

Article

Evolutionary Optimization of Control Strategies for Non-Stationary Immersion Environments

Alexander Musaev ^{1,2}, Andrey Makshanov ³ and Dmitry Grigoriev ^{4,*}

¹ St. Petersburg State Technological Institute (Technical University), 190013 St. Petersburg, Russia; amusaev@technolog.edu.ru

² St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, 199178 St. Petersburg, Russia

³ Department of Computing Systems and Computer Science, Admiral Makarov State University of Maritime and Inland Shipping, 198035 St. Petersburg, Russia; makshanov@oogis.ru

⁴ Center of Econometrics and Business Analytics (CEBA), St. Petersburg State University, 199034 St. Petersburg, Russia

* Correspondence: d.a.grigoriev@spbu.ru

Abstract: We consider the problem of evolutionary self-organization of control strategies using the example of speculative trading in a non-stationary immersion market environment. The main issue that obstructs obtaining real profit is the extremely high instability of the system component of observation series which implement stochastic chaos. In these conditions, traditional techniques for increasing the stability of control strategies are ineffective. In particular, the use of adaptive computational schemes is difficult due to the high volatility and non-stationarity of observation series. That leads to significant statistical errors of both kinds in the generated control decisions. An alternative approach based on the use of dynamic robustification technologies significantly reduces the effectiveness of the decisions. In the current work, we propose a method based on evolutionary modeling, which supplies structural and parametric self-organization of the control model.

Keywords: chaotic processes; control strategies; non-stationary environment; channel strategies; observation series; numerical studies; dynamic stability

MSC: 37M20; 37M10; 90C90



Citation: Musaev, A.; Makshanov, A.; Grigoriev, D. Evolutionary Optimization of Control Strategies for Non-Stationary Immersion Environments. *Mathematics* **2022**, *10*, 1797. <https://doi.org/10.3390/math10111797>

Academic Editors: Mihaela Neamțu, Eva Kaslik and Anca Rădulescu

Received: 31 March 2022

Accepted: 17 May 2022

Published: 24 May 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The problem of effective control in non-stationary immersion environments appears in a large number of applications. They are meteorology, the control of turbulent hydrodynamic flows, the stabilization of the state of non-stationary technological processes, asset management in capital markets, etc. The intricacy here is due to the unstable nature of the observed processes described by models of nonlinear chaotic dynamics [1–10]. Non-stationarity means that different observations have means, variances, and covariances that change over time. The non-stationary behavior can be a trend, a cycle, a random walk, or a combination of these. Non-stationary data is generally unpredictable and cannot be modeled or forecasted. Results obtained using non-stationary time series can be misleading because they may indicate a relationship between two variables when in fact there is none. For a better introduction to the subject, see the following Investopedia article [10].

High volatility and low predictability of chaotic processes significantly complicate the effective use of well-known control stabilization techniques based on traditional methods of adaptation and robustification. In particular, adaptive technologies turn out to be too inertial to close the feedback loop in time. Increasing the sensitivity of feedback leads to a high level of statistical errors of type II (“false alarms”). Robust control methods [11–15] focused on solutions with reduced sensitivity to statistical and dynamic variations of

the observed process in the conditions of a basic model significantly lose effectiveness compared to optimal a posteriori versions.

A control strategy based on sequential dynamic optimization of the control model in chaotic dynamics, as will be shown in this paper, also faces a number of well-known problems. Here we have, for example, bruteforcing control parameter values with a given step that determines the accuracy of the solution, exponential increasing amounts of calculations with an increase in the number of model parameters. Taking into account the high variability and unsteadiness of the observed process, the number of parameters of the optimized control model should not exceed five or six even for modern techniques. Hence, there is a need to study new techniques of sequential dynamic optimization of the control model based on suboptimal computational schemes.

In this paper, we use an observation series of currency pairs in the Forex market as a testing ground for studying the effectiveness and stability of control algorithms in chaotic immersion environment. Dynamic chaos in an observation series violates the fundamental premise of the repeatability of experiments under identical conditions. In particular, two geometrically similar observation segments in conditions of chaos may have completely different aftereffects [16,17]. As a result, traditional statistical data analysis technologies and control algorithms based on them are ineffective. In our studies, the choice of channel control strategies is determined primarily by their accessibility and interpretability.

This article considers the possibility of applying sequential optimization based on the method of evolutionary modeling in conditions of stochastic market chaos [18–21] etc. The method is based on the Darwinian concept of evolutionary self-organization and the theory of random search.

2. Methods

2.1. Observation Model and Problem Statement

A significant difference between the presented work and traditional studies in the field of asset management is their focus on the Wald's additive observation model [22–24]

$$y_k = x_k + v_k, k = 1, \dots, n \quad (1)$$

where $x_k, k = 1, \dots, n$ is the system component used in the process of making management decisions (i.e., open, close or retain current position), and $v_k, k = 1, \dots, n$ is the noise.

Currently, the prevailing point of view is that in market situations, the system component x_k in (1) is modeled as an output signal of a nonlinear system observed in the conditions of non-stationary and non-Gaussian interference v_k (dynamic chaos model) [5–13]. Lyapunov functions [25] and identification methods based on higher-order spectra [26] are used in order to substantiate such problem statements. There is a large area of research on the direct reconstruction of stochastic differential equations [27,28] for the model (1). Other points of view are based on nonlinear transformations of the y_k process, and, for example, on investigating fractal properties of the process trajectory [29–31].

Sequential filtering of the initial observation series $y_k, k = 1, \dots, n$ is usually used to isolate the system component $x_k, k = 1, \dots, n$ from (1) in real time. For this purpose, we utilize an exponential filter [32]:

$$x_k = \alpha y_k + (1 - \alpha)y_{(k-1)} = x_{(k-1)} + \alpha(y_k - x_{(k-1)}), k = 2, \dots, n \quad (2)$$

with a discounting coefficient $\alpha \in [0.01, 0.3]$. This range is our empirical finding for the Forex market. Simultaneously, it is possible to select α adaptively so it better corresponds to features of considered data. This filtering technique is not the best one, because an increase in the smoothing effect with a decrease of α leads to a significant bias of the generated estimates. However, it produces a satisfactory result for the considered examples, while also providing simplicity of interpretation of the extracted system component.

Note that the conventional statistical observation model is based on the assumption that its system component is an unknown deterministic process, and its noise component is

a stationary random process with independent increments [33]. Such a model was used in a wide class of trend analysis-based management strategies [34–37], though all of them are not resistant to possible dynamic variations in the process of changing quotes. One of the reasons for the unfeasible effectiveness of these trend indicators is the intrinsic inadequacy of the statistical approach for a chaotic observation series.

The main difference of the proposed model (1) is that its system component x_k , $k = 1, \dots, n$ is modeled as an oscillatory non-periodic process with a large number of local trends. This description indicates the possibility of interpreting this process as an implementation of the dynamic chaos model [5–13]. Its second distinctive feature is the noise component v_k , $k = 1, \dots, n$ being interpreted as a non-stationary random process described by an approximate Gaussian model with fluctuating parameters. In particular, correlations and spectral characteristics of this process change significantly over time [23,24].

The indicated features of observation series described by (1) violate the conditions of applicability of traditional statistical methods. In this case, numerical studies are the only approach to analyze the effectiveness of the developed management strategies. The statement of the asset management problem essentially coincides with the traditional formalization of the task to maximize the gain in the process of trading or investing capital.

Let $y_k = x_k + v_k$, $k = 1, \dots, n$ be a sequence of observations corresponding to a given time interval of asset management $T = n\Delta t$, where Δt is the selected interval between time counts. During the specified time, M operations are carried out in the trading process, each being determined by their start and finish $(k_{\text{open}}, k_{\text{close}})_j$, $j = 1, \dots, M$.

