Financial Statistical Journal (2023) Volume 6 Issue 2
doi: 10.24294/fsj.v6i2.2931

ORIGINAL RESEARCH ARTICLE

Model of stochastic auctions using level market index

Nikonov Maksim[1], Shishkin Alexei[1,*], Konev Dmitry[2], Dolmatov Aleksandr[1]

[1] *Moscow State University, Moscow 119991, Russia*

[2] *Risk Quants, Sberbank, Moscow 121170, Russia*

*** Corresponding author:** Shishkin Alexei, shishkin@cs.msu.ru

ABSTRACT

The following research paper is devoted to the complex topic of modeling stochastic financial markets using the example of auction markets. The presented model for market makers' behavior on stochastic auction markets contributes practically to the field of studying portfolio optimization, risk management, market participants' balance processes, and prediction problems via cutting-edge machine learning and statistics approaches. The reliability of the given model is proved practically with the help of modern machine learning methods of validation, namely, combinatorial splits. A client-server model for remote simulation was implemented, as well as interpreted language in C++. XGBoost, Catboost, LSTM, NN Ensemble, and H_2O Auto-ML models were considered in the course of building the decision model. Hyperparameters were obtained via Optuna. Besides that, the developed model was backtested on historical data of different financial assets, starting with stocks and ending with commodity prices and foreign exchange rates. Within all models, positive Sharpe ratios have been obtained, which indicates the robustness of the model. The paper offers a valuable framework for market maker decision-making stochastic modeling, examining its pricing mechanisms and financial risk management as crucial for exchanges, funds, and other financial institutions, which makes it relevant in the context of the current dynamics of the development of financial markets and the increase in trading volumes.

Keywords: auction market; Kalman filter; financial markets; machine learning; validation; stochastic modelling; modelling stochastic markets; backtesting

ARTICLE INFO

Received: 26 September 2023
Accepted: 27 October 2023
Available online: 17 November 2023

1. Introduction

Modeling of financial markets is an extremely complex and important topic that has been studied by mathematicians and economists for many decades. One of the key aspects of any market, or organized trade, is the pricing mechanism as well as the roles of different participants that drive the changes in prices. A structured and well-designed pricing mechanism ensures liquidity stability, trading efficiency, fair competition, and growth in dynamics of trades.

In an era of evolving auction platforms and digital marketplaces, this field continues to evolve, contributing to our understanding of market dynamics and participants' strategies. It is quite natural, given the heterogeneous behavior of investors, financial institution, and market makers, that the question of modelling of such auction markets is a complex and profound subject that needs comprehensive theoretical and practical development. There are several reasons for that. First of all, financial data often contains a lot of "noise", or random fluctuations, which are hard to distinguish from genuine, significant market signals. In the effort to capture complex patterns in data, machine learning models can become overfitted or too adapted to the historical data they

were trained on, impairing their ability to generalize and predict future trends. This is particularly noticeable in stock markets, where prices can swing unpredictably due to unforeseen events or changes in trader behavior. Moreover, financial data is often characterized by complex, non-linear interdependencies, which makes them very hard to model. Thus, understanding financial markets through machine learning and statistics can help prevent crises by predicting risky trends and bubbles, contributing to economic stability on a global scale. Finally, machine learning provides advanced tools for portfolio optimization, leveraging not just the statistical characteristics of assets but also analyzing market conditions and behavioral factors. This enables more effective risk management and the construction of robust investment strategies.

The auction market is a special type of market with a specific sales organization, which implies that the price of a good is determined by the process of competition between buyers. Sales are organized in accordance with the established rules, and the winner is the one with the highest bid price. The asset (stock, foreign exchange, commodity, or any other complex financial product), being the object of the auction, becomes the winner's own after all the settlements. Auction market modeling is a vital field within finance and economics, aiming to capture the dynamic interplay of buyers and sellers in auction-based trading environments. In such models, researchers introduce stochastic variables to account for factors like bidder behavior, information arrival, and price fluctuations. By doing so, one aims for better simulations of the inherent uncertainty of real-world auction markets, which can be influenced by a wide range of unpredictable events.

There are several important components that would help solve this problem. First, a client-server model would allow hypothesis testing to be done remotely without the processes of data preprocessing, data mining, model search, and hyperparameter selection.

Second, the proprietary interpreted language with pre-described machine learning algorithms and the Sharpe ratio calculation would contribute significantly to the uniformity of model backtesting systems and trading algorithms.

And, finally, the model of auction markets would allow us not to overthink the modeling and data processing processes for the above-mentioned problem.

All three components have been developed in this paper.

2. Relevant research and progress

The related works devoted to stochastic auction market modeling include some of the following research papers: Law and Viens[1] developed a high-frequency stochastic model under a limit order book and avoided several common assumptions regarding market participants' price processes (e.g., their independence, constant volumes of trade). The present paper will help examine this theorized approach from a more practical side, using our framework to implement the authors' model on real data and obtain error estimates of machine learning models for his constrained-order approach. Lux[2] tries to explain the characteristics of financial markets as emergent properties of interactions and dispersed activities of a large ensemble of agents populating the marketplace. His results have proposed simple structures that could reproduce the empirical findings to a high degree with statistics that are even quantitatively close to the empirical ones. Our model allows for including large ensembles of agents, which can help to prove his assertions empirically on a larger market data set. Kraft et al.[3] developed stochastic auction market modeling ideas in terms of the electricity (commodity) market and focused on the behavior of different market participants, taking into account their risk aversion. This model allows for a multi-stage stochastic optimization approach to determine optimal bids in financial commodity markets. Their results showed that it is crucial to consider statistical dependences between different underlying price processes during trading cycles. The present paper's model allows each market participant (or participants) to be modeled by a separate program in an interpreted programming language. In addition,

this approach can be used not only in commodity markets but also in other markets, such as stocks, FX, etc. This will help to check the authors' results on other different markets with more precise set-ups. Bubeck et al.[4] considered the revenue maximization problem in stochastic online auction markets. They generalized classical learning algorithms (experts, multi-armed bandits) to their multi-scale models in order to get the generalized approach appropriate to maximize the profits from the trading strategy on stochastic auctions. Our approach will help them validate their model on real market data and verify the profit maximization algorithm with a backtest. It will also allow one to tune parameters for market participants and test their impact on the maximization algorithm.

The stochastic auction market model given in the present research paper has hardly been studied before. The main articles operate with more general approaches to modeling markets.

This research paper introduces a new model with best practice algorithms: a publicly available interpreted programming language for algorithmic trading; the filtering problem without loss of generality; and the basic concepts of machine learning algorithms for the validation of the proposed model. Also, the robustness (in terms of resistance to exchange prices noises) of the model is tested on the real historical data of futures prices (transformed to fit the model inputs) using the Sharpe ratio. It may be useful for marketplaces and other different auction market platforms.

3. Theoretical model

3.1. Determining the model

First, it is necessary to develop the theoretical foundations for the model of stochastic behavior of auction markets.

1) There are total N $\in \mathbb{N}$ entities who act as real market participants. Each market participant has a unique index $1 \leq k \leq N$. Ek = const—the quantity of new products for the market, ε—random variable determining the costs, and depending on the number of participants at the cycle, a market commission is charged from each participant ω–r.v. determined by market level index. All participants have an initial capital equal to ηk for participant with index k, and $\delta(k)$ is a random variable determining the costs of purchasing a product on the market. The abovementioned is represented in the **Table 1** below.

Table 1. Market rules.

	Quantity	**Increment**	**Costs**
Stocks	Q_k	E_k	ε
Currency	η_k	?	$Q_k \times \delta(k)+\omega$

2) Each cycle, the market holds auctions to sell and purchase selected assets to/from the participants, choosing the most optimal offer depending on the transaction direction (the buyer requires the lowest price, the seller—the highest price). There exists a book of orders with limited purchasing and selling capacities; the participants need to choose the appropriate price, for instance, based on previous cycles. In the general case, the unit offer is determined by the following formula:

$$L_k=\alpha_k \times \mu_k \quad (1)$$

where $L_k \in \mathbb{Z}$, $\alpha_k \in \mathbb{N}$—quantity of assets, $\mu_k \in \mathbb{Z}$—prices set by the market participant.

3) The total supply is as follows:

$$\sum_{k=0}^{x} L_k \quad (2)$$

where x$\in \mathbb{N}$—number of participant's offers during the cycle.

