Visualization Design for Stock Market Simulation with Performance Analysis of Trading Algorithms

Vladimíra Kimlová[a,]

[a]*Department of Computer Science and Engineering, Faculty of Applied Sciences, University of West Bohemia, Plzeň, Czech Republic*

Abstract

The complexity and opacity of financial markets pose significant challenges for analysis, particularly in environments driven by high-frequency and algorithmic trading. This article presents two visualization tools developed within a stock market simulation platform: a web-based trading interface and a lightweight evaluation component. The web UI enables users to monitor real-time market activity and submit orders via an intuitive interface that mirrors professional trading terminals. Complementing this, the evaluation tool provides a structured statistical overview and visual snapshot for comparing the performance of different trading strategies. These visualizations enhance the interpretability of simulated market behavior and support both educational exploration and empirical analysis of trading dynamics and algorithm effectiveness.

Keywords: Trading Visualization, Stock Market Simulation, Algorithmic Trading, Trading Strategy Evaluation, Limit Order Book, Interactive Trading UI

1. Introduction

Financial markets are inherently complex, characterized by high-speed transactions, strategic behavior, and limited transparency. With the increasing prevalence of algorithmic trading, understanding the dynamics within a limit order book has become more challenging yet more critical – both for researchers and for practitioners. Analyzing such systems is difficult not only due to their technical nature, but also because relevant data are often proprietary, expensive, or deliberately obfuscated to conceal manipulative practices.

Simulators provide a valuable alternative, enabling controlled experimentation and reproducible analysis of market behavior. However, interpreting the raw data generated by such simulations can be non-trivial. Without suitable visualization tools, valuable insights into trading dynamics, agent behavior, and strategy performance can remain hidden.

This article introduces two visualization tools developed within a modular stock market simulator. The first is a real-time trading interface that allows users to observe market activity and place orders via an interactive web-based UI. The second is a compact evaluation module for post-simulation analysis, providing strategy comparison through tabular statistics and graphical summaries. These components aim to bridge the gap between raw simulation output and user understanding, facilitating both educational exploration and empirical research into algorithmic trading and market microstructure.

2. Related Work

Visualizations of financial market data play a crucial role in both academic research and practical trading applications. Existing solutions can be broadly divided into three categories: (i) real-time trading interfaces, (ii) visual analytics of order book data, and (iii) tools for evaluating algorithmic trading strategies.

Traditional trading platforms, such as Bloomberg Terminal [1] or MetaTrader [2], provide high-frequency price charts, order book depth views, and execution panels. These interfaces are designed for professional traders, offering low-latency updates and rich interaction options. However, their complexity and proprietary nature make them less suitable for educational or research-oriented use.

Academic research has explored various visualization techniques for order book dynamics. For instance, Ver-

Email address: jivl@students.zcu.cz (Vladimíra Kimlová)

hulst et al. [3] applied particle physics tools to visualize the limit order book, enhancing the understanding of market microstructure. Similarly, the OrderBookVis framework [4] employs 2.5D ridgeline plots to compare order books across different exchanges, facilitating the analysis of temporal developments in order book structures.

In the context of evaluating algorithmic trading strategies, platforms like QuantConnect [5] and Backtrader [6] provide performance summaries through return curves, risk metrics, and trade logs. However, these tools are often tightly coupled to historical data backtesting and lack direct integration with agent-based simulation environments.

While some research simulators include basic data exports, few provide user-friendly visual interfaces for observing real-time simulation progress or interpreting strategy outcomes. The tools presented in this work aim to address this gap by combining live market visualization with interpretable strategy evaluation, tailored specifically for agent-based simulation of limit order book markets.

3. Methods

To facilitate the observation and evaluation of trading dynamics within a simulated limit order book environment, two principal visualization modules were developed: an interactive trading interface and a post-simulation evaluation tool. These modules are designed not only to enhance the interpretability of market activity but also to support educational and experimental use cases.

