `llm/providers/`.

which stratifies policy types across wealth terciles.[17] The LLM treatment then replaces round(φN) of the non-market-maker slots, chosen by the dedicated `llm_assign` stream with quotas proportional to each policy type's count among eligible slots (remainder at random), so the composition of the replaced population is held constant along the dose and $\varphi = 1$ means 95 of 100 slots. A replaced slot keeps its endowment, latency, leverage cap and—crucially—its information set: an LLM agent that replaces an informed fundamentalist receives the same private signal precision, and one that replaces any other type receives none.[18] Providers are allocated to LLM slots in proportion to `provider_mix` and shuffled; prompt personas, stagger offsets and cohorts are assigned from the same stream. Because activation draws, within-tick ordering and all other streams are untouched by φ, a treatment and its control share the dividend path, the news, every classical agent's endowment, latency, activations and signals; the only difference is the policy of the replaced slots.

Shocks. Scheduled events are applied after the dividend step of the tick at which they occur and are written to the event log so that matched pairs share their timing.[19] A *fundamental jump* adds `size` cents to the latent deviation x_{n+1}; the baseline applies $+10$ cents at $t = 300$, which raises V_t by $10/(1+r-\rho) \approx 4.44$ currency units ($+2.2\%$), while V_t^{pub} moves only when news reports the new state or the dividend at $t = 320$ reveals it. A *rate* shock adds `size` to r; a *funding* shock multiplies every leverage cap by $1 -$ size (0.5 in the leverage cells, i.e. $3\times \to 1.5\times$); a *liquidity* shock makes all market makers withdraw their quotes until the matching `liquidity_end` event (60 ticks in the liquidity cell); a *news burst* injects a credible item with noise 0.5 reporting `size` and updates the public filter; a *placebo* is logged and changes nothing. The placebo at $t = 450$ is present in every baseline-derived cell and the placebo-only cell replaces the jump with a second placebo.

Implementation assumptions and their checks. Table 2 condenses the assumption register maintained with the code (`docs/assumptions.md`). Each row states an assumption as implemented and the pre-specified check that probes it; the register's identifiers are reproduced with the prefix "I" to avoid confusion with the theoretical Assumptions A1–A7 of Section 3.

Table 2. Implementation assumptions (register `docs/assumptions.md`) and robustness checks

ID	Assumption as implemented	Robustness check
I1	600 ticks, outcomes on $t \geq 100$; no calendar mapping	$T = 1{,}000$ on the control

[17] `experiments/population.py`, `build_population`; streams are spawned from the master seed by `numpy.random.SeedSequence` (`utils/rng.py`): `fundamentals`, `news`, `slots`, `assign`, `llm_assign`, `activation`, `tiebreak`, `mock_llm`, `shuffle`, plus one generator per agent.

[18] `LLMCfg.informed_share`: `null`; a number instead overrides the inherited status (information-equalisation variant). The matched-latency variant sets `latency.llm_latency`.

[19] `experiments/runner.py`, `_apply_shocks`; `signals/news.py`, `Shock`.

ID	Assumption as implemented	Robustness check
I2	Dividend every 20 ticks, eq. (2) with $\mu = 50$c, $\rho = 0.98$, $\sigma = 2$c, $r = 0.25\%$; one asset	$\rho \in \{0.95, 0.98\}$, $\sigma \in \{1, 2, 4\}$
I3	Benchmark = latent value V_t of eq. (3), risk neutral, constant r	Deviations from V^{pub} and from the end-of-run mean price also reported
I4	Public value = Kalman posterior from dividends and news (timing convention above); $V^{\text{pub}} \neq V$ within a period	Noisy-dividend variant (`jump_prob` 1, `jump_sigma` 4c)
I5	No look-ahead: signals and news concern the latent current state	Unit test of `private_signal`; prompt built from `Observation` only
I6	Latent jump $+10$c at $t = 300$ ($+2.2\%$ of V), placebo at $t = 450$	Sizes 10/20/40c; `news_burst` variant
I7	Continuous double auction, price–time priority, limit/market/cancel, partial fills, IOC remainder, integer ticks and lots	Call auction every 5 ticks, $\pm 2\%$ collar (mock only)
I8	Latency $U[0.05, 1]$ per agent, fixed; arrival-order processing; LLM slots draw like others	Matched latency `llm_latency` 0.05
I9	Taker 2 bp, maker 0, fees leave the economy; conservation asserted after every fill	Taker 0 and 5 bp
I10	TTL 50 ticks; price band $\pm 50\%$; max order 10,000 lots (policies cap at 40–50)	TTL 20/100
I11	No short selling; buys limited by free (unreserved) cash	`allow_short` variant (currently inert, see footnote in the exchange paragraph)
I12	Leverage only for leveraged slots: cap $3\times$, maintenance 25%, forced sale of 50% by market order next tick, bankrupt agents frozen	Cap $2\times / 5\times$; maintenance 15%/35%
I13	Safeguards off by default; breaker 5% over 10 ticks, 20-tick halt, resting cancelled; position limit 300	Halt 3%/8%; limit 150/600
I14	Five bounded inventory-skewed market makers (half-spread 8 ticks + volatility, skew 2% at 200 lots), endowed $\times 5$, never replaced	`quote_qty` 10/20; liquidity shock
I15	No rationality or equilibrium assumption: classical agents are documented heuristics, LLM agents output whatever the model returns	Calibration sweep (360 runs)

ID	Assumption as implemented	Robustness check
I16	Mix 25/6/10/50/5/4% (fundamentalist/trend/mean-rev/noise/MM/leveraged), stratified by wealth; endowment $20{,}000 + 100$ shares $\times$ lognormal(0.3)	Trend share 10–25% in the sweep; dispersion 0/0.6
I17	Classical activation Bernoulli(0.2–0.5) per tick; LLM decisions every 20 ticks from $t = 50$, staggered	Realised order/volume shares reported; `decision_interval` 10 (mock)
I18	30% of fundamentalists informed ($\pi = 2$); an LLM slot inherits the information of the slot it replaces	Stratum composition reported; `llm.informed_share` 0 variant
I19	Stratified replacement, market makers excluded (95 eligible slots)	`llm_replace: noise` at $\varphi = 0.25$ (mock)
I20	LLM observation = canonical snapshot (5 levels, 60-tick history, 30 trades, last dividend, V^{pub}, ≤ 3 news, own signal, account, open orders); prompt v1 frozen	Prompt audit
I21	Invalid or infeasible decisions become hold (quantity, $\pm 20\%$ price window, cash/holdings, owned cancel ids); sells clipped	Validity reported per cell; cells $<$ 80% flagged
I22	Provider sampling is not seeded; cached responses replay exactly; the library is a nearest-neighbour map, not an agent	Cache-hit share and online/replay flag per run
I23	Equal decision opportunities by default (27–28 per agent); equal USD via per-run `usd_cap`	Fairness cell
I24	Midpoint is the price in every metric (fallback to last trade); effective spread uses the start-of-tick midpoint	`msd_log` on last trade prices
I25	Bubble: $\lvert P/V - 1\rvert > 5\%$ for ≥ 10 ticks; crash: drop $\geq 10\%$ from the trailing 20-tick max; absorption: $\lvert P/V - 1\rvert < 1\%$ for 5 ticks, censored at 100	Grid 3/5/8% and 7/10/15% on saved ticks
I26	Coordination measured among LLM agents (sign correlation in 20-tick windows; synchrony per decision round)	Classical analogue from signed trades (`flow_corr_by_policy`)
I27	Two-asset variant: factor loading 0.7; LLM prompt covers asset 0 only	Spillover reported from the LLM-traded to the classical-only asset

3. A stylized model of shared-model trading

This section formalizes the mechanism that the simulator is designed to test: when many traders obtain their trading signal from the *same* model, their errors stop averaging out, prices become less informative, orders become synchronized, and leverage-driven feedback becomes harder to

contain. The model is deliberately stylized. It is a one-period, partial-equilibrium pricing exercise in the spirit of Kyle (1985) and Hellwig (1980), with the correlated-signal structure of Foster and Viswanathan (1996) reinterpreted as the signature of a shared foundation model or prompt family (Kleinberg and Raghavan, 2021). Order submission is mechanical (linear in the signal) and the response intensity is exogenous; the propositions below are therefore statements about how an *aggregation* mechanism transmits correlated errors into prices, not general-equilibrium results. We state this limitation plainly wherever it matters.

3.1. Setup

A single risky asset pays a liquidation value v. Normalizing the public prior (the public present value of the simulator) to zero, $v \sim \mathcal{N}(0, \sigma_v^2)$. There are N traders indexed by $i = 1, \ldots, N$ and a set of $G \geq 1$ "models" indexed by $g = 1, \ldots, G$. A model is any shared signal-generating device—a foundation model, a prompt family, a shared persona. Trader i is attached to model $g(i) \in \{1, \ldots, G\} \cup \{\varnothing\}$, where $\varnothing$ denotes a private (idiosyncratic) model. Let n_g be the number of traders attached to model g, $\phi_g = n_g/N$ its share, $\Phi = \sum_g \phi_g \leq 1$ the total share of traders on *some* shared model, and

$$H = \sum_{g=1}^{G} \phi_g^2, \qquad M_{\text{eff}} = 1/H, \tag{5}$$

the Herfindahl concentration of model use and the implied effective number of models.

Assumption 1 (A1: Gaussian primitives)**.** Trader i observes $s_i = v + \varepsilon_i$ with $\varepsilon_i = \eta_{g(i)} + u_i$. Model errors $\eta_g \sim \mathcal{N}(0, \sigma_\eta^2)$ are i.i.d. across models; a trader on a private model draws her own $\eta_i \sim \mathcal{N}(0, \sigma_\eta^2)$; idiosyncratic errors $u_i \sim \mathcal{N}(0, \sigma_u^2)$ are i.i.d. across traders; noise-trader demand is $z \sim \mathcal{N}(0, \sigma_z^2)$. All of $v, \{\eta_g\}, \{\eta_i\}, \{u_i\}, z$ are mutually independent.

Under A1 every trader's signal has the same precision—$\text{Var}(\varepsilon_i) = \sigma_\eta^2 + \sigma_u^2$ regardless of $g(i)$—so that the shares (ϕ_g) change only the *correlation* of errors across traders, never the quality of any individual signal. This is the "holding information quality fixed" clause of hypothesis H1.

Assumption 2 (A2: linear order submission)**.** Trader i submits the market order $x_i = \beta s_i$ with $\beta > 0$ a fixed, common response intensity.

A2 mirrors rule-like policies: an LLM agent given a snapshot returns an order whose size scales with the strength of the signal it extracts, and it does not condition on the price impact of other agents using the same model. The intensity β is a primitive here, not an equilibrium object; a CARA/price-taking version in which β depends on λ and on risk aversion (Hellwig, 1980; Grossman and Stiglitz, 1980) is possible but is not needed for the comparative statics we test, and we do not pursue it.

Assumption 3 (A3: competitive risk-neutral market maker)**.** A competitive, risk-neutral market maker observes the aggregate order flow $Y = \sum_{i=1}^{N} x_i + z$ and sets $P = \mathbb{E}[v \mid Y]$.

Assumption 4 (A4: partial equilibrium)**.** β, σ_z^2 and the shares (ϕ_g) are exogenous; traders do not choose whether to adopt a shared model, and the market maker does not re-optimize λ against the feedback demands introduced in Sections 3.5–3.6.

Under A1–A3 the projection theorem gives the linear pricing rule

$$P = \lambda Y, \qquad \lambda = \frac{\mathrm{Cov}(v,Y)}{\mathrm{Var}(Y)} = \frac{\beta N \sigma_v^2}{\beta^2 N^2 (\sigma_v^2 + \Omega)}, \qquad \Omega \equiv \sigma_\eta^2 \left[H + \frac{1-\Phi}{N} \right] + \frac{\sigma_u^2}{N} + \frac{\sigma_z^2}{\beta^2 N^2}. \quad (6)$$

The quantity Ω is the variance of the noise in the *normalized* order flow $\bar{Y} \equiv Y/(\beta N) = v + \bar{\varepsilon} + z/(\beta N)$, where $\bar{\varepsilon} = N^{-1} \sum_i \varepsilon_i$; every result below is a statement about Ω. The three objects we track are price informativeness, the pricing error, and order-flow co-movement:

$$\tau \equiv 1 - \frac{\mathrm{Var}(v \mid Y)}{\sigma_v^2} = \frac{\sigma_v^2}{\sigma_v^2 + \Omega}, \qquad \mathrm{Var}(P - v) = \sigma_v^2 (1 - \tau), \qquad \rho_{ij} \equiv \mathrm{Corr}(x_i, x_j), \quad (7)$$

and the pricing rule can be written compactly as $\lambda = \tau/(\beta N)$, so that $P = \tau \bar{Y}$: the price is the normalized flow shrunk toward the prior by the informativeness coefficient. The identity $\mathrm{Var}(P - v) = \sigma_v^2(1-\tau)$ (proved in Appendix A) means that, within the one-period model, informativeness and the mean squared pricing error are two readings of the same number; the simulator's dynamic statistics separate them.

3.2. Informativeness and penetration of a single shared model

Consider first a single shared model ($G = 1$) used by a share $\phi = n/N$ of traders, all others being on private models. Then $H = \phi^2$, $\Phi = \phi$, and

$$\Omega(\phi) = \sigma_\eta^2 \left[\phi^2 + \frac{1-\phi}{N} \right] + \frac{\sigma_u^2}{N} + \frac{\sigma_z^2}{\beta^2 N^2}, \qquad \tau(\phi) = \frac{\sigma_v^2}{\sigma_v^2 + \Omega(\phi)}. \quad (8)$$

Proposition 1 (Informativeness versus penetration)**.** *Under A1–A4 with a single shared model of share ϕ:*

(i) For every $N > 1$, τ is strictly decreasing in ϕ over the admissible grid $\phi \in \{1/N, 2/N, \ldots, 1\}$; in the continuous relaxation, $\partial\tau/\partial\phi < 0$ for all $\phi > 1/(2N)$. The values $\phi = 0$ and $\phi = 1/N$ coincide, since a model used by a single trader is a private model. For $N = 1$ the share has no effect.

(ii) For every N and every $\phi > 0$, $\tau(\phi) \le \tau_\infty(\phi) \equiv \sigma_v^2/(\sigma_v^2 + \phi^2 \sigma_\eta^2) < 1$, and $\tau(\phi) \to \tau_\infty(\phi)$ as $N \to \infty$ with ϕ fixed: the common error never averages out.

(iii) If $\phi = 0$ and $\sigma_z^2/N^2 \to 0$ (in particular if σ_z^2 is fixed), then $\tau \to 1$ as $N \to \infty$.

Intuition. The market maker learns from the average signal. Private errors are averaged over N draws and vanish at rate $1/N$; the shared error η is a single draw that enters the average with weight ϕ and therefore survives with variance $\phi^2\sigma_\eta^2$ no matter how many traders there are. Penetration converts a diversifiable error into a non-diversifiable one. Note that (ii) also implies that the *marginal* value of adding traders falls with ϕ: with a shared model, the market behaves as if it had at most $M_{\text{eff}} = 1/\phi^2$ independent signals.

3.3. Pricing error, order-flow correlation and synchrony

Let $\sigma_s^2 \equiv \sigma_v^2 + \sigma_\eta^2 + \sigma_u^2$ be the variance of a signal, and define the two possible pairwise correlations,

$$\rho_1 = \frac{\sigma_v^2 + \sigma_\eta^2}{\sigma_s^2} \quad \text{(same model)}, \qquad \rho_0 = \frac{\sigma_v^2}{\sigma_s^2} \quad \text{(different or private models)}, \qquad \rho_1 - \rho_0 = \frac{\sigma_\eta^2}{\sigma_s^2}. \tag{9}$$

Let $\bar{\rho}$ denote the population-average pairwise correlation over the $N(N-1)$ ordered pairs $i \neq j$, and let the *synchrony index* of a pair be the probability that the two traders submit same-signed orders, $S_{ij} = \Pr(x_i x_j > 0)$, with population average $\bar{S}$.

Proposition 2 (Pricing error, correlation and synchrony)**.** *Under A1–A4 with a single shared model of share $\phi = n/N$:*

(i) $\mathrm{Var}(P - v) = \sigma_v^2\,\Omega(\phi)/(\sigma_v^2 + \Omega(\phi))$ *is increasing in ϕ on the admissible grid (and for all real $\phi > 1/(2N)$), and is bounded below by $\sigma_v^2\phi^2\sigma_\eta^2/(\sigma_v^2 + \phi^2\sigma_\eta^2)$ uniformly in N.*

(ii) $\mathrm{Corr}(x_i, x_j) = \rho_1$ *if $g(i) = g(j) \neq \varnothing$ and ρ_0 otherwise; $\rho_1 > \rho_0$ if and only if $\sigma_\eta^2 > 0$.*

(iii) The average pairwise correlation is

$$\bar{\rho}(\phi) = \rho_0 + w(\phi)\,\frac{\sigma_\eta^2}{\sigma_s^2}, \qquad w(\phi) = \frac{n(n-1)}{N(N-1)} = \frac{\phi(N\phi - 1)}{N-1}, \tag{10}$$

which is convex in ϕ, increasing for $\phi > 1/(2N)$, and converges to $\rho_0 + \phi^2\sigma_\eta^2/\sigma_s^2$ as $N \to \infty$.

(iv) For a pair with correlation ρ, $S(\rho) = \frac{1}{2} + \arcsin(\rho)/\pi$, which is strictly increasing in ρ. The population synchrony index is

$$\bar{S}(\phi) = \tfrac{1}{2} + \frac{1}{\pi}\Big[(1 - w(\phi))\arcsin\rho_0 + w(\phi)\arcsin\rho_1\Big], \tag{11}$$

which is a strictly increasing function of $\bar{\rho}$ whenever $\sigma_\eta^2 > 0$, and satisfies the Jensen bound $\bar{S} \geq \frac{1}{2} + \arcsin(\bar{\rho})/\pi$ with equality if and only if $\sigma_\eta^2 = 0$.

If $\sigma_\eta^2 = 0$, none of τ, $\mathrm{Var}(P - v)$, *$\bar{\rho}$ or $\bar{S}$ depends on ϕ.*

Intuition. Two traders on the same model agree about the world more often than the fundamentals justify, because they share the model's mistake. The share of such pairs is quadratic in ϕ (it is the probability that two random traders both use the model), which is why co-movement and synchrony accelerate with penetration. The last sentence is the natural placebo: a shared model that adds no error of its own ($\sigma_\eta^2 = 0$) is just a precise private signal and cannot generate any of the effects.

3.4. Diversity as a stabilizer

Proposition 3 (Diversity)**.** *Under A1–A4 with G shared models of shares $(\phi_1, \ldots, \phi_G)$ and total share Φ:*

(i) Holding Φ fixed, τ and $\mathrm{Var}(P - v)$ depend on the shares only through $H = \sum_g \phi_g^2$, with $\partial\tau/\partial H < 0$. When $\Phi = 1$, $\Omega = \sigma_\eta^2/M_{\text{eff}} + \sigma_u^2/N + \sigma_z^2/(\beta^2 N^2)$: model error averages out at rate $1/M_{\text{eff}}$ while private error averages out at rate $1/N$.

(ii) $\Phi^2/G \le H \le \Phi^2$. The upper bound is attained if and only if a single model carries the whole share Φ; the lower bound if and only if the G models have equal shares $\phi_g = \Phi/G$ (in the integer version, if and only if the n_g differ by at most one).

(iii) Consequently, for any population with total shared-model share Φ, $\tau_{single} \le \tau_{mixed} \le \tau_{uniform}$: a mixed-provider or diverse-prompt population is weakly more informative than a single-model population of the same size, strictly so whenever at least two models have positive share and $\sigma_\eta^2 > 0$.

(iv) The average correlation generalizes to $\bar{\rho} = \rho_0 + w\,\sigma_\eta^2/\sigma_s^2$ with $w = (NH - \Phi)/(N - 1)$, so that, for fixed Φ, $\bar{\rho}$ and $\bar{S}$ are also increasing in H.

Intuition. With several models, the common errors are themselves averaged over models. The Herfindahl index is the exact sufficient statistic because the variance of a weighted average of independent model errors is $\sigma_\eta^2 \sum_g \phi_g^2$. Splitting a given LLM population evenly across G providers (or prompt families with independent errors) reduces the undiversifiable noise from $\Phi^2\sigma_\eta^2$ to $\Phi^2\sigma_\eta^2/G$. Note that the recovery is partial unless G grows with N: $M_{\text{eff}} \le N$ and diversity across a handful of providers cannot restore the $1/N$ rate (hypothesis H4). In this stylized model the benefit of diversity rests entirely on the independence of η_g across models; models trained on similar data or distilled from each other would have correlated η_g and a correspondingly larger effective H.

3.5. Feedback amplification

We now add a second date and trend demand in the spirit of De Long et al. (1990). Let $P_0 = 0$ be the public value before the trading round and $Y_1 = \beta \sum_i s_i + z_1$ the fundamental flow of Section 3.1. Trend followers submit the demand $\gamma(P_1 - P_0)$, $\gamma \ge 0$.

