https://doi.org/10.31891/csit-2023-4-4
UDC 004.8

Eugen FEDOROV, Maryna LESHCHENKO
Cherkasy State Technological University
Liubov KIBALNYK, Hanna DANYLCHUK
Cherkasy Bohdan Khmelnytskyi National University

CONNECTIONIST-METAHEURISTIC APPROACH TO THE ANALYSIS OF THE GLOBAL ECONOMY'S INVESTMENT ENVIRONMENT INDICATORS

The urgent task of using new approaches to analyze the indicators of foreign direct investment and macroeconomic indicators that affect the volume of their attraction to a particular country in the world economy was solved by a connectionist-metaheuristic approach.

The connectionist-metaheuristic approach solved the urgent task of using new approaches to analyze the foreign direct investment and macroeconomic indicators that affect the volume of their attraction to a particular country in the world economy. The proposed connectionist-metaheuristic system makes it possible to improve the quality of the approximation due to: the simplification of structural identification through the use of only one hidden layer of neural network models; reduction of the computational complexity of parametric identification and the ensuring good scalability through the use of batch mode for non-recurrent neural network models and multi-agent metaheuristics for recurrent neural network models; descriptions of non-linear dependencies through the use of neural network models; high approximation accuracy due to the use of recurrent neural network models; resistance to data incompleteness and data noise due to the use of metaheuristics for parametric identification of recurrent neural network models; lack of requirements for knowledge of distribution, homogeneity, weak correlation, and optimal factors' choice. In the case of a GPU, an LSTM-based neural network with the highest approximation accuracy should be chosen. For LSTM, the coefficient of determination using the gradient method is 0.785, and using metaheuristics (modified wasp colony optimization) is 0.835. The proposed approach makes it possible to expand the scope of approximation methods' application based on artificial neural networks and metaheuristics, which is confirmed by its adaptation for an economic problem and contributes to an increase in intelligent computer systems efficiency for general and special purposes.

Keywords: artificial neural networks, approximation methods, connectionist-metaheuristic approach, wasp swarm optimization, macroeconomic indicators.

Євген ФЕДОРОВ, Марина ЛЕЩЕНКО
Черкаський державний технологічний університет
Любов КІБАЛЬНИК, Ганна ДАНИЛЬЧУК
Черкаський національний університет імені Богдана Хмельницького

КОНЕКЦІОНІСТСЬКО-МЕТАЕВРИСТИЧНИЙ ПІДХІД ДО АНАЛІЗУ ПОКАЗНИКІВ ІНВЕСТИЦІЙНОГО СЕРЕДОВИЩА СВІТОВОЇ ЕКОНОМІКИ

Актуальне завдання використання нових підходів до аналізу показників прямих іноземних інвестицій та макроекономічних показників, що впливають на обсяги їх залучення в ту чи іншу країну світового господарства, було вирішено за допомогою коннекціоністсько-метаевристичного підходу.

У дослідженні використано коннекціоністсько-метаевристичний підхід в ході використання нових підходів до аналізу прямих іноземних інвестицій та макроекономічних показників, що впливають на обсяги їх залучення до тієї чи іншої країни світової економіки. Запропонований коннекціоністсько-метаевристичний підхід дає змогу підвищити якість апроксимації за рахунок: спрощення структурної ідентифікації за рахунок використання лише одного прихованого шару нейромережевих моделей; зниження обчислювальної складності параметричної ідентифікації та забезпечення гарної масштабованості за рахунок використання пакетного режиму для неповторних моделей нейронних мереж і багатоагентних метаевристик для рекурентних моделей нейронних мереж; описів нелінійних залежностей за допомогою нейромережевих моделей; високої точності апроксимації за рахунок використання рекурентних нейромережевих моделей; стійкості до неповноти даних і шуму даних за рахунок використання метаевристик для параметричної ідентифікації рекурентних моделей нейронних мереж; відсутності вимог щодо знання розподілу, однорідності, слабкої кореляції та вибору оптимальних факторів. У ході використання графічного процесора запропоновано вибрати нейронну мережу на основі LSTM, яка має найвищу точність апроксимації. Для LSTM коефіцієнт детермінації за допомогою градієнтного методу становить 0,785, а за допомогою метаевристики (модифікована оптимізація колонії ос) – 0,835. Запропонований підхід дає змогу розширити сферу застосування методів апроксимації на основі штучних нейронних мереж і метаевристик, що підтверджується його адаптацією до економічної задачі, та сприяє підвищенню ефективності інтелектуальних комп'ютерних систем загального та спеціального призначення.

Ключові слова: штучні нейронні мережі, методи апроксимації, коннекціоністсько-метаевристичний підхід, оптимізація осиного рою, макроекономічні показники.

