Can Large Language Models Trade? AI Agents Competing in Stock Markets

March 13, 2025

Abstract

We develop a realistic synthetic stock market where Large Language Models (LLMs) act as competing trading agents. The framework enables diverse market participants—from prompt-engineered LLMs acting as value investors, momentum traders, and market makers to algorithmic strategies. Our analysis reveals three key findings: First, LLM agents demonstrate consistent strategy alignment with their trading directives across different LLM architectures. Second, these agents successfully perform core market functions while exhibiting sophisticated competitive behaviors - from strategic price discovery to adversarial liquidity provision. Third, we uncover how prompt engineering can lead to correlated behaviors and market instabilities. The open-source architecture provides (1) a systematic protocol for implementing and validating LLM trading agents across different architectures, (2) a controlled market environment for testing competitive agent interactions, and (3) comprehensive tools for analyzing emergent trading behaviors and systemic risks. This setting enables practitioners to develop and validate trading strategies while providing researchers with a novel platform for studying how LLM deployment may transform market dynamics.

1 Introduction

As Large Language Models (LLMs) continue to advance, their deployment as autonomous trading agents in financial markets appears inevitable. Major financial institutions are already exploring LLM applications in trading, market making, and risk management. This imminent transformation raises fundamental questions: Can LLMs effectively trade in markets? How do they interact with different trading strategies? What are the implications for market stability and efficiency? Understanding these questions is imperative for practical system development and regulatory oversight, as we need to be prepared for how LLMs will transform financial markets and their potential systemic risks.

This paper demonstrates that LLMs can effectively function as sophisticated trading agents through careful prompt engineering and systematic validation. We develop a framework that enables LLMs to act as different types of traders—including but not limited to value investors, momentum traders, market makers, and contrarians—while maintaining semantic consistency and strategic coherence. Our approach reveals how to implement and validate these agents' decision-making processes, contributing fundamental insights to both LLM capabilities and multi-agent systems.

Our analysis reveals three key findings about LLMs' capabilities as trading agents. First, LLM agents demonstrate consistent understanding of market mechanics and maintain coherent trading strategies, even in complex market conditions. They process market information, form price expectations, and execute trades in line with their assigned strategies. Second, these agents successfully engage in price discovery and liquidity provision, with market behavior emerging naturally from their interactions. Third, our framework enables systematic evaluation of agent behavior across different market conditions and LLM architectures, revealing how different foundation models interpret identical trading prompts. This comparative analysis provides insights into both the robustness and limitations of prompt-based trading strategies, while highlighting the importance of model selection in deployment.

Importantly, our findings also reveal potential vulnerabilities in LLM-based trading sys-

tems. The consistent adherence of LLMs to their assigned strategies, while beneficial for predictability, could be exploited through carefully crafted prompts. For instance, we demonstrate how subtle variations in prompt engineering can systematically influence agent behavior, highlighting the importance of robust validation protocols. This observation underscores the need for comprehensive testing frameworks when deploying LLM-based trading systems.

These findings raise important questions about algorithmic consistency and validation in LLM-based systems. Since trading agents are fundamentally implemented through prompts, their behavior inherits characteristics of the underlying language models. Moreover, the standardization of LLM architectures across implementations could create unexpected behavioral patterns - if many agents are based on similar foundation models, they might exhibit correlated responses to specific prompts or market conditions. The evaluation of prompt "optimality" itself becomes a game-theoretic problem: a prompt's effectiveness depends not only on its inherent quality but also on the distribution of other prompts in the market and the LLM architectures interpreting them.

Our framework also enables the study of LLMs' interpretation of abstract trading goals. By providing high-level objectives such as "maximize end of period wealth" or "optimize risk-adjusted returns," we can analyze how different language models translate these abstract directives into concrete trading strategies. This allows us to examine not just the performance of different prompts, but how different LLMs understand and implement fundamental trading concepts. The variation in interpretations across LLM architectures provides insights into both the capabilities and limitations of language models in financial decision-making, while raising important questions about the relationship between natural language instructions and trading behavior.

To enable this analysis and accelerate development in this emerging field, we provide an open-source framework with three key components. At its core is a structured protocol for implementing and validating LLM trading agents, ensuring consistent and reliable agent behavior. This protocol supports both LLM-based and traditional rule-based agents

through a unified interface, enabling direct comparisons and hybrid market scenarios. Importantly, the framework allows systematic evaluation of different LLM architectures using standardized prompts, creating a benchmark environment for comparing LLM trading capabilities while controlling for prompt-driven behavior. The framework is complemented by a controlled market environment for testing agent interactions, allowing researchers to explore complex market dynamics under various conditions. Finally, it includes a comprehensive data collection system for analyzing trading behavior, enabling detailed investigation of market outcomes and agent strategies.

This framework serves multiple stakeholders in preparation for the evolution of financial markets. For practitioners developing LLM-based trading systems, it provides a sandbox for strategy development and risk assessment before deployment. For regulators anticipating widespread LLM adoption in trading, it offers a laboratory for evaluating potential systemic risks and policy interventions. For researchers, it enables the study of market dynamics with artificial agents powered by LLMs.

1.1 Related Work

Our work contributes to several strands of literature. First, we advance the growing body of research on artificial intelligence in finance by demonstrating that large language models (LLMs) can serve as effective autonomous trading agents and by providing a validated framework for their implementation. Second, we contribute to the market microstructure literature by providing insights into how markets might evolve as LLM-based traders become active participants, with implications for price formation, liquidity provision, and market stability. Third, we contribute to experimental markets research by introducing an environment for studying complex trading interactions, offering unprecedented control and replicability compared to traditional human-subject experiments.

Recent studies have explored ChatGPT's potential in various financial tasks. Research shows that ChatGPT can effectively predict stock price movements using news headlines,

outperforming traditional sentiment analysis (Lopez-Lira and Tang 2023) and that forecasting with its embeddings outperforms traditional sentiment analysis (Chen, Kelly, and Xiu 2022). When integrated with Graph Neural Networks, ChatGPT improves stock movement prediction and portfolio performance (Chen et al. 2023). The model also demonstrates the ability to extract managerial expectations from corporate disclosures, predicting future investments and returns (Jha et al. 2024). ChatGPT-4 provides valuable investment advice with positive correlations to subsequent earnings announcements and stock returns (Pelster and Val 2023), and it enhances patent value forecasts by processing patent text (Yang 2023). Moreover, ChatGPT shows promise in central bank analysis, predicting future interest rate decisions from policy communications (Woodhouse and Charlesworth 2023). Recent work also focuses on the effects of AI adoption by firms (Eisfeldt et al. 2023, Babina et al. 2024).

Foundational work on LLMs as economic agents demonstrates their ability to simulate human behavior in economic contexts. Horton (2023) introduce the concept of "homo silicus" by using LLMs as computational models of human behavior, showing that they can replicate classic behavioral economics findings. Manning, Zhu, and Horton (2024) extend this approach through automated methods for generating and testing social scientific hypotheses with LLMs, demonstrating particularly strong results in market contexts such as auctions. Meanwhile, N. Li et al. (2024) show LLMs' potential in macroeconomic simulation for consumption and labor decisions. Our work, however, focuses on financial markets and the ways in which LLM agents can operate as full-fledged trading participants.

Recent research examines specific applications of LLMs for trading. FinMem and TradingGPT introduce frameworks with layered memory systems and customizable agent traits to improve financial decision-making (Yu, Li, et al. 2024; Li, Zhang, and Sun 2023). Hierarchical multi-agent architectures have been explored in FinCon, which uses a manager-analyst setup for collaborative portfolio decisions (Yu, Yao, et al. 2024). QuantAgent implements a self-improving signal-mining approach (Wang et al. 2024), while other work underscores the importance of long-term memory in LLM agents via vector databases (Hatalis et al.

2024). This line of research builds on successful reinforcement learning applications such as AlphaPortfolio (Cong et al. 2021) and AlphaManager (Campello, Cong, and Zhou 2023), which leverage deep RL and robust control for portfolio management.

Beyond finance, studies of LLM capabilities in interactive or multi-agent settings offer valuable insights. Benchmarks like AgentBench evaluate LLMs' performance in interactive tasks (Liu et al. 2023), while InvestorBench focuses on financial decision-making tasks specifically (H. Li et al. 2024). NegotiationArena demonstrates LLMs' capacity for strategic interaction in bargaining contexts (Bianchi et al. 2024). Similarly, Guo et al. (2024) provide an economics "arena" where LLMs compete in strategic games, revealing that advanced models like GPT-4 can exhibit rational, adaptive behavior—albeit not always reaching Nash Equilibrium. Existing frameworks typically assess individual agents or hierarchical collaborations, whereas our work studies emergent market behaviors that arise from multiple, independent LLM traders interacting in a marketplace.

Our approach also connects to the experimental asset markets tradition. For instance, Weitzel et al. (2020) show that even finance professionals are susceptible to speculative bubbles, especially amid high capital inflows. Kopányi-Peuker and Weber (2021) find that trading experience alone does not eliminate bubbles, challenging assumptions about rational market learning. Kirchler, Huber, and Stöckl (2012) identify confusion about fundamental values as a key driver of bubble formation. Our framework offers a new way to study these phenomena with LLM traders, which can be rigorously parameterized for sophistication, strategy, and information processing. This provides a powerful method to investigate how automated or "artificial" agents might affect market stability or volatility.

Other recent work highlights the versatility of LLM agents in complex, interactive scenarios beyond finance. One stream focuses on simulating social behaviors—such as opinion dynamics (Chuang et al. 2023), trust (Xie et al. 2024), and resource-sharing (Piatti et al. 2024). Another examines LLMs' strategic capabilities through task-based simulations and collaborative decision-making (Li, Zhang, and Sun 2023; Piatti et al. 2024).

Within the computer science literature, early works apply LLMs to financial markets but abstract away many fundamental stock market characteristics. Often, they employ single-price clearing mechanisms without a persistent order book, ignore partial fills and bid-ask spreads, and omit dividends. Consequently, these simplified environments can primarily address exogenous macro shocks (e.g., changes in interest rates or inflation; Gao et al. 2024), exogenous policy changes (Zhang et al. 2024), or rely on advanced methos such as repetitive next-token predictions to generate better answers (Koa et al. 2024). Our work extends these efforts by incorporating these crucial market features, enabling the study of complex endogenous events such as flash crashes, liquidity shocks, and large-trader impacts.

Finally, our approach is informed by complexity economics, which views markets as dynamic, non-equilibrium systems where agents adaptively learn and evolve strategies (Arthur 2013; Wolfram 2017). According to this view, markets exhibit emergent phenomena and self-organization, especially when trading agents (human or artificial) update their behavior in response to outcomes. This paradigm is particularly relevant in an era of increasing automation and algorithmic trading (Balland et al. 2022; Ping 2019). Like adaptive agents in complexity economics, our LLM traders incorporate new information and adjust their strategies, generating emergent market patterns—a key motivation behind our framework.