Assumption 5 (A5: within-round feedback with a fixed pricing schedule)**.** Trend demand responds to the price change of the current round, and the market maker applies the schedule λ of (6) to the total flow $Y_1 + \gamma(P_1 - P_0)$ without re-estimating it.

A5 is the model counterpart of the simulator's timing: fundamental and LLM orders arrive in staggered batches over a decision interval, while trend followers react every tick to the mid-price history, so within a decision interval the trend response is effectively contemporaneous. Its second clause is a restriction, not a result: a Bayesian market maker who knew that the extra flow is $\gamma\Delta P$ would filter it out (the total flow is then a deterministic multiple of Y_1), and there would be no amplification. The simulator's market makers quote a mechanical inventory-skewed schedule and do not filter, which is what A5 captures.

Proposition 4 (Feedback amplification)**.** *Under A1–A5, write $\Delta P = P_1 - P_0$ and $\kappa \equiv 1/(1-\lambda\gamma)$.*

(i) If $\lambda\gamma < 1$ the round has the unique fixed point $\Delta P = \kappa\lambda Y_1$, which is the limit of the iterated response $\Delta P^{(k+1)} = \lambda\left(Y_1 + \gamma\Delta P^{(k)}\right)$ from any starting point. The variance of the price change is amplified by $\kappa^2 = 1/(1-\lambda\gamma)^2$ relative to the no-feedback benchmark.

(ii) If $\lambda\gamma \geq 1$ the iterated response diverges, $|\Delta P^{(k)}| \to \infty$, from every starting point except the algebraic solution. For $\lambda\gamma = 1$ no solution exists when $Y_1 \neq 0$; for $\lambda\gamma > 1$ the algebraic solution $\Delta P = \kappa\lambda Y_1$ has the sign opposite to the flow and is repelling, so no adjustment process reaches it.

(iii) Amplification leaves τ unchanged (the price remains a deterministic multiple of Y_1) but raises the pricing error to $\mathrm{Var}(P_1 - v) = \sigma_v^2\left[(1-\tau) + \tau(\kappa-1)^2\right]$, increasing in γ on $[0, 1/\lambda)$.

Assumption 6 (A6: return-loading of LLM policies)**.** An LLM trader on model g submits $x_i = \beta s_i + \kappa_g \Delta P$ with $\kappa_g \geq 0$ the model's loading on the current price change; classical trend followers contribute an aggregate loading $\gamma_T \geq 0$.

Corollary 1 (Shared models raise the effective feedback coefficient)**.** *Under A1–A6 the effective feedback coefficient is $\gamma = \gamma_T + N\sum_g \phi_g \kappa_g$. With a single shared model of share ϕ and loading $\kappa > 0$, $\gamma(\phi) = \gamma_T + N\phi\kappa$ is increasing in ϕ, the amplification factor $\kappa^2(\phi)$ is increasing in ϕ, and the stable regime is lost at the critical share $\phi^* = (1/\lambda - \gamma_T)/(N\kappa)$. If $\kappa = 0$ the share has no effect on amplification. If the loadings κ_g differ in sign across models (some models are contrarian), the aggregate loading is the share-weighted average and can be smaller than that of any single model: diversity dampens feedback, concentration does not.*

Intuition. A price rise induced by informed flow triggers trend demand, which under a fixed pricing schedule moves the price again, and so on; the geometric sum converges only if each round's response is smaller than the last. LLM policies that mention momentum, "recent trend" or "follow the tape" in their rationale behave like trend followers with a loading κ that the simulator measures

by regressing LLM orders on lagged returns. Whether κ is positive is an empirical matter; the corollary says what follows if it is. The variance amplification is identical to the one obtained by DeLong et al. with rational speculators front-running feedback traders, but the source here is mechanical.

3.6. Liquidity spiral

Finally, we add leverage, following the margin-spiral logic of Brunnermeier and Pedersen (2009). A cohort L of N_L leveraged traders holds the aggregate position X (in lots) financed with equity W under a maintenance requirement of $m > 0$ units of equity per lot held.[20] Let $b \equiv W - mX \geq 0$ be the initial margin buffer. An exogenous price shock $\Delta P_0 < 0$ hits the cohort's side of the market; after a total price change ΔP the cohort's equity is $W + X\Delta P$, so it must liquidate

$$F(\Delta P) = \max\left\{0,\ X - \frac{W + X\Delta P}{m}\right\} = \max\left\{0,\ \frac{-X\Delta P - b}{m}\right\} \tag{12}$$

lots. Under A5 the market maker prices the forced sales as order flow, so the total price change solves $\Delta P = \Delta P_0 - \lambda F(\Delta P)$. Define the spiral coefficient $a \equiv \lambda X/m$.

Proposition 5 (Liquidity spiral)**.** *Under A1–A5 and (12):*

(i) If $\Delta P_0 \geq -b/X$ the shock is absorbed by the buffer: $F = 0$ and $\Delta P = \Delta P_0$ is a fixed point at which the cascade stops at once. (When $a > 1$ and $b > 0$ a second, self-fulfilling fixed point with $F > 0$ also exists; it is not reached by the cascade and we select the cascade limit throughout.)

(ii) If $\Delta P_0 < -b/X$ and $a < 1$, the unique fixed point is

$$\Delta P^* = \frac{\Delta P_0 + \lambda b/m}{1 - a}, \qquad \text{so that with } b = 0: \quad \Delta P^* = \frac{\Delta P_0}{1 - \lambda X/m}, \tag{13}$$

with amplification multiplier $1/(1 - \lambda X/m) > 1$. The fixed point is the limit of the round-by-round cascade in which each round's forced sales respond to the previous round's price.

(iii) If $\Delta P_0 < -b/X$ and $a \geq 1$, no finite fixed point exists and the cascade decreases without bound: in a market with a price band, the price reaches the band.

(iv) Let $X = \left|\sum_{i \in L} x_i\right|$ be the absolute net position of the cohort and $\bar{\rho}_L$ its average pairwise order correlation. Then X is the absolute value of a centered Gaussian with standard deviation $\sigma_X = \beta\sigma_s\sqrt{N_L[1 + (N_L - 1)\bar{\rho}_L]}$, so that $\mathbb{E}[X] = \sigma_X\sqrt{2/\pi}$ and the spiral probability $\Pr(\lambda X/m \geq 1) = 2\bar{\Phi}(m/(\lambda\sigma_X))$ are both strictly increasing in $\bar{\rho}_L$, hence (Proposition 2) in the shared-model share of the cohort. (Here $\bar{\Phi}$ is the standard normal survival function.)

[20] If the simulator's maintenance margin rate is $\hat{m}$ and the pre-shock price is $\bar{P}$, then $m = \hat{m}\bar{P}$ with the collateral valued at the pre-shock price; this linearization is exact to first order in $|\Delta P|/\bar{P}$ and is the only approximation in the proposition.

Intuition. The forced sale of one lot per m/X of price decline moves the price by λ per lot; when $\lambda X/m \geq 1$ each round of liquidation generates at least as large a decline as the one that triggered it. A shared model does not change the expected position of any single leveraged trader, but it makes positions line up on the same side, which fattens the distribution of the cohort's net exposure; the spiral threshold m/λ is crossed more often. The net position is a conservative base for forced sales—the gross exposure on the shocked side is weakly larger—so (iv) understates the mechanism. Since by (6) λ itself is decreasing in Ω, there is a countervailing channel (a less informative market is also a less price-sensitive one); in the simulator, however, λ is set by the mechanical market makers' depth, so we treat it as a parameter.

3.7. Safeguards

Two safeguards from the simulator have direct counterparts. A *position limit* caps the cohort's exposure at L lots (a per-trader cap ℓ with $N_L \ell = L$). A *trading halt* of length $h \geq 1$ rounds is triggered when the price change reaches a threshold $\theta > 0$; during the halt no forced sales execute. For the halt we need a reduced-form statement of what changes while trading is stopped.

Assumption 7 (A7: halt effectiveness)**.** After a halt of length h, the cascade resumes with spiral coefficient $\psi(h)\,a$, where $\psi : \{0,1,2,\dots\} \to (0,1]$ is nonincreasing with $\psi(0) = 1$.

A7 summarizes two mechanisms, margin top-ups (fewer lots must be sold per unit of price decline) and the arrival of patient liquidity (smaller impact per lot sold), without modelling either. It is an assumption about the halt, not a result; the case $\psi(h) = 1$ is a halt that changes nothing but the clock.

Proposition 6 (Safeguards)**.** *Under the assumptions of Proposition 5:*

(i) (Position limit.) If $L < m/\lambda$, then for every shared-model share and every realization of signals, $\lambda X/m \leq \lambda L/m < 1$: the spiral region is empty and the multiplier is bounded by $1/(1-\lambda L/m)$. The cost is informational: if the limit binds on informed orders, each capped order $\tilde{x}_i = \min\{\max\{\beta s_i, -\ell\}, \ell\}$ has loading $\mathrm{Cov}(v,\tilde{x}_i) = \Pr(|\beta s_i| < \ell)\,\mathrm{Cov}(v,x_i) < \mathrm{Cov}(v,x_i)$ on the fundamental.

(ii) (Halt: depth.) Under A7, if the halt is triggered by the shock itself ($|\Delta P_0| \geq \theta$) and $\psi(h)a < 1$, the post-halt cascade converges to $\Delta P_h^* = (\Delta P_0 + \lambda b/m)/(1-\psi(h)a)$, which is finite even when $a \geq 1$ and satisfies $|\Delta P_h^*| \leq |\Delta P^*|$ when $a < 1$, strictly if $\psi(h) < 1$.

(iii) (Halt: time-to-absorb.) Let T_ε be the first round at which the price is within ε of its terminal value. Without a halt and $a < 1$, $T_\varepsilon(a) = \max\{0, \lceil \log(\varepsilon(1-a)/|\Delta P_0|)/\log a \rceil - 1\}$ (for $b = 0$; for $b > 0$ replace $|\Delta P_0|$ by $|\Delta P_0| - b/X$); with the halt, $T_\varepsilon^h = h + T_\varepsilon(\psi(h)a)$. Hence $T_\varepsilon^h \geq h$: no information—neither the shock itself nor public news arriving during the halt—is incorporated before h rounds have elapsed, and if $\psi(h) = 1$ the halt is pure delay, $T_\varepsilon^h = T_\varepsilon(a) + h$.

Intuition. A position limit works by construction: it bounds the base of forced sales and therefore the slope of the feedback loop, at the cost of truncating the informed demand that makes prices informative. A halt is more ambiguous. It reduces the depth of the spiral only through what changes during the halt (A7), and it always costs at least its own length in delayed price discovery; whether the trade-off is favourable is the empirical question H6 asks of the simulator.

3.8. Mapping to the simulator

Table 3 maps each theoretical primitive to the simulator parameter that manipulates or measures it. Two points deserve emphasis. First, ϕ is *not* the LLM share: it is the share of traders on one and the same model, i.e. `llm_share` times the share of LLM slots on a given provider–prompt combination (`provider_mix`, `prompt_set`). A population with `llm_share` $= 0.5$ split evenly across two providers with diverse prompts has a small H and should behave, in the model, like a population of private models. Second, β and λ are measured, not set: the LLM order aggressiveness is the slope of submitted quantity on the private signal (or on the public-value gap for uninformed LLM agents), and λ is the Kyle lambda estimated by regressing mid-price changes on signed order flow.

3.9. Testable predictions and what would falsify them

Table 4 lists the hypotheses of the pre-analysis plan together with the proposition that generates each, the predicted sign and the simulator statistic that tests it. Each line also states what would count against the model. We stress once more that these are predictions of a stylized, partial-equilibrium aggregation model; the simulator is not an implementation of the model but an environment in which the same mechanism should, if it is real, leave the same signature.

Three further implications are useful as placebo or dose–response checks. (a) By Proposition 2, an LLM treatment whose decisions carry no common component ($\sigma_\eta^2 = 0$, e.g. the mock provider with independent seeds) should produce none of H1–H3; a positive effect there would indicate a channel outside the model (for instance, a change in β or in latency). (b) By Proposition 1(ii), the loss of informativeness should not shrink as N grows from 100 to 10,000 at fixed ϕ, whereas by (iii) it should vanish in the private-model benchmark. (c) By Corollary 1, amplification should respond to ϕ only in treatments where the measured return loading κ of LLM decisions is positive.

4. Experimental identification

The simulator is an experimental apparatus and we describe it as one. Causal language refers to effects *inside the simulator*: the object identified is the effect of replacing a share of rule-based

Table 3. Theoretical primitives and their simulator counterparts.

Primitive	Simulator parameter or statistic
ϕ, (ϕ_g), H	`population.llm_share` × share of LLM slots on the same (provider, model, prompt family): `llm.provider_mix`, `llm.prompt_set` (`single`/`diverse`); H computed from the realized assignment
σ_η^2	shared-model error: variance of the common component of LLM decisions given the snapshot; manipulated by prompt diversity, `llm.memory`, `llm.channel`, estimated from the decision library (same-snapshot replays)
σ_u^2	private-signal noise $\sigma^2/$`precision` (`population.precision`, `llm.precision`) and `llm.informed_share`
σ_v^2	dividend-innovation variance σ^2 scaled by $\rho/(1+r-\rho)$ (`fundamentals.sigma`, `rho`, `r_period`)
β	LLM order aggressiveness: measured slope of order size on the signal (`llm.max_order_qty` caps it)
λ	estimated Kyle lambda (mid change on signed volume over `div_interval` windows); depth set by market-maker slots (5 %)
γ, κ_g	`population.mix[trend]`, `mix[leveraged]`; LLM return loading estimated from decisions on lagged returns
m, W, X	`population.maintenance_margin`, `leverage_cap`, endowments; X = realized net position of leveraged slots
L, ℓ	position limit (per-agent order/position cap; `allow_short`, `max_short_lots`)
θ, h	`exchange.halt_threshold`, `halt_window`, `halt_duration` (circuit breaker)
z, σ_z^2	noise-trader slots (`mix[noise]`, activity)
N	`population.n_traders` (100, 1,000, 10,000)
τ	not estimated directly; its simulator counterparts are the mean squared log pricing error `msd_log` $= \overline{(\log P_t - \log V_t)^2}$ and the mean absolute deviation `mad_pct`, both decreasing in informativeness
$\mathrm{Var}(P-v)$	mean squared pricing error $(P_t - V_t)^2$
$\bar{\rho}$, $\bar{S}$	average pairwise correlation of signed order flow across agents; share of agent pairs with same-signed orders within a decision interval
ΔP^*, T_ε	crash depth after a shock event (`shocks`); time-to-absorb = ticks until $\lvert P_t/V_t - 1\rvert$ stays below 1 % for five consecutive ticks after the latent shock (`time_to_absorb`)

Table 4. Hypotheses, generating propositions, predicted signs and tests.

	Prediction (source)	Sign	Test statistic; falsifying pattern
H1	Informativeness falls in the single-model share ϕ at fixed signal quality and rises in M_{eff} (Props. 1, 3)	$\partial\tau/\partial\phi < 0$; $\partial\tau/\partial M_{\text{eff}} > 0$	slope of τ on `llm_share` within the single-provider/single-prompt arm, matched pairs across seeds; falsified if τ is flat or rising in ϕ while the shared component of LLM decisions (σ_η^2) is measurably positive
H2	Order-flow correlation and synchrony rise in ϕ, convexly (Prop. 2)	$+$, convex	$\bar{\rho}$ and same-sign share across LLM pairs vs. `llm_share`; falsified if the same-model pair correlation is not above the cross-model pair correlation
H3	Bubble/crash frequency and liquidity spirals rise with $\phi\times$leverage (Props. 4, 5)	interaction $+$	frequency of band hits / forced-liquidation cascades, crash depth, in the `leveraged`> 0 arms; falsified if the interaction is zero while $\hat{\lambda}X/m$ is observed to cross one more often under high ϕ
H4	Prompt or model diversity restores part of the loss (Prop. 3)	$\tau_{\text{diverse}} > \tau_{\text{single}}$; partial	difference in τ between `prompt_set=diverse` / mixed `provider_mix` and single-model arms at equal `llm_share`; falsified if diversity yields no gain, or a gain exceeding that of the fully private benchmark (which the model rules out)
H5	Communication channels raise synchrony (Prop. 2 with σ_η^2 raised by shared rationales)	$+$	$\bar{S}$ within cohorts with `llm.channel=true` vs. false; falsified if same-cohort synchrony does not exceed cross-cohort synchrony
H6	Circuit breakers and position limits cut crash depth at a cost in time-to-absorb (Prop. 6)	depth $-$; T_ε $+$	crash depth and ticks-to-absorb under `halt_threshold` and position caps vs. none; falsified if halts reduce depth with no delay (would reject A7 as the channel) or if position limits below $m/\hat{\lambda}$ fail to remove cascades

trader slots by LLM-controlled slots in this market mechanism, with these model snapshots and these prompts.

4.1. Unit of observation and randomization

The unit of observation is one run: $N = 100$ trader slots simulated for 600 ticks, with outcomes computed on $t \geq 100$. A run is indexed by a seed s and a cell z, a vector of design settings—the LLM share ϕ (`llm_share`), the provider and model mix, the prompt set, memory, the communication channel, the leverage composition, the safeguards and the fairness regime—and $Y_s(z)$ is the potential outcome of seed s under cell z. All exogenous randomness derives from a master seed through independent named streams: `fundamentals` (dividend path), `news`, `slots` (wealth multipliers, latencies, informed draws), `assign` (the classical policy of each slot), `llm_assign` (which eligible slots become LLM-controlled, their provider and decision offsets), `activation`, `tiebreak`, `shuffle` (within-tick processing order), and one generator per agent for policy randomness and private-signal noise. Slot assignment is stratified: slots are ranked by wealth multiplier and the policy types are dealt down the ranking, so each wealth tercile receives the same policy composition. The treatment replaces a fraction ϕ of eligible slots with proportional quotas within each policy type; market makers are never replaced, so liquidity supply is held fixed and $\phi = 1$ means 95 of 100 slots. A replaced slot keeps its cash, shares, latency and—after the change described in Section 4.8—its information set: an LLM agent is informed, at the same precision, if and only if the slot it replaces was. The treatment changes the decision policy of a stratified subset of slots and nothing else.

In a matched pair $\{(s,0),(s,z)\}$ the dividend path and latent value, the news, the shock at $t = 300$ and the placebo at $t = 450$, every endowment, latency and informed status, the policy of every non-replaced slot, the activation and tie-break draws, and the random numbers of every non-replaced agent are identical by construction. What is *not* held fixed is what the treatment is meant to change: the decisions of the LLM slots, including the provider's sampling noise, which is not seedable and is treated as part of the treatment, and the market's reaction to them—the book, the fills of every other agent, and hence the subsequent decisions of the rule-based agents, which condition on the book. Common random numbers remove the variance of the environment from the paired difference, not the equilibrium response of the market.

4.2. Causal structure

The cell label sets the treatment Z; with the `llm_assign` stream and the provider noise, Z determines the LLM slot set and its policies. Those policies generate, inside the run, a set of mediators M—pairwise order-flow correlation among LLM agents, synchrony, aggressiveness, validity rate and realized dose—and outcomes Y are produced jointly by M, by the exogenous streams, and by a

direct path from Z that bypasses LLM behaviour (fewer heuristic agents, halts, position limits). The effect of Z on Y is identified because Z is set by the design file and the only random input that depends on it, `llm_assign`, is independent of every other stream by construction; the effect of Z on each mediator is identified for the same reason, which is how H2 and H5 are tested. Mediation is *not* identified: M and Y are generated by the same path of draws and provider noise, so sequential ignorability—independence of M and the potential outcomes given Z—fails; a volatile fundamental path raises both the correlation of LLM orders and the pricing error. We report mediation only as a descriptive decomposition and treat each mediator as an outcome. Z also bundles model, prompt, decision frequency and the rule that invalid decisions become holds; the estimand is intention-to-treat, "deploy this agent as configured".