The trading problem can be thus formulated as follows: select a management strategy S and construct a sequence of actions u_j , $j = 1, \dots, M$ to obtain maximum profit $G(S)$:

$$G(S) = \sum_{j=1}^M u_j(k_{\text{close}}) - u_j(k_{\text{open}}) = \max \quad (3)$$

In the simplest case, each management strategy is defined by the rules that determine the time of opening and closing a position $(k_{\text{open}}, k_{\text{close}})_j$, $j = 1, \dots, M$, and, in some cases, the lot size. The sum of the operation results at the k -th step $G_k(S)$ becoming smaller than the trader's available deposit G_0 means the management process resulted in complete loss.

The approach to determining the start time of an operation (the so-called “position opening”) is the defining characteristic of a management strategy. The operation can be finished (a “position” can be “closed”) when a specified profit level (TP, “take profit”) or loss level (SL, “stop loss”) is reached, or according to some different rules that could be more flexible.

This paper examines the technique of sequential evolutionary self-organization of the management strategy in the conditions of market chaos. Due to this, the simplest control schemes are used as basic control strategies, which makes it easy to visualize and interpret the obtained results. In particular, we used the so-called channel strategies [4,34–37]. Let us consider the simplest approaches to constructing such a strategy.

2.2. Channel Asset Management Strategies

We have a series of the trading asset's quote observations being modeled by (1). Let us define a “channel” as a range of observation values constrained by $y_k = x_k \pm B$, $k = 1, \dots, n$ [33]. Variations inside the channel $|y_k - x_k| = |\delta y_k| \leq B$, $k = 1, \dots, n$, are fluctuations that do not contain an obvious trend, in which case, the process can be referred to as a sideways trend or a flat. Channel width B can be selected depending on various considerations. It usually lies in the range from s_y to $3s_y$, where s_y is the estimate of the standard deviation (SD) δy_k , $k = 1, \dots, n$. In general, the choice of channel width depends on the nature of the data and the specificity of the selected management strategy. In some cases, the channel width may be some variable value $B_k = B_k(y_k)$, $k = 1, \dots, n$.

The observation series value y_k , $k = 1, \dots, n$ that breaks out of the channel is interpreted as the emergence of a trend in some management strategies. In the case of managing

assets according to the trend direction, such events give rise to a recommendation to open a position in accordance with the sign of the channel boundary. Due to strong variability, a trend is often considered to be present when the system component x_k , $k = 1, \dots, n$ formed by the exponential filter (2) with a given level of smoothing quits the channel. The values of the model parameters α , B , TP , SL are optional. Their selection depends on the knowledge and intuition of the trader, and they fully determine the management effectiveness. But it is often the case that intuition and other abilities of a human person appear to be ineffective in trading. Therefore, there is a need for strictly formalized and mathematically sound solutions.

We named the strategy of moving corresponding to the trend CSF (channel strategy forward). The idea behind it is simple: open a position up or down when the process breaks through the upper or lower bound of the channel respectively. The management algorithm contains two rules: Open Up position at $y_k > x_k + B$ or Open Dn at $y_k < x_k - B$. Otherwise, a position will be opened at each step outside the channel. In this regard, the more often used rules are based on determining the time of crossing the channel boundary ($y_{k-1} \leq x_{k-1} + B$ & ($y_k > x_k + B$) or ($y_{k-1} \geq x_{k-1} - B$) & ($y_k < x_k - B$), $k = 1, \dots, n$.

In the simplest case, a position is closed either when the $y_{close} > y_{open} + TP$ or $y_{close} < y_{open} - SL$ levels are reached (at Open Up) or when $y_{close} < y_{open} - TP$ or $y_{close} > y_{open} + SL$ (at Open Dn).

A more flexible control scheme allows for the obtaining of a model in which the upper and lower bounds of the channel are evaluated separately. An example of the implementation of the CSF management strategy for the Euro/Japanese Yen (EURJPY) currency pair on an observation interval of 1440 min counts (one day) is shown in Figure 1. In this case, the control model was defined by the parameters $\alpha = 0.04$, $B_1 = 15$, $B_2 = 15$, $TP = 15$, $SL = 15$. Blue diamonds indicate opening a position up, red diamonds denote opening a position down, and circles are positions being closed.

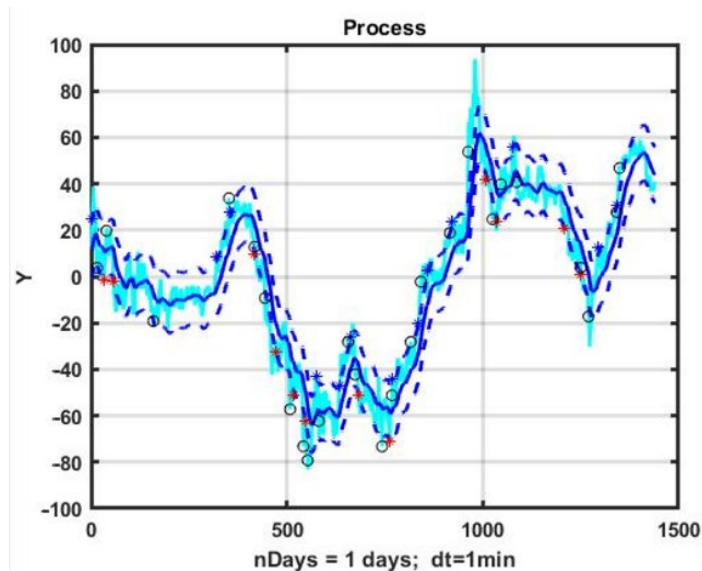


Figure 1. Example implementation of CSF on an observation interval of 1440 min counts.

Figure 2 presents management (3) effectiveness fluctuations in utilizing the channel strategy for the selected example. For the parameters that were used for the control model, the result turned out to be close to zero, and most of time during which trading was being carried out, it was negative. Note that in most cases of direct use of the CSF strategy, the result was negative.

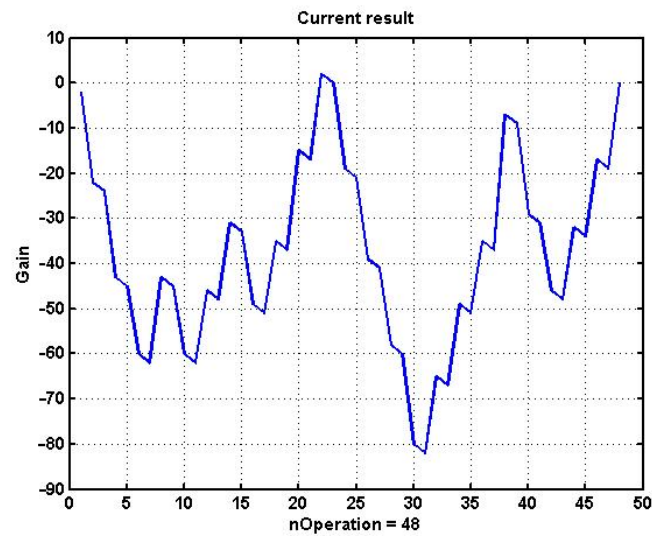


Figure 2. Performance of the CSF strategy for example in Figure 1.

Is it possible to achieve profit if there is a reliable forecast of the development of the process? To answer this question, we use brute-force posterior optimization of the parameters of the management model $M = \{\alpha, B1, B2, TP, SL\}$.

For each of the parameters, 15 iterations were carried out, starting from the values of $M_0 = \{0.01, 10, 10, 10, 10\}$. The values of the iteration step, respectively, were $Step = \{0.01, 1, 1, 1, 1\}$.