Introduction

International investment activity effectively complements international trade and contributes to the economic growth of national economies. The international investment flows from different countries intertwine and interact with each other, turning into global investment resources. An important condition for attracting investment resources is the presence of powerful economic potential, in which foreign investment flows are the foundation of global economy development. Important components of the economic potential and factors of the formation of a favorable investment environment are such as GDP volumes, the inflation rate, the unemployment rate, indicators of exports of goods and services, and others.

The object of the research is the process of analyzing statistical indicators that are indicators of the national economies’ development in the context of globalization. The subject of the research is a connectionist-metaheuristic approach for the analysis of economic indicators. The aim of the research is to increase the efficiency of the analysis of economic indicators based on the connectionist-metaheuristic approach to the approximation models’ creation.

The main tasks of the research:

1. To form a vector of economic indicators.
2. To create a recurrent neural network approximation model.
3. To develop a gradient method for identifying the parameters of recurrent neural network approximation models.
4. To create a metaheuristic method for the parametric identification of recurrent neural network approximation models.

Related works

The analysis of macroeconomic indicators using various methods is an important task of modern research. Nowadays various methods of economic analysis, including approximation are used to analyze such indicators.

The traditional methods of approximation are:

1. Statistical [1-3].
2. Analytical [4-6].
3. Method of group accounting of arguments (MGUA) [7, 8].

Existing traditional methods of approximation have one or more of the following disadvantages [9, 10]:

– approximation models are focused on linear dependencies;
– structural-parametric identification of the approximation model has a high computational complexity;
– approximation models cannot provide high accuracy;
– approximation models are sensitive to data noise and it is necessary to ensure data completeness for them;
– approximation models provide a priori knowledge of the type of distribution, weak correlation, uniformity, and preselection of factors.

In this regard, it is relevant to create an approximation approach that will eliminate these shortcomings.

At present, the neural network approach [11, 12] is popular, which allows using both conventional activation functions and wavelet-based activation functions. Since only a sequential learning mode is used for recurrent neural networks, metaheuristics [13-16] can be used to eliminate this drawback, which will allow them to be used in the analysis of economic indicators. Among the metaheuristics that allow parallelization on the GPU, there are multi-agent metaheuristics [17-20].

Research methods

1. Formation of the factors’ vector

We used the following indicators of the national economies' investment attractiveness to form the data array: the gross domestic product volume (GDP) per capita (per year, US dollars), inflation rate (according to the consumer price index, which reflects the annual percentage change in the cost for the average consumer of purchasing a basket of goods and services, per year, %), exports of goods and services indicators (total volume, per year, USD), labor force indicators (labor force is people aged 15 years and older who provide labor for the goods and services production, per year, number of people), income tax indicators (income tax is the amount of income taxes paid by an enterprise, % of the enterprises' commercial profit in a country, per year). All indicators were formed for the 2010-2019 period.

For neural network models, to improve the approximation accuracy, the factors and responses are normalized.

2. Creating a neural network model based on long short-term memory with a forgetting gate

A long short-term memory (LSTM) neural network with a forgetting gate is a recurrent two-layer ANN [LSTM, 25, 26].

It is considered that a memory block consists of only one memory cell to reduce computational complexity.

1. The calculation of the outputs of the input layer

$$y_i^{in}(n-1) = x_i,\ i \in \overline{1, N^{(0)}}$$

2. The calculation of the forgets gates outputs

$$y_j^{g\varphi}(n) = f\left(b_j^{g\varphi} + \sum_{i=1}^{N^{(0)}} w_{ij}^{in-g\varphi} y_i^{in}(n-1)\right),\ j \in \overline{1, N^{(1)}}$$

3. The calculation of the output signals of input gateways

$$y_j^{gin}(n) = f\left(b_j^{gin} + \sum_{i=1}^{N^{(0)}} w_{ij}^{in-gin} y_i^{in}(n-1)\right),\ j \in \overline{1, N^{(1)}}$$

4. The calculation of the input signals of memory cells

$$\tilde{y}_j^{c}(n) = 2g\left(b_j^{s} + \sum_{i=1}^{N^{(0)}} w_{ij}^{in-s} y_i^{in}(n-1)\right),\ j \in \overline{1, N^{(1)}}$$

5. The calculation of the output signals of exit gates

$$y_j^{gout}(n) = f\left(b_j^{gout} + \sum_{i=1}^{N^{(0)}} w_{ij}^{in-gout} y_i^{in}(n-1)\right),\ j \in \overline{1, N^{(1)}}$$

6. The calculation of the states of memory cells

$$s_j^{c}(n) = y_j^{g\varphi}(n) s_j^{c}(n-1) + y_j^{gin}(n) \tilde{y}_j^{c}(n),\ j \in \overline{1, N^{(1)}}$$

7. The calculation of the output signals of memory cells

$$y_j^{c}(n) = y_j^{gout}(n) g\left(s_j^{c}(n)\right),\ j \in \overline{1, N^{(1)}}$$