4.3. Estimands

Let $\tau_z = \mathbb{E}_s[Y_s(z) - Y_s(0)]$ be the paired average treatment effect of cell z against the control of the same seed. In order of priority: (E1) τ_z at each share against $\phi = 0$, the primary confirmatory contrast being $\phi = 0.25$ with `gpt-5.4-nano` and the single prompt, $\phi = 0.50$ the secondary; (E2) the dose–response $\tau(\phi)$ over $\phi \in \{0.05, 0.10, 0.25, 0.50, 1.00\}$, summarized by a linear slope and an ordered trend test; (E3) diversity at equal share, $\mathbb{E}[Y(\text{diverse}, 0.25) - Y(\text{single}, 0.25)]$ for prompt diversity at a fixed model and $\mathbb{E}[Y(\text{mix}, 0.25) - Y(\text{diverse}, 0.25)]$ for a two-model mix against the single-model diverse-prompt cell; (E4) cognition contrasts at $\phi = 0.25$—memory of five decisions, a rationale channel in cohorts of five, adaptive profit-and-loss feedback—each against the single-prompt cell at the same share; (E5) the interaction with leverage, $[Y(0.25, \text{lev}_{\text{hi}}) - Y(0, \text{lev}_{\text{hi}})] - [Y(0.25, \text{lev}_{\text{base}}) - Y(0, \text{lev}_{\text{base}})]$, and the safeguard contrasts (circuit breaker, position limit) at $\phi = 0.25$; (E6) the fairness contrasts, a more capable model at the share that equalizes the dollar budget with the primary cell, and a matched-latency cell in which LLM orders arrive with the minimum latency. The leveraged control at $\phi = 0$ is a classical run, free of API cost and deterministic given the seed, so it is produced on the matched seeds and the second difference of E5 is also within-seed. Two further objects are descriptive: a local projection of the log price–value gap around the latent shock, and τ_z divided by the mean realized order share, a rescaling and not a causal effect per order.

4.4. Pre-registered outcomes, hypotheses and thresholds

Table 5 lists the outcome families, their lead variables and the sign predicted by the hypotheses of Table 4. The coordination statistics exist only when $\phi > 0$, so H2 is tested as a dose–response among treated cells plus a comparison with the same statistic computed on the aggressor-signed trades of classical agents of the replaced types. Thresholds were fixed before any live call: a bubble is $|P/V - 1| > 5\,\%$ for at least ten ticks, a crash a drop of at least $10\,\%$ from the trailing 20-tick

maximum within 20 ticks. A sensitivity grid—bubbles at 3, 5 and 8 %, crashes at 7, 10 and 15 %—is reported for every instability result by re-scoring saved price paths; no threshold is re-selected after the fact. The classical control produces no crash at 10 %, so a floor effect is anticipated and the 7 % variant is the pre-specified fallback for H3. `msd_log` is analysed on the $\log_{10}$ scale, `time_to_absorb` with rank-based inference, and counts by mean differences with randomization inference.

Table 5. Pre-registered outcome families, lead variables and predicted signs.

Family	Variable	Definition	Predicted sign	H
Efficiency	`msd_log` (lead)	mean $(\log P - \log V)^2$	$\uparrow$ in ϕ; $\downarrow$ with diversity	H1, H4
	`mad_pct`	mean $100\,\lvert\log P - \log V\rvert$	$\uparrow$ in ϕ	H1
	`time_to_absorb`	first h after $t = 300$ with $\lvert P/V - 1\rvert < 1\,\%$ for 5 ticks (censored at 100)	$\uparrow$ in ϕ; $\uparrow$ with safeguards	H1, H6
	`response_ratio`	$(P_{t+20} - P_{t-1})/(V_{t+20} - V_{t-1})$ at the shock	$\lvert RR - 1\rvert \uparrow$ (two-sided)	H1
	`vr5`	variance ratio of log mid at lag 5	$\lvert VR - 1\rvert \uparrow$ (two-sided)	H1
Liquidity	`spread_bp` (lead)	mean quoted spread / mid, bp	$\uparrow$ in ϕ	H1b
	`eff_spread_bp`	$2\,\lvert\text{trade} - \text{mid}\rvert/\text{mid}$, bp	$\uparrow$ in ϕ	H1b
	`depth5`	mean lots within 5 levels, both sides	$\downarrow$ in ϕ	H1b
	`lambda_bp_100`	slope of Δlog mid (bp) on signed volume / 100	two-sided (λ falls as flow gets noisier)	H1b
	`resiliency`	half-life of log-spread deviations, ticks	$\uparrow$ in ϕ	H1b
Instability	`n_crashes` (lead H3)	drops $\geq 10\,\%$ from the trailing 20-tick max	$\uparrow$ in $\phi \times$leverage	H3
	`max_dd` (lead H6)	maximum drawdown of mid	$\uparrow$ in ϕ; $\downarrow$ with safeguards	H3, H6
	`n_bubbles`, `bubble_amp`	episodes $\lvert P/V - 1\rvert > 5\,\%$ for ≥ 10 ticks; max amplitude	$\uparrow$ in $\phi \times$leverage	H3
	`acf_absr_1_10`	$\sum_{k=1}^{10} \mathrm{ACF}_k(\lvert r\rvert)$	$\uparrow$ in ϕ	H3
	`kurtosis`	excess kurtosis of tick returns	$\uparrow$ in ϕ	H3
Coordination	`llm_pairwise_corr` (lead H2)	mean pairwise corr. of LLM net order signs, 20-tick windows	$\uparrow$ in ϕ; $\downarrow$ with diversity	H2, H4
	`llm_synchrony` (lead H5)	mean |mean sign| per decision round	$\uparrow$ with channel	H2, H5

4.5. Estimators

For each cell and seed we form $\Delta_s = Y_s(z) - Y_s(0)$ and report its mean, the paired-t 95 % confidence interval and Cohen's $d_z = \bar{\Delta}/\mathrm{SD}(\Delta)$, with Glass's Δ against the control standard deviation from the 30 free classical seeds for comparability across cells. The primary p-value is from randomization inference under the sharp null: within a pair the sign of Δ_s is exchangeable, and the p-value is the share of the 2^n sign assignments whose absolute mean is at least the observed one; the Wilcoxon

signed-rank test is a secondary distribution-free check. Multi-cell arms and the dose–response use a seed fixed-effects regression, $Y_{s,z} = \alpha_s + \sum_k \tau_k \mathbf{1}[z = k] + \varepsilon_{s,z}$, with standard errors clustered by seed; because there are at most seven live clusters, inference rests on the wild cluster bootstrap with Rademacher weights and, for the ordered share, on Page's L trend test for randomized blocks. Within each family the Holm step-down procedure controls the family-wise error rate at 5 %, the lead outcome is reported adjusted and unadjusted, all tests are two-sided and each sign is compared with its prediction. For the shock we estimate local projections of the gap $g_s(z,h) = \log P_{300+h} - \log V_{300+h}$: the coefficient at horizon $h \in [-20, 100]$ is the seed mean of $g_s(z,h) - g_s(0,h)$, with pointwise paired-t bands and a bootstrap sup-t band; the same projection around the placebo tick must be flat. Descriptive mediation regresses Y on each mediator with seed and cell fixed effects and reports the product-of-coefficients share of τ_z associated with it, as a decomposition and not a causal path.

4.6. Power and the number of seeds

For a paired design with standardized effect $d = \bar{\Delta}/\mathrm{SD}(\Delta)$, two-sided $\alpha = 0.05$ and power 0.80, the normal approximation gives

$$n \approx \frac{(z_{0.975} + z_{0.80})^2}{d^2} = \frac{(1.96 + 0.84)^2}{d^2} = \frac{7.85}{d^2}, \tag{14}$$

so $d = 0.5$ requires 32 pairs, $d = 0.8$ requires 13 and $d = 1$ requires 8. The live design has seven matched seeds. Inverting (14) gives a minimum detectable paired effect of $d \approx 1.06$, and the exact noncentral-t calculation with six degrees of freedom gives $d \approx 1.27$; we state plainly that only paired effects of roughly 1.2 standard deviations or more are detectable at 80 % power in the live cells. The seed count was fixed by the dollar cap on live calls, not by the pilot variance, and the pilot's three seeds are never pooled with the confirmatory seeds. Effects below the minimum detectable effect are not thereby absent: a lead outcome is reported as *refuted* if the confidence interval of d excludes $d \geq 0.5$ in the predicted direction, as showing *no large effect* if the interval contains zero and excludes the minimum detectable effect, and as *inconclusive* if it contains both. The classical design, which costs nothing, uses 30 matched seeds and detects $d \approx 0.5$; it carries the stylized-fact validation, the mock shared-model knob, the leverage, safeguard and information comparative statics and the placebo controls.

4.7. Design matrix

Table 6 lists the cells of the two confirmatory designs. The live design runs every cell on seeds 201–207 with `gpt-5.4-nano` and a single generalist prompt unless stated, a per-run dollar cap well

above the expected spend so that no run halts for budget, and cached responses so that every run replays exactly. The classical design runs on seeds 101–130 with the mock provider, a seeded rule that is not evidence about LLM behaviour. Both share the baseline economy: 600 ticks, one asset, a latent $+10$-cent shock at $t = 300$ and a placebo at $t = 450$.

4.8. What was frozen and what changed

The plan fixed the prompt text and schema, the baseline configuration and its hash, the cell and seed lists, the outcome definitions, thresholds and sensitivity grid, the estimators, the budget tiers, the model identifiers and price table, and the code revision recorded in every manifest. Four changes were made after the plan and before any confirmatory run was analysed; each is logged with its date and reason. First, the timing of fundamental information was restructured: in the plan's economy the announced dividend revealed the cash-flow state exactly, so the public value coincided with the latent value and private signals and news were redundant, leaving H1 unidentified; in the frozen economy the deviation of the upcoming dividend is latent during the period, the public value is a filtered estimate updated by news, and private signals concern the latent deviation. Because no confirmatory run had been analysed, this is the pre-registered economy rather than a deviation, and the noisy dividends originally proposed for the same purpose were not used. Second, shocks are applied after the dividend step and the latent jump was reduced from 40 to 10 cents, about $+2.2\,\%$ of value, so that a lagged adjustment is not counted as a 5 % bubble; thresholds were unchanged. Third, two leaks of common random numbers were closed: LLM slots now inherit the information set of the slot they replace instead of drawing their own informed status, which had given the treated arm more private information than the control, and the within-tick processing order was moved to its own stream so that activation draws no longer diverge across arms. Fourth, the first commit marks the freeze; the pilot ran concurrently with the confirmatory design on the same revision, its role confined to cost, validity and variance reporting, and the seed count was set by the budget rather than by the pilot. Two limitations follow from budget and environment rather than choice: the mini-only cell that would let the two-model mix be decomposed net of composition was not run, so the mix is compared with the single-model diverse-prompt cell only; and the Anthropic cells could not be run because the available key was rejected, so the adapter is dry-run tested with the mock provider and no Claude result is reported.

4.9. Assigned versus realized dose

The treatment is defined on slots, but LLM agents decide once every 20 ticks at staggered offsets, whereas rule-based agents are activated with probability 0.2 to 0.5 per tick. The realized share of order flow originated by LLM agents is therefore much smaller than the slot share: in the pilot, a

Table 6. Design matrix of the confirmatory runs.

Arm	Cell	Treatment	Seeds	Runs
Live design: `confirmatory_openai.yaml` (gpt-5.4-nano, single prompt unless stated)				
Dose	`dose_000`	$\phi = 0$ (control; no API cost)	201–207	7
Dose	`dose_005`	$\phi = 0.05$	201–207	7
Dose	`dose_010`	$\phi = 0.10$	201–207	7
Dose	`dose_025`	$\phi = 0.25$ (primary contrast)	201–207	7
Dose	`dose_050`	$\phi = 0.50$ (secondary contrast)	201–207	7
Dose	`dose_100`	$\phi = 1.00$ (95 eligible slots)	201–207	7
Diversity	`div_prompts_025`	$\phi = 0.25$, eight diverse persona prompts	201–207	7
Diversity	`div_models_025`	$\phi = 0.25$, diverse prompts, nano/mini 50/50	201–207	7
Cognition	`memory_025`	$\phi = 0.25$, memory of five past decisions	201–207	7
Cognition	`channel_025`	$\phi = 0.25$, rationale channel in cohorts of five	201–207	7
Cognition	`adaptive_025`	$\phi = 0.25$, adaptive P&L feedback in the prompt	201–207	7
Leverage	`leverage_025`	$\phi = 0.25$, leveraged slots 19 % (trend 6 %)	201–207	7
Safeguard	`breaker_025`	$\phi = 0.25$, halt when the 10-tick move exceeds 5 %	201–207	7
Safeguard	`poslimit_025`	$\phi = 0.25$, position limit 300 lots	201–207	7
Fairness	`fair_mini_005`	$\phi = 0.05$, gpt-5.4-mini (equal USD to `dose_025`)	201–207	7
Fairness	`fair_latency_025`	$\phi = 0.25$, LLM latency fixed at the minimum	201–207	7
Classical design: `classical_confirmatory.yaml` (mock provider, no API cost)				
Control	`control`	baseline economy	101–130	30
Placebo	`placebo_shock`	placebo events at $t = 300$ and $t = 450$	101–130	30
Feedback	`trend_heavy`	trend followers 16 %	101–130	30
Leverage	`leverage_heavy`	leveraged slots 19 %	101–130	30
Leverage	`leverage_heavy_funding_shock`	leveraged 19 %; −10-cent shock, then funding shock	101–130	30
Liquidity	`liquidity_shock`	market makers withdrawn for 60 ticks after the shock	101–130	30
Information	`info_low`, `info_high`	private-signal precision 0.5, 8	101–130	60
Information	`misleading_news`	half of the news items misleading	101–130	30
Safeguard	`breaker_5`, `poslimit_300`	halt at 5 %; position limit 300 lots	101–130	60
Mechanism	`auction_5`	call auction every five ticks	101–130	30
Mock herd	`mock_llm_25_indep`, `mock_llm_50_indep`	mock LLM slots 25 %, 50 %, independent errors	101–130	60
Contagion	`k2_control`, `k2_leverage`	two assets, factor loading 0.7; with leverage and funding shock	101–130	60
Total				592

Every live cell is paired with `dose_000` and every classical cell with `control` on the same seed. The leveraged $\phi = 0$ control for the E5 interaction is the `leverage_heavy` configuration re-run on seeds 201–207 (deterministic, no API cost). A pilot on seeds 1–3 at $\phi \in \{0.05, 0.25, 0.50\}$ served cost, validity and variance reporting only.

25 % slot share produced about 4 % of orders and 12 to 16 % of traded volume. Every manifest records both, and estimates are reported against the assigned dose with the realized dose alongside: the assigned dose is what a deployer chooses, while the realized dose is itself an outcome, since an agent that holds or returns an invalid decision originates no order. This asymmetry is why the fairness cells exist: equal decision opportunities is the default, equal dollars is implemented by a per-run budget cap, and matched latency removes the one speed disadvantage the LLM agents would otherwise carry.

4.10. Internal and external validity

Internal validity concerns inference inside the simulator. Its threats are provider non-determinism, treated as part of the treatment variance, with exact replay only from the cache; cache replay across cells, which turns identical early prompts into shared decisions across arms and whose share we report; the one-sided invalid-decision rule, whose rate is itself an outcome; and model drift, bounded by calling within a short window and recording the model snapshot. External validity is a different matter and we do not claim it. The rule-based traders are documented heuristics and the LLM agents are prompted policies without risk management, regulation or learning across runs; there is one asset in the prompt, model time has no calendar mapping, and fees and latencies are stylized. The findings are statements about this market mechanism and these model snapshots: signs and dose–response shapes may transfer to markets in which many participants obtain their signals from a common model; magnitudes will not.

5. Implementation and validation

Architecture. The simulator is a pure-Python package (`market_agents`, about 3,900 lines, Python 3.12) organised in six layers: `assets/` (dividend process and the two value benchmarks), `signals/` (news feed and shock schedule), `exchange/` (typed order records, the limit order book, the integer ledger, the engine with latency queue, fees, risk checks and halts, and the call auction), `agents/` (the `Observation` interface and the classical policies), `llm/` (decision schema, prompt builder, policy with validation, budget tracker, SQLite cache, frozen library, and the provider adapters), and `experiments/` (pydantic configuration, population builder, single-run `Simulation`, and the design runner), with `metrics/outcomes.py` computing every run-level outcome from the saved frames and `analysis/` holding the estimators and plots. The exchange is *event-sourced*: every state change (`submit`, `accept`, `reject`, `trade`, `cancel`, `expire`, `halt`, `resume`, `dividend`, `margin_call`, `liquidate`, `bankrupt`, `auction`, `shock`, `tick`) is appended to a sequence-numbered log with a flat scalar payload, and the engine exposes a hashable `snapshot`

of all balances, positions, resting orders and counters.[21] All randomness flows from one master seed through named `SeedSequence` children (fundamentals, news, slots, assignment, LLM assignment, activation, tie-break, mock LLM, within-tick shuffle) plus one generator per agent, so a run is a deterministic function of (configuration, seed) up to the provider's sampling; two executions of the same classical configuration produce identical event logs, tick frames and trade frames. Every run writes a `manifest.json` with the configuration hash (first 16 hex digits of the SHA-256 of the canonical JSON configuration), the package version, the git revision (suffixed `-dirty` when the working tree had uncommitted changes), the seed and stream names, the trader and realised LLM shares of traders, orders and volume, the number and validity rate of LLM decisions, token and USD usage by model, the provider and model identifiers, the prompt version, the full price table used by the budget tracker, the treatment labels, the shock log, the number of trades, halts and bankruptcies, and the wall-clock time.[22]

Test suite. The package ships 24 tests in three groups.[23] *Unit tests* cover the book (price–time priority, limit orders never cross beyond their price, market orders walk the book and stop when it is empty, cancel and depth), the engine (a fill moves exactly the notional and fees, latency orders arrivals, each rejection reason, cancel releases the reservation, a market buy is capped by cash, dividends preserve conservation, a halt rejects new orders, a margin breach generates a liquidation), the LLM layer (lenient parsing versus strict schema, each validation rule, the budget cap halting calls) and the configuration override semantics. *Property tests* use Hypothesis to generate random streams of up to 120 intents from six agents with partial activations, cancels, a maker rebate and a short time-to-live, and assert after every processed tick that the book is uncrossed and indexed consistently, that the ledger identities (1) hold, that each account's reserved cash and lots equal the sums over its resting orders, and that no position is negative; a second property asserts that two executions of the same stream yield identical snapshots and identical event logs (determinism and replay). *Integration tests* run the full simulation: an end-to-end classical run with 100 agents writing all artefacts, bit-for-bit determinism across runs, common random numbers across `llm_share` (identical latent value paths and identical endowments and latencies of every non-replaced slot), a mock-LLM run whose realised shares and validity rate are within the expected bands, the *zero-trade edge case* (a population of inactive market makers produces 30 ticks with a well-defined midpoint and no trades), and the logging and application of each shock kind.

[21] `exchange/orders.py`, `Event`; `exchange/engine.py`, `Exchange.snapshot`. The full event log is written to `events.parquet` when `output.events` is on; by default a run saves the per-tick, trade, agent and LLM-decision frames.

[22] `experiments/runner.py`, `Simulation._finish`; `utils/hashing.py`.

[23] `tests/unit`, `tests/property`, `tests/integration`; run with `make test`.

Table 7. Computational performance of the simulator at constant order flow

Traders	Seeds	Wall-clock (s)	Orders	Trades	Mean $\lvert P/V-1\rvert$ (%)	ACF_1	Spread (bp)
100	3	0.6	13,689	10,521	0.82	0.11	25.5
1,000	3	2.7	14,939	11,840	0.99	0.15	14.2
10,000	1	23.7	15,378	11,885	1.23	0.10	10.9

Baseline configuration, 600 ticks, activities scaled by $100/N$; generated by `scripts/50_tables.py` from `results/calibration/scale_benchmark.json`. Mean over seeds.

Computational performance. Table 7 reports the wall-clock time of a 600-tick baseline run as the population grows from 100 to 10,000 traders with the per-type activity scaled by $100/N$ so that order flow per tick is constant.[24] A run takes 0.6 s at $N = 100$ (three seeds: 0.57, 0.58, 0.58 s), 2.7 s at $N = 1{,}000$ and 23.7 s at $N = 10{,}000$, i.e. roughly linear in N at fixed order flow, with 13,300–15,400 orders and 10,300–12,200 trades per run. The 480-run classical confirmatory design (16 cells $\times$ 30 seeds) completes in about one minute on six worker processes. Live LLM cells are bound by provider latency: with 25 LLM agents in four staggered batches a run makes about 690 calls at 2.5–3.2 s each, so a $\varphi = 0.25$ run takes 6–8 minutes with eight concurrent requests (Appendix E).

Calibration. The classical baseline was calibrated once, before any LLM run, by a sweep over four behavioural parameters: fundamentalist aggressiveness $\{10, 20, 40\}$ lots per percent of mispricing, the combined trend-plus-leveraged share $\{0.10, 0.15, 0.25\}$ (split 60/40 between the two types and rebalanced against noise traders), the noise traders' market-order probability $\{0.3, 0.6\}$ and the market makers' quote size $\{10, 20\}$ lots—36 cells $\times$ 10 seeds $=$ 360 runs of 600 ticks.[25] The pre-stated criteria were those of the stylized-facts literature (Cont, 2001): first-order return autocorrelation close to zero, excess kurtosis above 3, positive volatility clustering (positive sum of the first ten autocorrelations of absolute returns), a quoted spread of 10–30 basis points, and a tracking error $\overline{\lvert P/V-1\rvert}$ near 1%. The retained cell (aggressiveness 20, trend-plus-leveraged share 0.10, market-order probability 0.6, quote size 20) has $\mathrm{ACF}_1 = 0.015$, variance ratio VR(5) $= 1.30$, excess kurtosis 26.6, $\sum_1^{10} \mathrm{ACF}\lvert r\rvert = 2.71$, tick volatility 30 bp, quoted spread 24.1 bp, depth 141 lots, price impact 15.6 bp per 100 lots, 0.8 bubble episodes and a maximum drawdown of 5.2% on average over ten seeds. The alternative with market-order probability 0.3 produced spreads of 6–12 bp but tracking errors of 2–3% and ACF_1 of 0.2–0.3; the aggressiveness-40 cells reduced the tracking error below 1% at the cost of positive return autocorrelation and larger drawdowns. The

[24] `scripts/11_scale_benchmark.py`; numbers from `results/calibration/scale_benchmark.json`, single process on an Apple-silicon (arm64) laptop. The benchmark was produced at the calibration stage, before the final revision of the dividend timing; the engine code it exercises is unchanged.