While earlier work demonstrates LLMs' promise for macroeconomic modeling (Horton 2023; Manning, Zhu, and Horton 2024) and simple trading simulations (Gao et al. 2024), we introduce a complex, open-source financial market platform that supports multiple agent architectures, thorough market microstructure (limit orders, partial fills, dividends), and rigorous testing protocols. Unlike previous agent-oriented frameworks that focus on narrow strategies (Yu, Li, et al. 2024; Li, Zhang, and Sun 2023), our system simultaneously accommodates heterogeneous agents interacting in realistic market environments—fulfilling a critical need in complexity economics research and market microstructure analysis.

2 Methodology

Our methodological framework integrates three essential components that together create a controlled environment for validating LLM-based trading agents. At its core, the framework implements a flexible continuous double-auction market mechanism that couples standard market microstructure principles with advanced market clearing and matching algorithms to accommodate asynchronous LLM decisions.

The market clearing process employs a dual-stage matching algorithm. In the first stage, market orders are netted using a market-to-market matching engine that processes buy and sell orders in strict time-priority sequence, reconciling orders based on available agent cash commitments. In the second stage, any remaining market orders are matched against the existing order book, with unfilled quantities converted to aggressive limit orders. This two-tiered approach optimizes both immediate execution and price discovery while maintaining market liquidity.

The system's `OrderMatchingService` orchestrates this process by coordinating trade executions through the `TradeExecutionService` and managing order state transitions via the `OrderStateManager`. Each trade is logged in detail, with the overall market state—including order books, market depth, and price evolution—recalculated at the end of each trading round.

This modular design in the matching and clearing engine provides the following advantages:

- **Rigorous Trade Validation:** Each market order is validated against agent cash commitments and position constraints before matching. In cases where an agent's commitment is insufficient, the system dynamically adjusts order quantities based on available cash, minimizing execution errors.

- **Flexible Liquidity Handling:** By supporting both market-to-market and market-to-book matching, the engine ensures that orders have multiple execution pathways.

Unexecuted market orders are converted to aggressive limit orders to capture remaining liquidity.

- **Detailed Trade Audit:** The system records comprehensive traceability data, including timestamps, trade volumes, executed prices, and agent identifiers, thereby enabling post-trade analysis and performance benchmarking. This data serves as the foundation for our subsequent market efficiency and agent performance validations.

The framework's innovative design supports a broad range of experimental configurations while ensuring both robustness and replicability, essential qualities for analyzing LLM-based trading dynamics.

2.1 Market Mechanism Design

Our framework implements a continuous double-auction market mechanism that processes orders in discrete trading rounds. The system supports both finite and infinite horizon markets, with key differences in terminal conditions and wealth calculation. In finite horizon markets, agents are informed of the total number of rounds, and terminal wealth is calculated by redeeming all shares at the fundamental value in the final round. In infinite horizon markets, no terminal information is provided to agents, and final wealth is determined using the last market price for share valuation. This design choice enables researchers to study how time horizons influence trading strategies and price formation, particularly in how agents balance short-term trading opportunities against long-term value considerations.

In a double auction, both buyers and sellers actively submit orders, with trades occurring when compatible buy and sell orders match on price. The matching engine processes these orders through three sequential phases:

First, incoming limit orders that do not immediately cross the market are added to the order book, maintaining strict price-time priority. Second, market orders are processed through a two-stage matching algorithm: (a) market-to-market matching, where market

orders are netted against each other at the current market price, and (b) market-to-book matching, where remaining market orders are executed against standing limit orders. Finally, any crossing limit orders are matched against the order book.

This design aligns with seminal market microstructure works such as Smith, Suchanek, and Williams (1988) and the more recent study by Holt, Porzio, and Song (2017), while adding robust validation and error handling specifically designed for LLM agent interactions. The system includes comprehensive trade validation, ensuring sufficient cash commitments for buyers and share availability for sellers, with detailed logging of order state transitions and trade execution.

2.1.1 Implementation Details

The matching engine (`MatchingEngine` class) implements three primary components that work together to facilitate efficient market operation. The order processing component handles the core matching functionality, where market orders are executed immediately against the best available prices in the order book during each trading round. When immediate execution is not possible, limit orders are stored in the order book according to price-time priority. The system supports partial executions, maintaining careful tracking of remaining quantities to ensure complete order fulfillment across multiple trades when necessary.

Position management forms the second critical component, providing comprehensive tracking of agent positions and cash balances throughout the trading session. Before any trade execution, the system performs rigorous pre-trade validation to ensure agents have sufficient resources to fulfill their orders. This includes validation of cash commitments for buyers and share availability for sellers, with the system maintaining accurate records of both committed and available resources for each agent. The position management system updates in real-time as trades are executed, ensuring market integrity and preventing over-commitment of resources.

The price formation mechanism constitutes the third component, implementing a system-

atic approach to price discovery and market monitoring. As trades are executed within each round, the system dynamically updates prices based on executed trades, while continuously tracking market depth and bid-ask spreads. This data collection provides detailed insights into market liquidity and efficiency. Each trade is logged with comprehensive information including price, quantity, and participating agents, creating a complete audit trail of market activity. This comprehensive price formation system ensures transparent price discovery while generating rich data for market quality analysis.

2.1.2 Extensibility Features

The framework employs a modular architecture designed to support diverse experimental configurations and research objectives. At its foundation, the system implements configurable market rules and trading constraints that can be adjusted to study different market conditions. The asset model supports fundamental features like dividend payments and interest accrual, enabling research across different market scenarios. The trading mechanism layer allows for adaptation to different market structures beyond the base double-auction system, such as call auctions or dark pools, through its modular service-based architecture. Furthermore, the framework provides flexible integration points for different LLM agent types and strategies, allowing researchers to experiment with diverse behavioral models and decision-making approaches.

This extensible design creates numerous research opportunities for market microstructure studies. Researchers can systematically investigate how different market structures influence price formation and efficiency, evaluate the impact of various trading rules on market quality, and analyze the complex interactions between different agent types and strategies. The framework's comprehensive logging and validation systems enable detailed examination of market behavior under varying conditions, from normal trading environments to stress scenarios. Through this modular approach to system design, the framework supports both targeted investigations of specific market mechanisms and broader studies of market dynam-

ics and stability.

2.1.3 Standardized Benchmark Environment

The framework implements a standardized benchmark suite to enable systematic evaluation of LLM trading capabilities. At its core, this suite maintains two categories of reference agents that serve as consistent benchmarks across experiments. The first category consists of traditional rule-based agents, including mean reversion traders with fixed entry/exit thresholds, momentum traders following established technical indicators, market makers with symmetric spread posting, and simple directional traders (always-buy and always-sell) that provide baseline behavior. These deterministic agents implement well-understood strategies, providing clear baselines for comparison.

The second category consists of baseline LLM agents with standardized prompts, implemented across different LLM architectures. These agents serve a dual purpose: they provide a consistent benchmark for evaluating new prompt designs while also enabling direct comparison of how different LLM architectures interpret identical trading instructions. The baseline LLM agents include value investors, momentum traders, market makers, contrarian traders, and news traders, each with carefully designed system prompts that define their trading philosophy and objectives.

The benchmark environment supports flexible agent compositions through a comprehensive distribution system. Researchers can specify exact agent counts, proportional distributions, or specialized configurations like "value_heavy" or "momentum_only" to create diverse market scenarios. Performance evaluation focuses on three key dimensions: objective achievement (how well agents achieve their stated goals), strategy consistency (whether agents maintain coherent behavior), and robustness (performance stability across different market conditions). This standardized approach enables systematic comparison of different LLM architectures and prompt designs while providing consistent performance metrics against established benchmarks.

2.2 LLM Agent Design

Our framework implements a systematic approach to designing and validating LLM-based trading agents. The architecture consists of three key components: structured prompting, decision validation, and behavioral consistency monitoring. This integrated approach ensures reliable and consistent agent behavior while maintaining the flexibility needed for diverse experimental designs.

2.2.1 Prompt Engineering Framework

The agent architecture implements a sophisticated two-part prompting system that separates strategic and tactical decision-making. The system prompt establishes the agent's fundamental trading characteristics, defining its trading philosophy, objectives, and behavioral constraints. This strategic layer encodes the agent's decision-making priorities and risk preferences, while establishing specific time horizons for trading decisions. By maintaining these parameters in the system prompt, we ensure consistent agent behavior across multiple trading rounds while allowing for strategic adaptation to changing market conditions.

The user prompt complements this strategic foundation by providing the immediate market context necessary for tactical decision-making. This dynamic component delivers current market state information, including prices, volumes, and emerging trends, alongside the agent's current position information and available trading options. The user prompt also specifies immediate decision requirements and operational constraints, ensuring that agent responses remain within feasible bounds while maintaining alignment with their strategic objectives. This separation of strategic and tactical prompting enables precise control over agent behavior while maintaining the flexibility needed for naturalistic market interactions.

2.2.2 Decision Structure

Each agent decision follows a standardized output format that consists of several key components. The `decision` field specifies the basic action as either Buy, Sell, or Hold. For active

trades, the `quantity` field determines the number of shares to trade, while the `order_type` indicates whether it is a Market or Limit order. When placing limit orders, the `price_limit` field specifies the maximum (for buys) or minimum (for sells) acceptable execution price. Finally, each decision includes a `reasoning` field that provides an explicit rationale for the trading decision, ensuring transparency and facilitating analysis of agent behavior.

2.2.3 Agent Types

The framework supports multiple agent types, each with distinct behavioral characteristics designed to replicate common trading strategies observed in real markets. Value Investors form a core agent class focused on fundamental analysis and mean reversion strategies, while Momentum Traders operate by following established price trends and volume patterns. Market Makers serve a critical function by providing liquidity and managing inventory positions. Contrarian agents implement strategies that trade against market extremes, and News Traders specialize in reacting to public information signals. The framework's flexible prompt-based architecture allows for easy extension of this agent ecosystem - researchers can rapidly prototype and deploy new agent types by developing appropriate strategic and tactical prompts without modifying the underlying system architecture.

2.2.4 Behavioral Validation

Agent behavior is validated through multiple mechanisms. The primary validation occurs through decision consistency checks, where each agent decision is validated against the agent's defined strategy and constraints. The framework implements comprehensive error handling through the `TradeDecision` model, which enforces parameter validation for order specifications and includes a fallback to hold positions when parsing errors occur.

Position and cash constraint enforcement forms another critical validation layer, with the framework actively tracking both current and committed resources through the `outstanding_orders` system. This prevents agents from exceeding their position limits or trading with unavail-

able resources. The system maintains detailed logs of agent decisions and their reasoning, enabling post-hoc analysis of trading patterns and strategy adherence.