8. The calculation of the output signal of the output layer

$$y^{out}(n) = b^{out} + \sum_{i=1}^{N^{(1)}} w_i^{c-out} y_i^{c}(n),$$

where $N^{(0)}$ – the length of the factors vector,

$N^{(1)}$ – is the number of neurons in the hidden layer memory blocks,

$b_j^{g\varphi}$ – is the threshold for the forgetting gate of the *j*-th block of memory,

b_j^{gin} – is the threshold for the input gateway of the *j*-th memory block,

b_j^{s} – is the threshold for the memory cell of the *j*-th memory block,

b_j^{gout} – is the threshold for the output gateway of the *j*-th memory block,

b^{out} – is the threshold for the neuron of the output layer,

$w_{ij}^{in-g\varphi}$ – is the weight of the connection between the *i*-th neuron of the input layer and the *j*-th memory block forgetting gate,

w_{ij}^{in-gin} – is the weight of the link between the *i*-th neuron of the input layer and the input gateway of the *j*-th memory block,

w_{ij}^{in-s} – is the weight of the link between the *i*-th neuron of the input layer and the input of the memory cell of the *j*-th memory block,

$w_{ij}^{in-gout}$ – is the weight of the link between the *i*-th neuron of the input layer and the output gateway of the *j*-th memory block,

w_i^{c-out} – is the weight of the link between the output of the memory cell of the *i*-th memory block and the output layer neuron,

$g(s) = \tanh(s)$ – is the activation function for memory cells,

$f(s) = \dfrac{1}{1+e^{-s}}$ – is the activation function for gateways.

3. Development of a gradient method for identifying the parameters of a neural network approximation model based on long short-term memory with a forgetting gate

1. Number of training iteration $n = 2$, initialization by uniform distribution on the interval (0,1) or [-0.5, 0.5] of offsets (thresholds) $b_j^{gin}(n), b_j^{g\varphi}(n), b_j^{gout}(n), b_{jv}^{s}(n)$ and weights $w_{ij}^{in-gin}(n)$, $w_{ij}^{in-g\varphi}(n)$, $w_{ij}^{in-gout}(n)$, $w_{ijv}^{in-s}(n)$, $i \in \overline{1, N^{(0)}}$, $j \in \overline{1, N^{(1)}}$, $v \in \overline{1, S_j}$, of offsets (thresholds) $b_j^{out}(n)$ and weights $w_{ivj}^{c-out}(n)$, $i \in \overline{1, N^{(1)}}$, $j \in \overline{1, N^{(2)}}$, $v \in \overline{1, S_j}$, where $N^{(0)}$ – the number of neurons in the input layer, $N^{(1)}$ – is the number of neurons in the hidden layer, $N^{(2)}$ – is the number of neurons in the output layer, S_j – is the number of cells in the j - th memory block.

2. The training set $\{(\mathbf{x}_\mu, \mathbf{d}_\mu) \mid \mathbf{x}_\mu \in R^{N^{(0)}}, \mathbf{d}_\mu \in R^{N^{(2)}}\}$, $\mu \in \overline{1, P}$ is specified where $\mathbf{x}_\mu$ – is the μ -th training input vector, $\mathbf{d}_\mu$ – is the μ -th training output vector, P – is the power of the training set. The number of the current pair from the training set is $\mu = 2$.

3. Initial computation of the output signal of the cell $s_{jv}^{c}(n-1) = 0$, $v \in \overline{1, S_j}$, $j \in \overline{1, N^{(1)}}$.

4. Output signal computation for each layer (forward run) $y_i^{in}(n-1) = x_{\mu-1,i}$,

$$y_j^{gin}(n) = f(net_j^{gin}(n)),\ j \in \overline{1, N^{(1)}},$$

$$f(s) = \frac{1}{1+e^{-s}},\ net_j^{gin}(n) = \sum_{i=0}^{N^{(0)}} w_{ij}^{in-gin}(n) y_i^{in}(n-1),$$

$$y_j^{g\varphi}(n) = f(net_j^{g\varphi}(n)),\ j \in \overline{1, N^{(1)}},$$

$$f(s) = \frac{1}{1+e^{-s}},\ net_j^{g\varphi}(n) = \sum_{i=0}^{N^{(0)}} w_{ij}^{in-g\varphi}(n) y_i^{in}(n-1),$$

$$\tilde{s}_{jv}^{c}(n) = g(net_{jv}^{c}(n)),\ v \in \overline{1, S_j},\ j \in \overline{1, N^{(1)}},$$

$$g(s) = 2\tanh(s),\ net_{jv}^{c}(n) = \sum_{i=0}^{N^{(0)}} w_{ijv}^{in-s}(n) y_i^{in}(n-1),$$