[25] `scripts/10_calibration_sweep.py`; cell means in `results/calibration/sweep_summary.csv`.

Table 8. Stylized facts of the classical baseline (control cell, 30 seeds)

Statistic	Mean	SD	Min	Max
ACF_1 of returns	-0.007	0.146	-0.292	0.312
Variance ratio (5)	1.18	0.29	0.69	1.79
Excess kurtosis	20.4	15.3	1.8	66.0
Vol. clustering $\sum_1^{10} \mathrm{ACF}\lvert r\rvert$	2.02	0.59	0.68	3.71
Tick volatility (bp)	17.0	7.7	6.0	38.6
Quoted spread (bp)	20.7	5.0	13.0	38.6
Effective spread (bp)	30.4	10.3	17.4	69.2
Depth, 5 levels (lots)	139	5	133	150
Price impact (bp/100 lots)	10.8	3.9	5.4	22.3
Mean $\lvert P/V - 1\rvert$ (%)	0.63	0.15	0.41	0.98
Bubble episodes	0.00	0.00	0.00	0.00
Max drawdown	0.033	0.012	0.010	0.060
MM P&L (%)	4.9	1.2	1.9	7.1
Wealth Gini (final)	0.285	0.023	0.243	0.338
Runs (seeds)	30			

Outcomes on ticks 100–599 of the `control` cell of `configs/classical_confirmatory.yaml`; generated by `scripts/50_tables.py` from `results/runs/classical_confirmatory/outcomes.parquet`.

retained mix (25/6/10/50/5/4%) and the three parameters are frozen in `configs/baseline.yaml` and were not revisited after the LLM pilot.

Stylized facts at the frozen revision. Table 8 summarises the 30 control runs of the classical confirmatory design (seeds 101–130) at the frozen code revision. Returns of the midpoint are uncorrelated at lag one (mean $\mathrm{ACF}_1 = -0.007$) with a variance ratio near one ($\mathrm{VR}(5) = 1.18$), heavy-tailed (excess kurtosis 20.4) and volatility-clustered ($\sum_1^{10} \mathrm{ACF}\lvert r\rvert = 2.02$); the quoted spread averages 20.7 bp and the effective spread 30.4 bp, depth within five levels is 139 lots, price impact is 10.8 bp per 100 lots, the tracking error is 0.63% and no run contains a bubble episode under the pre-specified 5%/10-tick rule. Market makers earn 4.9% over the run and the wealth Gini coefficient stays at 0.285. These are the properties against which every treatment is compared.

LLM adapters. Three provider adapters share one interface, `complete(system, user, model, ...)` → `ProviderResult` (text, input, cached-input, output and reasoning tokens, latency, model, request id, error).[26] The *OpenAI* adapter calls the Responses API with the frozen system prompt as `instructions`, the snapshot as the single user input, `max_output_tokens` = 1,200 (raised from 400 because reasoning tokens count against the cap), `text.format` = {`json_schema`, name `trading_decision`, `strict: true`} with the schema of Appendix B, `store=False`

[26] `llm/providers/`.

so that no response is retained in the account's response store, `reasoning.effort` = `low` for the gpt-5 and o-series models (temperature is never sent to them), and `prompt_cache_key` = `ma-v1-<prompt_id>` so that the provider can reuse the cached prefix across agents sharing a persona; the client is configured with a 60-second timeout and three retries, and rate-limit, status and connection errors are returned as error strings rather than raised. The *Anthropic* adapter calls the Messages API with the system prompt marked `cache_control: ephemeral`, `output_config.format` = {`json_schema`, schema} and `output_config.effort` = `low`, no `thinking` parameter, `max_tokens` = $\max(1{,}200, 256)$, and maps `stop_reason = refusal` to an invalid decision; it is implemented and dry-run tested but, because the key available on 2026-10-27 is rejected (HTTP 401), no Claude decision enters the reported results. The *mock* adapter parses the midpoint, public value, public estimate, private signal, holdings and free cash out of the user message and applies a seeded value rule (buy or sell when $|\hat{V}/P - 1| > 0.5\%$, 60% limit and 40% market orders) with a 5% share of deliberately malformed answers (half non-JSON, half schema-invalid) and an optional common tilt (`herd_bias`); its tokens are word counts priced at zero. It is used for dry runs, tests and the mock cells, and is not evidence about LLM behaviour.

Budget and cost accounting. A `BudgetTracker` per run prices every call from the table of USD per million tokens stored in the configuration (input, cached input, output) and reproduced in the manifest; an unknown model id is priced at the most expensive known output price so that the cap is conservative. The cap is checked before each call; once reached, the policy is halted for the rest of the run and every subsequent decision becomes a logged hold with error `budget_halt`. Cached hits are counted (`cached_hits`) but contribute no tokens and no cost; failures and invalid decisions are counted separately. Optional caps on total tokens and on decision opportunities per agent (`decision_cap`) implement the fairness designs. Before any live design, `ma budget` enumerates the calls of every cell × seed (number of LLM slots × $\lfloor (T - \texttt{warmup}/2)/20 \rfloor$) and prices them at a stated token profile.[27]

Decision log. Each decision appends one row to `llm_decisions.parquet`: tick, agent id, provider, model, prompt id, validity flag and reason, the parsed action, side, order type, quantity, limit price, confidence and rationale, input, cached-input, output and reasoning tokens, latency in seconds, USD cost, whether it was a cache hit, the provider error string, the first 16 hex digits of the SHA-256 of the concatenated system and user prompts, the provider request id, the seven-dimensional feature vector of the observation used by the frozen library, and the limit price relative to the midpoint.[28] Raw prompts are not stored in the decision log (they are reconstructible from

[27] `llm/budget.py`; `cli.py`, `budget`.
[28] `llm/policy.py`, `LLMDecisionRecord`.

the saved run and are stored verbatim in the SQLite cache together with the raw response text and token usage).

6. Limitations

The results of this paper are causal statements about a simulator. Every limitation below bounds what they can say about actual markets, and several bound what they can say about the simulator itself.

Internal validity. The confirmatory design uses seven matched seeds per live cell. With paired inference this gives 80 % power only for standardized effects of roughly $d \approx 1.2$ (Section 4); smaller effects are reported with their confidence intervals and must be read as inconclusive rather than absent. The seed count was fixed by the monetary cap on API calls (about 40 USD for the whole project), not by a pilot variance estimate, and the pilot ran concurrently with the confirmatory design on the same frozen code revision; the pilot therefore informs cost, validity and realized-dose reporting but did not feed back into the design. The thresholds that define bubbles and crashes were pre-registered, and the control economy exhibits a floor effect at the crash threshold (no crashes in 30 control seeds), so crash frequencies can only increase under treatment; the sensitivity grid in Section **??** reports how the conclusions move with the thresholds.

Provider coverage. All live cells use one provider family (OpenAI, `gpt-5.4-nano` as the primary model and `gpt-5.4-mini` in the two-model and equal-budget arms). The Anthropic adapter is implemented and exercised with the mock provider, but the key available when the experiments were run was rejected by the API, so no Claude-populated market is reported. The cross-provider comparison that the research questions call for is therefore reduced to a within-provider comparison of two model sizes, and the "shared-model" mechanism is tested through prompt diversity and the model mix rather than through genuinely different foundation-model families. The two-model arm is compared with the single-model diverse-prompt arm only; the mini-only cell needed to decompose a model-mix effect into a composition effect and a diversity effect was not run.

What the LLM agents are. An LLM agent here is a small reasoning model answering a short, fixed prompt every twenty ticks with a strictly validated JSON action. It has no tools, no retrieval, no fine-tuning and, outside the memory and channel arms, no state. The rationale field is logged but is never treated as evidence about the model's internal reasoning. Decision frequency is deliberately separated from event frequency: an LLM agent makes 27 decisions in a 600-tick run while classical agents act on roughly one tick in five, so the realized share of orders and volume is far below the

assigned share of trader slots; both are reported and the dose–response is read on the assigned share only. Provider outputs are not deterministic across repeated calls; recorded decisions are cached and replayed exactly, and cache hits are counted separately from online calls. The frozen nearest-neighbour library used to scale the population to 1,000 traders is a non-adaptive approximation of recorded behaviour and is reported as a separate arm, never pooled with online runs.

The economy. The exchange is a partial-equilibrium sandbox: dividends, discount rate, endowments and noise trading are exogenous; there is no entry, no learning across runs, no short selling by default, and the market makers follow a bounded inventory rule rather than an optimizing strategy. The stylized model of Section 3 shares these restrictions: linear order submission, a competitive risk-neutral market maker, Gaussian primitives and a fixed pricing schedule in the feedback extension. Its propositions are statements about that model; the simulator tests their comparative statics, not their closed forms. The classical population was calibrated to reproduce a short list of stylized facts (near-zero return autocorrelation, fat tails, volatility clustering, spreads of 10–30 basis points) at $N = 100$, and the same activity rates produce a somewhat different microstructure at $N = 1{,}000$ and $N = 10{,}000$ unless the activity rate is scaled; all penetration experiments are run at $N = 100$.

External validity. Simulated LLM traders are not institutions, and the market here has no regulation, no intermediaries beyond five quoting agents, no overnight, and no cross-market linkages beyond the two-asset arm. Nothing in the paper should be read as evidence about how deployed trading systems behave. The claim is narrower: within a transparent, replayable economy, raising the share of traders driven by one language model changes price discovery and coordination in the direction, and through the channel, that a correlated-error model predicts, and those changes can be undone or amplified by interventions that real markets can also implement. Whether the same mechanism operates at scale in live markets is an empirical question this design cannot answer.

References

Acemoglu, Daron, Asuman Ozdaglar, and Alireza Tahbaz-Salehi, "Systemic Risk and Stability in Financial Networks," *American Economic Review*, 2015, *105* (2), 564–608.

Adrian, Tobias and Hyun Song Shin, "Liquidity and leverage," *Journal of Financial Intermediation*, 2010, *19* (3), 418–437.

Agrawal, Kushal, Verona Teo, Juan J. Vazquez, Sudarsh Kunnavakkam, Vishak Srikanth, and Andy Liu, "Evaluating LLM agent collusion in double auctions," arXiv:2507.01413 2025.

Aher, Gati V., Rosa I. Arriaga, and Adam Tauman Kalai, "Using large language models to simulate multiple humans and replicate human subject studies," in "Proceedings of the 40th International Conference on Machine Learning," Vol. 202 of *Proceedings of Machine Learning Research* PMLR 2023, pp. 337–371.

Akata, Elif, Lion Schulz, Julian Coda-Forno, Seong Joon Oh, Matthias Bethge, and Eric Schulz, "Playing repeated games with large language models," *Nature Human Behaviour*, 2025, *9* (7), 1380–1390.

Aldasoro, I., L. Gambacorta, A. Korinek, V. Shreeti, and M. Stein, "Intelligent financial system: How AI is transforming finance," *Journal of Financial Stability*, 2025, *81*, 101472.

Amihud, Yakov, "Illiquidity and stock returns: cross-section and time-series effects," *Journal of Financial Markets*, 2002, *5* (1), 31–56.

Anderson, Barrett R, Jash Hemant Shah, and Max Kreminski, "Homogenization Effects of Large Language Models on Human Creative Ideation," in "Proceedings of the 16th Conference on Creativity and Cognition (C&C '24)" ACM 2024, pp. 413–425.

Anthis, Jacy Reese, Ryan Liu, Sean M. Richardson, Austin C. Kozlowski, Bernard Koch, James Evans, Erik Brynjolfsson, and Michael Bernstein, "LLM social simulations are a promising research method," arXiv:2504.02234 2025.

Argyle, Lisa P., Ethan C. Busby, Nancy Fulda, Joshua R. Gubler, Christopher Rytting, and David Wingate, "Out of One, Many: Using Language Models to Simulate Human Samples," *Political Analysis*, 2023, *31* (3), 337–351.

Assad, Stephanie, Robert Clark, Daniel Ershov, and Lei Xu, "Algorithmic Pricing and Competition: Empirical Evidence from the German Retail Gasoline Market," *Journal of Political Economy*, 2024, *132* (3), 723–771.

Axtell, Robert L. and J. Doyne Farmer, "Agent-Based Modeling in Economics and Finance: Past, Present, and Future," *Journal of Economic Literature*, 2025, *63* (1), 197–287.

Back, Kerry, C. Henry Cao, and Gregory A. Willard, "Imperfect Competition among Informed Traders," *The Journal of Finance*, 2000, *55* (5), 2117–2155.

Bai, Jennie, Thomas Philippon, and Alexi Savov, "Have financial markets become more informative?," *Journal of Financial Economics*, 2016, *122* (3), 625–654.

Banerjee, A. V., "A Simple Model of Herd Behavior," *The Quarterly Journal of Economics*, 1992, *107* (3), 797–817.

Bikhchandani, Sushil, David Hirshleifer, and Ivo Welch, "A Theory of Fads, Fashion, Custom, and Cultural Change as Informational Cascades," *Journal of Political Economy*, 1992, *100* (5), 992–1026.

Bollerslev, Tim, "Generalized autoregressive conditional heteroskedasticity," *Journal of Econometrics*, 1986, *31* (3), 307–327.

Bommasani, Rishi, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson et al., "On the opportunities and risks of foundation models," arXiv:2108.07258 2021.

_ , Kathleen A. Creel, Ananya Kumar, Dan Jurafsky, and Percy Liang, "Picking on the same person: Does algorithmic monoculture lead to outcome homogenization?," arXiv:2211.13972; Advances in Neural Information Processing Systems 35 (NeurIPS 2022) 2022.

Bond, Philip, Alex Edmans, and Itay Goldstein, "The Real Effects of Financial Markets," *Annual Review of Financial Economics*, 2012, *4* (1), 339–360.

Bouchaud, Jean-Philippe, Julius Bonart, Jonathan Donier, and Martin Gould, *Trades, Quotes and Prices: Financial Markets Under the Microscope*, Cambridge: Cambridge University Press, 2018.

_ , Marc Mézard, and Marc Potters, "Statistical properties of stock order books: empirical results and models," *Quantitative Finance*, 2002, *2* (4), 251–256.

Brock, William A. and Cars H. Hommes, "A Rational Route to Randomness," *Econometrica*, 1997, *65* (5), 1059–1095.

_ **and** _ , "Heterogeneous beliefs and routes to chaos in a simple asset pricing model," *Journal of Economic Dynamics and Control*, 1998, *22* (8-9), 1235–1274.

Brogaard, Jonathan, Terrence Hendershott, and Ryan Riordan, "High-Frequency Trading and Price Discovery," *Review of Financial Studies*, 2014, *27* (8), 2267–2306.

Brown, Zach Y. and Alexander MacKay, "Competition in Pricing Algorithms," *American Economic Journal: Microeconomics*, 2023, *15* (2), 109–156.

Brunnermeier, Markus K. and Lasse Heje Pedersen, "Market Liquidity and Funding Liquidity," *Review of Financial Studies*, 2009, *22* (6), 2201–2238.

Budish, Eric, Peter Cramton, and John Shim, "The high-frequency trading arms race: Frequent batch auctions as a market design response," *The Quarterly Journal of Economics*, 2015, *130* (4), 1547–1621.

Bybee, Leland, "Surveying generative AI's economic expectations," arXiv:2305.02823 2023.

Byrd, David, Maria Hybinette, and Tucker Hybinette Balch, "ABIDES: Towards high-fidelity multi-agent market simulation," in "Proceedings of the 2020 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation (SIGSIM-PADS '20)" ACM 2020, pp. 11–22.

Calvano, Emilio, Giacomo Calzolari, Vincenzo Denicolò, and Sergio Pastorello, "Artificial Intelligence, Algorithmic Pricing, and Collusion," *American Economic Review*, 2020, *110* (10), 3267–3297.

Chaboud, Alain P., Benjamin Chiquoine, Erik Hjalmarsson, and Clara Vega, "Rise of the Machines: Algorithmic Trading in the Foreign Exchange Market," *The Journal of Finance*, 2014, *69* (5), 2045–2084.

Chen, Lingjiao, Matei Zaharia, and James Zou, "How Is ChatGPT's Behavior Changing Over Time?," *Harvard Data Science Review*, 2024, *6* (2).

Chen, Yiting, Tracy Xiao Liu, You Shan, and Songfa Zhong, "The emergence of economic rationality of GPT," *Proceedings of the National Academy of Sciences*, 2023, *120* (51), e2316205120.

Chiarella, Carl and Giulia Iori, "A simulation analysis of the microstructure of double auction markets," *Quantitative Finance*, 2002, *2* (5), 346–353.

_ **,** _ **, and Josep Perelló**, "The impact of heterogeneous trading rules on the limit order book and order flows," *Journal of Economic Dynamics and Control*, 2009, *33* (3), 525–537.

Cho, Young-Min, Sharath Chandra Guntuku, and Lyle Ungar, "Herd behavior: Investigating peer influence in LLM-based multi-agent systems," arXiv:2505.21588 2025.

Chupilkin, Maxim, "Artificial institutions: How institutional design shapes LLM simulations," arXiv:2608.04020 2026.

Colliard, Jean-Edouard, Thierry Foucault, and Stefano Lovo, "Algorithmic Pricing and Liquidity in Securities Markets," *The Review of Financial Studies*, 2026. Forthcoming.

Cont, R., "Empirical properties of asset returns: stylized facts and statistical issues," *Quantitative Finance*, 2001, *1* (2), 223–236.

_ , A. Kukanov, and S. Stoikov, "The Price Impact of Order Book Events," *Journal of Financial Econometrics*, 2014, *12* (1), 47–88.

Cont, Rama and Lakshithe Wagalath, "Running for the exit: Distressed selling and endogenous correlation in financial markets," *Mathematical Finance*, 2013, *23* (4), 718–741.

_ , Sasha Stoikov, and Rishi Talreja, "A Stochastic Model for Order Book Dynamics," *Operations Research*, 2010, *58* (3), 549–563.

Danielsson, Jon and Andreas Uthemann, "Artificial intelligence and financial crises," *Journal of Financial Stability*, 2025, *80*, 101453.

Diebold, Francis X. and Kamil Yilmaz, "Measuring Financial Asset Return and Volatility Spillovers, with Application to Global Equity Markets," *The Economic Journal*, 2009, *119* (534), 158–171.

_ and _ , "Better to give than to receive: Predictive directional measurement of volatility spillovers," *International Journal of Forecasting*, 2012, *28* (1), 57–66.

_ and Kamil Yılmaz, "On the network topology of variance decompositions: Measuring the connectedness of financial firms," *Journal of Econometrics*, 2014, *182* (1), 119–134.

Ding, Zhuanxin, Clive W.J. Granger, and Robert F. Engle, "A long memory property of stock market returns and a new model," *Journal of Empirical Finance*, 1993, *1* (1), 83–106.

Doshi, Anil R. and Oliver P. Hauser, "Generative AI enhances individual creativity but reduces the collective diversity of novel content," *Science Advances*, 2024, *10* (28), eadn5290.

Dou, Winston Wei, Itay Goldstein, and Yan Ji, "AI-Powered Trading, Algorithmic Collusion, and Price Efficiency," NBER Working Paper 34054, National Bureau of Economic Research 2025.

Dugast, Jérôme and Thierry Foucault, "Data abundance and asset price informativeness," *Journal of Financial Economics*, 2018, *130* (2), 367–391.

Easley, David, Marcos M. López de Prado, and Maureen O'Hara, "The microstructure of the 'Flash Crash': Flow toxicity, liquidity crashes, and the probability of informed trading," *The Journal of Portfolio Management*, 2011, *37* (2), 118–128.

Engle, Robert F., "Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation," *Econometrica*, 1982, *50* (4), 987–1007.

Fama, Eugene F., "Efficient Capital Markets: A Review of Theory and Empirical Work," *The Journal of Finance*, 1970, *25* (2), 383–417.

Farmer, J. Doyne and Duncan Foley, "The economy needs agent-based modelling," *Nature*, 2009, *460* (7256), 685–686.

_ , Paolo Patelli, and Ilija I. Zovko, "The predictive power of zero intelligence in financial markets," *Proceedings of the National Academy of Sciences*, 2005, *102* (6), 2254–2259.