4) The optimal (the seller requires the highest price possible) supply for the sale is determined by the following formula:

$$S_p = \max(\sum_{i=0}^{N} \sum_{k=0}^{x} L_k),\ 0 \le p \le N \quad (3)$$

Therefore, the market purchases all the assets from the market participants who offer optimal prices, and from them, it will determine the next participant to buy from.

5) Similarly, the optimal purchase offer (the buyer requires the lowest price possible) is:

$$B_p = \min(\sum_{i=0}^{N} \sum_{k=0}^{x} L_k),\ 0 \le p \le N \quad (4)$$

The market will continue to satisfy the supply of participants as long as the quantity of products the market buys or sells is greater than zero.

The market situation can be at one of the m levels. The market offers to buy or sell assets from participants are determined depending on the market level, including pricemin and pricemax for one unit of an asset. The values are determined according to the table (**Table 2**) of market levels with the use of a random walk.

Table 2. Market levels.

	Purchase		**Sell**	
Level	**quantity**	**price$_{min}$**	**quantity**	**price$_{max}$**
1	$(V_s+\mu) * \sum_{k=1}^{N} I_A(k)$	$(W_s+\mu)$	$(V_s'+\mu) * \sum_{k=1}^{N} I_A(k)$	$(W_s'+\mu)$
2	$(V_s+2*\mu) * \sum_{k=1}^{N} I_A(k)$	$(W_s+2*\mu)$	$(V_s'+2*\mu) * \sum_{k=1}^{N} I_A(k)$	$(W_s'+2*\mu)$
3	$(V_s+3*\mu) * \sum_{k=1}^{N} I_A(k)$	$(W_s+3*\mu)$	$(V_s'+3*\mu) * \sum_{k=1}^{N} I_A(k)$	$(W_s'+3*\mu)$
4	$(V_s+4*\mu) * \sum_{k=1}^{N} I_A(k)$	$(W_s+4*\mu)$	$(V_s'+4*\mu) * \sum_{k=1}^{N} I_A(k)$	$(W_s'+4*\mu)$
…	…	…	…	…
m	$(V_s+m*\mu) * \sum_{k=1}^{N} I_A(k)$	$(W_s+m*\mu)$	$(V_s'+m*\mu) * \sum_{k=1}^{N} I_A(k)$	$(W_s'+m*\mu)$

The key issue lies in the market rules, which can differ from one market to another, and that is quite natural. Besides that, participants' behavior may also be different for various assets. To solve this problem, one may use the backtesting approach, which allows one to test the trading system on historical data in order to check its performance, keeping in mind that the participants' behavior and market rules are predetermined.

The topic of financial modeling and financial mathematics was particularly developed in the 1970s with the development of the options market and the emergence of other derivatives. One of the most famous and historically important results in financial mathematics was the famous Black-Scholes formula, named after its authors, Fischer Black and Myron Scholes, after their paper "The pricing of options and corporate liabilities" in 1973. The Black-Scholes formula allowed to price European calls and put options analytically and, despite all of its disadvantages, made a huge contribution to the development of financial markets and financial mathematics.

The Black-Scholes model[5] (more generally) has several assumptions, namely:

a) The underlying asset's price process follows the Geometric Brownian motion (GBM), with μ and σ being constants (drift and diffusion coefficients, respectively):

$$dS_t = \mu S_t dt + \sigma S_t dW_t \quad (5)$$

The equation (5) implies the lognormal distribution of the underlying

b) Short selling of assets is not restricted;

c) There are no transaction costs and taxes, and all the assets are perfectly divisible (say, one can buy or sell 1/5th of an asset);
d) The underlying asset pays no dividends (even though its introduction to the model is not difficult);
e) There are no arbitrage opportunities (the market is complete);
f) The assets are traded continuously;
g) The risk-free rate is r, and is applied to all maturities.

One of the key concepts in pricing financial instruments is risk-neutral valuation[6]. None of the parameters of the Black-Scholes model are affected by the preference for risk. Investors in the risk-neutral world would demand a yield equal to the risk-free rate (in other words, one would discount future cash flows using the risk-free rate of r), because all the financial derivatives and products could be replicated. Thus, each derivative is priced simply by taking the discounted expected value of the risk-neutral measure of the derivative's potential payout. Moreover, the price of a financial derivative will be a martingale under the risk-neutral measure that is equivalent to the real-world measure.

The model developed in this article is free of the problem of arbitrage opportunities since it is modeled in such a way that markets are efficient and a fair price is set by the market maker in the most optimal way.

As mentioned before, the underlying price process follows the lognormal dynamics under the GBM. At time T we have:

$$ln\, S_T - ln\, S_0 \sim \phi\left[(\mu - \sigma^2/2]T, \sigma\sqrt{T}\right] \quad (6)$$

$$ln\, S_T \sim \phi\left[l\, n\, S_0 + (\mu - \sigma^2/2)T, \sigma\sqrt{T}\right] \quad (7)$$

The lognormal dynamics of the stock path under the GBM may look like this (**Figure 1**):

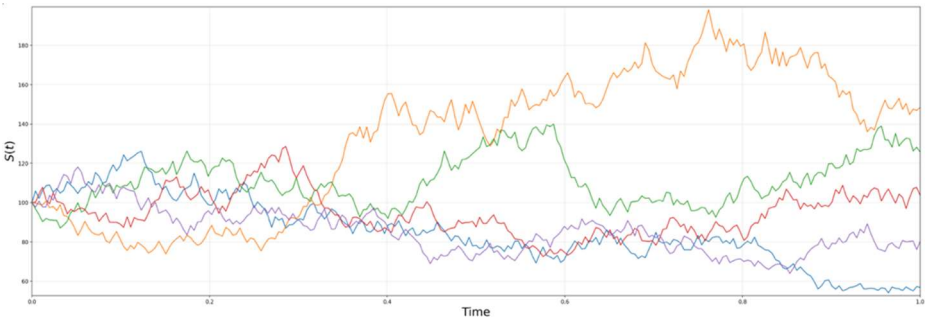

Figure 1. Five sample paths of the GBM under the lognormal dynamics.

6) The model of stochastic auctions with a level market index has a normal distribution of relative increments and follows the Geometric Brownian motion.

We can prove the normality of the transformed process $W_s + m\mu$, where W_s is the initial value of the process, m is a random variable from the Gaussian (normal) distribution, and μ is another random variable, and it is possible to leverage the properties of the sum of normal random variables.

First, we need to ensure that μ also follows the normal distribution. Let's assume that $\mu \sim N(\mu_\mu, \sigma_\mu^2)$. Now suppose that $m \sim N(\mu_m, \sigma_m^2)$. Then the product $m\mu$ will also follow the normal distribution since it is a liner combination of normally distributed random variables. From this, it follows that the sum $W_s + m\mu$ will also have a normal distribution since W_s is a constant. To find the parameters of this new distribution, we will need to find the parameters for the product first and then add the constant to the mean. This gives us the following parameters for $W_s + m\mu$:

$$\mu_{new} = W_s + \mu_m \mu_\mu$$
$$\sigma^2_{new} = \sigma^2_m \sigma^2_\mu + \sigma^2_m \mu^2_\mu + \sigma^2_\mu \mu^2_m \quad (8)$$

Thus, we can conclude that $W_s + m\mu \sim N(\mu_{new}, \sigma^2_{new})$. If μ, m are independent and have finite variance, and W_s is a stationary process (its variance and mean are constant over time), then the variance of X will be finite and not dependent on time s. If the additional condition of $Cov[W_s + m\mu, W_s] = 0$ holds, then variance of X will be simply $Var[W_s + m\mu] + Var[W_s]$.

Now if the mean and variance of X do not depend on time s and the normality conditions are satisfied, it can be concluded that the increments of the process $W_s + m\mu$ follow the normal distribution.

However, it is important to note that analytically proving normality can be complex and depends on the specific characteristics of the random variables μ, m, and the properties of W_s. Sometimes, numerical simulations and statistical tests may be required.

However, it is important to perform a statistical check for normality of increments: increments of S (the difference between values of S at neighboring time points) should follow a normal distribution.

This check is performed using the Shapiro-Wilk test. Its purpose is to test the hypothesis that the dataset was drawn from a normal distribution. This test is based on comparing observed data with theoretical expectations for a normally distributed sample. The Shapiro-Wilk test is sensitive to small deviations from normality, making it a commonly used method for assessing data normality. However, it's important to note that with large sample sizes, the test may become statistically significant even if deviations from normality are not substantial.