An example of the implementation of the CSF strategy with optimized parameter values $\alpha^* = 0.05$, $B1^* = 10$, $B2^* = 12$, $TP^* = 18$, $SL^* = 18$ at the same observation interval is presented in Figure 3. Note that this is the same EURJPY price history as in Figure 1, but the opening and closing moments are different. Figure 4 presents the change in performance during the use of CSF with optimized parameters. The management result for the day was $G = 248$ p.

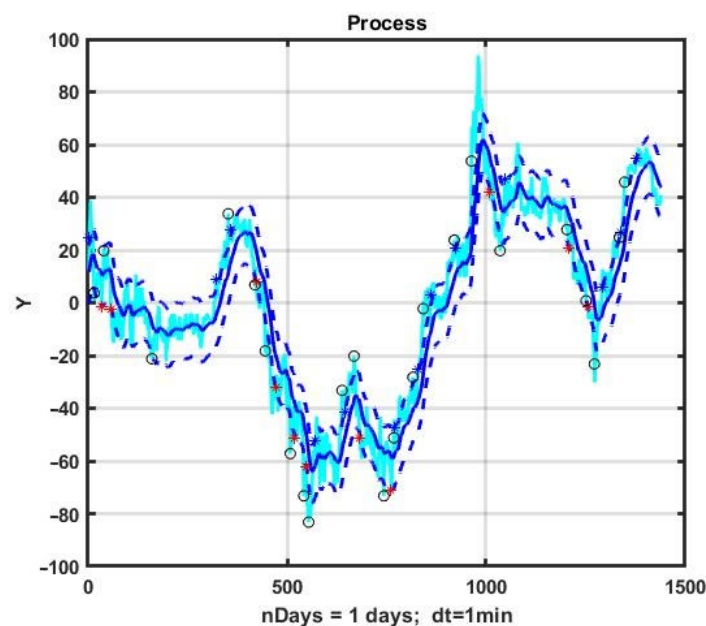


Figure 3. Example implementation of CSF with optimized parameters.

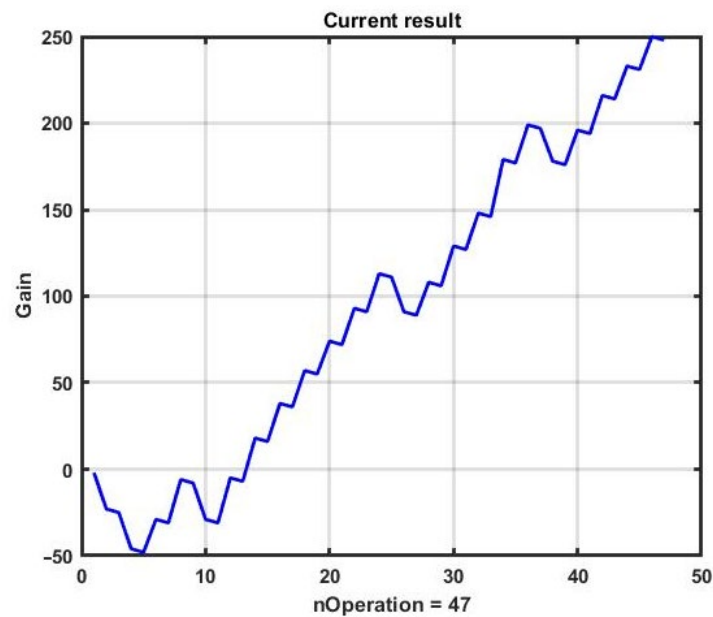


Figure 4. Performance of the CSF strategy with optimized parameters.

It is not difficult to see that this result is unstable. The results of applying the CSF strategy with the optimal parameters found for the next 10 days of management is shown in Figure 5. The obtained values completely confirm the instability of the found solution. None of the following nine days of observation produced a profitable result.

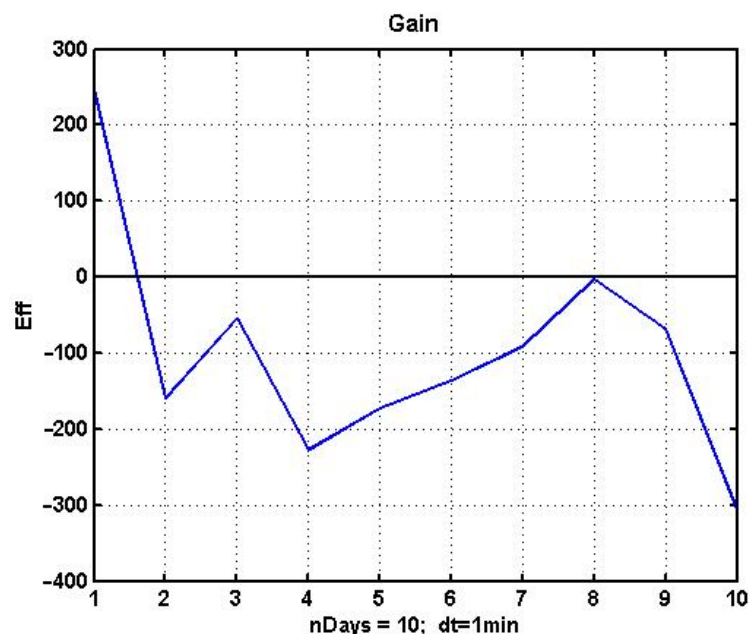


Figure 5. Performance of applying the CSF strategy with parameters optimal for the first day to the next nine days of management.

Comparing the results presented in Figure 6 with the dynamics of quotations of a financial instrument at the same observation interval, we can draw another conclusion. The structure of dynamics is more important than the degree of variability of the process. The minimum loss corresponds to the eighth day of observation, the most similar in structure to the first day for which the parameters of the control model were optimized. The worst result corresponds to the last, 10th day, with its strongly pronounced trend.

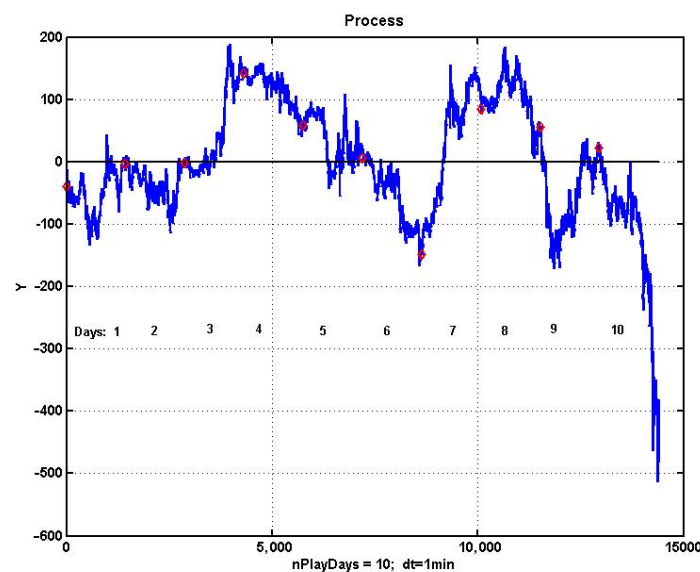


Figure 6. Dynamics of quotations during the indicated 10 days, corresponding to the performance of the CSF strategy in Figure 5.

Thus, in order to preserve, even if not the best, but at least positive result, it is necessary to significantly reduce the size of the time shift at which the optimal parameters are reassessed. This requirement significantly reduces the feasibility of using optimization based on brute forcing optional parameters. The problem is that the number of calculations shows an exponential increase along with an increase in the number of optimized parameters. It takes about 15 min on a 2.5 GHz 6-core INTEL Core i5 to optimize the model with five parameters by brute force on an observation interval of one day. A model with six parameters will take about 1.5 h.