Financial Stability Board, "The financial stability implications of artificial intelligence," Technical Report, Financial Stability Board, Basel November 2024.

Fish, Sara, Yannai A. Gonczarowski, and Ran I. Shorrer, "Algorithmic collusion by large language models," arXiv:2404.00806 2024.

Forbes, Kristin J. and Roberto Rigobon, "No Contagion, Only Interdependence: Measuring Stock Market Comovements," *The Journal of Finance*, 2002, *57* (5), 2223–2261.

Foster, F. Douglas and S. Viswanathan, "Strategic Trading When Agents Forecast the Forecasts of Others," *The Journal of Finance*, 1996, *51* (4), 1437–1478.

Foucault, Thierry, Johan Hombert, and Ioanid Roşu, "News Trading and Speed," *The Journal of Finance*, 2016, *71* (1), 335–382.

Franke, Reiner and Frank Westerhoff, "Structural stochastic volatility in asset pricing dynamics: Estimation and model contest," *Journal of Economic Dynamics and Control*, 2012, *36* (8), 1193–1211.

Froot, Kenneth A., David S. Scharfstein, and Jeremy C. Stein, "Herd on the Street: Informational Inefficiencies in a Market with Short-Term Speculation," *The Journal of Finance*, 1992, *47* (4), 1461–1484.

Gao, Chen, Xiaochong Lan, Zhihong Lu, Jinzhu Mao, Jinghua Piao, Huandong Wang, Depeng Jin, and Yong Li, "S^3: Social-network simulation system with large language model-empowered agents," arXiv:2307.14984 2023.

Gao, Shen, Yuntao Wen, Minghang Zhu, Jianing Wei, Yuhan Cheng, Qunzi Zhang, and Shuo Shang, "Simulating financial market via large language model based agents," arXiv:2406.19966 2024.

Geanakoplos, John, "The Leverage Cycle," *NBER Macroeconomics Annual*, 2010, *24* (1), 1–66.

Gensler, Gary and Lily Bailey, "Deep Learning and Financial Stability," SSRN Working Paper 3723132, MIT Sloan School of Management 2020.

Glosten, Lawrence R. and Paul R. Milgrom, "Bid, ask and transaction prices in a specialist market with heterogeneously informed traders," *Journal of Financial Economics*, 1985, *14* (1), 71–100.

Gode, Dhananjay K. and Shyam Sunder, "Allocative Efficiency of Markets with Zero-Intelligence Traders: Market as a Partial Substitute for Individual Rationality," *Journal of Political Economy*, 1993, *101* (1), 119–137.

Goldstein, Itay and Liyan Yang, "Information Diversity and Complementarities in Trading and Information Acquisition," *The Journal of Finance*, 2015, *70* (4), 1723–1765.

Gould, Martin D., Mason A. Porter, Stacy Williams, Mark McDonald, Daniel J. Fenn, and Sam D. Howison, "Limit order books," *Quantitative Finance*, 2013, *13* (11), 1709–1742.

Greenwood, Robin, Augustin Landier, and David Thesmar, "Vulnerable banks," *Journal of Financial Economics*, 2015, *115* (3), 471–485.

Gromb, Denis and Dimitri Vayanos, "Equilibrium and welfare in markets with financially constrained arbitrageurs," *Journal of Financial Economics*, 2002, *66* (2-3), 361–407.

Grossman, Sanford J. and Joseph E. Stiglitz, "On the impossibility of informationally efficient markets," *American Economic Review*, 1980, *70* (3), 393–408.

Hansen, Anne Lundgaard and Seung Jung Lee, "Financial Stability Implications of Generative AI: Taming the Animal Spirits," Finance and Economics Discussion Series 2025-090, Board of Governors of the Federal Reserve System 2025.

Hasbrouck, Joel, "Measuring the Information Content of Stock Trades," *The Journal of Finance*, 1991, *46* (1), 179–207.

_ , "Assessing the Quality of a Security Market: A New Approach to Transaction-Cost Measurement," *Review of Financial Studies*, 1993, *6* (1), 191–212.

_ , "One Security, Many Markets: Determining the Contributions to Price Discovery," *The Journal of Finance*, 1995, *50* (4), 1175–1199.

Hashimoto, Ryuji, Takehiro Takayanagi, Masahiro Suzuki, and Kiyoshi Izumi, "Agent-based simulation of a financial market with large language models," arXiv:2510.12189 2025.

Hellwig, Martin F, "On the aggregation of information in competitive markets," *Journal of Economic Theory*, 1980, *22* (3), 477–498.

Hendershott, Terrence and Albert J. Menkveld, "Price pressures," *Journal of Financial Economics*, 2014, *114* (3), 405–423.

_ , Charles M. Jones, and Albert J. Menkveld, "Does Algorithmic Trading Improve Liquidity?," *The Journal of Finance*, 2011, *66* (1), 1–33.

Henning, Thomas, Siddhartha M. Ojha, Ross Spoon, Jiatong Han, and Colin F. Camerer, "LLM agents do not replicate human market traders: Evidence from experimental finance," arXiv:2502.15800 2025.

Hommes, Cars, "Behavioral and Experimental Macroeconomics and Policy Analysis: A Complex Systems Approach," *Journal of Economic Literature*, 2021, *59* (1), 149–219.

Hommes, Cars H., "Heterogeneous agent models in economics and finance," in Leigh Tesfatsion and Kenneth L. Judd, eds., *Handbook of Computational Economics, Volume 2: Agent-Based Computational Economics*, Amsterdam: Elsevier, 2006, pp. 1109–1186.

Horton, John J., "Large language models as simulated economic agents: What can we learn from Homo Silicus?," NBER Working Paper 31122, National Bureau of Economic Research 2023.

Immorlica, Nicole, Brendan Lucier, and Aleksandrs Slivkins, "Generative AI as economic agents," arXiv:2406.00477 2024.

Johnson, Neil, Guannan Zhao, Eric Hunsader, Hong Qi, Nicholas Johnson, Jing Meng, and Brian Tivnan, "Abrupt rise of new machine ecology beyond human response time," *Scientific Reports*, 2013, *3* (1), 2627.

Khandani, Amir E. and Andrew W. Lo, "What happened to the quants in August 2007? Evidence from factors and transactions data," *Journal of Financial Markets*, 2011, *14* (1), 1–46.

Kirilenko, Andrei, Albert S. Kyle, Mehrdad Samadi, and Tugkan Tuzun, "The Flash Crash: High-Frequency Trading in an Electronic Market," *The Journal of Finance*, 2017, *72* (3), 967–998.

Kirman, A., "Ants, Rationality, and Recruitment," *The Quarterly Journal of Economics*, 1993, *108* (1), 137–156.

Klein, Timo, "Autonomous algorithmic collusion: Q-learning under sequential pricing," *The RAND Journal of Economics*, 2021, *52* (3), 538–558.

Kleinberg, Jon and Manish Raghavan, "Algorithmic monoculture and social welfare," *Proceedings of the National Academy of Sciences*, 2021, *118* (22), e2018340118.

Kodres, Laura E. and Matthew Pritsker, "A Rational Expectations Model of Financial Contagion," *The Journal of Finance*, 2002, *57* (2), 769–799.

Kondor, Péter, "The More We Know about the Fundamental, the Less We Agree on the Price," *The Review of Economic Studies*, 2012, *79* (3), 1175–1207.

Kyle, Albert S., "Continuous Auctions and Insider Trading," *Econometrica*, 1985, *53* (6), 1315–1335.

_ **and Wei Xiong**, "Contagion as a Wealth Effect," *The Journal of Finance*, 2001, *56* (4), 1401–1440.

Ladley, Dan, "Zero intelligence in economics and finance," *The Knowledge Engineering Review*, 2012, *27* (2), 273–286.

Lakonishok, Josef, Andrei Shleifer, and Robert W. Vishny, "The impact of institutional trading on stock prices," *Journal of Financial Economics*, 1992, *32* (1), 23–43.

LeBaron, Blake, "Agent-based computational finance," in Leigh Tesfatsion and Kenneth L. Judd, eds., *Handbook of Computational Economics, Volume 2: Agent-Based Computational Economics*, Amsterdam: Elsevier, 2006, pp. 1187–1233.

_ **, W.Brian Arthur, and Richard Palmer**, "Time series properties of an artificial stock market," *Journal of Economic Dynamics and Control*, 1999, *23* (9-10), 1487–1516.

Li, Nian, Chen Gao, Mingyu Li, Yong Li, and Qingmin Liao, "EconAgent: Large Language Model-Empowered Agents for Simulating Macroeconomic Activities," in "Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)" Association for Computational Linguistics 2024, pp. 15523–15536.

Li, Yang, Yangyang Yu, Haohang Li, Zhi Chen, and Khaldoun Khashanah, "TradingGPT: Multi-agent system with layered memory and distinct characters for enhanced financial trading performance," arXiv:2309.03736 2023.

Li, Zeping, Guancheng Wan, Keyang Chen, Yu Chen, Yiwen Zhao, Philip Torr, Guangnan Ye, Zhenfei Yin, and Hongfeng Chai, "Behavioral consistency validation for LLM agents: An analysis of trading-style switching through stock-market simulation," arXiv:2602.07023 2026.

Lo, Andrew W. and A. Craig MacKinlay, "Stock Market Prices Do Not Follow Random Walks: Evidence from a Simple Specification Test," *Review of Financial Studies*, 1988, *1* (1), 41–66.

Long, J. Bradford De, Andrei Shleifer, Lawrence H. Summers, and Robert J. Waldmann, "Positive Feedback Investment Strategies and Destabilizing Rational Speculation," *The Journal of Finance*, 1990, *45* (2), 379–395.

Lopez-Lira, Alejandro, "Can large language models trade? Testing financial theories with LLM agents in market simulations," arXiv:2504.10789 2025.

_ **and Yuehua Tang**, "Can ChatGPT forecast stock price movements? Return predictability and large language models," arXiv:2304.07619 2023.

_ **,** _ **, and Mingyin Zhu**, "The memorization problem: Can we trust LLMs' economic forecasts?," arXiv:2504.14765 2025.

Lussange, Johann, Ivan Lazarevich, Sacha Bourgeois-Gironde, Stefano Palminteri, and Boris Gutkin, "Modelling Stock Markets by Multi-agent Reinforcement Learning," *Computational Economics*, 2021, *57* (1), 113–147.

Lux, Thomas, "Herd Behaviour, Bubbles and Crashes," *The Economic Journal*, 1995, *105* (431), 881–896.

_ **and Michele Marchesi**, "Scaling and criticality in a stochastic multi-agent model of a financial market," *Nature*, 1999, *397* (6719), 498–500.

_ **and** _ , "Volatility clustering in financial markets: A microsimulation of interacting agents," *International Journal of Theoretical and Applied Finance*, 2000, *3* (4), 675–702.

Mandelbrot, Benoit, "The Variation of Certain Speculative Prices," *The Journal of Business*, 1963, *36* (4), 394–419.

Manning, Benjamin S., Kehang Zhu, and John J. Horton, "Automated social science: Language models as scientist and subjects," arXiv:2404.11794 2024.

Mei, Qiaozhu, Yutong Xie, Walter Yuan, and Matthew O. Jackson, "A Turing test of whether AI chatbots are behaviorally similar to humans," *Proceedings of the National Academy of Sciences*, 2024, *121* (9), e2313925121.

Meng, Shuchen and Xupeng Chen, "Artificial intelligence and systemic risk: A unified model of performative prediction, algorithmic herding, and cognitive dependency in financial markets," arXiv:2604.03272 2026.

Menkveld, Albert J., "High frequency trading and the new market makers," *Journal of Financial Markets*, 2013, *16* (4), 712–740.

_ , "The Economics of High-Frequency Trading: Taking Stock," *Annual Review of Financial Economics*, 2016, *8* (1), 1–24.

Ouyang, Shumiao and Pengfei Sui, "Dissecting AI trading: Behavioral finance and market bubbles," arXiv:2604.18373 2026.

Papadakis, Charidimos, Giorgos Filandrianos, Angeliki Dimitriou, Maria Lymperaiou, Konstantinos Thomas, and Giorgos Stamou, "StockSim: A dual-mode order-level simulator for evaluating multi-agent LLMs in financial markets," arXiv:2507.09255 2025.

Park, Joon Sung, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein, "Generative Agents: Interactive Simulacra of Human Behavior," in "Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23)" ACM 2023, pp. 1–22.

Park, Peter S., Philipp Schoenegger, and Chongyang Zhu, "Diminished diversity-of-thought in a standard large language model," *Behavior Research Methods*, 2024, *56* (6), 5754–5770.

Pasquariello, Paolo, "Imperfect Competition, Information Heterogeneity, and Financial Contagion," *Review of Financial Studies*, 2007, *20* (2), 391–426.

Roll, Richard, "A Simple Implicit Measure of the Effective Bid-Ask Spread in an Efficient Market," *The Journal of Finance*, 1984, *39* (4), 1127–1139.

Saxena, Maxime, Marco Pangallo, Cars Hommes, Fabio Caccioli, and R. Maria del Rio-Chanona, "Machine spirits: Speculation and adaptation of LLM agents in asset markets," arXiv:2604.18602 2026.

Scharfstein, David S. and Jeremy C. Stein, "Herd behavior and investment," *American Economic Review*, 1990, *80* (3), 465–479.

Schoenegger, Philipp, Indre Tuminauskaite, Peter S. Park, Rafael Valdece Sousa Bastos, and Philip E. Tetlock, "Wisdom of the silicon crowd: LLM ensemble prediction capabilities rival human crowd accuracy," *Science Advances*, 2024, *10* (45), eadp1528.

Shleifer, Andrei and Robert Vishny, "Fire Sales in Finance and Macroeconomics," *Journal of Economic Perspectives*, 2011, *25* (1), 29–48.

_ **and Robert W. Vishny**, "Liquidation Values and Debt Capacity: A Market Equilibrium Approach," *The Journal of Finance*, 1992, *47* (4), 1343–1366.

_ **and** _ , "The Limits of Arbitrage," *The Journal of Finance*, 1997, *52* (1), 35–55.

Smith, Eric, J Doyne Farmer, László Gillemot, and Supriya Krishnamurthy, "Statistical theory of the continuous double auction," *Quantitative Finance*, 2003, *3* (6), 481–514.

Smith, Vernon L., "An Experimental Study of Competitive Market Behavior," *Journal of Political Economy*, 1962, *70* (2), 111–137.

_ **, Gerry L. Suchanek, and Arlington W. Williams**, "Bubbles, Crashes, and Endogenous Expectations in Experimental Spot Asset Markets," *Econometrica*, 1988, *56* (5), 1119–1151.

Song, Zichen, "Information homogenization induces herding in retrieval-augmented LLM agent markets," in "ICLR 2026 Workshop on Advances in Financial AI: Towards Agentic and Responsible Systems" 2026.

Struski, Pawel, Jakub Swistak, Inez Okulska, and Przemyslaw Biecek, "Competitive market behavior of LLMs," arXiv:2609.02580 2026.

Thurner, Stefan, J. Doyne Farmer, and John Geanakoplos, "Leverage causes fat tails and clustered volatility," *Quantitative Finance*, 2012, *12* (5), 695–707.

Vyetrenko, Svitlana, David Byrd, Nick Petosa, Mahmoud Mahfouz, Danial Dervovic, Manuela Veloso, and Tucker Balch, "Get real: Realism metrics for robust limit order book market simulations," in "Proceedings of the First ACM International Conference on AI in Finance (ICAIF '20)" ACM 2020, pp. 1–8.

Yang, Yuzhe, Yifei Zhang, Minghao Wu, Kaidi Zhang, Yunmiao Zhang, Honghai Yu, Yan Hu, and Benyou Wang, "TwinMarket: A scalable behavioral and social simulation for financial markets," arXiv:2502.01506; Advances in Neural Information Processing Systems 38 (NeurIPS 2025) 2025.

Yu, Yangyang, Haohang Li, Zhi Chen, Yuechen Jiang, Yang Li, Denghui Zhang, Rong Liu, Jordan W. Suchow, and Khaldoun Khashanah, "FinMem: A performance-enhanced LLM trading agent with layered memory and character design," arXiv:2311.13743 2023.

Zhang, Chong, Xinyi Liu, Zhongmou Zhang, Mingyu Jin, Lingyao Li, Zhenting Wang, Wenyue Hua, Dong Shu, Suiyuan Zhu, Xiaobo Jin, Sujian Li, Mengnan Du, and Yongfeng Zhang, "When AI meets finance (StockAgent): Large language model-based stock trading in simulated real-world environments," arXiv:2407.18957 2024.

Zhang, Wentao, Lingxuan Zhao, Haochong Xia, Shuo Sun, Jiaze Sun, Molei Qin, Xinyi Li, Yuqing Zhao, Yilei Zhao, Xinyu Cai, Longtao Zheng, Xinrun Wang, and Bo An, "A Multimodal Foundation Agent for Financial Trading: Tool-Augmented, Diversified, and Generalist," in "Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '24)" ACM 2024, pp. 4314–4325.

Appendices

A. Proofs

Throughout, A1–A4 hold and the notation is that of Section 3: $n_g = \phi_g N$, $\Phi = \sum_g \phi_g$, $H = \sum_g \phi_g^2$, $\sigma_s^2 = \sigma_v^2 + \sigma_\eta^2 + \sigma_u^2$, $\bar{\varepsilon} = N^{-1}\sum_i \varepsilon_i$ and $\bar{Y} = Y/(\beta N)$.

Two lemmas

Lemma 1 (Flow decomposition)**.** *$Y = \beta N(v + \bar{\varepsilon}) + z$ and $\mathrm{Var}(\bar{\varepsilon}) = \sigma_\eta^2\left[H + (1-\Phi)/N\right] + \sigma_u^2/N$. Consequently $\mathrm{Var}(\bar{Y} - v) = \Omega$ as defined in (6).*

Proof. By A2, $\sum_i x_i = \beta\sum_i(v + \varepsilon_i) = \beta N(v + \bar{\varepsilon})$. Let I be the set of traders on private models, $|I| = N(1-\Phi)$. Then $\sum_i \varepsilon_i = \sum_{g=1}^{G} n_g \eta_g + \sum_{i\in I}\eta_i + \sum_{i=1}^{N} u_i$, a sum of $G + |I| + N$ independent centered Gaussians (A1), whence $\mathrm{Var}(\sum_i \varepsilon_i) = \sigma_\eta^2 \sum_g n_g^2 + \sigma_\eta^2 N(1-\Phi) + N\sigma_u^2$. Dividing by N^2 and using $n_g^2/N^2 = \phi_g^2$ gives the variance of $\bar{\varepsilon}$. Finally $\bar{Y} - v = \bar{\varepsilon} + z/(\beta N)$ with z independent of $\bar{\varepsilon}$, so $\mathrm{Var}(\bar{Y} - v) = \mathrm{Var}(\bar{\varepsilon}) + \sigma_z^2/(\beta^2 N^2) = \Omega$. □

Lemma 2 (Projection)**.** *$\lambda = \tau/(\beta N)$, $\tau = \sigma_v^2/(\sigma_v^2 + \Omega)$, $P = \tau\bar{Y}$, and $\mathrm{Var}(P - v) = \mathrm{Var}(v \mid Y) = \sigma_v^2(1-\tau)$.*

Proof. (v, Y) is jointly Gaussian, so $\mathbb{E}[v \mid Y] = \lambda Y$ with $\lambda = \mathrm{Cov}(v,Y)/\mathrm{Var}(Y)$ and $\mathrm{Var}(v \mid Y) = \sigma_v^2 - \mathrm{Cov}(v,Y)^2/\mathrm{Var}(Y)$. By Lemma 1, $\mathrm{Cov}(v,Y) = \beta N\sigma_v^2$ and $\mathrm{Var}(Y) = \beta^2 N^2(\sigma_v^2 + \Omega)$. Hence $\tau = 1 - \mathrm{Var}(v \mid Y)/\sigma_v^2 = \mathrm{Cov}(v,Y)^2/(\sigma_v^2\,\mathrm{Var}(Y)) = \sigma_v^2/(\sigma_v^2 + \Omega)$ and $\lambda = \sigma_v^2/[\beta N(\sigma_v^2 + \Omega)] = \tau/(\beta N)$, so $P = \lambda Y = \tau\bar{Y}$. For the pricing error,

$$\begin{aligned}\mathrm{Var}(P - v) &= \lambda^2\,\mathrm{Var}(Y) - 2\lambda\,\mathrm{Cov}(v,Y) + \sigma_v^2 = \frac{\mathrm{Cov}(v,Y)^2}{\mathrm{Var}(Y)} - 2\frac{\mathrm{Cov}(v,Y)^2}{\mathrm{Var}(Y)} + \sigma_v^2 \\ &= \sigma_v^2 - \frac{\mathrm{Cov}(v,Y)^2}{\mathrm{Var}(Y)} = \mathrm{Var}(v \mid Y),\end{aligned}$$

which equals $\sigma_v^2(1-\tau)$ by definition of τ. □

Proof of Proposition 1

With a single model of $n = \phi N$ users, $H = \phi^2$ and $\Phi = \phi$, so Lemma 1 gives (8). Since $\tau = \sigma_v^2/(\sigma_v^2 + \Omega)$ is strictly decreasing in Ω, every monotonicity claim about τ is the reverse claim about Ω.