$$s_{jv}^{c}(n) = y_j^{g\varphi}(n) s_{jv}^{c}(n-1) + y_j^{gin}(n) \tilde{s}_{jv}^{c}(n),\ v \in \overline{1, S_j},\ j \in \overline{1, N^{(1)}},$$

$$y_j^{gout}(n) = f(net_j^{gout}(n)),\ j \in \overline{1, N^{(1)}},$$

$$f(s) = \frac{1}{1+e^{-s}},\ net_j^{gout}(n) = \sum_{i=0}^{N^{(0)}} w_{ij}^{in-gout}(n) y_i^{in}(n-1),$$

$$y_{jv}^{c}(n) = y_j^{gout}(n) h(s_{jv}^{c}(n)),\ v \in \overline{1, S_j},\ j \in \overline{1, N^{(1)}},$$

$$h(s) = \tanh(s),$$

$$y_j^{out}(n) = f(net_j^{out}(n)),\ j \in \overline{1, N^{(2)}},$$

$$f(s) = \frac{1}{1+e^{-s}},\ net_j^{out}(n) = b_j^{out} + \sum_{i=1}^{N^{(1)}} \sum_{v=1}^{S_i} w_{ivj}^{c-out}(n) y_{iv}^{c}(n).$$

It is considered that $w_{0j}^{in-gin}(n) = b_j^{gin}(n), y_0^{in}(n-1) = 1$, $w_{0j}^{in-g\varphi}(n) = b_j^{g\varphi}(n), y_0^{in}(n-1) = 1$,

$$w_{0j}^{in-s}(n) = b_j^{s}(n), y_0^{in}(n-1) = 1,\ w_{0j}^{in-gout}(n) = b_j^{gout}(n), y_i^{gout}(n-1) = 1$$

5. Calculation of ANN error energy

$$E(n) = \frac{1}{2}\sum_{i=1}^{N^{(2)}} e_i^2(n),\ e_i(n) = y_i^{out}(n) - d_{\mu-1,i}.$$

6. The adjustment of synaptic weights based on the generalized delta rule and the RTRL rule (backward movement)

$$w_{ivj}^{c-out}(n+1) = w_{ivj}^{c-out}(n) - \eta \frac{\partial E(n)}{\partial w_{ivj}^{c-out}(n)},\ i \in \overline{1, N^{(1)}},\ v \in \overline{1, S_j},\ j \in \overline{1, N^{(2)}},$$

$$b_j^{out}(n+1) = b_j^{out}(n) - \eta \frac{\partial E(n)}{\partial b_j^{out}(n)},\ j \in \overline{1, N^{(2)}},$$

$$w_{ij}^{in-gout}(n+1) = w_{ij}^{in-gout}(n) - \eta \frac{\partial E(n)}{\partial w_{ij}^{in-gout}(n)},\ i \in \overline{1, N^{(0)}},\ j \in \overline{0, N^{(1)}},$$

$$w_{ijv}^{in-s}(n+1) = w_{ijv}^{in-s}(n) - \eta e_{jv}^{c}(n) \frac{\partial s_{jv}^{c}(n)}{\partial w_{ijv}^{in-s}(n)},\ i \in \overline{1, N^{(0)}},\ v \in \overline{1, S_j},\ j \in \overline{0, N^{(1)}},$$

$$w_{ij}^{in-gin}(n+1) = w_{ij}^{in-gin}(n) - \eta \sum_{v=1}^{S_j} e_{jv}^{c}(n) \frac{\partial s_{jv}^{c}(n)}{\partial w_{ij}^{in-gin}(n)},\ i \in \overline{1, N^{(0)}},$$

$$v \in \overline{1, S_j},\ j \in \overline{0, N^{(1)}},$$

$$w_{ij}^{in-g\varphi}(n+1) = w_{ij}^{in-g\varphi}(n) - \eta \sum_{v=1}^{S_j} e_{jv}^{c}(n) \frac{\partial s_{jv}^{c}(n)}{\partial w_{ij}^{in-g\varphi}(n)},\ i \in \overline{1, N^{(0)}},$$

$$v \in \overline{1, S_j},\ j \in \overline{0, N^{(1)}},$$

where η – is a parameter that determines the learning rate (with a large learning rate is faster, but the risk of getting the wrong solution increases), $0 < \eta < 1$.