3.1. Real-Time Trading Interface

The web-based trading interface was implemented using the `Bokeh` visualization library, which supports the creation of interactive, data-driven visualizations for browser-based applications in Python. The design emphasizes clarity and accessibility, enabling users to manually submit buy and sell orders, monitor live market conditions, and manage their portfolio without requiring programming knowledge.

The interface is structured into several functional panels:

- **Active Orders Panel:** Displays a ranked list of the top ten bid and ask orders, including associated prices and volumes. This view offers a concise summary of the current order book liquidity and market sentiment.
- **Price Chart**: Visualizes the temporal evolution of the best bid and ask prices (*Top of Book*). This helps users track short-term market trends and identify volatility. The chart supports `pan`, `box zoom`, `wheel zoom`, `reset`, and `hover` tools for detailed inspection.
- **Order Book Depth View**: Represents the aggregated distribution of buy and sell orders across discrete price levels, providing insight into market depth and potential price support or resistance. The plot includes an indication of the *mid-price*, defined as:

$$P_{\text{mid}}(t) = \frac{P_{\text{best\_bid}}(t) + P_{\text{best\_ask}}(t)}{2}$$

  This view is dynamically updated and also supports interactivity such as zooming, panning, and tooltips for precise inspection of individual order levels.
- **Market Statistics Panel:** Summarizes key market indicators, including daily price change, order book imbalance index, and market granularity. These metrics offer a high-level overview of current market conditions.
- **Trade History Table:** Presents a chronologically ordered list of recently executed trades, including transaction prices, quantities, and timestamps. The table supports dynamic sorting, enhancing usability and analytical flexibility.
- **Trading Panel:** Facilitates the submission of buy and sell orders, displays account balances, and allows for the management of open positions.

All interface components are updated in real time using asynchronous callbacks, enabling uninterrupted interaction with live market data. This interactivity not only enhances the user experience but also mirrors the responsiveness of real-world trading platforms, making the system suitable for experimentation, teaching, and behavioral research.

3.2. Post-Simulation Evaluation Tool

To support systematic comparison of trading strategies, a lightweight evaluation module was implemented using the `Plotly` library. This tool provides a dual-mode analysis: graphical time series visualizations and tabular performance summaries.

The core visualization is a multi-line chart displaying the net earnings trajectories of best individual agents

throughout the simulation. Net earnings are defined as the sum of realized profit from closed positions and the mark-to-market value of currently held assets:

$$\text{Earnings}_i(t) = \sum_{j=1}^{N_{\text{closed}}} \Pi_{ij} + q_i(t) \cdot p_{\text{mid}}(t),$$

where Π_{ij} denotes the profit from the j-th closed trade of agent i, $q_i(t)$ is the agent's current position size, and $p_{\text{mid}}(t)$ is the mid-price at time t. A reference curve indicating the mid-price evolution is included to contextualize agent behavior with respect to broader market trends. The evaluation interface is also interactive, allowing users to filter participants, zoom in on specific periods, and extract detailed comparisons between strategies.

Supplementing this visual analysis, a performance summary table reports key metrics for selected participants, including:

- `Final Balance`: Final account value after deducting messaging or trading fees.
- `Return (%)`: Relative return on the initial capital.
- `Avg Volume per Step` and `Max Volume`: Mean and peak position sizes held.
- `Avg Balance`: Time-averaged balance excluding current unrealized gains.
- `Num Orders`: Total number of submitted orders.

The tabular data supports interactive column-based sorting and filtering, further enhancing its analytical value. Together, the visualization components of the evaluation tool enable a nuanced understanding of strategy effectiveness and behavioral patterns across agents.

3.3. Design Rationale

The chosen visualization types – line charts for temporal evolution, depth views for order book inspection, tables for strategy metrics – were selected for their intuitive interpretability and widespread usage in financial analytics. The integration of real-time interactivity via `Bokeh` and high-resolution post-hoc analysis through `Plotly` creates a comprehensive environment for studying trading systems and evaluating algorithmic performance under simulated market conditions.

4. Results

To assess the utility and performance of the developed system, the key visual components of the interface alongside a quantitative summary of trading outcomes from selected algorithmic agents are presented. The figures below illustrate both the real-time interaction layer and the post-simulation evaluation tool.