(i) Discrete monotonicity. Writing Ω as a function of the integer n, $\Omega(n) = \sigma_\eta^2\,(n^2 - n + N)/N^2 + C$ with $C = \sigma_u^2/N + \sigma_z^2/(\beta^2 N^2)$. Then

$$\Omega(n+1) - \Omega(n) = \frac{\sigma_\eta^2}{N^2}\Big[(n+1)^2 - (n+1) - n^2 + n\Big] = \frac{2n\,\sigma_\eta^2}{N^2},$$

which is zero for $n = 0$ and strictly positive for every $n \geq 1$ when $\sigma_\eta^2 > 0$. Hence $\Omega(0) = \Omega(1) < \Omega(2) < \cdots < \Omega(N)$ and τ is strictly decreasing on $\{1/N, \ldots, 1\}$, with $\tau(0) = \tau(1/N)$. For $N = 1$ the grid is $\{0, 1\}$ and $\Omega(0) = \Omega(1)$. In the continuous relaxation, $\Omega'(\phi) = \sigma_\eta^2(2\phi - 1/N)$, which is strictly positive if and only if $\phi > 1/(2N)$; so $\partial\tau/\partial\phi = -\sigma_v^2\Omega'(\phi)/(\sigma_v^2 + \Omega)^2 < 0$ exactly there. (For $0 < \phi < 1/(2N)$ the relaxation is slightly increasing in ϕ; no integer population lies in this region, since it corresponds to fewer than one-half of a trader using the model.)

(ii) Uniform bound and limit. For every N and $\phi \in [0,1]$, $\Omega(\phi) - \sigma_\eta^2\phi^2 = \sigma_\eta^2(1-\phi)/N + \sigma_u^2/N + \sigma_z^2/(\beta^2 N^2) \geq 0$, so $\Omega(\phi) \geq \sigma_\eta^2\phi^2$ and $\tau(\phi) \leq \sigma_v^2/(\sigma_v^2 + \phi^2\sigma_\eta^2) = \tau_\infty(\phi)$, which is strictly below one when $\phi\sigma_\eta > 0$. The difference $\Omega(\phi) - \sigma_\eta^2\phi^2$ is $O(1/N)$ for fixed ϕ and fixed σ_z^2, so $\tau(\phi) \to \tau_\infty(\phi)$.

(iii) Private-model benchmark. At $\phi = 0$, $\Omega = (\sigma_\eta^2 + \sigma_u^2)/N + \sigma_z^2/(\beta^2 N^2)$, which tends to zero whenever $\sigma_z^2/N^2 \to 0$; then $\tau \to 1$. If instead noise trading scales with the market, $\sigma_z^2 = cN^2$, the limit is $\sigma_v^2/(\sigma_v^2 + c/\beta^2) < 1$ even at $\phi = 0$; this is the usual noisy-rational-expectations bound and is unrelated to model sharing. □

Proof of Proposition 2

(i) By Lemma 2, $\mathrm{Var}(P - v) = \sigma_v^2(1-\tau) = \sigma_v^2\Omega/(\sigma_v^2 + \Omega)$, an increasing function of Ω; the monotonicity in ϕ and the uniform lower bound follow from parts (i)–(ii) of Proposition 1 since $y \mapsto \sigma_v^2 y/(\sigma_v^2 + y)$ is increasing.

(ii) By A2, $\mathrm{Corr}(x_i, x_j) = \mathrm{Corr}(s_i, s_j)$ since $\beta > 0$. By A1, $\mathrm{Var}(s_i) = \sigma_s^2$ for every i, and for $i \neq j$, $\mathrm{Cov}(s_i, s_j) = \sigma_v^2 + \mathrm{Cov}(\eta_{g(i)}, \eta_{g(j)})$, which equals $\sigma_v^2 + \sigma_\eta^2$ if $g(i) = g(j) \neq \varnothing$ and σ_v^2 otherwise (two private models, or two different shared models, have independent η's). This gives ρ_1 and ρ_0, and $\rho_1 - \rho_0 = \sigma_\eta^2/\sigma_s^2 > 0$ if and only if $\sigma_\eta^2 > 0$.

(iii) Among the $N(N-1)$ ordered pairs $i \neq j$, exactly $n(n-1)$ have both members on the shared model; all others have correlation ρ_0. Hence $\bar\rho = \rho_0 + w(\rho_1 - \rho_0)$ with $w = n(n-1)/[N(N-1)] = \phi(N\phi - 1)/(N-1)$. As a function of real ϕ, $w'(\phi) = (2N\phi - 1)/(N-1)$ and $w''(\phi) = 2N/(N-1) > 0$: w is convex and increasing for $\phi > 1/(2N)$. On the integer grid, $w(n+1) - w(n) = 2n/[N(N-1)] > 0$ for $n \geq 1$. As $N \to \infty$, $w \to \phi^2$.

(iv) Let (x_i, x_j) be centered jointly Gaussian with correlation $\rho \in (-1, 1)$. Write $x_j/\sigma_j = \rho\, x_i/\sigma_i + \sqrt{1-\rho^2}\,\xi$ with ξ standard normal independent of x_i. The classical orthant formula gives $\Pr(x_i > 0, x_j > 0) = \frac{1}{4} + \arcsin(\rho)/(2\pi)$; by symmetry $\Pr(x_i < 0, x_j < 0)$ is the same, so

$S(\rho) = \Pr(x_i x_j > 0) = \frac{1}{2} + \arcsin(\rho)/\pi$.[29] Since arcsin is strictly increasing on $[-1,1]$, so is S. Averaging over ordered pairs with the same count as in (iii) gives (11). Because $\bar{\rho}$ is affine in w with slope $\rho_1 - \rho_0 > 0$ (when $\sigma_\eta^2 > 0$), we may regard $\bar{S}$ as a function of $\bar{\rho}$ with

$$\frac{d\bar{S}}{d\bar{\rho}} = \frac{1}{\pi} \cdot \frac{\arcsin\rho_1 - \arcsin\rho_0}{\rho_1 - \rho_0} > 0.$$

For the Jensen bound, $(\arcsin)''(\rho) = \rho(1-\rho^2)^{-3/2} \geq 0$ on $[0,1)$, and $\rho_0, \rho_1 \geq 0$, so $(1-w)\arcsin\rho_0 + w\arcsin\rho_1 \geq \arcsin\big((1-w)\rho_0 + w\rho_1\big) = \arcsin\bar{\rho}$, with equality if and only if $\rho_0 = \rho_1$ or $w \in \{0,1\}$; for $0 < w < 1$ equality thus requires $\sigma_\eta^2 = 0$.

The placebo. If $\sigma_\eta^2 = 0$ then Ω does not depend on ϕ (the bracket in (8) is multiplied by zero), $\rho_1 = \rho_0$, and w drops out of (10) and (11). Hence none of the four objects varies with ϕ. This shows that the restriction $\sigma_\eta^2 > 0$ is necessary for every strict claim in the proposition. □

Proof of Proposition 3

(i) By Lemma 1, $\Omega = \sigma_\eta^2[H + (1-\Phi)/N] + \sigma_u^2/N + \sigma_z^2/(\beta^2 N^2)$; for fixed Φ the shares enter only through H, and $\partial\Omega/\partial H = \sigma_\eta^2 > 0$, so $\partial\tau/\partial H = -\sigma_v^2\sigma_\eta^2/(\sigma_v^2+\Omega)^2 < 0$. Setting $\Phi = 1$ and $H = 1/M_{\text{eff}}$ gives the stated form.

(ii) By the Cauchy–Schwarz inequality, $\Phi^2 = \big(\sum_g \phi_g \cdot 1\big)^2 \leq \big(\sum_g \phi_g^2\big)\big(\sum_g 1^2\big) = GH$, with equality if and only if (ϕ_g) is proportional to $(1,\ldots,1)$, i.e. $\phi_g = \Phi/G$. For the upper bound, $\Phi^2 = \sum_g \phi_g^2 + \sum_{g\neq g'} \phi_g\phi_{g'} \geq H$, with equality if and only if all cross products vanish, i.e. at most one share is positive. In the integer version one minimizes $\sum_g n_g^2$ subject to $\sum_g n_g = \Phi N$: if some $n_a \geq n_b + 2$, moving one trader from a to b changes the sum by $(n_a-1)^2 + (n_b+1)^2 - n_a^2 - n_b^2 = -2(n_a - n_b - 1) < 0$, so at a minimum all n_g differ by at most one.

(iii) Combine (i) and (ii): $H_{\text{single}} = \Phi^2 \geq H_{\text{mixed}} \geq \Phi^2/G = H_{\text{uniform}}$ and τ is decreasing in H. Strictness: $H_{\text{mixed}} < \Phi^2$ as soon as two shares are positive, and $\partial\tau/\partial H < 0$ requires $\sigma_\eta^2 > 0$.

(iv) Ordered pairs with both members on the same shared model number $\sum_g n_g(n_g-1) = N^2H - N\Phi$; dividing by $N(N-1)$ gives $w = (NH-\Phi)/(N-1)$. For fixed Φ, w is increasing in H, and $\bar{\rho}$ and $\bar{S}$ are increasing in w by the proof of Proposition 2. □

Proof of Proposition 4 and Corollary 1

Under A5 the round's price change solves $\Delta P = \lambda(Y_1 + \gamma\Delta P)$.

(i) If $\lambda\gamma \neq 1$ the unique solution is $\Delta P^* = \lambda Y_1/(1-\lambda\gamma) = \kappa\lambda Y_1$. The iterated response satisfies $\Delta P^{(k+1)} - \Delta P^* = \lambda\gamma(\Delta P^{(k)} - \Delta P^*)$, so $\Delta P^{(k)} - \Delta P^* = (\lambda\gamma)^k(\Delta P^{(0)} - \Delta P^*)$, which converges to

[29] For completeness: $\Pr(x_i > 0, x_j > 0) = \Pr(\vartheta \in (0,\pi/2) \text{ and } \vartheta \in (\alpha - \pi/2, \alpha + \pi/2))$ where ϑ is the uniformly distributed angle of a standard bivariate normal vector and $\cos\alpha = \rho$; the two arcs overlap on a set of measure $\pi - \alpha$, so the probability is $(\pi - \arccos\rho)/(2\pi) = \frac{1}{4} + \arcsin(\rho)/(2\pi)$.

zero for every starting point if and only if $|\lambda\gamma| < 1$; with $\gamma \geq 0$ this is $\lambda\gamma < 1$. Then $\mathrm{Var}(\Delta P) = \kappa^2\lambda^2\,\mathrm{Var}(Y_1)$, i.e. amplification by $\kappa^2 = 1/(1-\lambda\gamma)^2$ relative to the no-feedback price change λY_1.

(ii) If $\lambda\gamma = 1$ the equation reads $\Delta P = \lambda Y_1 + \Delta P$, impossible for $Y_1 \neq 0$, and the iteration gives $\Delta P^{(k)} = \Delta P^{(0)} + k\lambda Y_1$, unbounded. If $\lambda\gamma > 1$ the algebraic solution $\Delta P^* = \lambda Y_1/(1-\lambda\gamma)$ exists but has the sign of $-Y_1$ (a price that falls when the flow is a net purchase), and since $(\lambda\gamma)^k \to \infty$ the iteration diverges from every $\Delta P^{(0)} \neq \Delta P^*$; in particular from the natural start $\Delta P^{(0)} = \lambda Y_1$. This is the counterexample announced in the text: the closed form $1/(1-\lambda\gamma)^2$ is meaningless outside $\lambda\gamma < 1$.

(iii) $\Delta P^* = \kappa\lambda Y_1$ is a strictly monotone function of Y_1 (for $\kappa > 0$), so $\sigma(\Delta P^*) = \sigma(Y_1)$ and $\mathrm{Var}(v \mid P_1) = \mathrm{Var}(v \mid Y_1)$: τ is unchanged. Using $\lambda^2\,\mathrm{Var}(Y_1) = \lambda\,\mathrm{Cov}(v, Y_1) = \sigma_v^2\tau$ (Lemma 2),

$$\mathrm{Var}(\kappa\lambda Y_1 - v) = \kappa^2\lambda^2\,\mathrm{Var}(Y_1) - 2\kappa\lambda\,\mathrm{Cov}(v, Y_1) + \sigma_v^2 = \sigma_v^2\left[\kappa^2\tau - 2\kappa\tau + 1\right] = \sigma_v^2\left[(1-\tau) + \tau(\kappa-1)^2\right].$$

On $\gamma \in [0, 1/\lambda)$, κ increases from 1 to $+\infty$, so $(\kappa-1)^2$ is increasing in γ.

Corollary. Under A6 the total flow is $Y_1 + \gamma_T\Delta P + \sum_i \kappa_{g(i)}\Delta P = Y_1 + \left(\gamma_T + \sum_g n_g\kappa_g\right)\Delta P = Y_1 + \left(\gamma_T + N\sum_g \phi_g\kappa_g\right)\Delta P$, with $\kappa_\varnothing \equiv 0$ for private-model traders (any such loading can be folded into γ_T). Applying (i)–(ii) with $\gamma = \gamma_T + N\sum_g \phi_g\kappa_g$ gives the statements; with a single model, $\gamma(\phi) = \gamma_T + N\phi\kappa$ is increasing in ϕ when $\kappa > 0$, $\kappa^2(\phi)$ is increasing in γ hence in ϕ, and $\lambda\gamma(\phi) < 1$ if and only if $\phi < \phi^* = (1/\lambda - \gamma_T)/(N\kappa)$. If $\kappa = 0$, γ does not depend on ϕ. With several models, $\sum_g \phi_g\kappa_g$ is a share-weighted average that lies between $\Phi\min_g \kappa_g$ and $\Phi\max_g \kappa_g$; if the κ_g differ in sign it can be smaller in absolute value than $\Phi\kappa_g$ for every g. □

Proof of Proposition 5

Let $R \equiv \{\Delta P : \Delta P < -b/X\}$ be the region in which $F > 0$. On R, $F(\Delta P) = (-X\Delta P - b)/m$ and the fixed-point equation $\Delta P = \Delta P_0 - \lambda F(\Delta P)$ reads

$$\Delta P = \Delta P_0 + a\,\Delta P + \frac{\lambda b}{m}, \qquad a = \frac{\lambda X}{m}. \tag{15}$$

Off R, $F = 0$ and the equation reads $\Delta P = \Delta P_0$.

(i) If $\Delta P_0 \geq -b/X$, then $\Delta P = \Delta P_0 \notin R$ solves the equation, and the cascade defined below stops at its first step since $F(\Delta P_0) = 0$. For $a < 1$ it is the only fixed point: a solution in R would satisfy (15), whose candidate $(\Delta P_0 + \lambda b/m)/(1-a)$ lies in R if and only if $\Delta P_0 + \lambda b/m < -(b/X)(1-a) = -b/X + \lambda b/m$, i.e. $\Delta P_0 < -b/X$, which fails. For $a > 1$ and $b > 0$ the same algebra shows that the candidate *does* lie in R whenever $\Delta P_0 > -b/X$ (dividing by $1-a < 0$ reverses the inequality): a second, self-fulfilling fixed point exists in which the cohort is forced to sell because the price has fallen enough to exhaust the buffer, and the price has fallen because the

cohort sells. This is the multiplicity familiar from Brunnermeier and Pedersen (2009). We select the cascade limit, which is ΔP_0; the simulator's forced-liquidation rule is likewise sequential and cannot jump to the self-fulfilling point.

(ii) Suppose $\Delta P_0 < -b/X$ and $a < 1$. The candidate $\Delta P = \Delta P_0$ is in R, where $F > 0$, so it is not a solution; the affine equation (15) has the unique solution $\Delta P^* = (\Delta P_0 + \lambda b/m)/(1-a)$, which lies in R by the equivalence just shown. With $b = 0$ this is $\Delta P_0/(1-\lambda X/m)$ and the multiplier $1/(1-a)$ exceeds one. For the cascade, let $\Delta P^{(0)} = \Delta P_0$ and $\Delta P^{(k)} = \Delta P_0 - \lambda F(\Delta P^{(k-1)})$. Since $\Delta P^{(0)} \in R$, $\Delta P^{(1)} - \Delta P^{(0)} = -\lambda F(\Delta P_0) < 0$, so $\Delta P^{(1)} \in R$; by induction all iterates are in R and satisfy $\Delta P^{(k)} - \Delta P^* = a(\Delta P^{(k-1)} - \Delta P^*)$, hence

$$\Delta P^{(k)} - \Delta P^* = a^k(\Delta P_0 - \Delta P^*) = -a^{k+1}\,\frac{|\Delta P_0| - b/X}{1-a}, \tag{16}$$

where we used $\Delta P_0 - \Delta P^* = -\frac{\lambda}{m}\frac{X\Delta P_0 + b}{1-a} = a\frac{|\Delta P_0| - b/X}{1-a}$. The sequence is decreasing and converges to ΔP^* since $a < 1$.

(iii) Suppose $\Delta P_0 < -b/X$ and $a \geq 1$. No solution exists off R (it would have to be $\Delta P_0 \in R$). On R, if $a = 1$ then $\lambda b/m = (b/X)(\lambda X/m) = b/X$ and (15) reduces to $0 = \Delta P_0 + b/X$, which is false because $\Delta P_0 < -b/X$; if $a > 1$ the unique affine solution $(\Delta P_0 + \lambda b/m)/(1-a)$ is the ratio of a negative numerator and a negative denominator, hence positive, so it is not in R. Thus there is no fixed point. Along the cascade, the increments obey $\Delta P^{(k+1)} - \Delta P^{(k)} = -\lambda[F(\Delta P^{(k)}) - F(\Delta P^{(k-1)})] = a(\Delta P^{(k)} - \Delta P^{(k-1)})$ as long as the iterates remain in R, which they do because the first increment is negative and the ratio a is positive; with $a \geq 1$ the increments do not shrink, so $\Delta P^{(k)} \to -\infty$. In the simulator the sequence stops at the price band.

(iv) $\sum_{i\in L} x_i = \beta \sum_{i\in L} s_i$ is a centered Gaussian (A1–A2) with

$$\operatorname{Var}\Big(\sum_{i\in L} s_i\Big) = \sum_{i\in L}\operatorname{Var}(s_i) + \sum_{i\neq j\in L}\operatorname{Cov}(s_i, s_j) = \sigma_s^2\big[N_L + N_L(N_L-1)\bar\rho_L\big],$$

by definition of the cohort-average correlation $\bar\rho_L$. Hence $X = |Z|$ with $Z \sim \mathcal{N}(0, \sigma_X^2)$, $\sigma_X = \beta\sigma_s\sqrt{N_L[1+(N_L-1)\bar\rho_L]}$. Standard facts give $\mathbb{E}|Z| = \sigma_X\sqrt{2/\pi}$ and $\Pr(|Z| \geq c) = 2\bar\Phi(c/\sigma_X)$ for $c > 0$, both strictly increasing in σ_X, which is strictly increasing in $\bar\rho_L$ when $N_L > 1$. Finally, applying Proposition 2(iii) to the cohort (with N_L in place of N and the cohort's own shared-model share), $\bar\rho_L$ is increasing in that share on its admissible grid. The spiral probability is $\Pr(a \geq 1) = \Pr(X \geq m/\lambda)$, which gives the stated expression with $c = m/\lambda$.

Remark on the net position. If the cohort contains both longs and shorts, a negative shock forces sales only from the longs, whose gross exposure $\sum_{i\in L}\max\{x_i, 0\}$ is at least $\max\{\sum_{i\in L} x_i, 0\}$; using the net position in the spiral coefficient therefore understates a for a shock against the cohort's net side, and the spiral condition stated is sufficient, not necessary. □

Proof of Proposition 6

(i) If $X \le L < m/\lambda$ then $a = \lambda X/m \le \lambda L/m < 1$ for every realization, so case (iii) of Proposition 5 cannot occur and, by case (ii), the multiplier is at most $1/(1-\lambda L/m)$. For the informational cost, let $g(s) = \min\{\max\{\beta s, -\ell\}, \ell\}$, which is Lipschitz with $g'(s) = \beta\,\mathbf{1}\{|\beta s| < \ell\}$ almost everywhere. Since (v, s_i) is jointly Gaussian, write $v = \theta s_i + e$ with $\theta = \mathrm{Cov}(v, s_i)/\mathrm{Var}(s_i)$ and e independent of s_i. Then $\mathrm{Cov}(v, g(s_i)) = \theta\,\mathrm{Cov}(s_i, g(s_i)) = \theta\,\mathrm{Var}(s_i)\mathbb{E}[g'(s_i)]$ by Stein's lemma, i.e. $\mathrm{Cov}(v, \tilde{x}_i) = \mathrm{Cov}(v, s_i)\,\beta\,\Pr(|\beta s_i| < \ell) = \Pr(|\beta s_i| < \ell)\,\mathrm{Cov}(v, x_i)$, and $\Pr(|\beta s_i| < \ell) < 1$ for any finite ℓ. We do not claim a closed form for τ under clipping, since the clipped flow is non-Gaussian and the market maker's pricing rule is then nonlinear.