Hence, there is a need to switch to suboptimal computational schemes for sequential optimization, which significantly reduces the calculation time to a level that makes possible real-time sequential data processing.

2.3. Features of Evolutionary Optimization for Chaotic Immersion Environments

In this paper, an algorithm based on the method of evolutionary modeling is proposed as a suboptimal computational scheme for optimizing the management strategy [38]. Modern computing techniques and applications based on this method can be found in [39–46]. Unlike evolutionary modeling, evolutionary optimization of management strategies is not interested in the degree of similarity of the mathematical model to the real data obtained via monitoring the managed object and the parameters of the immersion environment. Its task is to choose a management model that produces the best solution according to (3). Genetic algorithms decrease the volume of computations by about 40% according to estimates given in a number of referenced works.

At the same time, the control strategy itself, as a set of decision rules, can also be modified. The implementation of nonparametric mutations of the management strategy consists in choosing the structure of the model and management rules from the a priori knowledge bank. As a set of decision rules is selected, the list of parameters to be changed, their critical values and ranges of changes are modified. The advantage of this approach is its feasibility. However, at the same time, the arbitrariness of machine choice is limited, and there is no possibility of obtaining radically new strategies that are not provided by the programmer. Any regularization and any set of restrictions can block access to unexpected original solutions.

Moreover, the complete removal of restrictions in the process of random modifications of the structure of strategies leads to a huge number of meaningless decision rules. Waiting for any reasonable solution to appear will take time comparable with real biological evolution. At the same time, the question of artificial generation of management strategies remains open.

Evolutionary technology, like the entire probabilistic-statistical paradigm, is implicitly oriented towards a comfortable hypothesis about the repeatability of experiments in unchanging or slowly changing conditions. The transition to non-stationary, and even chaotic processes, inevitably destroys the optimality of statistical solutions, including those constructed via evolutionary modeling. However, chaos, in general, contains regularizing effects that reduce the degree of total uncertainty. If evolutionary technology can identify, at least not explicitly, and use such hidden patterns, then the task of constructing a winning strategy may be feasible. In addition, using an evolutionary computational scheme will help us answer the question of the fundamental admissibility of particular classes of management strategies.

The paper uses the basic evolutionary modeling algorithm described in [31]. The return to the original version of this concept is due to the fact that it does not introduce additional restrictions on the mechanism of variability and leaves a wide range of opportunities for its formation. For example, genetic algorithms are focused on the model of bisexual reproduction, which is very important in the implementation of variability in biology. However, for the models under consideration, there is no need to limit the process of variability to the mechanisms of gene exchange. The same can be said about the method of differential evolution. The formation of a new genome, as a mutant vector formed from other parental genomes, also introduces unnecessary restrictions for this case.

In econometric models, one can make any modifications to the structure of genes if they do not contradict common sense and the laws of the market. Therefore, it is reasonable to use the computational scheme of evolutionary optimization, which corresponds to the traditional concept of evolutionary modeling. At the same time, the mechanism of variability is based on the well-known mechanism for extracting a random variable from the range of permissible variations in genome parameters.

2.4. Algorithm of Evolutionary Optimization of the Management Model

Consider a set of ancestor strategies $S_A = \{S_{A1}, \dots, S_{AN_a}\}$ with N_a elements, each of which is defined by its structure R (the decision-making rule) and a set of corresponding numerical parameters a , i.e., $S = \{R, a\}$. The effectiveness of a strategy $Eff(s)$ is assessed via applying it to the time series of observations $Y(t)$, which together form an experimental retrospective dataset. We introduce two nonlinear operators.

1. The operator of variability and multiplication of strategies:

$$Var(S_A) : S_A = \{S_{A1}, \dots, S_{AN_a}\} \Rightarrow \{S_{D1}, \dots, S_{DN_D}\} : N_D = k_b N_a, k_b > 1$$

Here $\{S_{D1}, \dots, S_{DN_D}\}$ is a set of descendant strategies, each of which is created by modifying one of the ancestor strategies, N_D is the number of descendant strategies in one generation, and $k_b > 1$ is the multiplication coefficient of strategies. The union of ancestor strategies and descendant strategies is a generation of size $N_g = N_a + N_D = N_a (1 + k_b)$:

$$S_G = \{S_{G1}, \dots, S_{GN_g}\} = \{S_{A1}, \dots, S_{AN_a}\} \cup \{S_{D1}, \dots, S_{DN_D}\}$$

2. A selection operator that selects the “surviving” strategies from the generation $S_G = \{S_{G1}, \dots, S_{GN_g}\}$ that become the ancestors of the next generation:

$$Sel(S_1, \dots, S_{N_g}) = \{S_1, \dots, S_{N_g}\} \Rightarrow \{S_{\langle 1 \rangle}, \dots, S_{\langle N_a \rangle} : Eff(S_{\langle 1 \rangle}) \geq \dots \geq Eff(S_{\langle N_a \rangle}) \geq Eff(S_j), \forall j > N_a\}$$

Let $S_0 = \{K_0, p_0, a_0\}$ be a particular management strategy with the specified parameters, adopted as the basic “parent strategy”. Then evolutionary optimization is reduced to a cyclic execution of an operator sequence:

$$\begin{aligned} S_0 \Rightarrow \text{Var}(S_0) = S_A = \{S_1, \dots, S_{N_A}\} \Rightarrow \text{Var}(S_A) = S_D = \{S_1, \dots, S_{N_D}\} \\ \uparrow \\ \text{Sel}(S_1, \dots, S_{N_g}) = \{S_1, \dots, S_{N_g}\} \Leftarrow S_G = (S_A) \cup (S_D) \end{aligned} \quad (4)$$

where arrows denote the sequential order of operators. Since selection is carried out by superiority, the optimality of the final solution is not guaranteed. However, it will be the best of the whole set created during the implementation of evolutionary modeling.

The process of evolutionary optimization is obviously converging to more effective strategies by virtue of its very construction. This is due to the fact that the new generation always includes ancestor strategies in its composition. Consequently, the most effective strategies in principle cannot be discarded by the accepted selection and selection procedure. However, a high convergence rate cannot be expected due to the randomness of the modification process. The convergence rate will be close to the convergence rate of a random search, and depends on the size N_g of the generation being formed. It can be assumed that the convergence rate will be higher if the multiplication coefficient k_b is made variable so that the number of descendant strategies N_d depends on the effectiveness of parent strategies, i.e., $N_d = k(\text{Eff}(S_a))$, $k > 1$. In other words, a more effective ancestor can produce more offspring. However, this statement requires additional verification.

Other regularization methods aimed at increasing the convergence rate of evolutionary optimization are also possible.

2.5. Computational Aspects of the Evolutionary Optimization Algorithm

The functional structure of the algorithm of evolutionary optimization of management strategies is shown in Figure 7. The sequence of evolution is represented by a diagram of a process developing from the bottom up.

Let us consider the presented algorithm. At the preliminary stage, some basic strategy (prototype) is formed. Its structure is chosen either randomly or based on existing a priori experience in asset management. With the help of the variability mechanism, the basic prototype strategy is modified, giving rise to parent strategies.

Then, in accordance with (4), the program loops for the number of successive generations, which are usually called epochs.

At the first step of this loop, the first generation of descendant strategies is formed using the variability generator, which, combined with their parents, form the first generation of strategies S_G . Furthermore, each of the first-generation strategies undergoes a testing by being applied to a set of retrospective observations $\{Y(t), Y(t - Y)\}$, where T is the size of the validation dataset. The created strategies are ranked by their effectiveness $\text{Eff}(S_i)$, $i = 0, \dots, N_g$ and a specified number N_a of “surviving” strategies that are allowed for further “reproduction” (modification) are selected. The selected strategies are the parents of a new set of modified descendant strategies and together with them form the second generation.