For the Brownian motion:

Increments of S follow a normal distribution (p-value = 0.7704195380210876).

For our model's process:

Increments of S follow a normal distribution (p-value = 0.10352202504873276).

In order to simulate the auction market, we need to account for auction sessions that happen randomly over time. The dynamics of trades (if a single asset is considered) may be simulated using the lognormal GBM. There are also trading sessions that may happen randomly (e.g., uniformly distributed in time). If only one single session is examined, different paths of market participants may be modelled using the mean-reverting Ornstein-Uhlenbeck process (the Gaussian process). The SDE is as follows:

$$dS_t = \kappa(\theta - S_t)dt + \sigma dW_t \quad (9)$$

where $\kappa > 0$ is the mean-reversion speed, θ—the long-term mean to which the process converges, $\sigma > 0$—constant volatility.

The solution to the Ornstein-Uhlenbeck SDE is given by:

$$S_T = e^{-\kappa T} S_0 + \theta(1 - e^{-\kappa T}) + \sigma \int_0^T e^{-\kappa(T-u)}\, dW_u \quad (10)$$

This process may be used to simulate paths inside one trading cycle by setting the initial point (S_0) equal to the price where the cycle begins, and the long-term mean (θ) may be set equal to the next trading point.

The illustration of the abovementioned nested simulation is represented below (**Figure 2**, with the black path representing the dynamics of trade and the red ones representing inner paths representing different market participants or order books):

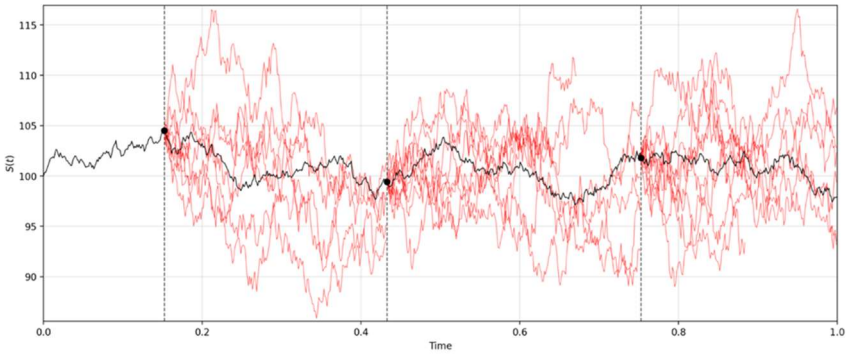

Figure 2. Illustration of auction market simulation.

Interruptions of trajectories within each trading cycle are caused by the exit of a participant from the market, for example, bankruptcy due to the chosen strategy. A market with several assets may look as follows (**Figure 3**—with colored paths as inner participants' processes):

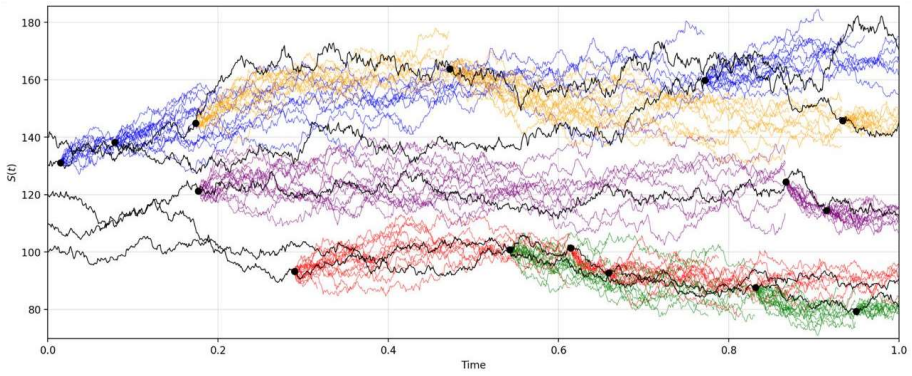

Figure 3. Illustration of several auction market simulations at a time.

3.2. Filtration problem

A linearized Kalman filter can be used to solve the problem of predicting and correcting the selling price.

Consider a complete probability space with filtering $(\Omega, \mathcal{F}, \mathrm{P}, \{\mathcal{F}_t\}_{t\in\mathbb{Z}_+})$, $\mathcal{F}_0$-adapted random vector X_0, $\mathcal{F}_t$-adapted sequences of random vectors $\{\mathrm{V}_t\}$ и $\{\mathrm{W}_t\}$ (X_0, $\{\mathrm{V}_t\}$ and $\{\mathrm{W}_t\}$ are generally independent). Also consider the stochastic dynamic system with discrete time:

$$\begin{cases} \mathrm{X}_t = \mathrm{a}_t(\mathrm{X}_{t-1}) + \mathrm{b}_t\mathrm{V}_t, \quad t \in \mathbb{N}, \quad \mathrm{X}_0 \sim \pi_0(\mathrm{x}): \mathrm{E}[\mathrm{X}_0] = \mathrm{m}_0, \mathrm{cov}(\mathrm{X}_0, \mathrm{X}_0) = \mathrm{D}_0 \\ \mathrm{Y}_t = \mathrm{A}_t(\mathrm{X}_t) + \mathrm{B}_t\mathrm{W}_t, \quad t \in \mathbb{N}. \end{cases} \tag{11}$$

The abovementioned stochastic system has:

- the first equation—the dynamics equation,
- the second equation—the observations model,
- $\mathrm{X}_t \in \mathbb{R}^{\mathrm{N}}$—unobservable state of the system; $\mathrm{a}_t(\mathrm{x}): \mathbb{R}^{\mathrm{N}} \to \mathbb{R}^{\mathrm{N}}, \mathrm{b}_t \in \mathbb{R}^{\mathrm{N}\times\mathrm{N}}$—discrete drift and diffusion in dynamics; $\mathrm{V}_t \in \mathbb{R}^{\mathrm{N}}$—sequence of random vectors—diffusions in dynamics; X_0—initial condition;
- $\mathrm{Y}_t \in \mathbb{R}^{\mathrm{M}}$—process of accessible observation; $\mathrm{A}_t(\mathrm{x}): \mathbb{R}^{\mathrm{N}} \to \mathbb{R}^{\mathrm{M}}, \mathrm{B}_t \in \mathbb{R}^{\mathrm{M}\times\mathrm{M}}$—additive useful signal and noise intensity; $\mathrm{W}_t \in \mathbb{R}^{\mathrm{M}}$—sequence of observation errors,
- All conditions of the dynamic system with discrete time are satisfied for a given model.

Let $\mathcal{Y}_\mathrm{T} = \sigma\{\mathrm{Y}_1, \dots, \mathrm{Y}_\mathrm{T}\}$—σ-algebra generated by observations on time interval [1, T].

Let us derive the equations of optimal filtration under some additional assumptions:

a) Matrices b_t and B_t are non-singular;
b) $V_t \in \mathbb{R}^N$—sequence of independently distributed random vectors with distribution density $p_V(v)$.
c) $W_t \in \mathbb{R}^M$—sequence of independently distributed random vectors with distribution density $p_W(w)$.
d) Initial condition X_0 has the distribution density $\pi_0(x)$.
e) X_0, $\{V_t\}$ and $\{W_t\}$ are independent in general.