$$\frac{\partial E(n)}{\partial w_{ivj}^{c-out}(n)} = y_{iv}^{c}(n)\delta_j^{out}(n),$$

$$\frac{\partial E(n)}{\partial b_j^{out}(n)} = \delta_j^{out}(n),$$

$$\frac{\partial E(n)}{\partial w_{ij}^{in-gout}(n)} = y_i^{in}(n-1)\delta_j^{gout}(n),$$

$$\frac{\partial s_{jv}^{c}(n)}{\partial w_{ijv}^{in-s}(n)} = \begin{cases} \dfrac{\partial s_{jv}^{c}(n-1)}{\partial w_{ijv}^{in-s}(n-1)} y_j^{g\varphi}(n) + y_i^{in}(n-1) y_j^{gin}(n) g'(net_{jv}^{c}(n)), & n > 2 \\ y_i^{in}(n-1) y_j^{gin}(n) g'(net_{jv}^{c}(n)), & n = 2 \end{cases},$$

$$\frac{\partial s_{jv}^{c}(n)}{\partial w_{ij}^{in-g\varphi}(n)}=\begin{cases}\dfrac{\partial s_{jv}^{c}(n-1)}{\partial w_{ij}^{in-g\varphi}(n-1)}y_{j}^{g\varphi}(n)+ & n>2\\ +y_{j}^{in}(n-1)s_{jv}^{c}(n-1)f'(net_{j}^{\varphi}(n)), & \\ y_{j}^{in}(n-1)s_{jv}^{c}(n-1)f'(net_{j}^{\varphi}(n)), & n=2\end{cases},$$

$$\frac{\partial s_{jv}^{c}(n)}{\partial w_{ij}^{in-gin}(n)}=\begin{cases}\dfrac{\partial s_{jv}^{c}(n-1)}{\partial w_{ij}^{in-gin}(n-1)}y_{j}^{g\varphi}(n)+ & n>2\\ +y_{j}^{in}(n-1)g(net_{jv}^{c}(n))f'(net_{j}^{gin}(n)), & \\ y_{j}^{in}(n-1)g(net_{jv}^{c}(n))f'(net_{j}^{gin}(n)), & n=2\end{cases},$$

$$e_{jv}^{c}(n)=y_{j}^{gout}(n)h'(s_{jv}^{c}(n))\sum_{l=1}^{N^{(2)}}w_{jvl}^{c-out}(n)\delta_{l}^{out}(n),$$

$$\delta_{j}^{out}(n)=f'(net_{j}^{out}(n))(y_{j}^{out}(n)-d_{\mu-1,j}),$$

$$\delta_{j}^{gout}(n)=f'(net_{j}^{gout}(n))\sum_{v=1}^{S_j}h(s_{jv}^{c}(n))\sum_{l=1}^{N^{(2)}}w_{jvl}^{c-out}(n)\delta_{l}^{out}(n).$$

7. Terminating condition check

If $(n-1)\bmod P>0$, then $\mu=\mu+1$, $n=n+1$, go to step 4.

If $(n-1)\bmod P=0$ and $\frac{1}{P}\sum_{s=1}^{P}E(n-P+s)>\varepsilon$, then $n=n+1$, go to step 2.

If $(n-1)\bmod P=0$ and $\frac{1}{P}\sum_{s=1}^{P}E(n-P+s)<\varepsilon$, then terminate.

4. Development of a metaheuristic method for identifying the parameters of a neural network approximation model based on long short-term memory with a forgetting gate

For the LSTM neural network model, due to its recurrence, there is no parametric identification in the batch mode, which reduces the learning rate [29-31, 32-34]. To eliminate this shortcoming, this paper proposes a modified method for optimizing the wasp colony.

For this method, a criterion (goal function) is used, which means the choice of such values of the vector of parameters (thresholds and weights) that deliver the minimum of the mean square error (the difference between the model output and the test output).

$$F=\frac{1}{P}\sum_{\mu=1}^{P}(y_{\mu}^{out}-d_{\mu})^{2}\rightarrow\min_{\theta},$$

where y_{μ}^{out} – μ-th model response,

d_{μ} – μ-th test response.

Wasp colony optimization is based on the social behavior of wasps. The position of each wasp in space corresponds to a solution. The goal of the algorithm is to locate the optima in a multidimensional space using all the pivots. At each iteration, the strongest wasps will be determined based on the goal function. These wasps and their environment are modified. Weak wasps die and are replaced by wasps produced by crossing the best wasps. The weakest wasps are replaced with random wasps. It is proposed to decrease the parameter for generating a new position as the iteration number increases, in order to ensure the convergence of the method, in contrast to the traditional method.

1. Initialization

1.1. To set a parameter δ for generating a new position, and $0 < \delta < 1$.

1.2. To set the maximum number of iterations N, population size K, the maximum number of best wasps Q, neighborhood size Z, the length of the wasp position vector M (corresponds to the number of parameters of the neural network), the minimum and maximum values for the position vector $x_j^{\min}, x_j^{\max}$, $j \in \overline{1,M}$.

1.3. To generate randomly a vector of the best position

$$x^* = (x_1^*,...,x_M^*),\ x_j^* = x_j^{\min} + (x_j^{\max} - x_j^{\min})U(0,1),$$

where $U(0,1)$ – a function returning a uniformly distributed random number in a range $[0,1]$.