Furthermore, the cycle is repeated until either a specified number of generations is reached, or, according to some other criterion, for example, when the effectiveness of management does not improve during a given number of epochs.

The variability mechanism used in the program randomly selects changes made to the strategy from the following set of options:

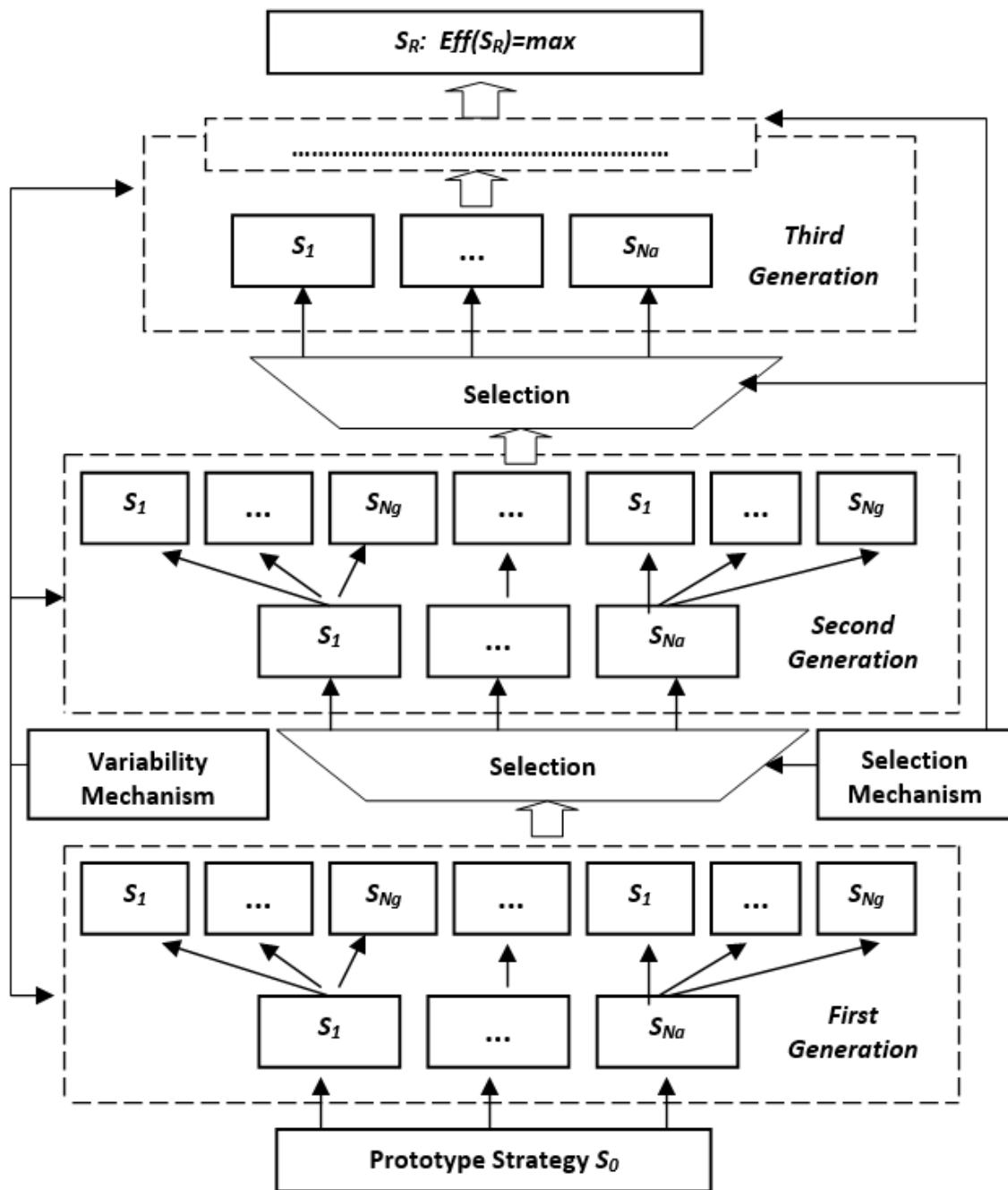


Figure 7. General functional structure of the evolutionary optimization algorithm.

1. Small single changes. In the strategy undergoing modification, relatively small changes are made to only one parameter (gene) selected by a random draw. The size of the change range depends on the parameter. This one-time change does not usually exceed 10% of the original value. The choice of the parameter is carried out by a random draw, similar to how it happens in the Monte Carlo method.
2. Small group changes. They are carried out similarly to the previous case, but are made to several gene parameters at once instead. Their amount and their numbers are selected via a random draw.
3. Strong single mutation or parametric mutation. The gene number is selected via a random draw. Usually, the number of mutations in a generation is small, and the probability of their occurrence does not exceed 2–3%. The size of the change field

also depends on the parameter, usually a single mutation can reach 30–50% of the original value.

4. Strong group changes are similar to the previous case, but are made immediately in several parameters, as in case 2.
5. Structural (nonparametric) mutations. The parent strategy with some relatively small probability (usually less than 0.01) may undergo nonparametric mutation. In this case, the number of genes in the original genome may change, or, in a more radical case, the management strategy itself may be completely modified. The most rational way in this case consists of randomly choosing a management strategy from an a priori created knowledge base.

It should be expected that such mutations will be quite constructive, since the strategies included in the data bank, to one degree or another, have already been pre-selected by at least the common sense of their developer. For example, there is a transition from a channel strategy to management based on trend analysis, etc. However, this approach limits the evolution to the level of the programmer's constructive imagination.

A more radical approach consists in stochastic synthesis of new strategies using artificial intelligence technologies. In this case, the evolution program gets out of the strict control of the developer not only at the parametric, but also at the structural level. This approach, in theory, enables the generation of completely new, unexpected solutions. However, in most cases, such mutations will generate ineffective strategies that will be immediately eliminated by the selection mechanism, without generating variants of management strategies in the next generation. In this regard, it may make sense to deliberately preserve such mutations for a given number of epochs and allow them to generate variants of descendants in a mutant zoo.

The selection mechanism, as already noted, ranks a set of strategies according to their effectiveness, and selects among them the best strategies that become the parents of the next generation. It is important to note the principle of incomplete or open solutions, meaning that at each cycle of selection, not the only best option is selected, but a group of strategies is as well. This approach allows us to come to the best solution along a chain consisting of intermediate options other than optimal, which is especially important when searching for effective management strategies in chaotic immersion environments.

3. Results

As an example of the implementation of the evolutionary optimization method, consider the asset management task based on the CSF management strategy. In order to be able to compare with the best solution found by brute force, we will consider the same one-day observation interval as in the examples shown in Figures 1–4. As optimization parameters (the genome) we will use $M = \{\alpha, B1, B2, TP, SL\}$. From the above optimization example (Figures 3 and 4) it follows that the best parameters of the control model are $\alpha^* = 0.05$, $B1^* = 10$, $B2^* = 12$, $TP^* = 18$, $SL^* = 18$.

Figure 8 shows an example of CSF implementation with evolutionary optimization of the management model for 10 epochs. The best result was 177 points with the parameters of the model being $\alpha^* = 0.05$, $B1^* = 12$, $B2^* = 12$, $TP^* = 15.5$, $SL^* = 14.8$. The convergence to the found result is shown in Figure 9.

The dependence of the effectiveness of the CSF strategy during the epoch is shown in Figure 10.

The plots show that the obtained result, as expected, is inferior to the gain obtained by brute force. However, this search only took 0.26 s with the same CPU and with the observation interval the same as in the example with brute force optimization, and half of this time was spent on the implementation and output of graphics.