The derivation of recurrence relations of filtration is based on the property of the conditional density of distribution:

$$\pi_{X|Y,Z}(x|y,z) = \frac{\pi_{X,Y|Z}(x,y|z)}{\int \pi_{X,Y|Z}(u,y|z)du} \tag{12}$$

The equations are derived by the method of mathematical induction. We will denote $\hat{\pi}_t(x) = \hat{\pi}_t(x|Y_1, \dots, Y_t)$ as a state distribution density X_t conditional on $\mathcal{Y}_t$ (filtration estimation density).

a) $t = 0$. In this case:

$$\begin{cases} \hat{\pi}_0(x) = \pi_0(x) \\ \hat{X}_0 = E[X_0 | \mathcal{Y}_0] = E[X_0] = \int_{\mathbb{R}^N}^{?} x\pi_0(x)dx \end{cases} \tag{13}$$

Equation (13) is a formula for calculating the initial condition.

b) Let for the time $t-1$ the density of filtration estimation $\hat{\pi}_{t-1}(x) = \hat{\pi}_{t-1}(x|\mathcal{Y}_{t-1})$ of the state X_{t-1} conditional on $\mathcal{Y}_{t-1}$ and the corresponding filtration estimate $\hat{X}_{t-1} = E[X_{t-1}|\mathcal{Y}_{t-1}] = \int_{\mathbb{R}^N}^{?} x\,\pi_{t-1}(x)dx$ to be known. Then the density $\bar{\pi}_t(x) = \bar{\pi}_t(x|\mathcal{Y}_{t-1})$ of the state X_t conditional on $\mathcal{Y}_{t-1}$ (distribution density of the one-step forecast) is given by the formula:

$$\bar{\pi}_t(x) = \int_{\mathbb{R}^N}^{?} \left|\det^{-1}\left(b_t(u)\right)\right| p_V\left(b_t^{-1}(u)(x - a_t(u))\right) \hat{\pi}_{t-1}(u)du \tag{14}$$

c) Conditional joint density $\bar{\rho}_t(x,y) = \rho_t(x,y|\mathcal{Y}_{t-1})$ of the distribution of pair (X_t, Y_t) conditional $\mathcal{Y}_{t-1}$ looks like:

$$\bar{\rho}_t(x,y) = \left|\det^{-1}\left(B_t(x)\right)\right| p_W\left(B_t^{-1}\left(y - A_t(x)\right)\right) \bar{\pi}_t(x) \tag{15}$$

Then the density of filtration estimation at step t is equal to:

$$\hat{\pi}_t(x) = \frac{\bar{\rho}_t(x, Y_t)}{\int_{\mathbb{R}^N}^{?} \bar{\rho}_t(u, Y_t)du} \tag{16}$$

And the sought estimate of optimal filtering:

$$\hat{X}_t = \frac{1}{\int_{\mathbb{R}^N}^{?} \bar{\rho}_t(v, Y_t)dv} \int_{\mathbb{R}^N}^{?} u\,\bar{\rho}_t(u, Y_t)du \tag{17}$$

The problem of optimal filtering is to construct $E[X_t|\mathcal{Y}_t]$. However, the realization of the corresponding formulas is a rather complex computational problem. The following additional information is known about the equations:

a) Functions $a_t(x)$, $A_t(x)$ are continuously differentiable over x,

b) A reference trajectory of the state $\{x_t\}$ is known, the true trajectory of the system $\{X_t\}$ is located in the neighborhood.

Let's decompose the equation in the neighborhood $\{x_t\}$ into the Taylor series:

$$\begin{cases} X_t = a_t(x_{t-1}) + \left.\frac{\partial a_t(x)}{\partial x}\right|_{x_{t-1}} (X_{t-1} - x_{t-1}) + \bar{o}(\|X_{t-1} - x_{t-1}\|) + b_t V_t, \\ Y_t = A_t(x_t) + \left.\frac{\partial A_t(x)}{\partial x}\right|_{x_t} (X_t - x_t) + \bar{o}(\|X_t - x_t\|) + B_t W_t. \end{cases} \quad (18)$$

And ignore the higher-order terms $\bar{o}(\|X_{t-1} - x_{t-1}\|)$ и $\bar{o}(\|X_t - x_t\|)$. Then the equations are replaced by their linear approximations:

$$\begin{cases} X_t = \left.\frac{\partial a_t(x)}{\partial x}\right|_{x_{t-1}} X_{t-1} + \left(a_t(x_{t-1}) - \left.\frac{\partial a_t(x)}{\partial x}\right|_{x_{t-1}} x_{t-1}\right) + b_t V_t, \\ Y_t = \left.\frac{\partial A_t(x)}{\partial x}\right|_{x_t} X_t + \left(A_t(x_t) - \left.\frac{\partial A_t(x)}{\partial x}\right|_{x_t} x_t\right) + B_t W_t \end{cases} \quad (19)$$

Linearized Kalman filter is applied to the linearized systems of observations:

a) Initial condition:

$$\begin{cases} \hat{X}_0 = m_0, \\ k_0 = D_0. \end{cases} \quad (20)$$

b) Forecast:

$$\begin{cases} \tilde{X}_t = \left.\frac{\partial a_t(x)}{\partial x}\right|_{x_{t-1}} \hat{X}_{t-1} + \left(a_t(x_{t-1}) - \left.\frac{\partial a_t(x)}{\partial x}\right|_{x_{t-1}} x_{t-1}\right), \\ \tilde{k}_t = \left.\frac{\partial a_t(x)}{\partial x}\right|_{x_{t-1}} k_{t-1} \left(\left.\frac{\partial a_t(x)}{\partial x}\right|_{x_{t-1}}\right)^T + b_t b_t^T. \end{cases} \quad (21)$$

c) Adjustment:

$$\hat{X}_t = \tilde{X}_t + \tilde{k}_t \left(\left.\frac{\partial A_t(x)}{\partial x}\right|_{x_t}\right)^T \left(\left.\frac{\partial A_t(x)}{\partial x}\right|_{x_t} \tilde{k}_t \left(\left.\frac{\partial A_t(x)}{\partial x}\right|_{x_t}\right)^T + B_t B_t^T\right)^{-1} \left(Y_t - \left.\frac{\partial A_t(x)}{\partial x}\right|_{x_t} \tilde{X}_t - \left(A_t(x_t) - \left.\frac{\partial A_t(x)}{\partial x}\right|_{x_t} x_t\right)\right), \quad (22)$$

$$k_t = \tilde{k}_t - \tilde{k}_t \left(\left.\frac{\partial A_t(x)}{\partial x}\right|_{x_t}\right)^T \left(\left.\frac{\partial A_t(x)}{\partial x}\right|_{x_t} \tilde{k}_t \left(\left.\frac{\partial A_t(x)}{\partial x}\right|_{x_t}\right)^T + B_t B_t^T\right)^{-1} \left.\frac{\partial A_t(x)}{\partial x}\right|_{x_t} \tilde{k}_t. \quad (23)$$

Selecting a "successful" reference trajectory $\{x_t\}$ is a key factor affecting the accuracy of filtering estimates.

The most important for understanding all nonlinear Kalman filters, including the linearized Kalman filter, is the fact that $\tilde{k}_t$—is not the forecast error covariance matrix, and k_t—is not the covariance matrix of the filtering estimation error. These are only some of their approximations.

Usage of H^∞ filter:

Let us consider another kind of Kalman filter for a "symmetric" observation system.

1) Consider a complete probability space with filtration $(\Omega, \mathcal{F}, P, \{\mathcal{F}_t\}_{t\in\mathbb{Z}_+})$, $\mathcal{F}_0$-adapted random vector X_0, $\mathcal{F}_t$-adapted sequences of independent Gaussian random vectors $\{V_t\}$ и $\{W_t\}$ (X_0, $\{V_t\}$ and $\{W_t\}$) which are independent. A pair of Markov recurrent stochastic sequences:

$$X_t = a_t X_{t-1} + b_t V_t, \qquad t \in \mathbb{N}, \qquad X_0 \sim \mathcal{N}(m_0, D_0) \tag{24}$$

$$Y_t = A_t X_{t-1} + B_t W_t, \qquad t \in \mathbb{N} \tag{25}$$

is called a linear Gaussian (stochastic dynamical) observation system with discrete time. The abovementioned stochastic system has:

- (24)—dynamics equation;
- (25)—model of observations;
- $X_t \in \mathbb{R}^N$ —unobservable state of the system; $a_t: \mathbb{N} \to \mathbb{R}^{N\times N}$, $b_t: \mathbb{N} \to \mathbb{R}^{N\times N}$ —sequences of deterministic matrices (discrete drift and diffusion in dynamics); $V_t \in \mathbb{R}^N$—the sequence of random vectors–diffusions in dynamics; X_0—initial condition;
- $Y_t \in \mathbb{R}^M$ —process of accessible observation; $A_t: \mathbb{N} \to \mathbb{R}^{M\times N}$, $B_t: \mathbb{N} \to \mathbb{R}^{M\times M}$ —sequences of deterministic functions (additive useful signal and noise intensity); $W_t \in \mathbb{R}^M$ —sequence of observation errors; it is assumed that $B_t B_t^T > 0$.

2) Optimal state filtering problem X_t linear Gaussian observation system with discrete time is to find $\widehat{X}_t = E[X_t|\mathcal{Y}_t]$, where $\mathcal{Y}_t = \sigma\{Y_1, \dots, Y_t\}$.