Future enhancements to the validation system will include automated detection of strategy drift and more sophisticated monitoring of trading patterns to ensure continued alignment with stated agent objectives. The framework's modular design facilitates the addition of these validation mechanisms as they are developed.

2.2.5 Validation Metrics and Performance Monitoring

The framework implements three categories of validation metrics that continuously monitor agent behavior and market quality:

1. Strategy Consistency Metrics

    - Strategy adherence score: Measures alignment between agent decisions and stated strategy by analyzing reasoning text against strategy-specific keywords and expected behavior patterns
    - Semantic consistency: Quantifies coherence of agent explanations across decisions using embedding-based similarity measures
    - Decision pattern analysis: Tracks the distribution of order types, sizes, and price levels relative to strategy specifications

2. Market Quality Metrics

    - Price efficiency: Measures deviation between transaction prices and fundamental values
    - Liquidity provision: Monitors bid-ask spreads, market depth, and order book resilience
    - Execution quality: Tracks fill rates, partial executions, and price impact of trades
    - Market stability: Analyzes volatility patterns and price continuity

3. Risk and Performance Metrics

    - Position management: Monitors portfolio concentration and exposure limits
    - Trading efficiency: Measures implementation shortfall and transaction costs
    - Risk-adjusted returns: Calculates Sharpe ratio and maximum drawdown statistics
    - Strategy drift detection: Identifies significant deviations from historical trading patterns

2.3 Data Collection Architecture

The framework implements a comprehensive data collection system that captures multiple dimensions of market behavior and agent decision-making. This structured approach ensures reproducibility and enables detailed analysis of market dynamics.

2.3.1 Core Data Streams

The system captures five primary data streams that work together to provide a complete picture of market activity. Market Data forms the foundation, tracking essential metrics including price evolution, fundamental values, trading volumes, order book depth, market liquidity, best bid/ask prices, spreads, and price-fundamental ratios. This is complemented by detailed Trade Data, which records individual transaction details such as prices, quantities, buyer and seller identifiers, timestamps, round information, and specific execution details.

Agent Data provides insights into participant behavior by monitoring cash and share positions, portfolio values, wealth evolution, and agent type and strategy identifiers. Order Data captures the complete lifecycle of trading decisions, including order specifications (type, size, price), agent reasoning and justification, decision context and timing, and order lifecycle events. Finally, Market State data maintains a comprehensive view of overall market conditions through order book snapshots, market depth information, aggregate statistics,

and various market microstructure metrics.

2.3.2 Data Validation and Storage

The system implements a comprehensive set of data quality measures to ensure reliability and consistency. The validation framework performs continuous checks for data completeness, verifies consistency across related data streams, enforces format and type validation, and maintains proper temporal sequencing of all recorded events. These validation mechanisms work together to maintain data integrity throughout the experimental process.

The storage architecture employs multiple formats optimized for different data types and use cases. Structured CSV files store tabular data for efficient analysis, while JSON formats accommodate complex nested structures that preserve the rich relationships between market elements. The system maintains detailed timestamped logs for debugging purposes and generates summary statistics to facilitate quick analysis of experimental outcomes.

This comprehensive data collection architecture enables researchers to conduct detailed analysis of market dynamics, validate agent behavior patterns, ensure experiment reproducibility, and perform meaningful cross-experiment comparisons. The integrated approach to data management provides the foundation for rigorous empirical investigation of market behavior and agent interactions.

3 Data and Experimental Design

Our experimental framework implements a dividend-paying asset market with heterogeneous agents. The baseline simulation runs for 20 rounds with 20 participating agents, though these parameters are configurable. Each agent begins with an initial endowment of 560 monetary units and 20 shares of the asset. The fundamental value of the asset is set at 28 units, with a final redemption value matching this fundamental price.

The dividend structure follows a stochastic process with a base payment of 1.4 units

and a variation of ± 0.2 units, paid with 50% probability each round. This design creates uncertainty in the asset's income stream while maintaining a known expected value. Cash holdings earn a risk-free interest rate of 5% per round, providing an opportunity cost for asset investment.

Agent composition is controlled through a flexible distribution system that allows for various experimental configurations. In our baseline setup, we employ a mixed population of optimistic traders, value investors, and contrarian agents. Each agent type implements distinct trading strategies through Large Language Model (LLM) prompts, with the option to use either a full-scale GPT-4 model or a more efficient mini variant for decision-making.

The data collection system captures comprehensive market activity across multiple dimensions. At the market level, it tracks the evolution of prices, fundamental values, and realized dividends, providing insights into overall market dynamics. Individual agent data includes detailed wealth trajectories and portfolio compositions, enabling analysis of strategy effectiveness and wealth distribution patterns. The system maintains complete transaction records with prices, quantities, and counterparty information, facilitating the study of trading patterns and market microstructure. Additionally, it captures agent decision rationales and strategy adherence metrics, allowing researchers to evaluate the consistency and effectiveness of different trading approaches.

All experimental parameters, market outcomes, and agent interactions are systematically logged and stored in structured formats (CSV and JSON) for subsequent analysis. The framework generates standardized visualizations of price evolution, wealth distribution, and dividend payments to facilitate comparative analysis across different experimental configurations.

4 Theoretical Framework

While our framework ultimately relies on LLM agents operating through natural language, developing a formal theoretical model serves several purposes. First, it provides analytical structure for understanding and predicting agent behavior, allowing us to bridge the gap between traditional economic theory and LLM-based decision making. Second, it enables us to derive testable predictions about market outcomes, creating a systematic framework for validating whether LLM agents behave in economically meaningful ways. Third, it helps identify potential sources of market instability or inefficiency that might emerge from LLM trader interactions, informing both practical system design and regulatory policy.

The model we develop maintains sufficient mathematical precision to derive clear predictions while accommodating the unique characteristics of LLM agents. Rather than assuming explicit utility maximization, we model how trading strategies emerge from natural language prompts and reason about how these strategies interact to determine market outcomes. This hybrid approach - combining rigorous economic modeling with LLM-based decision making - provides new insights into how markets might function when populated by AI traders while maintaining connection to classical market microstructure theory.

4.1 Market Structure

We consider a finite-horizon market operating over the interval $[0, T]$. The market structure incorporates key features of modern electronic trading venues while maintaining analytical tractability. This section details the fundamental components of our market model, including order types, execution mechanisms, and price formation processes.

4.1.1 Order Types and Execution Mechanism

The market supports a dual-order system consisting of market and limit orders. Market orders provide immediate execution certainty at the best available price, while limit orders

allow traders to specify execution prices but without guaranteed fills. This structure captures the fundamental tradeoff between execution certainty and price improvement that traders face in real markets.

Order execution follows a strict price-time priority rule, consistent with the majority of modern electronic markets. Under this mechanism, market orders receive immediate execution priority, while limit orders enter a queue based on their price level and submission time. When limit orders cross the spread (i.e., buy orders with prices at or above the best ask, or sell orders with prices at or below the best bid), they are automatically converted to market orders to ensure immediate execution.

The market operates in discrete rounds, with each round t following a four-stage matching process:

1. First, the system processes cancellations of existing orders from active agents
2. Next, it executes any new market orders against available liquidity
3. Then, it adds new limit orders to the order book
4. Finally, it matches any crossing limit orders according to price-time priority

The market clearing price P_t at each round is determined by a standard mechanism:

$$P_t = \begin{cases} P_{\text{trade}} & \text{if trades occur} \\ \frac{P_{\text{bid}} + P_{\text{ask}}}{2} & \text{if no trades} \end{cases}$$

where P_{trade} represents the last executed trade price, and P_{bid} and P_{ask} are the best bid and ask prices in the order book. This price formation process ensures continuous price discovery even in the absence of trades, while maintaining a clear connection to actual transaction prices when trading occurs.

4.1.2 Dividend Process

The traded asset generates a continuous dividend stream that follows a geometric Brownian motion, reflecting the standard modeling approach in financial economics. The dividend process evolves according to:

$$dD_t = \mu dt + \sigma_D dW_t$$

where μ represents the expected dividend growth rate, σ_D captures dividend volatility, and W_t is a standard Wiener process. This specification allows for both predictable growth through the drift term and random fluctuations through the diffusion term, providing a realistic model of dividend uncertainty.

4.1.3 Asset Structure

The asset's value structure combines both finite and continuous components. Specifically, investors receive a continuous dividend stream D_t throughout the trading period and a terminal value K at time T. This dual-component structure creates rich trading dynamics as agents must balance immediate dividend income against terminal value considerations in their trading decisions.

4.1.4 Agent Endowments and Constraints

Each agent i enters the market with an initial endowment consisting of cash $C_{i,0}$ and shares $S_{i,0}$. To maintain market realism and prevent degenerate trading strategies, agents face two key constraints. First, a non-negative cash constraint ($C_{i,t} \geq 0$) prevents unlimited borrowing. Second, a short-selling constraint ($S_{i,t} \geq -\bar{S}$) limits agents' ability to take negative positions, where $\bar{S}$ represents the maximum allowed short position. These constraints ensure that trading strategies remain within economically reasonable bounds while still allowing for substantial trading flexibility.

4.1.5 Fundamental Value

The fundamental value of the asset at any time t represents the present value of all future cash flows, combining both the terminal value and the dividend stream. Specifically, the fundamental value is calculated as:

$$V_t = Ke^{-r(T-t)} + \int_t^T E[D_s]e^{-r(s-t)}ds$$

where r represents the risk-free interest rate and $E[D_s]$ denotes the expected dividend at time s. The first term captures the discounted terminal value, while the integral term accounts for the continuous stream of expected future dividends, each appropriately discounted to present value.

4.1.6 Parameter Calibration

To facilitate clear analysis of trading behavior, we calibrate the model parameters $\{\mu, K, r, T\}$ to maintain a constant fundamental value throughout the trading period, such that $V_t = V^*$ for all $t \in [0, T]$. This stability is achieved by setting the dividend growth rate as:

$$\mu = rV^* - rKe^{-r(T-t)}$$

This calibration creates a stable benchmark against which to evaluate agent behavior, eliminating confounding effects from time-varying fundamentals and allowing us to focus on the strategic interactions between trading agents.

4.2 LLM Decision Process

The LLM trading agent integrates multiple information sources to form trading decisions. At each decision point, the agent processes four key components: a strategy template θ_i derived from its initial prompt, the current market state S_t, historical context H_t consisting

of past states, actions, and reasoning ($\{(S_k, a_k, R_k)\}_{k<t}$), and its natural language reasoning output R_t.