The question arises of how much the quality of management can be improved by increasing the number of epochs. Figures 11 and 12 show changes in the effectiveness of the best versions of the management model obtained during, 50 and 200 iterations, respectively.

The gain in the above examples reached, 197 p. and 203 p., respectively, did not achieve the effectiveness of brute force (248 p.). However, the time spent was 0.203 s in the first case, and 0.541 s in the second. A further search for the best option was quite slow, and this can be seen from the section of the horizontal line in Figure 12, starting from about 70 epochs. Increasing the number of epochs to 500 increased the profit to only 209 p., and spending to 1.17 s.

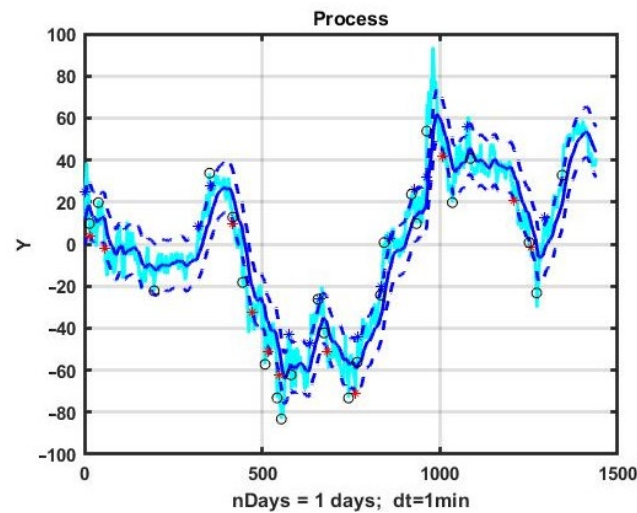


Figure 8. An example implementation of the CSF strategy with parameters obtained via evolutionary optimization.

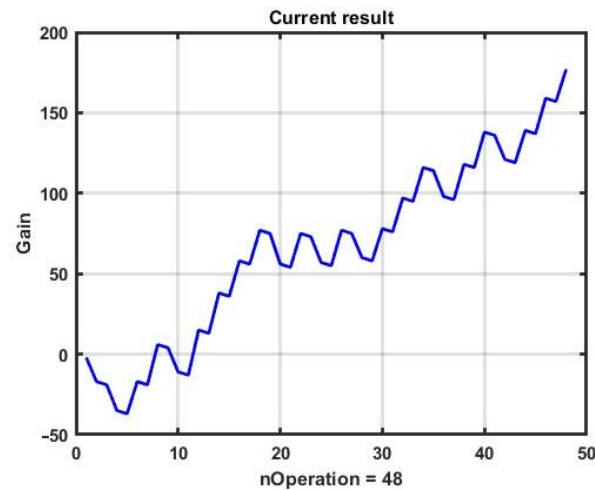


Figure 9. Effectiveness of the CSF strategy with evolutionary optimization.

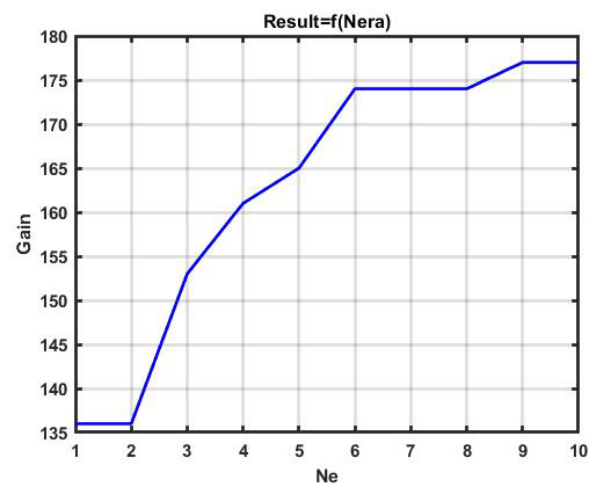


Figure 10. Effectiveness of the CSF strategy with evolutionary optimization for 10 epochs.

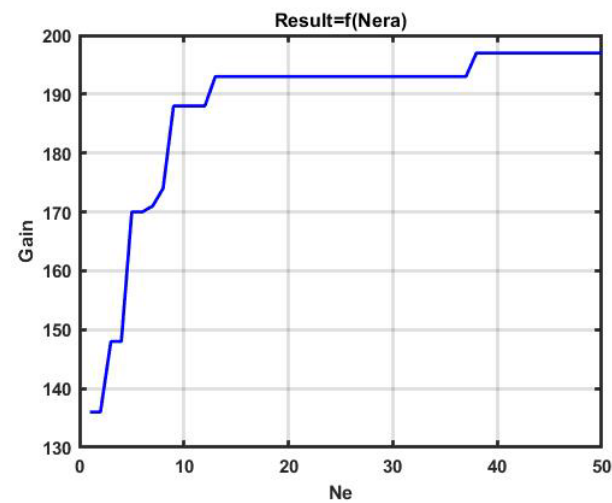


Figure 11. Effectiveness of evolutionary optimized CSF strategy, 50 epochs.

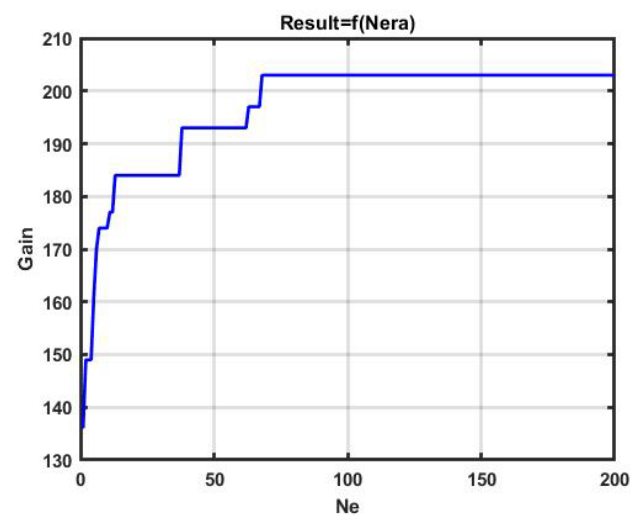


Figure 12. Effectiveness of evolutionary optimized CSF strategy, 200 epochs.

Note that in the considered examples, a simplified version of the variability operator was used. For example, structural mutation was not used at all. However, even in the

proposed case, it becomes possible to conclude that this algorithm is suitable for sequential optimization with new data arriving every 0.5–1 s.

4. Discussion

The chaotic dynamics of quotations, characteristic for electronic capital markets, obstruct the effective use of traditional methods of increasing the stability of management strategies. As shown in the examples in Figures 5 and 6, the parameters that are optimal for a given day of observing the dynamics of quotations on the very next day lead to an unpredictable loss. In this regard, the management model should be dynamically corrected as new results of monitoring the market situation become available.

At the same time, the most powerful optimization tool based on brute force incurs significant time costs even for modern high-performance processors. This is due to the fact that the time complexity here is described by a power function with an increase in the number of iteration steps, and by an exponential function with an increase in the number of model parameters. As a result, even with a relatively simple model with five to six parameters, the required computational resources exceed the capabilities of modern processors. At the same time, increasing the time interval between successive calculations of the best parameters is unacceptable, given that the dynamics of market assets is absolutely non-inertial and can be discontinuous even within one minute of observation. Due to this, we proposed an optimization of the management model based on evolutionary modeling.

We analyzed the effectiveness of this approach using the example of real data obtained by monitoring the currency exchange market. The results of numerical studies have shown that maximum gain decreased by about 15–20% compared to the result obtained by brute forcing parameter values. At the same time, the number of operations decreased to such a level that model parameters can be optimized even with a one-second interval of monitoring results.