3) (Kalman filter). Optimal estimation $\widehat{X}_t$ of the state of (24) and the covariance matrix of its error $k_t = \mathrm{cov}\left(\widehat{X}_t - X_t, \widehat{X}_t - X_t\right)$ are computed using the following two-step recurrence algorithm:

a) Smoothing step:

$$\begin{gathered}\overline{X}_{t-1} = E[X_{t-1}|\mathcal{Y}_t] \\ = \widehat{X}_{t-1} + k_{t-1}A_t^T\left(A_t k_{t-1} A_t^T + B_t B_t^T\right)^{-1}\left(Y_t - A_t \widehat{X}_{t-1}\right),\end{gathered} \tag{26}$$

$$\begin{gathered}\overline{k}_{t-1} = \mathrm{cov}(\overline{X}_{t-1} - X_{t-1}, \overline{X}_{t-1} - X_{t-1}) = \\ = k_{t-1} - k_{t-1}A_t^T\left(A_t k_{t-1} A_t^T + B_t B_t^T\right)^{-1} A_t k_{t-1}\end{gathered} \tag{27}$$

b) Forecast step:

$$\widehat{X}_t = a_t \overline{X}_{t-1} \tag{28}$$

$$k_t = a_t \overline{k}_{t-1} A_t^T + b_t b_t^T \tag{29}$$

c) Initial condition:

$$\widehat{X}_0 = m_0 \tag{30}$$

$$k_0 = D_0 \tag{31}$$

Since the second filtering step is quite simple, sometimes the algorithm (26)–(29) is written in the following equivalent form:

$$\widehat{X}_t = a_t\widehat{X}_{t-1} + a_t\beta_t\left(Y_t - A_t\widehat{X}_{t-1}\right) \tag{32}$$

$$\beta_t = k_{t-1}A_t^T\left(A_t k_{t-1}A_t^T + B_t B_t^T\right)^{-1} \tag{33}$$

$$k_t = a_t\bar{k}_{t-1}a_t^T + b_t b_t^T \tag{34}$$

$$\bar{k}_{t-1} = (I - \beta_t)A_t k_t \tag{35}$$

After simple transformations using the matrix inversion lemma, the two-step procedure for calculating k_t can be represented in a one-step form:

$$k_t = a_t k_{t-1}\left(I + A_t^T\left(B_t B_t^T\right)^{-1}A_t k_t\right)^{-1}a_t^T + b_t b_t^T \tag{36}$$

And the gain coefficient is in the form:

$$\beta_t = k_{t-1}\left(I + A_t^T\left(B_t B_t^T\right)^{-1}A_t k_{t-1}\right)^{-1}A_t^T\left(B_t B_t^T\right)^{-1} \tag{37}$$

4) Conditional optimization problem under equality type constraints:

$$J(x) \to \min \;\; s.t.\, x \in \mathbb{R}^n : f(x) = 0, \qquad f: \mathbb{R}^n \to \mathbb{R}^m \tag{38}$$

Consider the Lagrangian function

$$J_L(x, \lambda) = J(x) + \lambda^T f(x) \tag{39}$$

When smoothness conditions of $J(x)$ and $f(x)$ hold so that the point x^* was the solution to the problem (41) it is necessary for there to be such vector λ^* that would be the solution to the system

$$\left.\begin{aligned} \left.\frac{\partial J_L}{\partial x}\right|_{(x^*,\lambda^*)} &= 0 \\ \left.\frac{\partial J_L}{\partial \lambda}\right|_{(x^*,\lambda^*)} &= 0 \end{aligned}\right\} \tag{40}$$

5) Conditional optimization problem under inequality-type constraints:

$$J(x) \to \min \;\; s.t.\, x \in \mathbb{R}^n : f(x) \le 0, \qquad f: \mathbb{R}^n \to \mathbb{R}^m \tag{41}$$

Consider the same Lagrangian function (39). When smoothness conditions of $J(x)$ and $f(x)$ hold so that the point x^* was the solution to the problem (41), it is necessary for there to be such vector λ^* that would be the solution to the system.

$$\left.\begin{aligned} \left.\frac{\partial J_L}{\partial x}\right|_{(x^*,\lambda^*)} &= 0, \\ \lambda_i^* f_i(x^*) = 0, \qquad i = 1, \dots, m - &\text{ complementary rigidity condition}, \\ \lambda_i^* \ge 0, \qquad i = 1, \dots, m. & \end{aligned}\right\} \tag{42}$$

The method of Lagrange multipliers for solving conditional optimization problems, represented visually in the **Figure 4** below, can be extended to the case of dynamic systems with discrete time. Of course, such a problem can be reduced to an ordinary problem of high dimensionality, but there is still a possibility of an iterative solution.

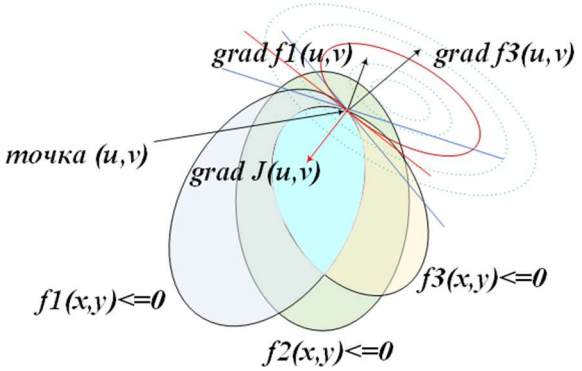

Figure 4. Conditional optimization problem under constraints of equality type with the Lagrangian function.

Consider a deterministic dynamical system:

$$X_{t+1} = a_t X_t + V_t, \qquad t = 0, \dots, T-1 \tag{43}$$

And the optimization criterion takes the form:

$$J = \psi(X_0) + \sum_{t=0}^{T-1} D_t(X_t, V_t) \tag{44}$$

The Lagrangian function in this case takes the form (only the equations of state act as constraints) (43)):

$$\begin{gathered} J_L = \psi(X_0) + \sum_{t=0}^{T-1} \left[D_t(X_t, V_t) + \lambda_{t+1}^T (a_t X_t + V_t - X_{t+1}) \right] = \\ \psi(X_0) + \sum_{t=0}^{T-1} \left[D_t(X_t, V_t) + \lambda_{t+1}^T (a_t X_t + V_t) \right] - \sum_{t=0}^{T} \lambda_t^T X_t + \lambda_0^T X_0 \end{gathered} \tag{45}$$

Defining the Hamiltonian:

$$\mathcal{H}_t(X_t, V_t, \lambda_{t+1}) = D_t(X_t, V_t) + \lambda_{t+1}^T (a_t X_t + V_t), \quad t = 0, \dots, T-1 \tag{46}$$

The Lagrangian function can be expressed as:

$$J_L = \psi(X_0) + \sum_{t=0}^{T-1} \left[\mathcal{H}_t - \lambda_t^T X_t \right] - \lambda_T^T X_T + \lambda_0^T X_0 \tag{47}$$

The necessary conditions of optimality in this case have the form:

$$\frac{\partial J_L}{\partial X_0} = 0, \qquad \frac{\partial J_L}{\partial X_T} = 0, \frac{\partial J_L}{\partial X_t} = 0, \qquad t = 1, \dots, T-1, \tag{48}$$

$$\frac{\partial J_L}{\partial V_t} = 0, \qquad t = 0, \dots, T-1, \frac{\partial J_L}{\partial \lambda_t} = 0, \qquad t = 0, \dots, T. \tag{49}$$

Accounting for (47), the optimality conditions take the form:

$$\lambda_0^T + \frac{\partial \psi_0}{\partial X_0} = 0 \tag{50}$$

$$-\lambda_T^T = 0 \tag{51}$$

$$\lambda_t^T = \frac{\partial \mathcal{H}_t}{\partial X_t}, \qquad t = 1, \dots, T-1 \tag{52}$$

$$\frac{\partial \mathcal{H}_t}{\partial V_t} = 0, \qquad t = 0, \dots, T-1 \tag{53}$$

$$X_{t+1} = a_t X_t + V_t, \qquad t = 0, \dots, T-1 \tag{54}$$

Formulation of the problem of H^∞ filtration:

$$J_H\left(\{\hat{Z}_t\}, X_0, \{V_t\}, \{W_t\}\right) \to \inf_{\{\hat{Z}_t\}} \sup_{X_0, \{V_t\}, \{W_t\}} \tag{55}$$

If we consider a linear transformation $(X_0, \{V_t\}, \{W_t\}) \xrightarrow{\hat{Z}} \{\Delta_t^Z\}$, then (55) may be expressed as

$$\sup_{X_0, \{V_t\}, \{W_t\}} J_H\left(\{\hat{Z}_t\}, X_0, \{V_t\}, \{W_t\}\right) = \left\|\hat{Z}\right\|^2 \to \inf \tag{56}$$

6) (H^∞ filter). The following expressions are given:

a) system of observations (45), (46) and the estimated output (47),

b) price function—(50), where matrices P_0, Q_t, R_t, S_t are symmetric positively defined (chosen according to the practical problem to be solved (it is important that they are not directly related to the statistical characteristics of noise).