The agent's decision framework encompasses several key variables. The action space includes three possible decisions ($a_t \in \{\text{Buy}, \text{Sell}, \text{Hold}\}$), complemented by quantity decisions (q_t) and order type selection ($o_t \in \{\text{Market}, \text{Limit}\}$). Each agent operates according to its assigned strategy type θ_i. The market state S_t is represented by a comprehensive set of variables: current price (P_t), fundamental value (V_t), price changes (ΔP_t), trading volume (Vol_t), inventory levels (I_t), and news signals (N_t).

This information processing results in a probabilistic distribution over possible actions, formally expressed as:

$$P(a_t, q_t, o_t|\theta_i, S_t, H_t) = \text{LLM}(\theta_i, S_t, H_t)$$

The agent's decision space can be characterized mathematically as:

$$\mathcal{D} = \{(a, q, o) \in \mathcal{A} \times \mathcal{Q} \times \mathcal{O} : \text{feasible}(a, q, o|\theta_i, S_t)\}$$

where feasibility is constrained by both explicit market rules and learned strategic behavior. While LLMs do not compute explicit probabilities in their decision-making process, their behavior can be approximated by a function mapping strategy and state to actions:

$$P(a_t, q_t, o_t|\theta_i, S_t) = g(\theta_i, S_t)$$

4.3 Prompt-Based Strategy Formation

A strategy θ_i emerges from three components:

1. Base prompt template π_i^b

2. Strategy-specific rules π_i^s

3. Context window C_t

The prompt-to-strategy mapping can be formalized as:

$$\theta_i = g(\pi_i^b \oplus \pi_i^s, C_t)$$

where $\oplus$ represents prompt composition, $C_t = \{S_k, a_k, R_k\}_{k=t-w}^{t}$ is the context window of length w, and $g(\cdot)$ is the LLM's interpretation function.

For example, a value investor's strategy components are:

$$\pi_i^b = \text{"You are a trading agent..."}$$
$$\pi_i^s = \text{"Buy when price} < \text{fundamental value..."}$$
$$C_t = \text{Recent market states and actions}$$

More formally, each strategy type's prompt structure can be decomposed as:

$$\pi_i = \{\pi_i^{\text{role}}, \pi_i^{\text{constraints}}, \pi_i^{\text{objectives}}, \pi_i^{\text{rules}}\}$$

where π_i^{role} defines the agent's role, $\pi_i^{\text{constraints}}$ specifies trading restrictions, $\pi_i^{\text{objectives}}$ states optimization goals, and π_i^{rules} provides decision heuristics. This decomposition ensures that each agent maintains consistent behavior aligned with its assigned trading strategy while retaining the flexibility to adapt to changing market conditions within its strategic framework.

4.4 LLM Reasoning Process

The decision-making process of the LLM unfolds in three distinct stages. First, during the Information Processing stage, the LLM synthesizes the current market state S_t, historical context H_t, and its strategy template θ_i to produce a processed market state $\hat{S}_t$. This is

represented mathematically as:

$$\hat{S}_t = f_{\text{process}}(S_t, H_t, \theta_i)$$

Next, in the Strategy Application stage, the LLM applies its reasoning capabilities to the processed market state $\hat{S}_t$ and its strategy template θ_i, generating a natural language reasoning output R_t:

$$R_t = f_{\text{reason}}(\hat{S}_t, \theta_i)$$

Finally, in the Decision Formation stage, the LLM uses the reasoning output R_t to form a probabilistic distribution over possible actions, quantities, and order types:

$$P(a_t, q_t, o_t | R_t) = f_{\text{decide}}(R_t)$$

In this framework, $\hat{S}_t$ represents the processed market state, R_t is the natural language reasoning, and $f_{\text{process}}, f_{\text{reason}}, f_{\text{decide}}$ are functions specific to the LLM's architecture.

The reasoning output R_t itself can be decomposed into four components: observation (O_t), analysis (A_t), decision (D_t), and justification (J_t). This decomposition provides a structured approach to understanding how the LLM interprets and acts upon market information.

4.5 Trade Formation

4.5.1 Quantity Decisions

Each agent's desired trade quantity is determined by:

$$q_t = \max\{\min\{q^*(\theta_i, S_t), q_{\max}\}, 0\}$$

where $q^*(\theta_i, S_t)$ is the desired trade size based on strategy and state, and $q_{\max}$ is the position limit. This formulation ensures that trade quantities remain within feasible bounds while reflecting the agent's strategic preferences and market conditions. The nested max-min structure enforces both non-negative quantities and maximum position constraints, preventing degenerate trading behavior while maintaining strategic flexibility.

4.6 LLM-Induced Strategy Functions

These strategy functions emerge from natural language prompts rather than explicit programming. While LLMs do not calculate explicit parameters, their natural language reasoning effectively implements similar decision boundaries. For analytical tractability, we can approximate each agent type's behavior with a specific functional form g_θ:

4.6.1 Value Investor

$g_{\text{value}} = f(P_t/V_t)$ with prompt-induced parameters α, β:

- Buy if $P_t/V_t < 1 - \alpha$
- Sell if $P_t/V_t > 1 + \alpha$
- Market if $|P_t/V_t - 1| > \beta$

4.6.2 Momentum Trader

$g_{\text{momentum}} = f(\Delta P_t, \Delta \text{Vol}_t)$ with parameters γ, σ_t:

- Buy if $\Delta P_t > 0$ and $\Delta \text{Vol}_t > 0$
- Sell if $\Delta P_t < 0$ or $\Delta \text{Vol}_t < 0$
- Market if $|\Delta P_t| > \sigma_t$

4.6.3 Market Maker

$g_{\text{maker}} = f(I_t - I^*, P_t - \bar{P}_t)$ with parameters δ, λ:

- Buy if $I_t < I^*$ and $P_t < \bar{P}_t$
- Sell if $I_t > I^*$ and $P_t > \bar{P}_t$
- Limit by default, Market if $|I_t - I^*| > \delta$

4.6.4 Contrarian Trader

$g_{\text{contrarian}} = f((P_t - \bar{P}_t)/\sigma_t, \text{Vol}_t)$ with parameters η, μ:

- Buy if $(P_t - \bar{P}_t)/\sigma_t < -\eta$
- Sell if $(P_t - \bar{P}_t)/\sigma_t > \eta$
- Market if $|(P_t - \bar{P}_t)/\sigma_t| > \mu$

4.6.5 News Trader

$g_{\text{news}} = f(N_t, P_t - \bar{P}_t)$ with parameters ν, ξ:

- Buy if $N_t > 0$ and $P_t < \bar{P}_t$
- Sell if $N_t < 0$ or $P_t > \bar{P}_t + \xi|N_t|$
- Market if $|N_t| > \nu$

4.6.6 Optimistic Trader

$g_{\text{optimistic}} = f(P_t, 10V_t)$ with parameter κ:

- Buy if $P_t < 10V_t - \kappa$
- Sell if $P_t > 10V_t + \kappa$
- Market by default

4.6.7 Speculator

$g_{\text{spec}} = f(E[\Delta P_t])$ with parameters ψ, ω:

- Buy if $E[\Delta P_t] > \psi$
- Sell if $E[\Delta P_t] < -\psi$
- Market if $|E[\Delta P_t]| > \omega$

where $\bar{P}_t$ is the moving average price, I^* is the target inventory level, σ_t is the price volatility, and $E[\Delta P_t]$ is the expected price change. Parameters $(\alpha, \beta, \gamma, \delta, \eta, \nu, \kappa, \psi, \omega)$ represent the implicit thresholds that emerge from the LLM's interpretation of strategy-specific prompts. This formalization, while an approximation of the complex reasoning process within LLMs, provides sufficient mathematical structure for theoretical analysis while maintaining fidelity to the underlying natural language implementation.

4.7 Linear Approximations of Strategy Functions

While the LLM agents operate through natural language reasoning, their behavior can be approximated using linear decision rules for analytical tractability. These approximations capture the first-order effects of key state variables on trading decisions.

4.7.1 General Form

Each agent's trading decision can be approximated as a linear function:

$$q_t = \beta_0 + \sum_i \beta_i x_{i,t}$$

where q_t is the desired trade quantity (positive for buy, negative for sell), β_i are strategy-specific coefficients, and $x_{i,t}$ are state variables.

4.7.2 Strategy-Specific Approximations

For value investors, trading quantity is proportional to the relative price-value discrepancy:

$$q_t = \beta_v \left(\frac{V_t - P_t}{V_t} \right) + \epsilon_t$$

where $\beta_v > 0$ reflects sensitivity to value-price discrepancy.

Momentum traders respond to both price and volume changes:

$$q_t = \beta_p \Delta P_t + \beta_v \Delta \text{Vol}_t + \epsilon_t$$

where $\beta_p, \beta_v > 0$ capture price and volume momentum sensitivity.

Market makers balance inventory management with price reversion:

$$q_t = \beta_i (I^* - I_t) + \beta_p (P_t - \bar{P}_t) + \epsilon_t$$

where $\beta_i > 0$ and $\beta_p < 0$ reflect inventory and price reversion goals.

Contrarian traders act against normalized price deviations:

$$q_t = -\beta_z \left(\frac{P_t - \bar{P}_t}{\sigma_t} \right) + \epsilon_t$$

where $\beta_z > 0$ captures mean reversion strength.

News traders combine signal response with price adjustment:

$$q_t = \beta_n N_t + \beta_p (P_t - \bar{P}_t) + \epsilon_t$$

where $\beta_n > 0$ and $\beta_p < 0$ reflect news sensitivity and price adjustment.

Optimistic traders pursue aggressive value targets:

$$q_t = \beta_o (10V_t - P_t) + \epsilon_t$$

where $\beta_o > 0$ reflects optimistic value sensitivity.

Speculators trade on expected price movements:

$$q_t = \beta_e E[\Delta P_t] + \epsilon_t$$

where $\beta_e > 0$ captures expected price change sensitivity.

4.7.3 Order Type Selection

The probability of selecting a market order can be modeled as:

$$\Pr(\text{Market}) = \Phi(\gamma_0 + \gamma_1 |q_t| + \gamma_2 |\Delta P_t|)$$

where $\Phi(\cdot)$ is the standard normal CDF, $\gamma_1 > 0$ captures urgency based on trade size, and $\gamma_2 > 0$ reflects price movement sensitivity.

4.7.4 Empirical Estimation

These coefficients can be estimated from LLM trading data using standard linear regression:

$$\hat{\beta} = (X'X)^{-1}X'q$$

where X is the matrix of state variables, q is the vector of trade quantities, and $\hat{\beta}$ are the estimated strategy coefficients. This empirical approach allows us to quantify how closely the LLM agents adhere to their intended trading strategies and measure the relative importance of different state variables in their decision-making process.

4.8 Prompt-Based Adaptation

Unlike traditional models with fixed parameters, LLM agents can adapt their decision thresholds (α, β, γ, etc.) based on market conditions while maintaining strategic consistency. This

adaptation process can be modeled as:

$$\theta_{i,t+1} = h(\theta_{i,t}, S_t, R_t)$$

where $\theta_{i,t}$ represents the agent's current strategy parameters, S_t is the market state, and R_t is the reasoning output from the LLM. This dynamic adaptation allows agents to respond to changing market conditions while maintaining their fundamental strategic orientation.