It is important to note that the potential of evolutionary optimization is by no means exhausted by the version of its implementation considered in the paper. First of all, the above example did not use structural mutations. In essence, we are talking about utilizing qualitatively different strategies, which can give a result that significantly exceeds the mechanistic optimization based on brute forcing parameter values. Moreover, new opportunities arise, such as artificial intelligence technologies: the computer independently generates management strategies that are not contained in the a priori knowledge base.

The second point related to the advantage of using evolutionary optimization is due to the fact that the drawing of parameter values is carried out on a scale of counts close to continuous. The discreteness is essentially determined only by the mantissa of the bit grid. This means that the conducted adjustment is finer than that obtained with a fixed iteration step. The advantage of fine-tuning is due to the extremely high sensitivity of the management result to variations in the values of the model parameters, which is generally characteristic of a chaotic series of observations.

It should be noted that we used a very simple channel strategy as a demonstration example in this work. This was done in order to ensure the clarity and interpretability of the results. However, in practice, such strategies are not used as is, because in most cases they only lead to loss.

In [33] it was shown that for almost any variations of the observed dynamic process, profitable solutions exist for this class of strategies. However, their implementation requires fine-tuning, minor deviations from which, as follows from the theory of dynamic chaos, lead to loss. Nevertheless, with a small time shift of observation series relative to the optimization tuning interval, the positive gain on average is preserved. This means that it is fundamentally possible to build a dynamic self-adjusting asset management system. One of the variants of such a solution is considered in this paper.

Other formulations of asset management tasks that also use genetic algorithms for optimization can be seen, for example, in [47,48]. At the same time, various algorithms

based on random search in some cases lead to completely unexpected and, often, anti-intuitive results [49,50].

A development of these studies being focused on constructing a dynamic self-adjusting asset management system with an increased level of resistance to variations in the dynamic and statistical characteristics of observation series implies considering the following points:

- the study of potential characteristics of self-adjusting asset management systems for various sets of dynamic properties of observation intervals of chaotic processes;
- the development of a knowledge base of management strategies and its application for implementing structural mutations of the management model in the mechanism of variability of the evolutionary optimization algorithm;
- the development of randomized synthesis of management strategies using multi-expert data analysis [51];
- the use of composite algorithms combining the capabilities of robustification and adaptation in management decision-making.
- the effectiveness of the application of evolutionary optimization in markets and periods that differ in the degree of market efficiency within the Efficient Market Hypothesis (EMH) [52]. It is supposed that the greatest profit can be made in a highly inefficient market. At the same time various exchange markets all have multifractal structural properties with different levels in the sample and sub-samples that cause inefficiency with different levels in these foreign exchange markets [30]. Another work reveals that the efficiency of the cryptocurrency markets varies over time, which is consistent with adaptive market hypothesis (AMH) [53]. The question about the level of current market inefficiency which is acceptable for self-adjusting asset management systems needs to be investigated.
- the use of external add-ons that carry information exogenous to technical analysis on expected trends of the considered financial instrument and market mood in general.
- The outlined issues constitute the subject of our further research.

Author Contributions: Conceptualization, Methodology, A.M. (Andrey Makshanov); Validation, Writing, A.M. (Alexander Musaev); Review and Editing, Investigation, Programming, Visualization, Administration, Scientific Discussions, Supervision, Funding Acquisition, D.G. All authors have read and agreed to the published version of the manuscript.

Funding: The research of Alexander Musaev described in this paper is partially supported by the state research FFZF-2022-0004. The research of Dmitry Grigoriev for this paper is supported by Saint Petersburg State University, project ID: 93024916.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The source of data is Finam.ru (<https://www.finam.ru/>, accessed on 21 April 2022).

Acknowledgments: The authors are grateful to participants at the Center for Econometrics and Business Analytics (ceba-lab.org, CEBA) seminar series for helpful comments and suggestions.

Conflicts of Interest: The authors declare that they have no conflict of interest.

References

1. Layek, G.C. *An Introduction to Dynamical Systems and Chaos*; Springer: New Delhi, India, 2015; Volume 449.
2. Kim, E.J. Intermittency and self-organisation in turbulence and statistical mechanics. *Entropy* **2019**, *21*, 574. [[CrossRef](#)]
3. Inglada-Perez, L. A comprehensive framework for uncovering non-linearity and chaos in financial markets: Empirical evidence for four major stock market indices. *Entropy* **2020**, *22*, 1435. [[CrossRef](#)]
4. Gregory-Williams, J.; Williams, B.M. *Trading Chaos: Maximize Profits with Proven Technical Techniques*, 2nd ed.; John Wiley & Sons: New York, NY, USA, 2004; p. 251.
5. Feldman, D. *Chaos and Dynamical Systems*; Princeton University Press: Princeton, NJ, USA, 2019.
6. Bensaïda, A.; Litimi, H. High level chaos in the exchange and index markets. *Chaos Solitons Fractals* **2013**, *54*, 90–95. [[CrossRef](#)]
7. Klioutchnikov, I.; Sigova, M.; Beizerov, N. Chaos theory in finance. *Procedia Comput. Sci.* **2017**, *119*, 368–375. [[CrossRef](#)]