If at each time step the following inequality is satisfied:

$$P_t^{-1} - \theta \bar{S}_t + A_t^T R_t^{-1} A_t > 0 \tag{57}$$

Then the price function can be made smaller $\frac{1}{\theta}$ using the following recurrent estimation algorithm:

$$\bar{S}_t = L_t^T S_t L_t \tag{58}$$

$$\beta_t = P_t\left(I - \theta \bar{S}_t P_t + A_t^T R_t^{-1} A_t P_t\right)^{-1} A_t^T R_t^{-1} \tag{59}$$

$$\hat{X}_{t+1} = a_t \hat{X}_t + a_t \beta_t \left(Y_t - A_t \hat{X}_t\right) \tag{60}$$

$$P_{t+1} = a_t P_t\left(I - \theta \bar{S}_t P_t + A_t^T R_t^{-1} A_t P_t\right)^{-1} a_t^T + Q_t \tag{61}$$

Suitable tasks for the H^∞ filter:

a) Systems for which robustness must be guaranteed or the quality of the worst-case evaluation is prioritized.
b) Models with unpredictable changes in which identification or optimization of gain is difficult or resource intensive.
c) Systems whose models are not fully known.

Below one can see the Kalman filter applied to the auction market path (**Figure 5**).

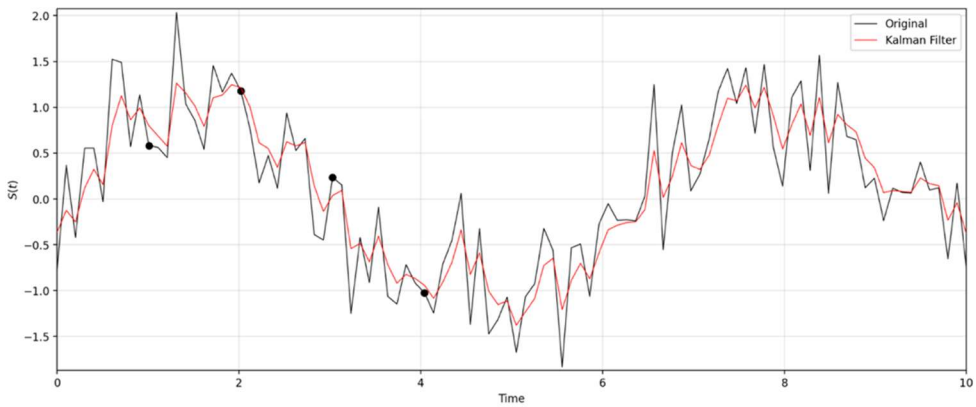

Figure 5. The Kalman filter applied to the auction market path.

4. Practical examination

4.1. Program implementation

Users make bids in parallel and independently from each other. Only after all the users have declared the end of the move, the cycle is terminated. One can see the commands for the server in the **Table 3**.

Table 3. User teams with market level modeling.

Name	Param. 1	Param. 2	Example	Comment
market	–	–	market	Displays market level parameters
player	id	–	player 1	Displays user-specific settings
online	–	–	online	Displays the number of non-bankrupt users
me	–	–	me	Displays the current user's settings
exit	–	–	exit	Server disconnection
prod	count	–	prod 2	Produce 2 units of good from 2 units of raw materials
buy	count	price	buy 2 2000	Buy 2 units of good for 2000 units of currency
sell	count	price	sell 2 2000	Sell 2 units of good for 2000 units of currency
build	count	–	build 2	Build 2 production facilities
turn	–	–	turn	Finalization of offers
help	–	–	help	Displays program commands

At the end of the cycle, the market reports to users' full information about the trading results, namely, the total closing price, the number of bought and sold units of trade relations for all users, and the number of currency units for all users. The program terminates when, for one reason or another, only one user remains in the program. A detailed model can be found in the research paper[4] by the author.

The statistical bot model is to use trivial estimation or estimation by the mathematical expectation of winning bets.

1) The current bid for the buy has the form:

$$\mathrm{E}|\sum_{k=0}^{N} buy_k| \tag{62}$$

where buy_k—bids to buy from all players on the previous turn, N $\in \mathbb{N}$—number of auction participants.

2) The current bid for the sale is of the form:

$$\mathrm{E}|\sum_{k=0}^{N} sell_k| \tag{63}$$

where buy_k—bids to sell from all players on the previous turn, N$\in\mathbb{N}$—number of auction participants.

The starting parameters of the interpreter program are the IP address and TSP port number of the server, as well as the name of the file containing the program in the model language. The script determines the further behavior of the bot. Thus, the bot program is a combination of a client program and a model language interpreter program.

Similarly, with the server, the start parameters are set on the command line. The input language of the bot should allow the use of all the information that is available to a normal user, as well as issuing all the commands

that a normal user would issue. The language should be algorithmically complete and have the ability to create non-trivial strategies.

All whitespace characters must be equal, and any number of whitespace characters must also be allowed anywhere in the program.

The language contains the following operators: assignment, unconditional transition, conditional operator, auction operator, and debug print operator. The language supports arithmetic and comparison operations. The language must define the priority of operations.

Operands can be constants, variables, arrays, and calls to built-in functions. Expressions may contain parentheses of any nesting. A lexical and syntactic analyzer must be used for implementation.

The result of the analyzer's work should be an internal representation of the scenario convenient for further interpretation—RPN (**Figure 6**).

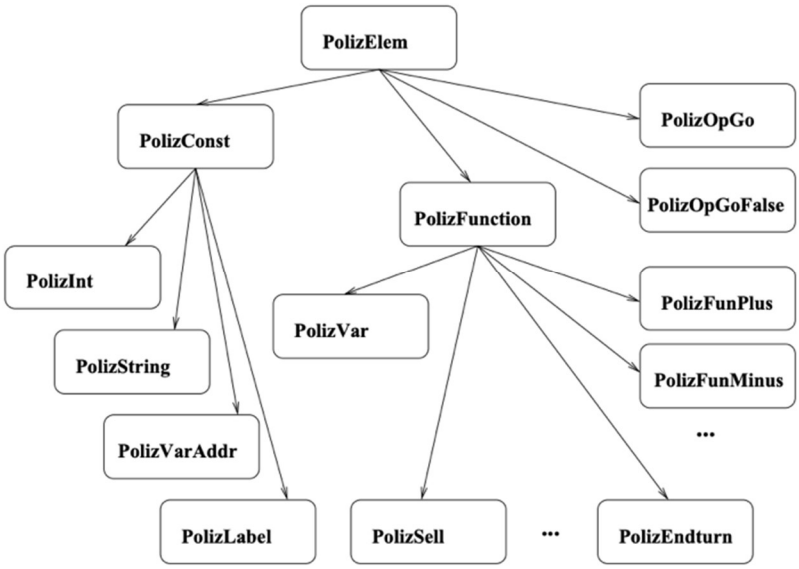

Figure 6. Hierarchy of classes for representing RPN.

Syntax of the proprietary language of trading strategies:
?—internal function indicator;
$—variable indicator;
;—end of operation indicator;
{}—initialization similar to C/C++-type languages;
{}end—indicator for the beginning and end of the entire program.

The language is algorithmic complete. Below, one can see an example of a simple trading strategy implementation program (**Figure 7**):

```
Start
|-- {
|-- $winsrawcount = 0
|-- $users = ?active_players()
|-- $count_fac = ?factories(?my_id())
|
|-- while ?money(?my_id()) < $demand * 2000
|    |
|    |-- $count_fac = $count_fac - 1
|
|-- ?prod($count_fac)
|-- $k = 1
|
|-- while $k < $users
|    |
|    |-- if ?result_prod_bought($k) >= 1
|    |    |
|    |    |-- $winsprodcount[$j] = ?result_prod_bought($k)
|    |    |-- $winsprodprice[$j2] = ?result_prod_price($k)
|    |    |-- $j = $j + 1
|    |    |-- $j2 = $j2 + 1
|    |
|    |-- if ?result_prod_bought($k) < 1
|    |    |
|    |    |-- $failsprodcount[$j3] = ?result_prod_bought($k)
|    |    |-- $failsprodprice[$j4] = ?result_prod_price($k)
|    |    |-- $j3 = $j3 + 1
|    |    |-- $j4 = $j4 + 1
|    |
|    |-- $k = $k + 1
|--}
|-- end
```

Figure 7. A simple trading strategy implementation program.