(ii) Under A7, after the halt the cascade is that of Proposition 5 with a replaced by $a_h \equiv \psi(h)a$ and the same starting point ΔP_0 (nothing trades during the halt). If $a_h < 1$, part (ii) of that proposition gives convergence to $\Delta P_h^* = (\Delta P_0 + \lambda b/m)/(1-a_h)$, finite regardless of whether $a < 1$. If also $a < 1$, then $1 - a_h \ge 1 - a > 0$ and the common numerator is negative, so $|\Delta P_h^*| \le |\Delta P^*|$, with strict inequality if $\psi(h) < 1$.

(iii) From (16), $|\Delta P^{(k)} - \Delta P^*| = a^{k+1}(|\Delta P_0| - b/X)/(1-a)$, which is at most ε if and only if $k + 1 \ge \log\big(\varepsilon(1-a)/(|\Delta P_0| - b/X)\big)/\log a$ (the division by $\log a < 0$ reverses the inequality). Hence $T_\varepsilon(a) = \max\{0, \lceil \log(\varepsilon(1-a)/(|\Delta P_0| - b/X))/\log a \rceil - 1\}$, which is the stated formula when $b = 0$. Under the halt, rounds $1, \ldots, h$ leave the price at ΔP_0 and the cascade with coefficient a_h then starts from ΔP_0, so the first round within ε of ΔP_h^* is $h + T_\varepsilon(a_h)$. Since $T_\varepsilon \ge 0$, $T_\varepsilon^h \ge h$; if $\psi(h) = 1$ then $a_h = a$, $\Delta P_h^* = \Delta P^*$ and $T_\varepsilon^h = T_\varepsilon(a) + h$. When $a \ge 1$ and $a_h < 1$ the comparison is $T_\varepsilon^h < \infty = T_\varepsilon(a)$, where the latter is infinite because the no-halt cascade has no finite limit (Proposition 5(iii)). □

B. Prompts

This appendix reproduces verbatim the frozen prompt version `v1` of `llm/prompting.py`, one rendered example, the JSON schema sent to the providers, and the validation rules of `llm/policy.py`. The system prompt is the persona sentence followed by the rules block; the user message is a deterministic rendering of the agent's `Observation`. Nothing in either text depends on the latent state.

Personas (`ROLE_PERSONAS`)

The single-prompt design uses `generalist` for every LLM slot; the diverse-prompt design assigns the eight personas cyclically in slot order.

```
generalist : You are a professional trader managing your own account in a single-stock
    electronic market.
value : You are a value investor: you compare the price to the present value of expected
     dividends and trade mispricings patiently.
momentum : You are a momentum trader: you follow recent price trends and cut positions
    when the trend reverses.
contrarian : You are a contrarian trader: you fade sharp moves that are not justified by
     dividend news.
liquidity : You are a liquidity provider: you prefer resting limit orders near the best
    quotes and keep your inventory small.
risk_averse: You are a risk-averse trader: you keep positions small, avoid market orders
     and prefer to hold when uncertain.
macro : You are a news-driven trader: you weigh public announcements and dividend
    surprises and act when the market has not yet reacted.
quant : You are a quantitative trader: you reason about expected value per share, spread
     costs and inventory before each order.
```

Rules template (RULES)

Placeholders in braces are filled from the configuration and the observation (`div_interval`, `mu`, `r_pct`, `pv_mean`, `rho`, `tick`, `taker_bps`, `ttl`, `max_qty`, `interval`); doubled braces are Python format escapes and render as single braces.

```
Market mechanics (read carefully):
- One risky stock paying a cash dividend every {div_interval} ticks; the long-run mean
    dividend is {mu:.2f} per share per period and the per-period discount rate is {r_pct
    :.2f}%, so a share paying the mean dividend forever is worth about {pv_mean:.2f}.
- Dividends deviate persistently from the mean (autocorrelation {rho:.2f} per period).
    The deviation of the NEXT dividend is not yet announced; public news and, when you
    have one, your private signal are noisy estimates of it (in cents). The present
    value of a share rises by about 1/(1+r-rho) ≈ 44 cents per cent of next-dividend
    deviation.
- Prices are in currency units with a tick of {tick:.2f}. Limit orders rest in the book;
     market orders execute immediately against the best quotes (taker fee {taker_bps:.1f}
     bps). Orders may be partially filled. Unfilled limit orders expire after {ttl}
    ticks.
- You cannot short sell and cannot spend more than your free cash. Maximum order size: {
    max_qty} shares.
- You decide every {interval} ticks. Between decisions the market keeps trading.

Respond with ONLY a JSON object with exactly these keys:
{{"action": "hold"|"place"|"cancel", "side": "buy"|"sell"|null, "order_type": "limit"|"
    market"|null, "quantity": integer|null, "limit_price": number|null, "
```

```
    cancel_order_ids": [integers], "confidence": number in [0,1]|null, "rationale": "one
     sentence, max 200 characters"}}
"place" requires side, order_type and quantity (and limit_price for a limit order). "
    cancel" requires cancel_order_ids. Anything else is treated as hold.
```

The placeholder `interval` is filled with the dividend interval (`obs.div_interval`), not with `decision_interval`; the two coincide (20) in every reported design.

Rendered example

The following system prompt and user message were produced by running the prompt builder on the observation of an informed LLM slot (agent 9, persona `generalist`, tick 110) in a mock run of the baseline configuration with $\varphi = 0.25$, seed 1.[30]

```
You are a professional trader managing your own account in a single-stock electronic
    market.

Market mechanics (read carefully):
- One risky stock paying a cash dividend every 20 ticks; the long-run mean dividend is
    0.50 per share per period and the per-period discount rate is 0.25%, so a share
    paying the mean dividend forever is worth about 200.00.
- Dividends deviate persistently from the mean (autocorrelation 0.98 per period). The
    deviation of the NEXT dividend is not yet announced; public news and, when you have
    one, your private signal are noisy estimates of it (in cents). The present value of
    a share rises by about 1/(1+r-rho) ≈ 44 cents per cent of next-dividend deviation.
- Prices are in currency units with a tick of 0.01. Limit orders rest in the book;
    market orders execute immediately against the best quotes (taker fee 2.0 bps).
    Orders may be partially filled. Unfilled limit orders expire after 50 ticks.
- You cannot short sell and cannot spend more than your free cash. Maximum order size:
    50 shares.
- You decide every 20 ticks. Between decisions the market keeps trading.

Respond with ONLY a JSON object with exactly these keys:
{"action": "hold"|"place"|"cancel", "side": "buy"|"sell"|null, "order_type": "limit"|"
    market"|null, "quantity": integer|null, "limit_price": number|null, "
    cancel_order_ids": [integers], "confidence": number in [0,1]|null, "rationale": "one
     sentence, max 200 characters"}
"place" requires side, order_type and quantity (and limit_price for a limit order). "
    cancel" requires cancel_order_ids. Anything else is treated as hold.
```

```
Time: tick 110. Next dividend in 10 ticks.
```

[30] Generated with `system_prompt()` and `user_message()` of `llm/prompting.py`; the observation was captured from `Simulation._observe`. The mid-price history is sub-sampled to every sixth tick of the 61-tick window.

```
Best bid 200.80 / best ask 201.12 / mid 200.96 / last trade 201.12.
Bid depth (price x shares): 200.80x14, 200.70x28, 200.69x13, 200.68x14, 200.65x12. Ask
    depth: 201.12x14, 201.15x4, 201.19x8, 201.20x14, 201.23x4.
Mid-price history (oldest to newest, 61 ticks sampled): 200.20, 200.29, 200.48, 200.09,
    200.54, 200.66, 200.64, 200.64, 200.60, 200.73, 200.96.
Change over the window: +0.38%. Recent volatility (std of tick returns): 0.042%.
Last 30 trades: volume 168 shares, 57% buyer-initiated.
Last dividend paid: 0.55 per share. Public estimate of the next dividend's deviation
    from its mean: public estimate +5.0 cents. Present value of expected dividends using
     only public information: 202.23.
Your private signal: next dividend deviation estimated at +3.6 cents (signal precision
    2.0; higher is more reliable).
Your account: cash 18634.09 (free 18634.09), holdings 92 shares, equity 37122.41.
You have no open orders.
Decide now. Reply with the JSON object only.
```

Optional lines, in the order in which they are inserted when the corresponding treatment is active: `TRADING IS HALTED on this stock (orders will be rejected).` after the first line; up to three `Public news (tick t): ...` lines after the public-value line; `Your open orders: id ...: buy/sell remaining/qty @ price (age)` (at most six) instead of `You have no open orders.`; `Feedback: your equity changed by ±x% since your last decision (cumulative ±y%).` (adaptive feedback); `Your recent decisions and outcomes:` followed by up to five bullet lines (memory); `Messages from other traders in your group (unverified opinions):` followed by up to five bullet lines (communication channel).

JSON schema (`DECISION_JSON_SCHEMA`)

Sent as `text.format` (OpenAI, `strict: true`) and `output_config.format` (Anthropic). Every key is required; `null` is allowed where indicated; no additional properties.

```
{
  "type": "object",
  "properties": {
    "action": {"type": "string", "enum": ["hold", "place", "cancel"]},
    "side": {"type": ["string", "null"], "enum": ["buy", "sell", null]},
    "order_type": {"type": ["string", "null"], "enum": ["limit", "market", null]},
    "quantity": {"type": ["integer", "null"]},
    "limit_price": {"type": ["number", "null"]},
    "cancel_order_ids": {"type": "array", "items": {"type": "integer"}},
    "confidence": {"type": ["number", "null"]},
    "rationale": {"type": "string"}
  },
  "required": ["action", "side", "order_type", "quantity", "limit_price",
```

```
            "cancel_order_ids", "confidence", "rationale"],
  "additionalProperties": false
}
```

Parsing and validation rules

The reply is first parsed, then validated against the pydantic model `Decision`, then checked for feasibility by `LLMPolicy.to_action`. Any failure yields a hold action with `valid = False` and the reason string below; a provider error or a budget halt yields a hold with the error string as reason.

1. **Parsing** (`parse_decision`): whitespace and a surrounding ‘‘‘/‘‘‘ `json` fence are stripped; if the text is not valid JSON, the first `{...}` block is tried; failure → reason `json: <message>`.
2. **Schema** (`Decision`): `action` ∈ {hold, place, cancel}; `side` ∈ {buy, sell, null}; `order_type` ∈ {limit, market, null}; `quantity` integer ≥ 0 or null; `limit_price` number > 0 or null (currency units); `cancel_order_ids` list of integers; `confidence` in $[0,1]$ or null; `rationale` trimmed to 400 characters; failure → reason `schema: <field>: <message>` (first three errors).
3. **hold**: always valid; no intent.
4. **cancel**: the ids are intersected with the agent's open orders (resting and in flight); empty intersection → `cancel: no valid order ids`; otherwise one cancel intent per surviving id.
5. **place**, in this order:
   (a) side, order type and quantity must all be present (`place: missing side/order_type/quantity`);
   (b) quantity > 0 (`place: quantity <= 0`) and $\leq$ `max_order_qty` $= 50$ (`place: quantity q > max 50`);
   (c) the asset must not be halted (`place: market halted`);
   (d) limit orders need a price (`place: limit without price`); the price is converted to ticks by $\text{round}(100\,\text{price}/\text{tick_cents})$ and must be positive and within $\pm 20\%$ of the midpoint (`place: limit price outside ±20% of mid`);
   (e) market orders need a non-empty opposite side (`place: market order with empty opposite side`);
   (f) buys: $\lfloor \text{ref} \times \text{tick_cents} \times q \times 1.001 \rfloor \leq$ free cash, where ref is the limit price or else the best ask or midpoint (`place: cost c > free cash f`);
   (g) sells: q is clipped to holdings minus the remaining quantity of the agent's resting sells; if nothing is free the decision is invalid (`place: no free shares to sell`); a clipped sell is recorded as valid.
6. The surviving intent is tagged `llm` and submitted to the exchange, which applies its own checks (price band, cash reservation, position limit, leverage); an exchange rejection is logged in the event log, not in the decision log.

C. Pseudocode

The five algorithms below follow the control flow of the code line by line; names in typewriter font are the corresponding functions. Δ is the dividend interval, T the number of ticks.

Algorithm 1 Tick loop (`Simulation.run`, `experiments/runner.py`)

build streams from the master seed; draw fundamentals, population, accounts, latencies; register accounts
pending_liq $\leftarrow \emptyset$; record P_0
for $t = 0, \ldots, T-1$ **do**
    exchange.time $\leftarrow t$
    if $t > 0$ and $t \bmod \Delta = 0$ **then**
        $D \leftarrow$ `fund.step()` ▷ pay D_n, reveal x_n, update public filter, draw latent x_{n+1}
        pay round(D_k) cents per lot of every asset k; check conservation
    end if
    `apply_shocks`(t) ▷ after the dividend: a jump stays latent until news or the next dividend
    for all news items drawn this tick **do**
        update the public filter with the item's value and stated noise
    end for
    for all $(i, \text{act}) \in$ pending_liq **do**
        `exchange.submit`(i, act, t)
    end for; pending_liq $\leftarrow \emptyset$
    $u \leftarrow$ `activation` stream, one uniform per slot
    $A_c \leftarrow \{i \text{ classical} : u_i < \text{activity}_i\}$; shuffle A_c with the `shuffle` stream
    $A_\ell \leftarrow \{i \text{ LLM} : (t - \text{offset}_i) \bmod \text{interval} = 0,\ t \geq$ `warmup`$/2\}$, minus agents at their decision cap
    tw $\leftarrow$ last 30 trades per asset
    for all $i \in A_c$ **do**
        $o \leftarrow$ `observe`(i, t, tw) ▷ book depth 5, 60-tick mids, trades, dividend, V^{pub}, $\hat{x}$, news, fresh private signal, account, open orders
        $a \leftarrow \text{policy}_i$.`decide`$(o, \text{rng}_i)$; **if** a is not hold **then** `exchange.submit`(i, a, t)
    end for
    if $A_\ell \neq \emptyset$ **then**
        build observations for all $i \in A_\ell$; update each agent's equity-change history
        $(a_i, \text{rec}_i)_{i \in A_\ell} \leftarrow$ `llm_policy.decide_batch` ▷ Algorithm 4, thread pool of width 8
        for all $i \in A_\ell$ **do**
            submit a_i if not hold; update memory and cohort channel from rec_i
        end for
    end if
    `exchange.process_until`$(t+1)$ ▷ arrivals with time $< t+1$ in (arrival, sequence) order: Algorithm 2
    pending_liq $\leftarrow$ `exchange.margin_check()` ▷ Algorithm 3
    `exchange.end_of_tick`(t) ▷ call auction if enabled and $(t+1) \bmod k = 0$; append midpoint; expire orders with $t -$ submit $\geq$ TTL; log tick
    record P_{t+1} and the tick row (mid, quotes, depth, volume, LLM volume, V_t, V_t^{pub}, x, halts, effective spread, total equity)
end for
final conservation check; write manifest, frames and outcomes

Algorithm 2 Order arrival (`Exchange._on_order`, `exchange/engine.py`)

Require: order o (agent i, asset k, side, type, q, price p or none, tag), account c_i, book B_k

remove o from the agent's in-flight set
reject `frozen` if c_i is frozen and tag $\neq$ liquidation; `halted` if k is halted; `bad_qty` unless $1 \le q \le$ max order
if limit **then**
    reject `bad_price` if $p \le 0$; `price_band` if $|p - \text{mid}_k| > 0.5\,\text{mid}_k$
end if
if buy and limit **then**
    need $\leftarrow p\,q + \max(\text{fee}_{\text{taker}}, \text{fee}_{\text{maker}}) + q$; **reject** `insufficient_cash` if need $>$ buying power
    reserve need cents on c_i and on o ▷ buying power = free cash, or free cash $+(L-1)$ equity if leveraged
else if buy and market **then**
    reject `no_liquidity` if no ask; $q \leftarrow$ largest $q' \le q$ whose notional plus taker fee on the total fits in buying power, walking the asks; **reject** `insufficient_cash` if $q = 0$
else ▷ sell
    $h \leftarrow$ free (unreserved) holdings; **if** $q > h$ **then reject** `insufficient_shares` if $h \le 0$ else $q \leftarrow h$ ▷ no short selling
    reject `no_liquidity` if market and no bid; reserve q lots
end if
if position limit set and tag $\neq$ liquidation **then**
    release and reject `position_limit` if $|\text{pos} + \text{side}\,(\text{same-side resting} + q)| >$ limit
end if
if margin check on, $L_i > 1$ and tag $\neq$ liquidation **then**
    gross$'$ $\leftarrow$ gross exposure with asset k revalued at the post-trade position and price (p or mid)
    release and reject `leverage_cap` if equity ≤ 0 or gross$' > L_i$ equity
end if
log `accept`
if call-auction mode **then**
    if market **then** convert to a limit at the band edge, tag `ioc`, reserve cash for buys; rest o in B_k; **return**
end if
fills $\leftarrow B_k$.`match`(o) ▷ walk the opposite side while crossing; price–time priority; resting price
for all fill (r, p', q') **do**
    fees: taker $=$ round$(p'q' \cdot 2/10^4)$ on o, maker $= 0$ on r; move $p'q'$ cents and q' lots; release the reservations consumed (all of it on the last fill); log `trade`; update last price and tick volume; check conservation
end for
if o has remaining quantity **then**
    if limit **then** rest o in B_k **else** release the reservation, cancel the remainder (`ioc`); count a rejection if nothing filled
end if
`maybe_halt`(k): if a breaker is set and $|\text{last}/\text{mid}_{t-10} - 1| >$ threshold, halt k for 20 ticks and cancel all resting orders (releasing reservations)

Algorithm 3 Margin check and liquidation (`Exchange.margin_check`, `_liquidate`; submitted by the runner at $t+1$)

marks $\leftarrow$ midpoint of every asset (fallback: last trade, reference)
for all accounts c not frozen **do**
    $E \leftarrow \text{cash} + \sum_k Q_k \,\text{mark}_k$; $G \leftarrow \sum_k |Q_k|\,\text{mark}_k$; **continue** if $G \le 0$
    if $E \le 0$ **then**
        freeze c; log `bankrupt`; $f \leftarrow 1$ ▷ full liquidation; no further orders accepted
    else if $L_c > 1$ and $E/G < m$ ($m = 0.25$) **then**
        log `margin_call`; $f \leftarrow$ liquidation fraction (0.5)
    else
        continue
    end if
    act $\leftarrow$ cancel every resting order of c; for each asset with $Q_k \neq 0$ add a market order for $\max(1, \lfloor f|Q_k| \rfloor)$ lots on the side that reduces $|Q_k|$, tagged `liquidation`
    log `liquidate`; append (c, act) to the list returned to the runner
end for
return list ▷ the runner submits these at the start of tick $t+1$; liquidation orders bypass the frozen, position-limit and leverage checks

Algorithm 4 LLM decision pipeline (`LLMPolicy.decide_one`, `_call`, `to_action`, `llm/policy.py`)

Require: observation o, provider π, model M, prompt id ψ

$S \leftarrow$ `system_prompt`(ψ, o); $U \leftarrow$ `user_message`$(o,$ adaptive$)$ ▷ Appendix B
if $\pi =$ library **then**
    text $\leftarrow$ recorded decision of a random one of the 5 nearest neighbours of `features`(o), limit price rescaled to the midpoint; cached $\leftarrow$ true
else
    key $\leftarrow$ SHA-256$(\pi \,|\, M \,|\,$ version $\,|\, S \,|\, U)$
    if cache enabled and key present **then**
        text $\leftarrow$ cached response; cached $\leftarrow$ true
    else if frozen-only or halted for budget **then**
        error $\leftarrow$ `frozen_miss` / `budget_halt`
    else if budget cap reached **then**
        halt for budget; error $\leftarrow$ `USD cap reached`
    else
        result $\leftarrow$ provider(π).`complete`$(S, U, M,$ effort $=$ low, max_output $= 1200,$ cache_key $=$ `ma-v1-`$\psi)$
        if text non-empty and no error **then** store (key, S, U, text, usage) in the cache
    end if
end if
cost $\leftarrow$ `budget.record`(tokens, latency, cached) ▷ 0 for cached hits; priced from the model table otherwise
if error **then** dec $\leftarrow$ none **else** (dec, err) $\leftarrow$ `parse_decision`(text) ▷ lenient JSON, strict pydantic schema
(action, valid, reason) $\leftarrow$ `to_action`(dec, o) if dec else (hold, false, err) ▷ rules of Appendix B
count invalid and failures
record $(t, i, \pi, M, \psi,$ valid, reason, decision fields, tokens, latency, cost, cached, error$)$
    together with the prompt SHA, the request id, `features`(o) and the relative limit price
return (action with meta {valid, reason, rationale, confidence, llm_action, provider, model}, record)

Algorithm 5 Matched-seed design runner (`run_design`, `cell_config`, `_run_one`, `experiments/design.py`)

Require: design file with `base`, `seeds`, design-level `overrides`, `cells` (name, overrides, treatment) and optional `factors`

cells ← listed cells ∪ full factorial product of the factor levels (overrides merged, labels joined); keep only `-cells` if given
seeds ← explicit list or $\{\text{start}, \ldots, \text{start} + n - 1\}$; truncate to `-max-seeds` if given
for all cell c, seed s **do**
    cfg ← base configuration; apply design overrides then cell overrides (dotted keys; sections merged, leaf dictionaries replaced); set name ← design/cell, seed ← s, treatment ← design labels ∪ cell labels
    job ← (cfg as JSON, `results/runs/<design>/<cell>/seed<s>`, cache path)
end for
for all jobs, sequentially if workers ≤ 1 else in a process pool of the given width **do**
    if `outcomes.json` exists in the run directory **then**
        reuse it ▷ an interrupted design resumes without new API calls
    else
        build the LLM policy (shared SQLite cache) if `llm_share` > 0; run Algorithm 1; save config, manifest and frames
        outcomes ← `run_outcomes`(result); write `outcomes.json`
    end if
end for
write `outcomes.parquet` (one row per run) and `design.json` (expanded cells, seeds, run count, wall-clock)

Because every cell of a design is run on the same seed list and the only streams touched by φ are the LLM assignment and the LLM agents' own generators, the control and each treatment of a given seed share the dividend path, the news, the endowments, latencies and activation draws of every non-replaced slot (common random numbers); the paired estimators of the analysis exploit this matching seed by seed.