8. Gu, E.G. On the existence of chaos in a discontinuous area-preserving map arising in financial markets. *Int. J. Bifurc. Chaos* **2018**, *28*, 1850177. [\[CrossRef\]](#)
9. Sarkar, A.; Chakrabarti, G.; Sen, C. Volatility, long memory, and chaos: A discussion on some “Stylized facts” in financial markets with a focus on high frequency data. In *Development and Sustainability*; Springer: New Delhi, India, 2013; pp. 71–101.
10. Iordanova, T. An Introduction to Non-Stationary Processes. Available online: <https://www.investopedia.com/articles/trading/07/stationary.asp> (accessed on 21 April 2022).
11. Riani, M.; Hubert, M. Editorial, special issue on “Advances in robust statistics”. *Metron* **2021**, *79*, 121–125. [\[CrossRef\]](#)
12. Llanos, C.; Galletti, R.C.; Maronna, R. On-line process monitoring using a robust statistics based methodology. *Lat. Am. Appl. Res.-Int. J.* **2019**, *49*, 111–116. [\[CrossRef\]](#)
13. Maronna, R.A.; Martin, R.D.; Yohai, V.J.; Salibián-Barrera, M. *Robust Statistics: Theory and Methods (with R)*; John Wiley & Sons: New York, NY, USA, 2019.
14. Rousseeuw, P.J.; Hubert, M. Anomaly detection by robust statistics. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2018**, *8*, e1236. [\[CrossRef\]](#)
15. Petrelli, M. Introduction to robust statistics. In *Introduction to Python in Earth Science Data Analysis*; Springer: Cham, Switzerland, 2021; pp. 181–194.
16. Musaev, A.; Grigoriev, D. Machine learning-based cyber-physical systems for forecasting short-term state of unstable systems. In *Cyber-Physical Systems: Intelligent Models and Algorithms. Studies in Systems, Decision and Control*; Springer: Cham, Switzerland, 2022; Volume 417.
17. Musaev, A.; Makshanov, A.; Grigoriev, D. Forecasting multivariate chaotic processes with precedent analysis. *Computation* **2021**, *9*, 110. [\[CrossRef\]](#)
18. Jin, Y.; Wang, H.; Chugh, T.; Guo, D.; Miettinen, K. Data-driven evolutionary optimization: An overview and case studies. *IEEE Trans. Evol. Comput.* **2018**, *23*, 442–458. [\[CrossRef\]](#)
19. Zhang, Y.; Gong, D.W.; Sun, J.Y.; Qu, B.Y. A decomposition-based archiving approach for multi-objective evolutionary optimization. *Inf. Sci.* **2018**, *430*, 397–413. [\[CrossRef\]](#)
20. Abu Khurma, R.; Aljarah, I.; Sharieh, A.; Abd Elaziz, M.; Damaševičius, R.; Krilavičius, T. A review of the modification strategies of the nature inspired algorithms for feature selection problem. *Mathematics* **2022**, *10*, 464. [\[CrossRef\]](#)
21. Kalyuzhnaya, A.V.; Nikitin, N.O.; Hvatov, A.; Maslyayev, M.; Yachmenkov, M.; Boukhanovsky, A. Towards generative design of computationally efficient mathematical models with evolutionary learning. *Entropy* **2020**, *23*, 28. [\[CrossRef\]](#)
22. Boer, K.; Kaymak, U.; Spiering, J. From discrete-time models to continuous-time, asynchronous modeling of financial markets. *Comput. Intell.* **2007**, *23*, 142–161. [\[CrossRef\]](#)
23. Hastie, T.J.; Tibshirani, R.J. *Generalized Additive Models*; Routledge: Abingdon, UK, 2017.
24. Musaev, A.; Grigoriev, D. Analyzing, modeling, and utilizing observation series correlation in capital markets. *Computation* **2021**, *9*, 88. [\[CrossRef\]](#)
25. Bezruchko, B.P.; Smirnov, D.A. *Extracting Knowledge from Time Series: An Introduction to Nonlinear Empirical Modeling*; Springer: New York, NY, USA, 2010.
26. Petropulu, A.P. Higher-order spectral analysis. In *The Digital Signal Processing Handbook, Second Edition: Wireless, Networking, Radar, Sensor Array Processing, and Nonlinear Signal Processing*; CRC Press: Boca Raton, FL, USA, 2009; pp. 18–21.
27. Cremers, J.; Hübler, A. Construction of differential equations from experimental data. *Z. Nat. A* **1987**, *42*, 797–802. [\[CrossRef\]](#)
28. Crutchfield, J.P.; McNamara, B.S. Equations of motion from a data series. *Complex Syst.* **1987**, *1*, 417–452.
29. Cheng, C. Multi-scale Gaussian process experts for dynamic evolution prediction of complex systems. *Expert Syst. Appl.* **2018**, *99*, 25–31. [\[CrossRef\]](#)
30. Han, C.; Wang, Y.; Ning, Y. Comparative analysis of the multifractality and efficiency of exchange markets: Evidence from exchange rates dynamics of major world currencies. *Phys. A Stat. Mech. Appl.* **2019**, *535*, 122365. [\[CrossRef\]](#)
31. Kirichenko, L.; Radivilova, T.; Bulakh, V. Binary classification of fractal time series by machine learning methods. In *International Scientific Conference Intellectual Systems of Decision Making and Problem of Computational Intelligence*; Springer: Cham, Switzerland, 2019; pp. 701–711.
32. Gardner, E.S.J. Exponential smoothing: The state of the art—Part II. *Int. J. Forecast.* **2006**, *22*, 637–666.
33. Musaev, A.; Makshanov, A.; Grigoriev, D. Numerical studies of channel management strategies for nonstationary immersion environments: EURUSD case study. *Mathematics* **2022**, *10*, 1408. [\[CrossRef\]](#)
34. Nti, I.K.; Adekoya, A.F.; Weyori, B.A. A systematic review of fundamental and technical analysis of stock market predictions. *Artif. Intell. Rev.* **2020**, *53*, 3007–3057. [\[CrossRef\]](#)
35. Niederhoffer, V.; Kenner, L. *Practical Speculation*; John Wiley & Sons: New York, NY, USA, 2005.
36. Colby, R.W.; Meyers, T.A. *The Encyclopedia of Technical Market Indicators*; IRWIN Professional Publishing: Burr Ridge, IL, USA, 2012.
37. Chordia, T.; Goyal, A.; Saretto, A. *P-Hacking: Evidence from Two Million Trading Strategies*; Swiss Finance Institute: Zürich, Switzerland, 2018.
38. Lu, Y. Artificial intelligence: A survey on evolution, models, applications and future trends. *J. Manag. Anal.* **2019**, *6*, 1–29. [\[CrossRef\]](#)
39. Jia, L.U.; Shazemeen, N.M.; Martinkute-Kauliene, R. Portfolio decision using time series prediction and multi-objective optimization. *Rom. J. Econ. Forecast.* **2020**, *23*, 118.

40. Eltaieb, T.; Mahmood, A. Differential evolution: A survey and analysis. *Appl. Sci.* **2018**, *8*, 1945. [[CrossRef](#)]
41. Mamanis, G. A comparative study on multi-objective evolutionary algorithms for tri-objective mean-risk-cardinality portfolio optimization problems. In *Computational Management*; Springer: Cham, Switzerland, 2021; pp. 277–303.
42. Masich, I.S.; Kulachenko, M.A.; Stanimirović, P.S.; Popov, A.M.; Tovbis, E.M.; Stupina, A.A.; Kazakovtsev, L.A. Formation of fuzzy patterns in logical analysis of data using a multi-criteria genetic algorithm. *Symmetry* **2022**, *14*, 600. [[CrossRef](#)]
43. Slowik, A.; Kwasnicka, H. Evolutionary algorithms and their applications to engineering problems. *Neural Comput. Appl.* **2020**, *32*, 12363–12379. [[CrossRef](#)]
44. Bansal, J.C.; Singh, P.K.; Pal, N.R. *Evolutionary and Swarm Intelligence Algorithms*; Springer: Cham, Switzerland, 2019; Volume 779.
45. Tian, Y.; Cheng, R.; Zhang, X.; Li, M.; Jin, Y. Diversity assessment of multi-objective evolutionary algorithms: Performance metric and benchmark problems. *IEEE Comput. Intell. Mag.* **2019**, *14*, 61–74. [[CrossRef](#)]
46. Cai, X.; Zhao, H.; Shang, S.; Zhou, Y.; Deng, W.; Chen, H.; Deng, W. An improved quantum-inspired cooperative co-evolution algorithm with multi-strategy and its application. *Expert Syst. Appl.* **2021**, *171*, 114629. [[CrossRef](#)]
47. Asjad, M.; Khan, S. Analysis of maintenance cost for an asset using the genetic algorithm. *Int. J. Syst. Assur. Eng. Manag.* **2017**, *8*, 445–457. [[CrossRef](#)]
48. Sarijaloo, A.; Moradbakloo, A. Asset management using genetic algorithm: Evidence from Tehran stock exchange. *Manag. Sci. Lett.* **2014**, *4*, 221–226. [[CrossRef](#)]
49. Fogel, G.B. Evolutionary computation for the inference of natural evolutionary histories. *IEEE Connect.* **2005**, *3*, 11–14.
50. Yang, X.S. Nature-inspired optimization algorithms: Challenges and open problems. *J. Comput. Sci.* **2020**, *46*, 101104. [[CrossRef](#)]
51. Musaev, A.; Grigoriev, D. Multi-expert systems: Fundamental concepts and application examples. *J. Theor. Appl. Inf. Technol.* **2022**, *100*, 336–348.
52. Singh, J.E.; Babshetti, V.; Shivaprasad, H.N. Efficient market hypothesis to behavioral finance: A review of rationality to irrationality. *Mater. Today Proc.* 2021, in press. [[CrossRef](#)]
53. Chu, J.; Zhang, Y.; Chan, S. The adaptive market hypothesis in the high frequency cryptocurrency market. *Int. Rev. Financ. Anal.* **2019**, *64*, 221–231. [[CrossRef](#)]