4.2. Model ensembles and results

The input of each neural network is raw data in any format—categorical attributes are translated using label encoding, so any initial date format with any set of attributes can be used for the neural network approach. Then each model uses two hidden layers, and the output layer represents the transaction date, optimal price, and signal (buy or sell) (**Figure 8**).

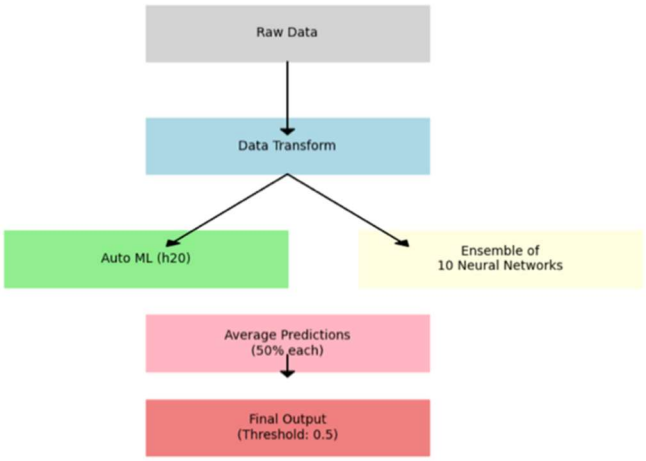

Figure 8. Steps in the implementation of a neural network ensemble with the auto ML method.

4.3. Model validation

The following validation method is called Combinatorial Splits[7]. Consider T observations partitioned into N groups with shuffling, where groups $n = 1, \ldots, N-1$ are of size T/N, the N-th group is of size $T - [T/N](N-1)$, and $[\cdot]$ is the floor or integer function. For a testing set of size k groups, the number of possible training/testing splits is C_k^{N-k}.

Since each combination involves k tested groups, the total number of tested groups is kC_N^{N-k}. And since we have computed all possible combinations, these tested groups are uniformly distributed across all N (each group belongs to the same number of training and testing sets). The implication is that from k-sized testing sets on N groups we can back backtest a total number of paths $\phi[N,k] = \frac{k}{N}C_N^{N-k}$. For numerical results, see **Table 4**.

Table 4. Different models' performance metric.

	RMSE	**MSE**	**MAE**	**MAPE**
Trivial expectation	1.2	1.44	1.1	10.2
XGBOOST	0.8	0.64	0.7	6.3
CATBOOST	0.75	0.56	0.65	5.8
LSTM	0.9	0.81	0.8	7.4
AUTO ML H_2O	0.7	0.49	0.6	5.0
Kalman filter	0.85	0.72	0.75	6.9
H_2O and NN ensemble	0.6	0.36	0.5	4.5

4.4. Backtesting problem

In order to backtest the model, one needs to perform the following steps[8]:

1) Form a matrix M by collecting the performance series from the N trials.

Each column $n = 1, \dots, N$ represents a vector of PnL (mark-to-market profits and losses) over $t = 1, \dots, T$ observations associated with a particular model configuration.

2) Partition M across rows, into an even number S of disjoint submatrices of equal dimensions. Each of these submatrices M_s, with $s = 1, \dots, S$ is of order $\left(\frac{T}{S}N\right)$.

3) Form all combinations C_s of M_s, taken in groups of size $\frac{S}{2}$, which gives a total number of $C_S^{S/2}$ combinations.

4) For each combination $c \in C_s$ we:

a) Form the training set J by joining the $S/2$ submatrices M_s that constitute c. J is a matrix of order $\frac{T}{S}\frac{S}{2} \times N = \frac{T}{2} \times N$;

b) Form the testing set $\bar{J}$ as the complement of J in M. In other words, $\bar{J}$ is the $\frac{T}{2} \times N$ matrix formed by all rows of M that are not part of J;

c) Form a vector R of performance statistics of order N where the n-th item of R reports the performance associated with the n-th column of J (the training set);

d) Determine the element $n*$ such that $R_n \leq R_{n*}, \ \forall n = 1, \dots, N$. In other words, $n* = arg\ \max_n R_n$;

e) Form a vector $\bar{R}$ of performance statistics of order N, where the n-the item of $\bar{R}$ reports the performance associated with the n-th column of J (the testing set);

f) Determine the relative rank of $\overline{R_{n*}}$ within $\bar{R}$. We denote this relative rank as $\overline{\omega_c}$, where $\overline{\omega_c} \in (0,1)$. This is the relative rank of the out-of-sample (OOS) performance associated with the trail chosen in-sample (IS). If the strategy optimization procedure does not overfit, we should observe that $\overline{R_{n*}}$ systematically outperforms $\bar{R}$ (OOS), just as R_{n*} outperformed R (IS);

g) Define the logit $\lambda_c = log\ \frac{\overline{\omega_c}}{1-\overline{\omega_c}}$. This presents the property that $\lambda_c = 0$ when $\overline{R_{n*}}$ coincides with the median of $\overline{R}$. High logit values imply a consistency between IS and OOS performance, which indicates a low level of backtest overfitting.

5) Compute the distribution of ranks OOF by collecting all the λ_c, for $c \in CS$. The probability distribution function $f(\lambda)$ is then estimated as the relative frequency at which λ occurred across all C_s with $\int_{-\infty}^{\infty} f(\lambda)d\lambda$. Finally, the PBO is estimated as $PBO = \int_{-\infty}^{0} f(\lambda)d\,\lambda$ as that is the probability associated with IS optimal strategies that underperform OOS. See performance degradation below (**Figure 9**) and stochastic dominance (**Figure 10**).

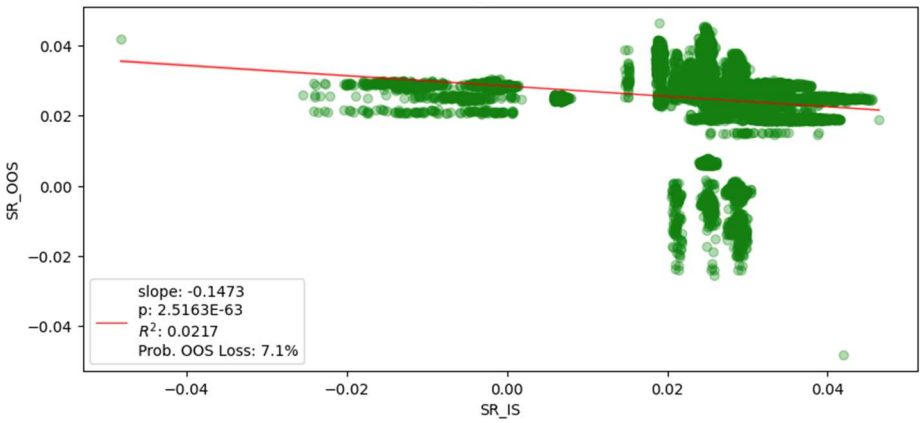

Figure 9. Performance degradation, IS vs OOS.

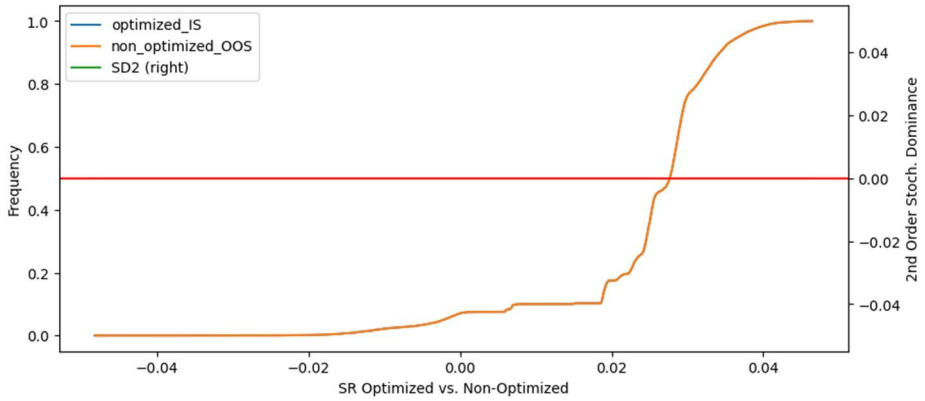

Figure 10. Stochastic dominance.