1.4. To create an initial population $P^{(0)}$

1.4.1. The wasp number $k = 1$, $P^{(0)} = \varnothing$

1.4.2. To generate randomly a position vector x_k

$$x_k = (x_{k1},...,x_{kM}),\ x_{kj} = x_j^{\min} + (x_j^{\max} - x_j^{\min})U(0,1)$$

1.4.3. $P^{(0)} = P^{(0)} \bigcup \{x_k\}$,

1.4.4. If $k \le K$, then $k = k + 1$, go to step 1.4.2

2. Iteration number $n = 0$.

3. To sort $P^{(n)}$ by target function, i.e., $F(x_k) < F(x_{k+1})$

4. To determine the best wasp in terms of the goal function

$$k^* = \arg\min_k F(x_k)$$

5. If $F(x_{k^*}) < F(x^*)$, then $x^* = x_{k^*}$

6. To apply the crossover operator Q to the best (first) wasps

6.1. The wasp number $k = 1$, $P^{(n+1)} = \varnothing$

6.2. To select randomly the -th wasp different from the k-th wasp from the best wasps, i.e. $m = round(1 + (Q-1)U(0,1))$, $k \ne m$,

where $round()$ – is the function that rounds a number to the nearest whole number.

6.3. To perform a medium crossover over x_k and x_m, and get the vector $\tilde{x}_k$

$$\tilde{x}_k = (\tilde{x}_{k1},...,\tilde{x}_{kM}),\ \tilde{x}_{kj} = 0.5(x_{kj} + x_{mj})$$

6.4. If $F(x_k) < F(\tilde{x}_k)$, then $\tilde{x}_k = x_k$

6.5. $P^{(n+1)} = P^{(n+1)} \bigcup \{\tilde{x}_k\}$

6.6. If $k < Q$, then $k = k + 1$, go to step 6.2

7. To create a neighborhood for each of the Q best wasps and apply a crossover operator to these neighborhoods.

7.1. The wasp number $k = 1$

7.2. To create a neighborhood U_{x_k} for the k-th wasp

7.2.1. $z = 1$

7.2.2. Solution generation u_z from decision x_k

7.2.2.1. $u_{zj} = x_{kj} + \delta\left(\frac{N-n}{N}\right)(x_j^{\max} - x_j^{\min})(-1 + 2U(0,1))$, $j \in \overline{1,M}$

7.2.2.2. $u_{zj} = \max\{x_j^{\min}, u_{zj}\}$, $u_{zj} = \min\{x_j^{\max}, u_{zj}\}$, $j \in \overline{1,M}$

7.2.3. If $u_z \notin U_{x_k}$, then $U_{x_k} = U_{x_k} \bigcup \{u_z\}$, $z = z + 1$

7.2.4. If $z \le Z$, then go to step 7.2.2

7.3. To apply crossover to neighborhood of k-th wasp

7.3.1. $z = 1$

7.3.2. To choose randomly different from the z-th wasp m-th axis from the neighborhood U_{x_k}, i.e. , $m = round(1 + (Z - 1)U(0,1))$, $z \ne m$

7.3.3. To perform a medium crossover over u_z and u_m, and get a wasp $\tilde{x}_{Q+(k-1)Z+z}$

$$\tilde{x}_{Q+(k-1)Z+z} = (\tilde{x}_{Q+(k-1)Z+z,1},...,\tilde{x}_{Q+(k-1)Z+z,M}),$$

$$\tilde{x}_{Q+(k-1)Z+z,j} = 0.5(\hat{x}_{kj} + \hat{x}_{mj})$$

7.3.4. If $F(u_z) < F(\tilde{x}_{Q+(k-1)Z+z})$, then $\tilde{x}_{Q+(k-1)Z+z} = u_z$

7.3.5. $P^{(n+1)} = P^{(n+1)} \bigcup \{\tilde{x}_{Q+(k-1)Z+z}\}$

7.3.6. If $z \le Z$, then $z = z + 1$, go to step 7.3.2

7.4. If $k < Q$, then $k = k + 1$, go to step 7.2

8. To apply the substitution operator, i.e., replace $K - (Q + Q \cdot Z)$ the worst wasps with random wasps

8.1. The wasp number $k = Q + Q \cdot Z + 1$

8.2. To select randomly two different wasps l and m from the best wasps, i.e. $l = round(1 + (Q - 1)U(0,1))$, $m = round(1 + (Q - 1)U(0,1))$, $l \ne m$

8.3. To execute crossover over x_m and x_l, and get wasp $\tilde{x}_k$

$$\tilde{x}_k = (\tilde{x}_{k1},...,\tilde{x}_{kM}), \ \tilde{x}_{kj} = 0.5(x_{mj} + x_{lj})$$

8.3. $P^{(n+1)} = P^{(n+1)} \bigcup \{\tilde{x}_k\}$

8.4. If $k < K$, then $k = k + 1$, go to step 8.2

9. If $n < N - 1$, then $n = n + 1$, go to step 3

The result is x^*.