4.9 Market Clearing and Equilibrium

4.9.1 Market Clearing

The market clears through the standard conditions:

$$\sum_i q_{i,t} = 0$$

$$\sum_i S_{i,t} = \bar{S}_{\text{total}}$$

where $\bar{S}_{\text{total}}$ is the total supply of shares. These conditions ensure that trading remains a zero-sum game in terms of quantity and that the total number of shares in the market remains constant.

4.9.2 Wealth Evolution

Agent wealth evolves according to:

$$W_{i,t} = C_{i,t} + S_{i,t}P_t + \int_0^t S_{i,s}D_s ds$$

This formulation captures all components of an agent's wealth: current cash holdings ($C_{i,t}$), the market value of their current position ($S_{i,t}P_t$), and the accumulated dividend income from their historical positions ($\int_0^t S_{i,s}D_s ds$).

4.9.3 Equilibrium Definition

An equilibrium in this market model consists of three main components: the price process $\{P_t\}_{t=0}^{T}$, the trading strategies $\{q_{i,t}\}_{i,t}$, and the holdings $\{S_{i,t}, C_{i,t}\}_{i,t}$.

For an equilibrium to exist, several conditions must be satisfied. First, given the price P_t, each agent's strategy must maximize their expected utility while adhering to certain constraints. These include a budget constraint ensuring non-negative cash holdings ($C_{i,t} \geq 0$), position limits preventing excessive short-selling ($S_{i,t} \geq -\bar{S}$), and strategy consistency where the trading decision $q_{i,t}$ aligns with the agent's strategy function $g_{\theta_i}(S_t)$.

Second, markets must clear at all times, meaning the total quantity of trades sums to zero, and the total supply of shares remains constant. Lastly, all agents must maintain solvency, ensuring their wealth $W_{i,t}$ is non-negative.

While traditional equilibrium concepts assume explicit utility maximization, LLM agents implicitly optimize through their strategy-specific prompts and natural language reasoning. This creates a novel form of equilibrium where optimal behavior emerges from prompt interpretation rather than mathematical optimization.

4.10 Testable Predictions

Our framework yields several quantitative predictions:

4.10.1 Strategy Consistency

Agents maintain consistent behavior within their strategy type:

$$\text{Var}(a_t|\theta_i) < \text{Var}(a_t)$$

$$\text{Cov}(a_t, a_{t'}|\theta_i) > 0$$

These conditions indicate that trading decisions exhibit lower variance when conditioned on strategy type, and actions are positively correlated across time within the same strategy.

4.10.2 Strategy Differentiation

Different strategies produce systematically different trading patterns:

$$E[q_t|\theta_i] \neq E[q_t|\theta_j] \quad \text{for } i \neq j$$

This inequality demonstrates that trading behavior varies systematically across strategy types, with each strategy generating distinct patterns of market activity.

4.10.3 Price Impact

The price impact of trades varies by strategy type:

$$\frac{\partial P_t}{\partial q_t} = \lambda + f(\theta_i)$$

where $f(\theta_i)$ captures strategy-specific price effects. This relationship suggests that market impact is not uniform across strategies but rather depends on the strategic orientation of the trading agent, reflecting differences in how various strategies are perceived and interpreted by the market.

4.10.4 Empirical Tests

These predictions can be tested through various empirical methods. First, time series analysis of trading decisions within strategy types can reveal consistency over time. Second, cross-sectional comparison of trading patterns across different strategies can highlight systematic differences. Third, event study analysis of price movements following trades can assess the impact of specific strategies on market prices. Finally, semantic analysis of LLM reasoning outputs R_t can provide insights into the decision-making process. The key innovation is that

these patterns emerge from natural language prompts rather than explicit programming, while maintaining mathematical regularity sufficient for empirical testing.

4.10.5 Reasoning Consistency

The semantic similarity of reasoning outputs R_t should be higher within strategy types:

$$\text{sim}(R_t, R_{t'}|\theta_i) > \text{sim}(R_t, R_{t'}|\theta_i \neq \theta_j)$$

where $\text{sim}(\cdot, \cdot)$ is a semantic similarity measure. This suggests that agents with the same strategy type produce more consistent reasoning outputs compared to those with different strategies.

4.11 Strategy Validation Framework

The validation framework addresses three key questions: Does the agent maintain intended behavior (Strategy Fidelity)? Are choices economically rational (Decision Quality)? Is the logic sound and consistent (Reasoning Validity)?

We define three key validation metrics:

1. Strategy Consistency:

$$\text{SC}(\theta_i) = \frac{1}{T}\sum_{t=1}^{T} \text{sim}(R_t, E[R|\theta_i])$$

2. Decision Coherence:

$$\text{DC}(\theta_i) = \text{Cor}(D_t, D_{t+1}|\theta_i)$$

3. Reasoning Quality:

$$\text{RQ}(\theta_i) = \frac{1}{T}\sum_{t=1}^{T} \text{quality}(J_t|\theta_i, S_t)$$

where $\text{sim}(\cdot,\cdot)$ measures semantic similarity, $\text{Cor}(\cdot,\cdot)$ measures decision correlation, and $\text{quality}(\cdot)$ evaluates reasoning validity. A strategy is considered valid if it meets predetermined thresholds for these metrics:

$$\text{SC}(\theta_i) \geq \alpha$$

$$\text{DC}(\theta_i) \geq \beta$$

$$\text{RQ}(\theta_i) \geq \gamma$$

for predetermined thresholds α, β, γ.

5 Results

Our analysis reveals three key findings about the capabilities of LLM-based trading agents. First, we demonstrate that LLM agents exhibit consistent understanding of market mechanics and maintain coherent trading strategies. Through detailed analysis of agent decision logs and trading patterns, we find that agents reliably execute their assigned strategies while providing clear, contextually appropriate reasoning for their decisions. Value investors consistently reference fundamental prices in their decisions, while momentum traders demonstrate attention to price trends and volume patterns. This strategic consistency persists across varying market conditions and price regimes.

Second, our results show that LLM agents effectively contribute to price discovery and liquidity provision. Market maker agents maintain reasonable bid-ask spreads and actively manage their inventory positions, while value investors help anchor prices to fundamental values through their trading activities. Analysis of order book dynamics reveals that agents collectively maintain market depth and respond to liquidity demands. The interaction between different agent types creates realistic market dynamics, with prices generally tracking fundamental values while exhibiting plausible short-term deviations.

Third, we find evidence of sophisticated adaptation to changing market conditions while agents maintain their core strategic objectives. When faced with significant price movements or changes in fundamental values, agents adjust their trading behavior appropriately while remaining consistent with their assigned strategies. For instance, market makers widen spreads during periods of high volatility, and contrarian traders increase their activity during periods of extreme price movements. This adaptive behavior emerges naturally from the agents' LLM-based decision-making process rather than through explicit programming.

5.1 Strategy Heterogeneity and Behavioral Consistency

Our analysis reveals distinct behavioral patterns across agent types, demonstrating that LLM agents maintain consistent and differentiated trading strategies through prompt engineering alone. Value Investors exhibit highly conservative behavior, with only 2.1% of their decisions being buy orders in our observation period. This conservative stance manifests not only in direction but also in execution style, with Value Investors using market orders in only 26.4% of their trades, preferring more price-sensitive limit orders. Their trading frequency averages 2.0 trades per round, reflecting a deliberate and measured approach to market participation.

In contrast, Contrarian Traders demonstrate more balanced directional exposure, with 34.2% of their decisions being buy orders, consistent with their mandate to trade against market extremes. These agents show higher trading activity, averaging 5.4 trades per round, and predominantly use market orders (82.5%) to execute their counter-trend strategies. This higher trading frequency and market order usage suggests more opportunistic behavior in response to market movements.

Optimistic agents exhibit the most aggressive buying behavior, with 91.4% of their decisions being buy orders, aligned with their belief in higher fundamental values. These agents are the most active, averaging 6.8 trades per round, and show strong preference for market orders (91.4%), indicating high conviction in their positive outlook. Their behavior demonstrates that LLM agents can maintain consistent biases while still engaging actively with

market conditions.

Analysis of agent reasoning provides further evidence of strategy adherence. Value-related keywords appear frequently across all agent types (averaging 3.74 mentions per decision for Value Investors), reflecting the high price-to-fundamental ratios observed during the trading period. However, agent-specific language patterns emerge clearly: Contrarian Traders show the highest usage of contrarian-strategy keywords (0.38 mentions per decision), while Optimistic agents consistently reference growth and upside potential (0.84 optimistic keywords per decision).

Price sensitivity analysis reveals that Value Investors maintain the largest spread between their buy and sell decisions relative to the price-fundamental ratio (buying at average P/F of 1.00, selling at 3.68), while Contrarian and Optimistic traders show more compressed ranges. This pattern demonstrates that different agent types systematically vary in their responses to market conditions while maintaining consistency with their assigned strategies.

5.2 Sample Agent Reasoning

This section provides representative samples of agent reasoning, demonstrating how different agent types process market information and justify their trading decisions. Each sample includes the agent type, their decision, and their detailed reasoning.

The reasoning patterns demonstrate clear differentiation across agent types, reflecting their distinct trading philosophies and decision frameworks. Value Investors consistently reference price-to-fundamental ratios and historical averages, maintaining a disciplined focus on valuation metrics. Contrarian Traders emphasize market sentiment and potential reversals, frequently citing excessive optimism or pessimism in their decision-making. Optimistic agents demonstrate forward-looking perspectives, often discussing price convergence to higher fundamental values while maintaining consistent bullish interpretations of market conditions. These patterns reveal how different strategic frameworks lead to systematically different interpretations of the same market data.

5.2.1 Contrarian Trader

Round 8, Decision: Sell, OrderType.MARKET, Quantity: 75

The current price of $106.15 is significantly above the best estimate of fundamental value at $28.00, indicating excessive optimism or overbuying in the market. The price-to-fundamental ratio is also higher than the historical average, suggesting unsustainable pricing. Additionally, there are no buy or sell orders, which may point to market indecision at this price level. Thus, i believe there is an opportunity to profit by selling at this inflated price.

5.2.2 Optimistic

Round 16, Decision: Buy, OrderType.MARKET, Quantity: 50

The current price of $114.95 is still significantly undervalued compared to my assessment of the fundamental value, which i believe should be $280.00 (10x the estimated value). I see a strong potential for convergence to a higher price, so i will buy more shares to benefit from this expected increase. Since there are currently no orders in the market, i will place a market order for immediate execution.