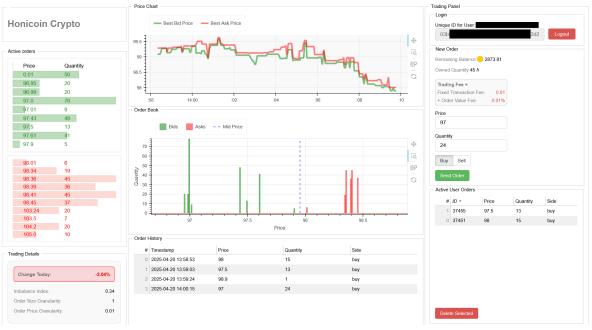

Figure 1: Web-based real-time trading interface with order book, market chart, and order controls.

Figure 1 shows the web UI used by participants during the simulation. The interface displays current market activity, submitted orders, and basic user statistics. In this example, we can observe a user who has placed six orders, two of which are still active. The design allows users to monitor and react to market changes quickly while tracking their own trading activity.

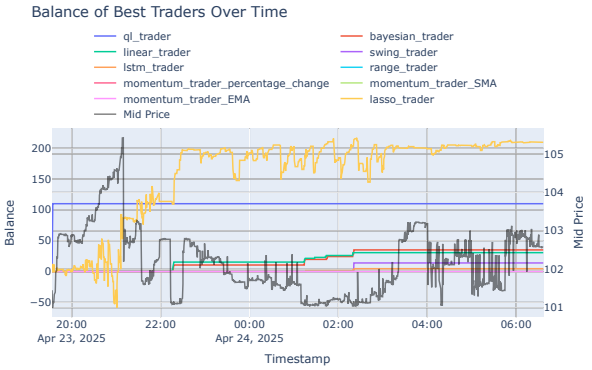

Figure 2: Evaluation dashboard showing trader performance over time.

Figure 2 presents a visual summary of selected traders' performance after the simulation. While some traders, such as the lasso agent, show strong early performance, their final ranking does not always reflect this due to factors like transaction costs. The visualization helps highlight not only final outcomes but also trading style and risk-taking behavior during the session.

Table A.1 in Appendix A complements the visual analysis by providing a detailed summary of key performance metrics for all algorithmic agents. Beyond absolute profitability, the table highlights differences in trading behavior, such as order frequency and volume, which are crucial for evaluating strategy efficiency. For instance, the Q-learning agent stands out with the highest final balance, reflecting a relatively balanced trading approach. In contrast, the Lasso-based agent, which appears strong in the time-series plot (Figure 2), ultimately ranks lower due to an overly aggressive submission pattern that leads to high transaction fees. Similarly, the scalping strategy, despite its high activity level, suffers substantial losses – demonstrating how trading costs and behavioral patterns significantly impact net performance.

5. Discussion

The system demonstrates how well-designed visual tools can enhance the interpretability of trading behavior and strategy performance within a simulated environment. The live trading interface offers users an intuitive way to interact with the market, providing immediate feedback through components such as the order book depth view and trade execution history. These real-time elements support situational awareness and decision-making, making the interface suitable for both educational and experimental purposes.

The post-simulation evaluation tool further complements the experience by visualizing the evolution of agent performance across the trading session. As shown in Figure 2, the plot makes it possible to identify strategies that dominate in terms of profitability, while also revealing trading intensity over time. However, visual clarity can be challenged when many agents are active simultaneously, motivating a need for better filtering or grouping capabilities.

While the core visualizations are functional and informative, the current design is largely static in terms of what is displayed. This limits users' ability to tailor the interface to specific analysis needs. Introducing more customization options – such as toggling chart components, changing visual mappings, or integrating new market views – would further enrich the analytical potential of the system.

6. Future Work

Future development should focus on expanding the system's flexibility and analytical depth. On the evaluation side, implementing filters to select specific agents or strategy types would allow for clearer comparisons and reduce visual clutter in dense simulations. Additional performance indicators – such as Sharpe ratio, drawdown, or order fill rates – could provide a more nuanced view of trading behavior beyond absolute returns.