Below are the backtesting results. Simple visualization is represented below (**Figure 11**):

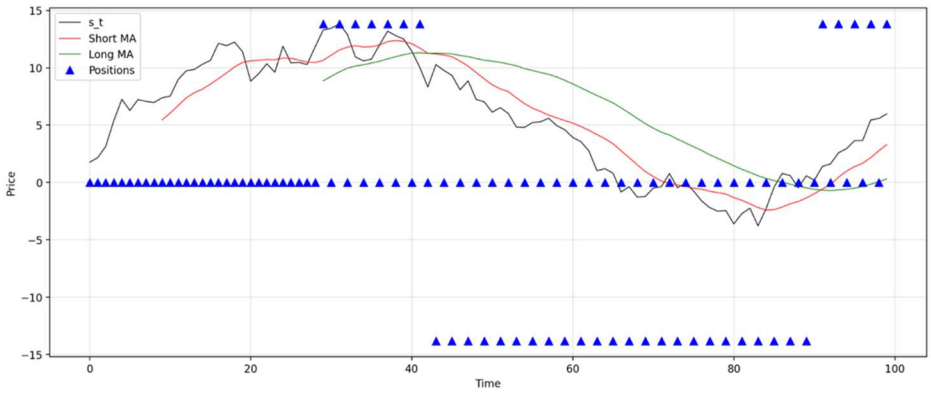

Figure 11. Backtesting on one path and one strategy.

The objects above are:

a) short_ma (short-term moving average) represents the short-term moving average of the asset prices. A short-term moving average is calculated as the average of asset prices over a certain short time interval. In the context of the strategy, the short-term moving average is used to smooth asset prices and identify short-term changes in trend.

b) long_ma (long-term moving average) represents the long-term moving average of the asset prices. A long-term moving average is calculated similarly to the short-term moving average, but with a wider time interval. In the context of the strategy, the long-term moving average is used to smooth asset prices and identify long-term trends.

c) positions indicate the current position in the strategy. The values of this variable can be as follows:

- 1: long position (purchase) signals that the short-term moving average has crossed the long-term moving average from top to bottom, which can be considered a buy signal;
- –1: short position (selling) signals that the short-term moving average has crossed the long-term moving average from below upwards, which can be considered as a signal to sell the asset;
- 0: out of position means that there is no clear signal to buy or sell, and the participant is out of the market.

In order to calculate a value (metric) to assess the performance of the strategy, one can use Sharpe ratio, which measures the ratio of the return and its risk. Sharpe ratio is one of the most popular metrics for backtesting trading strategies. This ratio is calculated as follows (64):

$$SharpeRatio = \frac{R_p - R_f}{\sigma_p} \tag{64}$$

where R_p—average annualized return of the strategy, R_f—risk-free rate (e.g., government bonds, or bank account), σ_p—standard deviation of the strategy (risk).

High value of Sharpe ratio indicates a good relative return of the trading strategy compared to the risk taken.

In addition to synthetic data, there were used data of AAPL, USD/RUB, S&P500, GOLD, and Crude Oil quotations (**Table 5**) for the time range from 31/12/21 to 10/04/22, sourced from finam.ru:

Table 5. Backtest on historical data.

Strategy	Average return	Risk (Std. deviation)	Sharpe ratio
Trivial expectation	0.05	0.1	0.2
XGBOOST	0.07	0.12	0.33
CATBOOST	0.08	0.11	0.45
LSTM	0.09	0.13	0.46
AUTO ML H_2O	0.06	0.09	0.33
Kalman filter	0.07	0.1	0.4
H_2O and NN ensemble	0.08	0.11	0.45

The similar, but not the same approach of using machine learning methods and their implementation on big data was represented in one of the authors' papers[9]. However, the main focus of that paper was on working with the data and preparation for analysis.

The backtest results demonstrate the robustness of our algorithms not only on synthetic data, but also on real futures data adjusted for the auction markets model. To improve the backtest results, one may pay attention to the following:

- Adjustment for real market conditions: backtesting on real futures data allows you to take into account actual market conditions such as liquidity, spreads, commissions, and other costs. This provides a more realistic view of strategy performance.
- Analysis on different time horizons: backtest results can be presented for different time horizons, such as daily, weekly, and monthly data. This will help understand how the strategy behaves over different time horizons.
- Volatility analysis: one can look at how a strategy responds to changes in market volatility. This is important for determining the level of risk and capital management.
- Different instruments and assets: a backtest can be conducted not only on stock futures, but also on other assets such as currencies, bonds, or commodities, which will help determine which markets the strategy works best in.
- Auction market analysis: Modeling auction markets can include consideration of market opening and closing times, trading volumes, and value dynamics.
- Comparison to broad market indices: comparing a strategy's performance to underlying indices, such as the S&P 500 or other indices, can provide insight into whether or not the strategy is outperforming the market.
- Analysis of resilience to different market regimes: it is important to evaluate how a strategy behaves in different market regimes, such as bullish, bearish, or sideways. This allows you to understand how resilient the strategy is to a variety of market conditions.
- Risk assessment and performance metrics: in addition to returns, one may also consider risk metrics such as drawdown, maximum deviation, and performance metrics such as Sharpe ratio, Treynor ratio, and others.
- Automated monitoring: for long-term strategies, it is important to set up automated monitoring of the strategy on real data to quickly identify and react to changes in market conditions.

It is quite important to remember that backtesting only serves as a tool to evaluate the potential performance of a strategy on historical data. Actual results may vary significantly depending on current market conditions and other factors.

5. Conclusion

In conclusion, it is desirable to describe the main results of the work: a proprietary model of stochastic auctions using the level market index has been developed. A risk-neutral measure has been introduced, making the market arbitrage-free. A minor theorem has been proven, stating that in such a model, the auction closing price will follow a Brownian motion. The filtration problem has been described and implemented. A proprietary interpretable programming language has been developed in C++ for testing trading algorithms on this model. The algorithms have been validated using a powerful machine learning algorithm. Quality metrics for the predictive models have been obtained. Robustness metrics (resistance to noise in real data) of the models have been obtained through backtesting on real, modified data to fit the described model. A market-making task has been formulated for the model, which can employ reinforcement learning methods, a topic that has been relevant in recent years.

Author contributions

Conceptualization, NM and KD; methodology, NM and KD; software, NM; validation, MN, KD and DA; formal analysis, NM; investigation, SA; resources, SA; data curation, SA and DA; writing—original draft preparation, NM and KD; writing—review and editing, SA; visualization, KD; supervision, SA; project administration, SA and DA; funding acquisition, SA. All authors have read and agreed to the published version of the manuscript.

Conflict of interest

The authors declare that there is no conflict of interest.

References

1. Law B, Viens F. Market making under a weakly consistent limit order book model. High Frequency. 2019; 2(3-4): 215-238. doi: 10.1002/hf2.10050
2. Lux T. Stochastic Behavioral Asset Pricing Models and the Stylized Facts. Kiel Institute for the World Economy; 2008.
3. Kraft E, Russo M, Keles D, Bertsch V. Stochastic Optimization of Trading Strategies in Sequential Electricity Markets. Karlsruhe Institute of Technology; 2021.
4. Bubeck S, Devanur N, Huang Z, Niazadeh R. Multi-scale online learning: Theory and applications to online auctions and pricing. Journal of Machine Learning Research. 2019; 20: 1-37.
5. Hull JC. Options, Futures, and Other Derivatives, 5th ed. Pearson College Div; 2002. pp. 234-248.
6. Björk T. Arbitrage Theory in Continuous Time, 3rd ed. Oxford University Press; 2009. p. 103.
7. Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning Data Mining, Inference, and Prediction, 2nd ed. Springer; 2016. p. 241.
8. de Prado ML. Advances in Financial Machine Learning. Wiley; 2018. pp. 73-75.
9. Nikonov MV, Shmitov MO. Modern methods of distributed intellectual data analysis self-developed own stochastic financial market model. In: Proceedings of the III International Research Contest; 2022. pp. 80-97.