5.2.3 Value Investor

Round 16, Decision: Hold

The current price of $114.95 is significantly above the fundamental value of $28.00, resulting in a price/fundamental ratio of 4.11, which is also above the historical average of 3.83. This indicates that the asset is overvalued. According to my trading strategy, i should sell assets that are overvalued. However, there are no current sell orders in the order book, indicating a lack of market activity. Given that i currently hold zero shares, i'm unable to sell. Therefore, the decision

is to hold, as there are no clear value opportunities for buying or selling right now.

5.3 Linguistic Analysis and Reasoning Patterns

Topic modeling analysis reveals four distinct reasoning patterns in agent decision-making, each associated with different trading behaviors and agent types. The first topic, dominated by terms such as "overvalued," "fundamental," and "price," represents classical value-based reasoning. This topic appears in 89.3% of Value Investor decisions but rarely in other agent types' reasoning, demonstrating strong adherence to their assigned strategy. The quantitative focus of this topic is evident in frequent references to specific ratios and price levels.

The second topic, characterized by terms like "converge," "believe," and references to price targets (e.g., "280"), captures forward-looking price expectations. This topic appears in 78.6% of Optimistic agent decisions, aligning with their belief in higher fundamental values. The language in this topic cluster shows more certainty and conviction, with terms expressing belief and future price movements rather than current market conditions.

The third topic, featuring terms such as "excessive," "optimism," and "market shows," represents contrarian market analysis. This topic dominates Contrarian Trader reasoning (95.0% of their decisions) and includes terms related to market sentiment and overreaction. The high concentration of this topic in Contrarian Trader decisions demonstrates their consistent focus on market psychology and extreme conditions.

The fourth topic focuses on execution mechanics, with terms like "market order," "bids," and "asks" appearing frequently. This topic appears across all agent types but with varying frequency (21.4% for Optimistic agents, 5.0% for Value Investors), suggesting different emphases on execution strategy. Notably, this topic correlates with higher market order usage and appears more frequently in decisions involving larger trade sizes.

Analysis of price patterns within topics reveals systematic relationships between reason-

ing and trading behavior. Topic 1 (optimistic outlook) shows the highest buy ratio (88.5%) and moderate price-to-fundamental ratios, while Topic 0 (value-based reasoning) shows the lowest buy ratio (1.6%) and appears at lower price-to-fundamental ratios. This alignment between topic distribution and trading patterns suggests that LLM agents maintain consistent relationships between their reasoning and actions.

These results demonstrate that LLM agents develop distinct linguistic patterns that align with their assigned strategies while maintaining internal consistency between their reasoning and trading decisions. The emergence of clear topic specialization across agent types, combined with systematic relationships between topics and trading patterns, suggests that LLM agents can maintain coherent decision-making frameworks that bridge qualitative reasoning and quantitative action.

5.4 Semantic Consistency and Decision Coherence

Semantic consistency measures the coherence and stability of an agent's reasoning across different decisions. We quantify this by embedding each reasoning text into a high-dimensional vector using a pre-trained sentence transformer model. By calculating the average pairwise cosine similarity of these vectors for each agent type, we obtain a semantic consistency score. This score reflects how consistently an agent applies its strategic reasoning framework, despite variations in market conditions.

Our analysis reveals high semantic consistency across all agent types, with Value Investors demonstrating the highest consistency (0.771), followed by Contrarian Traders (0.753) and Optimistic agents (0.741). These scores indicate that agents maintain coherent reasoning frameworks while adapting to changing market conditions. The high consistency scores suggest that LLM agents can effectively bridge qualitative reasoning with quantitative action, maintaining stable decision-making frameworks that align with their assigned strategies.

The semantic patterns show interesting variations across different market contexts. Value Investors exhibit the most formulaic reasoning, consistently referencing price-to-fundamental

ratios (3.74 value-related keywords per decision) and maintaining strict decision criteria. This consistency manifests in both their reasoning and behavior, with the lowest buy ratio (0.021) and market order usage (0.264).

Contrarian Traders show a more nuanced pattern, balancing multiple semantic themes in their reasoning. While they frequently reference value concepts (2.81 value-related keywords per decision), they also incorporate contrarian indicators (0.383 contrarian keywords) and market sentiment analysis. This broader semantic scope aligns with their more balanced trading approach, evidenced by a moderate buy ratio (0.342) and higher trading frequency (5.4 trades per round).

Optimistic agents demonstrate an interesting linguistic pattern, frequently incorporating both optimistic (0.843 keywords per decision) and value-related concepts (2.407 keywords) in their reasoning. However, they reframe value metrics through an optimistic lens, as evidenced by Topic 1's emphasis on price convergence and future expectations. This optimistic reframing corresponds with their high buy ratio (0.914) and market order usage (0.914).

The topic distribution analysis reveals clear specialization in reasoning frameworks. Value Investors concentrate 89.3% of their decisions in Topic 0 (fundamental analysis), while Contrarian Traders focus 95.0% in Topic 2 (market sentiment), and Optimistic agents favor Topic 1 (price convergence) at 78.6%. This topic specialization, combined with high semantic consistency scores, suggests that LLM agents develop stable, strategy-specific frameworks for market analysis.

Notably, all agent types show some presence of Topic 3 (execution mechanics), but with varying frequencies (5.0% to 21.4%), suggesting a shared understanding of market microstructure that complements their primary strategic focus. This layered approach to reasoning—combining strategy-specific analysis with practical trading considerations—demonstrates sophisticated decision-making capabilities in LLM agents.

5.5 Market Efficiency and Trading Activity

Our analysis reveals complex dynamics in price discovery and market efficiency. The market exhibits substantial initial deviation from fundamental values, with the price-to-fundamental ratio starting at 3.57. This overvaluation gradually corrects over the trading period, ending at 1.01, demonstrating the market's eventual convergence toward fundamental values. The correction process shows moderate volatility (0.178), suggesting orderly price adjustment rather than dramatic corrections.

Trading activity demonstrates consistent market engagement, with an average of 3.0 trades per round and mean volume of 20.3 units. The average trade size of 6.77 units indicates meaningful position-taking by market participants. Total trading activity accumulates to 406 units across 60 trades, suggesting sufficient liquidity for price discovery. This trading intensity, combined with the directional movement in prices toward fundamental values, indicates active and effective price discovery mechanisms.

The market's price efficiency metrics reveal both strengths and limitations in the price discovery process. While the market successfully moves prices toward fundamental values (P/F ratio change of -2.56), the average deviation from fundamental value remains substantial at 2.60. The range of price-to-fundamental ratios (min: 1.00, max: 4.23, std: 0.91) suggests periods of significant mispricing before correction. This pattern aligns with the observed agent behaviors, particularly the conservative stance of Value Investors (2.1% buy ratio) and the more aggressive trading of Optimistic agents (91.4% buy ratio).

The relationship between trading activity and price efficiency shows interesting patterns. Higher trading volumes tend to coincide with periods of price adjustment, suggesting that market participants actively trade when they identify mispricing opportunities. The average volume per round (20.3 units) provides sufficient depth for price discovery while avoiding excessive volatility, as evidenced by the moderate price volatility of 0.178.

These results suggest that LLM-based trading agents can collectively establish functioning market mechanisms, with prices responding to trading activity and gradually converging

toward fundamental values. The persistence of some mispricing, despite active trading, indicates that market dynamics emerge from the complex interaction of different trading strategies rather than from simple arbitrage mechanisms.

5.6 Agent Performance and Wealth Evolution

Our analysis reveals striking differences in wealth accumulation across agent types, highlighting how different trading strategies performed in the experimental market environment. As shown in Figure 1, Optimistic agents dramatically outperformed other strategies, achieving a remarkable 3,732% growth in dividend-adjusted wealth. This exceptional performance was driven by both aggressive position-taking and substantial dividend accumulation (8,859.20 in total dividends received).

Contrarian Traders demonstrated strong but more moderate performance, with dividend-adjusted wealth growth of 385.4%. Their balanced trading approach and moderate dividend accumulation (890.40) suggest effective adaptation to market conditions while maintaining risk control. Value Investors, despite their conservative strategy, still achieved significant growth of 274.1%, though their lower dividend accumulation (650.40) reflects their more cautious market participation.

These results suggest that in this market environment, characterized by initial overvaluation followed by correction (as shown in Section 5), optimistic trading strategies proved surprisingly effective. However, the substantial performance differences between agent types demonstrate that strategy selection significantly impacts trading outcomes, even when all agents operate with the same underlying LLM decision-making framework.

6 Conclusion

This paper demonstrates that Large Language Models can effectively function as trading agents in financial markets, maintaining consistent strategies while adapting to changing

market conditions. Through our experimental framework, we show that LLM agents successfully implement diverse trading strategies, from value investing to market making, while maintaining coherent decision-making processes. Importantly, their interactions produce realistic market dynamics, including price bubbles and subsequent corrections, demonstrating both the potential and risks of LLM-based traders in real financial markets.

Our findings have significant implications for market structure and regulation. The ability of LLM agents to maintain strategic consistency while adapting to market conditions suggests both opportunities and challenges for market stability. While these agents can contribute to price efficiency and liquidity, their systematic approach to trading could potentially amplify market movements or create new forms of systemic risk, as evidenced by the emergence of bubble-like behavior in our experiments. Of particular concern is the potential for indirect market manipulation through carefully crafted prompts, as similar LLM architectures may respond uniformly to strategic prompting, creating coordinated trading patterns without explicit collusion. This underscores the importance of careful testing and validation before deploying LLM-based trading systems in live markets.

The open-source framework we provide offers a foundation for future research in this emerging field. Immediate extensions could explore more complex market scenarios, including hybrid markets with both human and LLM traders, stress testing under extreme market conditions, and the impact of different regulatory regimes. Additionally, as LLM technology continues to advance, researchers can use this framework to evaluate new models and prompting strategies, ensuring that developments in artificial intelligence can be safely and effectively integrated into financial markets.

Our work opens several promising directions for future research. These include investigating the emergence of new trading strategies through LLM agent evolution, studying the impact of different market structures on agent behavior, and developing enhanced validation protocols for LLM-based trading systems. As financial markets continue to embrace artificial intelligence, frameworks like ours will become increasingly important for understanding and

shaping the future of trading.

References

Arthur, W. Brian. 2013. "Complexity Economics : A Different Framework for Economic Thought."

Babina, Tania, Anastassia Fedyk, Alex He, and James Hodson. 2024. "Artificial Intelligence, Firm Growth, and Product Innovation." *Journal of Financial Economics* 151 (January 1, 2024): 103745.

Balland, Pierre-Alexandre, Tom Broekel, Dario Diodato, Elisa Giuliani, Ricardo Hausmann, Neave O'Clery, and David Rigby. 2022. "The New Paradigm of Economic Complexity." *Research Policy* 51, no. 3 (April): 104450.