Numerical research

The numerical study of the proposed approach was carried out using the TensorFlow module and its Keras submodule. The Pandas module was used for tabular data I/O and missing value recovery through linear interpolation.

The World Bank economic indicators database (https://databank.worldbank.org/home.aspx) was used in the work. The economic indicators of 145 countries for 10 years were used. The size of the initial sample was equal to 1450. The initial sample was divided into training (P=928), verification (232) and test (290).

The length of the input layer $N^{(0)}$ for all networks was 5. The length of the hidden layer $N^{(1)}$ for the LSTM-based neural network was $N^{(0)}$. We propose a parametric identification of the LSTM neural network model based on the Adam method in batch mode to increase the learning rate. The parameter η, which determines the learning rate, was 0.001.

There is no parametric identification of these models based on the Adam method in batch mode for the LSTM neural network model, due to its recurrence, which reduces the learning rate. We propose to use metaheuristics to eliminate this shortcoming [33; 34].

It is proposed to decrease the parameter for generating a new position as the iteration number increases, in order to ensure the convergence of the method, in contrast to the traditional wasp colony optimization method. For the modified wasp colony optimization method, the parameter for generating a new position was 0.1, the population K size was 120, the maximum number of best wasps Q was 10, and the neighborhood size Z was 10. The maximum number of iterations N was 100.

Research results

To assess the quality of the approximation, the determination coefficient is calculated in the form

$$R^2 = 1 - \frac{\sum_{\mu=1}^{P} (d_\mu - f(\mathbf{x}_\mu, \mathbf{w}))^2}{\sum_{\mu=1}^{P} (d_\mu - \bar{d})^2}, \; R^2 \in [0,1],$$

$$\bar{d} = \sum_{\mu=1}^{P} d_\mu$$

The results of comparing the quantitative characteristics of the proposed methods for the parametric identification of neural network models are presented in Table 1, where *M* – is the number of parameters for a specific neural network.

Table 1

Comparison of the qualitative characteristics of the proposed methods for parametric identification of neural network models

Neural networks	Characteristics of the methods of parametric identification	
	Coefficient of determination	Computational complexity
LSTM-based neural network without the use of metaheuristics and GPU	0.785	~*NPM*, *M*=226 (forward / reverse)
LSTM based neural network using metaheuristics and GPU	0.835	~*NK*

An LSTM-based neural network using metaheuristics and a GPU has a higher determination coefficient, i.e., the highest approximation accuracy due to random search, and lower computational complexity due to the use of GPU.

Conclusions

The urgent task of using new approaches to analyze the indicators of foreign direct investment and macroeconomic indicators that affect the volume of their attraction to a particular country in the world economy was solved by a connectionist-metaheuristic approach.

The proposed connectionist-metaheuristic approach makes it possible to increase the approximation efficiency by:

- simplifying structural identification by using only one hidden layer of neural network models;
- reducing the computational complexity of parametric identification by using multi-agent metaheuristics for recurrent neural network models;
- high approximation accuracy due to the use of recurrent connections in neural network models;
- the possibility of non-linear approximation through the use of neural network models;
- lack of requirements for a priori knowledge of the distribution, weak correlation, homogeneity and preliminary selection of factors.
- resistance to data noise and data incompleteness due to metaheuristic parametric identification of recurrent neural networks models.

The proposed approach makes it possible to expand the scope of application of approximation methods based on the connectionist approach and metaheuristics, which is confirmed by its adjustment for an economic problem and contributes to an increase in the efficiency of intelligent computer systems related to the economy.

Prospects for further research are the study of the proposed approach for various problems of artificial intelligence, as well as the creation of methods for analyzing economic.

The research was carried out in accordance with the priority direction of the development of science and technology in Ukraine "Information and Communication Technologies" and contain some results of the research "Development of models and methods of biometric identification of people" (state registration number 0119U002860).

References

1. Baillie R.T., Kapetanios G., Papailias F. Modified information criteria and selection of long memory time series models. *Computational Statistics and Data Analysis*. 2014. Vol. 76. P. 116–131

2. Bidyuk P., Prosyankina-Zharova T., Terentiev O. Modelling nonlinear nonsta-tionary processes in macroeconomy and finances. *Advances in Intelligent Systems and Computing* .2019. Vol. 754. P. 735-745.

3. Wu L., Liu S., & Yang,Y. Grey double exponential smoothing model and its application on pig price forecasting in China. *Applied Soft Computing*. 2016. Vol. 39. P. 117–123.

4. Srivastava R.B., Srivastava P.K. Comparison of Lagrange's and Newton's interpolating polynomials. *Journal of Experimental Sciences*. 2012. Vol. 3(1): 01-04.