For the real-time trading interface, introducing new types of visualizations (e.g., candlestick charts, price impact heatmaps) and customizable layout components would give users more control over how market information is displayed. This would make the platform more adaptable to different user profiles, ranging from novice traders to researchers studying microstructural patterns.

Finally, enabling users to configure which market metrics or strategy outputs are visible in the interface would further personalize the experience and support more focused exploratory analysis.

7. Conclusion

This work presents an interactive simulation platform for evaluating trading strategies in synthetic markets, with a strong emphasis on visual interpretability. Through a combination of real-time user interfaces and post-simulation analytic tools, the system bridges the gap between experimental flexibility and educational clarity. The use of dynamic visualization libraries such as `Bokeh` and `Plotly` enables an engaging user experience while supporting rigorous evaluation of agent performance.

The platform proves effective not only for demonstrating individual strategy outcomes, but also for exploring broader market phenomena, such as the effects of high-frequency trading or fee-sensitive behavior. Despite some current limitations, the system lays a strong foundation for future experimentation and pedagogical use. It represents a practical and extensible tool for researchers, students, and developers interested in algorithmic trading and market dynamics.

Acknowledgment

I would like to express my sincere gratitude to my thesis supervisor, doc Ing. Jan Pospíšil, Ph.D., for his valuable guidance, support, and constructive feedback throughout the course of this work. His expertise and insightful suggestions were instrumental in shaping both the technical and conceptual aspects of the thesis.

References

[1] Bloomberg Professional Services, Bloomberg Terminal, https://www.bloomberg.com/professional/products/bloomberg-terminal/, 2025. Accessed April 2025.

[2] M. S. Corp., MetaTrader 4 trading platform, https://www.metatrader4.com/, 2025. Accessed April 2025.

[3] M. E. Verhulst, P. Debie, S. Hageboeck, J. M. E. Pennings, C. Gardebroek, A. Naumann, P. van Leeuwen, A. A. Trujillo-Barrera, L. Moneta, When two worlds collide: Using particle physics tools to visualize the limit order book, Journal of Futures Markets 41 (2021) 1715–1734. URL: http://dx.doi.org/10.1002/fut.22251. doi:10.1002/fut.22251.

[4] A. Jobst, D. Atzberger, R. Henker, W. Scheibel, J. Döllner, Orderbookvis: A visualization approach for comparing order books from centralized crypto exchanges, 2023. doi:10.1109/ICBC56567.2023.10174944.

[5] QuantConnect Team, QuantConnect platform documentation, https://www.quantconnect.com/docs/, 2024. Accessed April 2025.

[6] Backtrader Community, Backtrader: Python backtesting library, https://www.backtrader.com/, 2024. Accessed April 2025.

Appendix A. Agent Performance Statistics

Table A.1: Summary of trading performance statistics for algorithmic agents.

	Strategy	Balance	Ret [%]	AvgVol	MaxVol	AvgBal	Ords
1	Q-learning	10109.17	1.09	54.88	55	-5519.80	37
2	Bayes. reg.	10034.05	0.34	11.19	23	-1130.97	18
3	Linear reg.	10029.61	0.30	11.09	20	-1120.54	19
4	Swing	10013.61	0.14	3.09	9	-312.67	3
5	LSTM	10003.44	0.03	1.13	3	-114.02	27
6	Range	10000.00	0.00	0.00	0	0.00	0
7	Mom. %	9998.82	-0.01	0.00	0	0.00	27
8	Mom. SMA	9998.82	-0.01	0.00	0	0.00	27
9	Mom. EMA	9998.82	-0.01	0.00	0	0.00	27
10	Lasso reg.	9958.25	-0.42	59.84	89	-5987.86	2180
11	Mom. RSI	9879.97	-1.20	83.30	89	-8541.93	3214
12	RF reg.	9829.06	-1.71	57.13	88	-5788.70	2291
13	Ridge reg.	9791.07	-2.09	65.77	88	-6723.03	2021
14	Scalping	7702.54	-22.97	36.14	82	-3971.76	18547