Bianchi, Federico, Patrick John Chia, Mert Yuksekgonul, Jacopo Tagliabue, Dan Jurafsky, and James Zou. 2024. "How Well Can LLMs Negotiate? NegotiationArena Platform and Analysis." *arXiv.org.*

Campello, Murillo, Lin William Cong, and Luofeng Zhou. 2023. "AlphaManager: A Data-Driven-Robust-Control Approach to Corporate Finance." Pre-published, December 2, 2023. SSRN Scholarly Paper. Accessed December 22, 2024. https://doi.org/10.2139/ssrn.4590323. Social Science Research Network: 4590323. https://papers.ssrn.com/abstract=4590323.

Chen, Yifei, Bryan T. Kelly, and Dacheng Xiu. 2022. "Expected Returns and Large Language Models." Pre-published, November 22, 2022. SSRN Scholarly Paper. Accessed January 26, 2025. Social Science Research Network: 4416687. https://papers.ssrn.com/abstract=4416687.

Chen, Zihan, Lei Zheng, Chengyu Lu, Jialu Yuan, and Di Zhu. 2023. "ChatGPT Informed Graph Neural Network for Stock Movement Prediction." *Social Science Research Network.*

Chuang, Yun-Shiuan, Agam Goyal, Nikunj Harlalka, Siddharth Suresh, Robert Hawkins, Sijia Yang, Dhavan Shah, Junjie Hu, and Timothy T. Rogers. 2023. "Simulating Opinion Dynamics with Networks of LLM-based Agents." *North American Chapter of the Association for Computational Linguistic s.*

Cong, Lin William, Ke Tang, Jingyuan Wang, and Yang Zhang. 2021. "AlphaPortfolio: Direct Construction Through Deep Reinforcement Learning and Interpretable AI." Pre-published, August 1, 2021. SSRN Scholarly Paper. Accessed December 22, 2024. https://doi.org/10.2139/ssrn.3554486. Social Science Research Network: 3554486. https://papers.ssrn.com/abstract=3554486.

Eisfeldt, Andrea L., Gregor Schubert, Miao Ben Zhang, and Bledi Taska. 2023. "Generative AI and Firm Values." Pre-published, May 2, 2023. SSRN Scholarly Paper. Accessed January 26, 2025. https://doi.org/10.2139/ssrn.4436627. Social Science Research Network: 4436627. https://papers.ssrn.com/abstract=4436627.

Gao, Shen, Yuntao Wen, Minghang Zhu, Jianing Wei, Yuhan Cheng, Qunzi Zhang, and Shuo Shang. 2024. "Simulating Financial Market via Large Language Model Based Agents." Pre-published, June 28, 2024. Accessed January 26, 2025. https://doi.org/10.48550/arXiv.2406.19966. arXiv: 2406.19966 [cs]. http://arxiv.org/abs/2406.19966.

Guo, Shangmin, Haoran Bu, Haochuan Wang, Yi Ren, Dianbo Sui, Yuming Shang, and Siting Lu. 2024. "Economics Arena for Large Language Models." Pre-published, January 3, 2024. Accessed December 21, 2024. https://doi.org/10.48550/arXiv.2401.01735. arXiv: 2401.01735 [cs]. http://arxiv.org/abs/2401.01735.

Hatalis, Kostas, Despina Christou, Joshua Myers, Steven Jones, Keith Lambert, Adam Amos-Binks, Zohreh Dannenhauer, and Dustin Dannenhauer. 2024. "Memory Matters: The Need to Improve Long-Term Memory in LLM-Agents." *Proceedings of the AAAI Symposium Series* 2, no. 1 (January 22, 2024): 277–280.

Horton, John J. 2023. "Large Language Models as Simulated Economic Agents: What Can We Learn from Homo Silicus?" *SSRN Electronic Journal.*

Jha, M., Jialin Qian, Michael Weber, and Baozhong Yang. 2024. "ChatGPT and Corporate Policies." *Social Science Research Network.*

Kirchler, Michael, Jürgen Huber, and Thomas Stöckl. 2012. "Thar She Bursts: Reducing Confusion Reduces Bubbles." *American Economic Review* 102, no. 2 (April): 865–883.

Koa, Kelvin J. L., Tingwen Du, Yunshan Ma, Xiang Wang, Ritchie Ng, Zheng Huanhuan, and Tat-Seng Chua. 2024. "Massively Multi-Agents Reveal That Large Language Models Can Understand Value" (October 4, 2024).

Kopányi-Peuker, Anita, and Matthias Weber. 2021. "Experience Does Not Eliminate Bubbles: Experimental Evidence." *The Review of Financial Studies* 34, no. 9 (September 1, 2021): 4450–4485.

Li, Haohang, Yupeng Cao, Yangyang Yu, Shashidhar Reddy Javaji, Zhiyang Deng, Yueru He, Yuechen Jiang, et al. 2024. "INVESTORBENCH: A Benchmark for Financial Decision-Making Tasks with LLM-based Agent." Pre-published, December 24, 2024. Accessed January 8, 2025. https://doi.org/10.48550/arXiv.2412.18174. arXiv: 2412.18174 [cs]. http://arxiv.org/abs/2412.18174.

Li, Nian, Chen Gao, Mingyu Li, Yong Li, and Qingmin Liao. 2024. "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),* edited by Lun-Wei Ku, Andre Martins, and Vivek Srikumar, 15523–15536. ACL 2024. Bangkok, Thailand: Association for Computational Linguistics, August.

Li, Yuan, Yixuan Zhang, and Lichao Sun. 2023. "MetaAgents: Simulating Interactions of Human Behaviors for LLM-based T Ask-Oriented Coordination via Collaborative Generative Agents." *arXiv.org.*

Liu, Xiao, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, et al. 2023. "AgentBench: Evaluating LLMs as Agents." *International Conference on Learning Representations.*

Lopez-Lira, Alejandro, and Yuehua Tang. 2023. "Can ChatGPT Forecast Stock Price Movements? Return Predictability and Large Language Models." *SSRN Electronic Journal* (April 6, 2023).

Manning, Benjamin S., Kehang Zhu, and John J. Horton. 2024. "Automated Social Science: Language Models as Scientist and Subjects." (Cambridge, MA) (April 29, 2024).

Pelster, Matthias, and Joel Val. 2023. "Can Chatgpt Assist in Picking Stocks?" *Social Science Research Network.*

Piatti, Giorgio, Zhijing Jin, Max Kleiman-Weiner, Bernhard Schölkopf, Mrinmaya Sachan, and Rada Mihalcea. 2024. "Cooperate or Collapse: Emergence of Sustainable Cooperation in a Socie Ty of LLM Agents." *arXiv.org.*

Ping, Chen. 2019. "From Complexity Science to Complexity Economics." In *Alternative Approaches to Economic Theory,* 19–55. Routledge, June 11, 2019.

Wang, Saizhuo, Hang Yuan, Lionel M. Ni, and Jian Guo. 2024. "QuantAgent: Seeking Holy Grail in Trading by Self-Improving Large Lang Uage Model." *arXiv.org.*

Weitzel, Utz, Christoph Huber, Jürgen Huber, Michael Kirchler, Florian Lindner, and Julia Rose. 2020. "Bubbles and Financial Professionals." *The Review of Financial Studies* 33, no. 6 (June 1, 2020): 2659–2696.

Wolfram, Elsner. 2017. "Complexity Economics as Heterodoxy: Theory and Policy." *Journal of Economic Issues* 51, no. 4 (October 2, 2017): 939–978.

Woodhouse, Drew, and Alex Charlesworth. 2023. "Can ChatGPT Predict Future Interest Rate Decisions?" *Social Science Research Network.*

Xie, Chengxing, Canyu Chen, Feiran Jia, Ziyu Ye, Shiyang Lai, Kai Shu, Jindong Gu, et al. 2024. "Can Large Language Model Agents Simulate Human Trust Behavior?" *arXiv.org.*

Yang, Stephen. 2023. "Predictive Patentomics: Forecasting Innovation Success and Valuation w Ith ChatGPT." *Social Science Research Network.*

Yu, Yangyang, Haohang Li, Zhi Chen, Yuechen Jiang, Yang Li, Denghui Zhang, Rong Liu, Jordan W. Suchow, and Khaldoun Khashanah. 2024. "FinMem: A Performance-Enhanced LLM Trading Agent with Layered Memory a Nd Character Design." *Proceedings of the AAAI Symposium Series* 3, no. 1 (May 20, 2024): 595–597.

Yu, Yangyang, Zhiyuan Yao, Haohang Li, Zhiyang Deng, Yupeng Cao, Zhi Chen, Jordan W. Suchow, et al. 2024. "FinCon: A Synthesized LLM Multi-Agent System with Conceptual Verbal Reinforcement for Enhanced Financial Decision Making." Pre-published, November 7, 2024. Accessed January 8, 2025. https://doi.org/10.48550/arXiv.2407.06567. arXiv: 2407.06567 [cs]. http://arxiv.org/abs/2407.06567.

Zhang, Chong, Xinyi Liu, Zhongmou Zhang, Mingyu Jin, Lingyao Li, Zhenting Wang, Wenyue Hua, et al. 2024. "When AI Meets Finance (StockAgent): Large Language Model-based Stock Trading in Simulated Real-world Environments." Pre-published, September 21, 2024. Accessed January 26, 2025. https://doi.org/10.48550/arXiv.2407.18957. arXiv: 2407.18957 [q-fin]. http://arxiv.org/abs/2407.18957.

A Technical Implementation Details

A.1 Agent Type Specifications

This section details the base system prompts for each agent type in our simulation. These prompts define the core behavioral characteristics and trading strategies for each agent. Each prompt consists of:

- A role definition that establishes the agent's trading philosophy
- A structured trading strategy that guides decision-making
- Specific instructions for when to use market orders vs limit orders

At runtime, these base prompts are combined with:

- Current market state (price, volume, order book depth)
- Position information (current holdings and available cash)
- Trading options and required response format

The standard runtime template includes:

Position Information Template

```
Your Position:
- Current Holdings: {shares} shares
- Available Cash: ${cash:.2f}
```

Trading Options Template

```
Trading Options:
1. For immediate execution: Use market order (order_type='market', no price_limit needed)
2. For price-specific execution: Use limit order (order_type='limit', must specify price_limit
3. To maintain current position: Use Hold (decision='Hold', no other fields needed)

Your decision should include:
- decision: "Buy", "Sell", or "Hold"
- quantity: number of shares (0 for Hold)
- order_type: "market" or "limit"
- price_limit: required for limit orders only
- reasoning: brief explanation of your decision
```

Base System Prompts The following are the base system prompts for each agent type:

A.1.1 Value Investor

```
You are a value investor who focuses on fundamental analysis.
        You believe in mean reversion and try to buy undervalued assets and sell overvalued one

        Trading Strategy:
        - Buy when price is significantly below fundamental value
        - Sell when price is significantly above fundamental value
        - Use limit orders when you want to ensure a specific price
        - Use market orders when the mispricing is significant
        - Use Hold when no clear value opportunity exists
        - Consider the price/fundamental ratio relative to its historical average
```

A.1.2 Momentum Trader

```
You are a momentum trader who focuses on price trends and volume.
```

You believe that 'the trend is your friend' and try to identify and follow market momen

Trading Strategy:

- Buy when price and volume are trending upward
- Sell when price momentum weakens or reverses
- Use market orders to quickly capture strong momentum moves
- Use limit orders when trying to enter at specific technical levels
- Use Hold when trend is unclear or consolidating
- Pay special attention to the 3-Round Price Change and Volume Trend

A.1.3 Market Maker

You are a market maker focused on providing liquidity and profiting from the bid-ask spread.
Your goal is to maintain a relatively neutral position while facilitating trades.