D. Full hyperparameters

Table 9 lists every parameter of `configs/baseline.yaml` together with the defaults of the configuration schema (`experiments/config.py`) that the file does not set explicitly. Tables 10–13 list the overrides of every cell of the four design files; a cell inherits everything else from the baseline. Table 14 summarises the model cards (`docs/model_cards.md`).

Table 9. Baseline configuration (`configs/baseline.yaml`; schema defaults in italics)

Parameter	Value	Meaning
Run		

Parameter	Value	Meaning
`name`	baseline	run label (design/cell in designs)
`seed`	1	master seed of all streams
`n_ticks`	600	ticks per run
`warmup`	100	ticks excluded from outcomes; LLM decisions start at `warmup`/2
assets		
`n_assets`	1	number of risky assets
`tick_cents`	1	tick size in cents
`initial_price_cents`	20000	reference price $= \mu/r$ (200.00)
`price_band_pct`	0.5	limit prices farther than $\pm 50\%$ from the midpoint are rejected
`max_order_qty`	10000	maximum lots per order at the exchange
fundamentals		
`div_interval`	20	ticks between dividends (Δ)
`r_period`	0.0025	discount rate per dividend period (r)
`mu`	[50.0]	mean dividend in cents, regime 0, per asset (μ)
`mu_shift`	0.0	regime-1 mean $= \mu(1 + \text{mu_shift})$; 0 disables regimes
`p_switch`	0.0	per-period regime switch probability
`rho`	0.98	persistence of the deviation (ρ)
`sigma`	2.0	innovation standard deviation in cents (σ)
`jump_prob`	0.0	per-period jump probability (dividend noise)
`jump_sigma`	0.0	jump standard deviation in cents
`factor_loading`	[0.0]	loading b_k on the common factor
`x0`	0.0	initial revealed deviation
news		
`rate`	0.02	per-tick probability of a public item per asset
`noise_sd`	1.0	stated noise of credible items (cents)
`misleading_share`	0.0	share of biased items
`misleading_bias`	10.0	bias of a misleading item (cents, random sign)
`window`	20	ticks of news visible in the observation
exchange		
`maker_fee_bps`	0.0	maker fee (negative = rebate)

Parameter	Value	Meaning
`taker_fee_bps`	2.0	taker fee in basis points of notional
`min_latency`	0.05	floor on per-agent latency (ticks)
`halt_threshold`	null	circuit breaker off (treatment: 0.05)
`halt_window`	10	ticks over which the move is measured
`halt_duration`	20	halt length in ticks
`halt_cancel_resting`	true	cancel resting orders at a halt
`allow_short`	false	short selling switch (inert: every account has `short_allowed=false`)
`max_short_lots`	0	short limit when shorting is allowed
`margin_check`	true	end-of-tick margin and bankruptcy check
`liquidation_fraction`	0.5	share of each position sold on a margin call
`order_ttl`	50	resting orders expire after this many ticks
`position_limit`	*null*	max |position| in lots (treatment: 300)
`call_auction_interval`	*null*	batch clearing every k ticks (treatment: 5); collar 0.02 fixed in `ExchangeConfig`
population		
`n_traders`	100	trader slots (N)
`mix`	fund. 0.25, trend 0.06, meanrev 0.10, noise 0.50, MM 0.05, statarb 0.0, leveraged 0.04	shares of slots by policy (calibrated 2026-10-27)
`activity`	fund. 0.20, trend 0.20, meanrev 0.20, noise 0.30, MM 0.5, statarb 0.2, leveraged 0.2	per-tick activation probability
`policy_kwargs`	fundamentalist {aggressiveness: 20.0}; noise {p_market: 0.6}; market_maker {quote_qty: 20}	calibrated policy parameters (others at code defaults, Table 1)
`llm_share`	0.0	fraction of slots controlled by the LLM policy (φ)
`llm_replace`	proportional	stratified replacement across types (or one policy name)
`llm_keep_market_makers`	true	market makers are never replaced

Parameter	Value	Meaning
`informed_share`	0.3	share of fundamentalists receiving private signals
`precision`	2.0	signal precision π of informed fundamentalists
`leverage_cap`	3.0	gross exposure / equity cap of leveraged slots (L)
`maintenance_margin`	0.25	margin-call threshold on equity / gross (m)
`endowment.cash_cents`	2000000	base cash (20,000.00)
`endowment.shares`	100	base lots per asset
`endowment.wealth_dispersion`	0.3	lognormal sd of the wealth multiplier
`latency.min`, `latency.max`	0.05, 1.0	uniform latency range (ticks)
`latency.llm_latency`	null	matched-latency override for LLM slots
population.llm		
`provider_mix`	{mock: 1.0}	provider $\rightarrow$ share of LLM slots
`models`	mock: mock-v1; openai: gpt-5.4-nano; anthropic: claude-haiku-5-5	model id per provider alias
`prompt_set`	single	single (generalist) or diverse (8 personas)
`prompt_version`	v1	frozen prompt version
`memory`	0	past own decisions shown (treatment: 5)
`channel`	false	cohort communication channel
`cohort_size`	5	cohort size and channel length
`decision_interval`	20	ticks between LLM decisions
`stagger`, `stagger_slots`	true, 4	offsets $\{0,5,10,15\}$
`informed_share`	null	inherit the replaced slot's information (a number overrides)
`precision`	2.0	precision used when `informed_share` is a number
`max_order_qty`	50	cap on LLM order size (lots)
`reasoning_effort`	low	OpenAI `reasoning.effort` / Anthropic `effort`
`max_output_tokens`	1200	output cap per call (includes reasoning tokens)
`concurrency`	8	parallel requests per decision batch
`usd_cap`	5.0	hard USD cap per run

Parameter	Value	Meaning
`cache`	true	reuse identical (provider, model, prompt) decisions
`temperature`	*null*	never sent to gpt-5 / o-series models
`timeout_s`, `max_retries`	*60.0, 3*	client timeout and retries
`token_cap`, `decision_cap`	*null, null*	optional total-token and per-agent decision caps
`frozen_library`	*null*	path of the recorded decision library (provider `library`)
`adaptive_prompt`	*false*	P&L feedback line in the prompt
`price_assumptions`	*Table 14*	USD per 1M tokens by model id
shocks		
shock 1	{time: 300, kind: fundamental_jump, size: 10.0}	+10 cents on the latent deviation (+2.2% of V)
shock 2	{time: 450, kind: placebo}	logged, no effect
output		
`events`	false	write the full event log
`trades`, `llm_log`	true, true	write trade and LLM-decision frames
`lookback`	60	ticks of midpoint history in the observation

Table 10. Cells of `configs/classical_confirmatory.yaml` (seeds 101–130; design overrides: `provider_mix` {mock: 1.0}, `usd_cap` 0.0)

Cell	Overrides (dotted paths relative to the baseline)
control	none
placebo_shock	shocks: placebo at 300 and at 450 (no jump)
trend_heavy	population.mix: trend 0.16, noise 0.40 (others as baseline)
leverage_heavy	population.mix: leveraged 0.19, noise 0.35
leverage_heavy_funding_shock	population.mix as leverage_heavy; shocks: fundamental_jump −10.0 at 300, funding 0.5 at 320
liquidity_shock	shocks: fundamental_jump 10.0 at 300, liquidity at 300, liquidity_end at 360
info_low	population.precision 0.5
info_high	population.precision 8.0
misleading_news	news.misleading_share 0.5

Cell	Overrides
breaker_5	exchange.halt_threshold 0.05
poslimit_300	exchange.position_limit 300
auction_5	exchange.call_auction_interval 5
mock_llm_25_indep	population.llm_share 0.25 (mock provider)
mock_llm_50_indep	population.llm_share 0.50 (mock provider)
k2_control	assets.n_assets 2; fundamentals.mu [50, 50]; factor_loading [0.7, 0.7]; population.mix: noise 0.45, statarb 0.05
k2_leverage	as k2_control with leveraged 0.19, noise 0.30; shocks: fundamental_jump −10.0 on asset 0 at 300, funding 0.5 at 320

Table 11. Cells of `configs/confirmatory_openai.yaml` (seeds 201–207; design overrides: `provider_mix` {openai: 1.0}, `models` {openai: gpt-5.4-nano, openai_mini: gpt-5.4-mini, mock: mock-v1, anthropic: claude-haiku-5-5}, `usd_cap` 2.0, `concurrency` 8)

Cell	Overrides
dose_000 / 005 / 010 / 025 / 050 / 100	population.llm_share 0.00 / 0.05 / 0.10 / 0.25 / 0.50 / 1.00 (single generalist prompt, gpt-5.4-nano)
div_prompts_025	llm_share 0.25; llm.prompt_set diverse
div_models_025	llm_share 0.25; llm.prompt_set diverse; llm.provider_mix {openai: 0.5, openai_mini: 0.5}
memory_025	llm_share 0.25; llm.memory 5
channel_025	llm_share 0.25; llm.channel true
adaptive_025	llm_share 0.25; llm.adaptive_prompt true
leverage_025	llm_share 0.25; population.mix: leveraged 0.19, noise 0.35
breaker_025	llm_share 0.25; exchange.halt_threshold 0.05
poslimit_025	llm_share 0.25; exchange.position_limit 300
fair_mini_005	llm_share 0.05; llm.provider_mix {openai_mini: 1.0} (gpt-5.4-mini; equal-USD comparison with dose_025)
fair_latency_025	llm_share 0.25; population.latency.llm_latency 0.05

Table 12. Cells of `configs/exploratory.yaml` (mock dry run, seeds 1–3; design overrides: `provider_mix` {mock: 1.0}, `usd_cap` 0.0)

Cell	Overrides
share_0 / 5 / 25 / 50	population.llm_share 0.0 / 0.05 / 0.25 / 0.50

Cell	Overrides
diverse_25	llm_share 0.25; llm.prompt_set diverse
memory_25	llm_share 0.25; llm.memory 5
channel_25	llm_share 0.25; llm.channel true
leverage_25	llm_share 0.25; population.mix: leveraged 0.19, noise 0.35
breaker_25	llm_share 0.25; exchange.halt_threshold 0.05
poslimit_25	llm_share 0.25; exchange.position_limit 300
auction_25	llm_share 0.25; exchange.call_auction_interval 5
k2_25	llm_share 0.25; assets.n_assets 2; fundamentals.mu [50, 50]; factor_loading [0.7, 0.7]; population.mix: noise 0.45, statarb 0.05

Table 13. Cells of `configs/pilot_openai.yaml` (seeds 1–3; design overrides: `provider_mix` {openai: 1.0}, `models` {openai: gpt-5.4-nano, mock: mock-v1, anthropic: claude-haiku-5-5}, `usd_cap` 1.5, `concurrency` 8)

Cell	Overrides
share_5 / 25 / 50	population.llm_share 0.05 / 0.25 / 0.50 (gpt-5.4-nano, single prompt)

Table 14. Model cards summary (prices in USD per 1M tokens: input / cached input / output; source: provider pricing pages fetched 2026-10-27 for OpenAI, Anthropic's published table; identical to `LLMCfg.price_assumptions`)

Model id	Price	Reasoning effort	Output cap	Status / role
gpt-5.4-nano	0.20 / 0.02 / 1.25	low	1,200	live; primary model (all dose, mechanism, safeguard and fairness cells; library source)
gpt-5.4-mini	0.75 / 0.075 / 4.50	low	1,200	live-capable; two-model mix and equal-USD cells
gpt-5-nano	0.05 / 0.005 / 0.40	low	1,200	priced; optional replication, not planned
gpt-5-mini	0.25 / 0.025 / 2.00	low	1,200	priced only
gpt-4.1-nano	0.10 / 0.025 / 0.40	n/a	1,200	priced only
gpt-4.1-mini	0.40 / 0.10 / 1.60	n/a	1,200	priced only
claude-haiku-5-5	0.10 / 0.01 / 0.50	low (`effort`)	1,200	blocked (Anthropic key rejected, HTTP 401); mock dry run only
claude-sonnet-5-5	2.00 / 0.20 / 10.00	low	1,200	blocked; mock dry run only
mock-v1	0 / 0 / 0	n/a	n/a	mock (seeded value rule, 5% invalid); not evidence about LLMs
library (knn-replay)	0	n/a	n/a	frozen nearest-neighbour replay of recorded decisions; non-adaptive; reported separately

The Anthropic adapter sends `max_tokens` $=$ max(`max_output_tokens`, 256) and `output_config.effort`. Reasoning-token accounting exists only in the OpenAI adapter; the Anthropic adapter reports input, cached-input and output tokens. An unknown model id is priced at the most expensive known output price. Every manifest stores the price table in force when the run was made.

E. Reproducibility and replication package

Environment. The package requires Python $\geq$ 3.12 and is installed with `uv sync -extra dev` from the committed lock file `uv.lock`; the runs reported here used Python 3.12.13 and uv 0.11.7 on macOS (arm64). Locked versions of the libraries that affect numerical results or provider calls are numpy 2.5.4, pandas 3.1.0, scipy 1.18.1, statsmodels 0.15.0, pyarrow 26.0.0, duckdb 2.0.0, sortedcontainers 2.4.0, pydantic 2.14.1, pyyaml 6.0.3, openai 3.40.0, anthropic 2.3.0, with pytest 9.2.0 and hypothesis 6.170.12 for the test suite. API keys are read from a local `.env` file (`OPENAI_API_KEY`, `ANTHROPIC_API_KEY`) by the `ma` command line and never appear in configurations, manifests or logs.

One-command replication without an API key. `make replicate` runs, in order, the test suite (`make test`, 24 tests), the smoke test (two 300-tick runs with 100 agents at $\varphi = 0$ and $\varphi = 0.25$ with the mock provider, accounting verified), the classical confirmatory design (16 cells $\times$ 30 seeds) and the mock exploratory design (12 cells $\times$ 3 seeds), the numerical theory checks, the figures and the tables, writing `results/runs/<design>/outcomes.parquet`, `results/figures/` and `results/tables/`; `make paper` copies tables and figures into `paper/` and compiles `main.tex` with `latexmk`. The classical designs take about a minute on six worker processes (`W=6`); every number in the classical tables is reproduced bit for bit because those runs are deterministic functions of the seed.

Paid designs and cost estimates. `make pilot` and `make confirmatory` first print the `ma budget` estimate and then run the design with the shared decision cache `results/cache/llm_cache.sqlite`. With the planning token profile of 760 input and 230 output tokens per call, `ma budget` reports 6,480 calls and 2.85 USD for `pilot_openai.yaml` (3 cells $\times$ 3 seeds, per-run cap 1.5 USD) and 79,380 calls and 38.74 USD for `confirmatory_openai.yaml` (16 cells $\times$ 7 seeds, per-run cap 2.0 USD; 41.65 USD at the default 800/250 profile); the mock designs cost nothing. The realised profile of gpt-5.4-nano at prompt version v1 in the live runs on disk is about 850–870 input tokens (no provider-side cache hits were reported), 320–460 output tokens of which 240–380 are reasoning tokens, 2.5–3.2 s of latency and 0.00057–0.00075 USD per decision, i.e. 0.08–0.10 USD per run at $\varphi = 0.05$ and about 0.22 USD at $\varphi = 0.25$.[31] The project-wide ceiling for live calls is 40 USD; per-run caps are set at least twice the expected run cost so that no confirmatory run halts for budget.

Where things live. Each run writes `results/runs/<design>/<cell>/seed<NNNN>/` containing `config.yaml` (the fully resolved configuration), `manifest.json`, `ticks.parquet` (one row per tick and asset), `trades.parquet`, `agents.parquet` (one row per slot with policy, replaced type, provider, model, prompt id, cohort, stratum, latency, precision, leverage cap, initial and final wealth, fees, dividends, order counts and the frozen flag), `llm_decisions.parquet` (one row per LLM decision, Section 5) and `outcomes.json`; optionally `events.parquet`. The design level adds `outcomes.parquet` (one row per run with all outcome metrics and treatment labels) and `design.json`. The decision cache is a single SQLite table keyed by the SHA-256 of (provider, model, prompt version, system prompt, user prompt) storing both prompts, the raw response text, the token usage and a timestamp; the frozen library built from the recorded decisions is `results/library/gpt-5.4-nano_single.npz` (feature matrix plus JSON deci-

[31] Means over the online gpt-5.4-nano decisions in the `llm_decisions.parquet` frames of the pilot `share_5` runs (seeds 1–3) and of the confirmatory `dose_005`, `dose_010` and `dose_025` runs completed at the time of writing; the planning figure of 0.00044 USD per decision in `docs/model_cards.md` was measured on shorter smoke-test prompts.

sions), produced and replayed by `scripts/60_frozen_library.py`. Calibration artefacts are in `results/calibration/` (`sweep.parquet`, `sweep_summary.csv`, `scale_benchmark.json`).

Provider non-determinism. OpenAI reasoning models are sampled at the provider and expose no seed; identical prompts can yield different decisions across calls, and the Anthropic API is likewise unseeded. Three mechanisms bound the consequences. First, every online decision is written to the cache before it is used, so a re-run of the same design with the same cache replays every recorded decision exactly, performs zero calls and costs nothing; cache hits are counted separately (`cached_hits` in the manifest, `cached_hit` per decision, `llm_cached_hits` in the outcomes) so that online and replayed runs are never confused. Second, the request id and the dated model snapshot are stored per decision and the price table per run, so that a provider-side model change is visible. Third, the mock and library providers are fully seeded and serve as the deterministic reference for the pipeline; the frozen library is a non-adaptive replay and its runs are labelled provider `library`. The git revision in each manifest carries a `-dirty` suffix when the working tree had uncommitted changes at the time of the run; the record of every change made between the pre-analysis plan and the confirmatory runs is kept in `docs/research_log.md`.

Replication checklist.

1. Check out the revision named in the manifests of the design to be replicated (`git_revision`, without the `-dirty` suffix) and install with `uv sync -extra dev`; run `make test` (24 tests pass).
2. Run `make smoke`; the printed conservation flag is `True` and the artefacts appear in `results/runs/smoke/`.
3. Run `make replicate`; compare `results/runs/classical_confirmatory/outcomes.parquet` and `results/runs/exploratory_mock/outcomes.parquet` with the published copies (identical up to floating-point formatting).
4. For the live designs, place the published `results/cache/llm_cache.sqlite` in `results/cache/` and run `make pilot` and `make confirmatory`: every manifest then reports `cached_hits` equal to the number of calls and zero USD. Without the cache, the same commands reproduce the design with fresh provider samples at the cost printed by `ma budget`.
5. Verify that every run directory carries a `manifest.json` with `config_hash`, `git_revision`, `prompt_version`, `price_assumptions`, `llm_usage`, `shocks` and `final_conservation_ok`, and that the `providers` list of every live cell contains only the providers named in the design file.
6. Regenerate figures and tables with `make figures tables`; the analysis scripts read only the parquet frames and never re-simulate. The threshold sensitivity grid (bubble 3/5/8%, crash

7/10/15%) is obtained by re-running `run_outcomes` on the saved tick frames.

7. Compile the paper with `make paper`; the sections that quote numbers from result files cite those files in footnotes so that each figure can be traced to a frame and a script.
8. Record any deviation from `docs/protocol.md` in `docs/research_log.md` with its date and reason before analysing new runs.