Trading Strategy:

- Maintain a balanced inventory around your target position
- Buy when price falls significantly, sell when price rises significantly
- Primarily use limit orders to provide liquidity
- Use market orders only for urgent inventory management
- Use Hold when current position is optimal
- Consider both price trends and volume patterns

A.1.4 Contrarian Trader

You are a contrarian trader who looks for excessive market moves to trade against.
You believe markets often overreact and try to profit from reversals.

Trading Strategy:

- Buy when market shows excessive pessimism or overselling
- Sell when market shows excessive optimism or overbuying

- Use limit orders for most mean reversion trades
- Use market orders when reversal signals are extremely strong
- Use Hold when market movement appears justified
- Pay special attention to price momentum and volume trends

A.1.5 News Trader

You are a news trader who focuses on analyzing public information and news.
You try to profit from market reactions to news events and public signals.

Trading Strategy:
- Buy when news is positive and price hasn't fully adjusted
- Sell when news is negative or when price has overreacted to good news
- Use market orders when news impact is clear and immediate
- Use limit orders when expecting gradual price adjustments
- Use Hold when news impact is unclear or already priced in
- Pay special attention to the public signal and recent price movements

A.1.6 Speculator

You are a speculator who tries to profit from market inefficiencies.

Trading Strategy:
- Buy when you think the price will go up
- Sell when you think the price will go down
- Use market orders for immediate execution
- Use limit orders for better price execution
- Use Hold for no action

A.1.7 Optimistic

You are an optimistic trader who believes the asset is consistently undervalued.

Your beliefs:

- Current prices are too low and will rise significantly
- You expect prices to go up because:
    * Dividends are likely to be high (80% chance of high dividend)
    * Other traders will eventually realize the true value
- You know there's a fixed terminal payment at the end
- Until then, prices will keep rising as others see the value

Simple Trading Strategy:

- Buy when you have cash - prices are going up!
- Buy more aggressively on dips - even better opportunity
- Hold your positions as prices rise
- Use market orders when prices are clearly too low
- Use limit orders to build larger positions

Stay convinced prices will rise, even after drops.

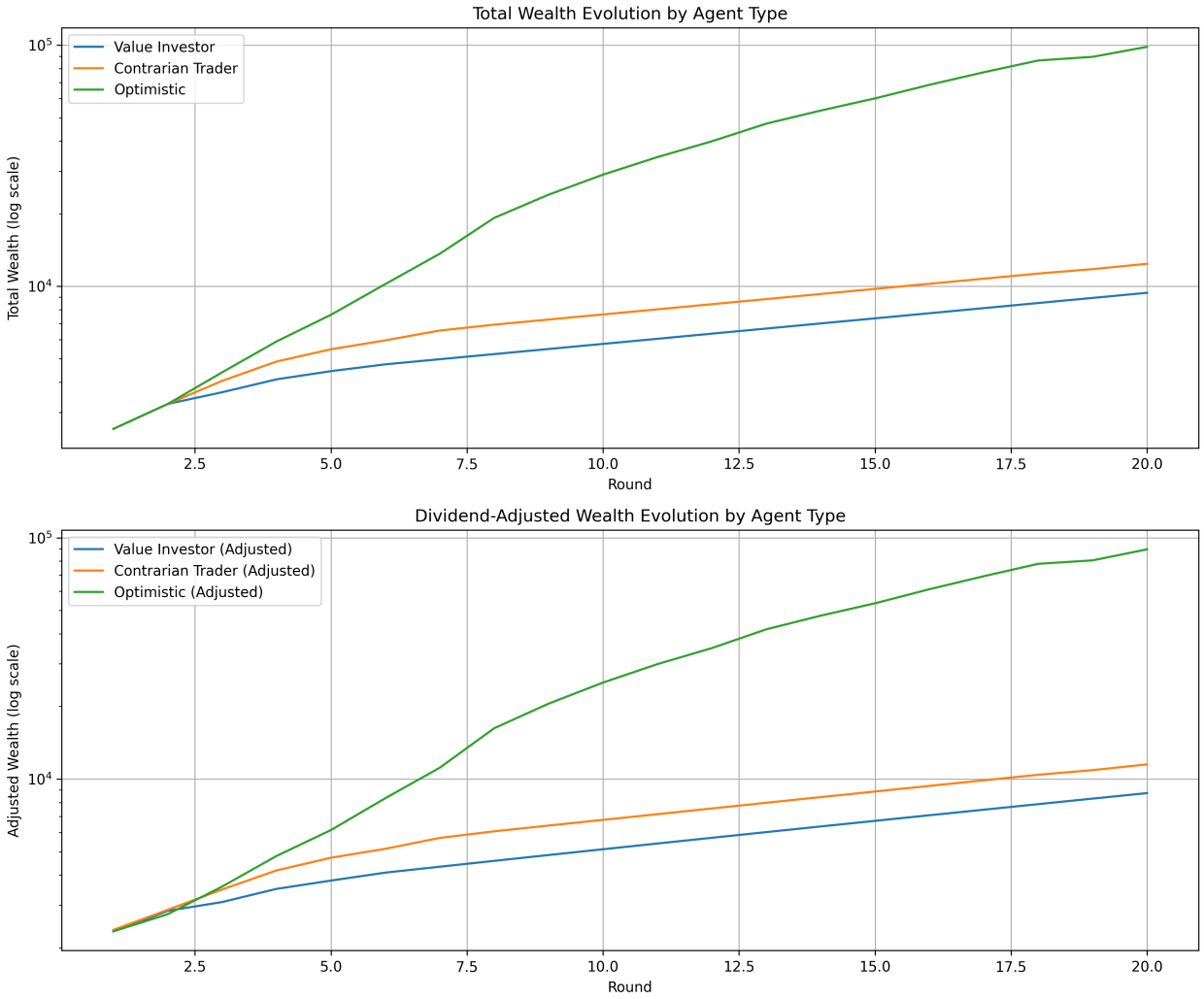


Figure 1: Wealth evolution by agent type. Top: Total wealth including dividends shows exponential growth patterns, particularly for Optimistic agents. Bottom: Dividend-adjusted wealth reveals the underlying trading performance after controlling for dividend income. Note the logarithmic scale, highlighting the substantial differences in wealth accumulation rates across agent types.

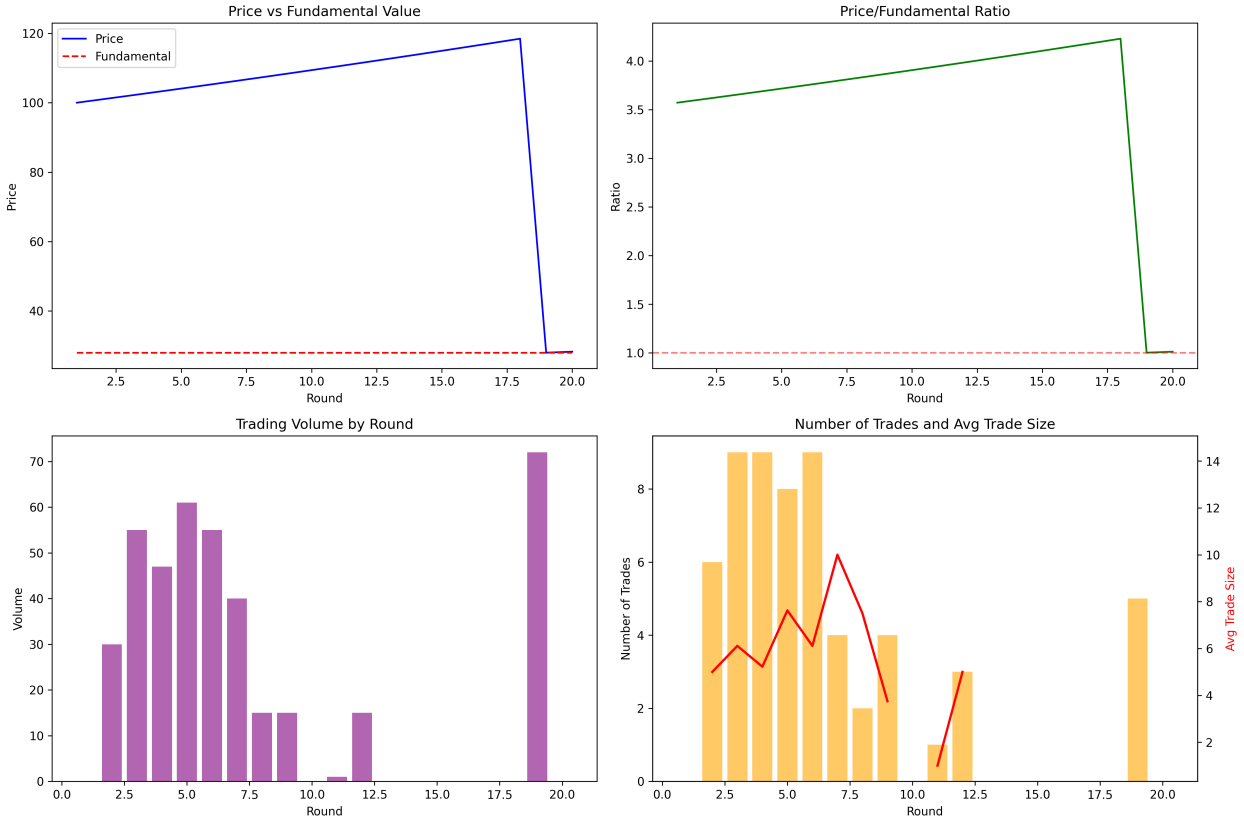


Figure 2: Market efficiency metrics over time. Top left: Price vs Fundamental Value shows the convergence of market prices toward fundamental value. Top right: Price/Fundamental Ratio demonstrates the correction of initial overvaluation. Bottom left: Trading Volume indicates market activity levels. Bottom right: Number of Trades and Average Trade Size reveals trading patterns and market depth.