5. Yang Y., Gordon Sh.P. Visualizing and Understanding the Components of Lagrange and Newton Interpolation. *PRIMUS*. 2016. Vol. 26:1. P. 39-52

6. Burden R., Faires J. Numerical Analysis. Boston, MA: Brooks/Cole, 2010.

7. Li, R.Y.M., Fong S.; Chong K.W.S. Forecasting the REITs and stock indices: Group Method of Data Handling Neural Network approach. *Pacific Rim Property Research Journal*. 2017. Vol. 23 (2). P. 123–160.

8. Fong S., Nannan Z., Wong R.K., Yang X.S. Rare events forecasting using a residual feedback GMDH neural network. *Data Mining Workshops (ICDMW)*: 12th International Conference on Data Mining Workshops, IEEE Computer Society Washington, DC, USA, 2012. P. 464–473

9. Andras P. Function approximation using combined unsupervised and supervised learning. *IEEE Transactions on Neural Networks and Learning Systems*. 2014. Vol. 25. P.495-505.

10. Hastie T., Tibshirani R., Friedman J. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer, 2009.

11. Schmidhuber J. Deep learning in neural networks: An overview. *Neural Networks*. 2015. Vol. 61. P. 85–117.

12. Sivanandam S.N., Sumathi S., Deepa S.N. Introduction to Neural Networks using Matlab 6.0. New Delhi: The McGraw-Hill Comp., Inc., 2006. 660 p.

13. Nakib A., Talbi El-G. Metaheuristics for Medicine and Biology. Berlin: Springer-Verlag, 2017. 211 p.

14. Yang X.-S. Nature-inspired Algorithms and Applied Optimization. Charm: Springer, 2018. 330 p.

15. Subbotin S., Oliinyk A., Levashenko V., Zaitseva E. Diagnostic Rule Mining Based on Artificial Immune System for a Case of Uneven Distribution of Classes in Sample. *Communications*. 2016. Vol.3. P.3-11.

16. Blum C., Raidl G. R. Hybrid Metaheuristics. Powerful Tools for Optimization. Charm: Springer, 2016. 157 p.

17. Bozorg-Haddad O., Solgi M., Loaiciga H. Meta-heuristic and Evolutionary Algorithms for Engineering Optimization. Hoboken, New Jersey: Wiley & Sons, 2017. 293 p.

18. Chopard B., Tomassini M. An Introduction to Metaheuristics for Optimization. New York: Springer, 2018. 230 p.

19. Radosavljević J. Metaheuristic Optimization in Power Engineering. New York: The Institution of Engineering and Technology, 2018. 536 p.

20. Alba E., Nakib A., Siarry P. Metaheuristics for Dynamic Optimization. – Berlin: Springer-Verlag, 2013. – 398 p.

21. Du K.-L., Swamy K.M.S. Neural Networks and Statistical Learning, London: Springer-Verlag, 2014. 824 p.

22. Haykin S. Neural Networks and Learning Machines. Upper Saddle River, New Jersey: Pearson Education, Inc., 2009. 906 p.

23. Warsito B., Santoso R.R., Suparti S.S., Yasin H. Cascade Forward Neural Network for Time Series Prediction. *Journal of Physics Conference Series*. 2018. Vol. 1025, № 1. P.1-8.

24. Aggarwal C.C. Neural Networks and Deep Learning. Cham, Switzerland: Springer, 2018. 497 p.

25. Sundermeyer M., Schluter R., Ney H. Lstm neural networks for language modeling. *Thirteenth Annual Conference of the International Speech Communication Association*, 2012.

26. Potash P., Romanov A., Rumshisky A. Ghostwriter: using an lstm for automatic rap lyric generation. *In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. 2015. P. 1919– 1924.

27. Chung J., Gulcehre C., Cho K., Bengio Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555, 2014.

28. Dey R., Salem F.M. Gate-Variants of Gated Recurrent Unit (GRU) Neural Networks. arXiv:1701.05923, 2017.

29. Pinto C., Runkler T.A., Sousa J.M. Wasp Swarm Algorithm for Dynamic MAX-SAT Problems. *Proceedings of the 8th international conference on Adaptive and Natural Computing Algorithms*. 2007. Part I. P. 350-357.

30. Engelbrecht A.P. Computational Intelligence: an introduction. Chichester, West Sussex: Wiley & Sons, 2007. 630 p.

31. Du K.-L., Swamy M.N.S. Search and Optimization by Metaheuristics. Techniques and Algorithms Inspired by Nature. Charm: Springer, 2016. 434 p.

32. Roy R. Ph., Thiruthuvanathan M. M. Enhanced Stock Market Prediction Using Hybrid LSTM Ensemble. *Proceedings of Congress on Intelligent Systems*. 2021. Vol. 1, P. 49-61.

33. Kumar S., Mahapatra R.P. Analytical Analysis of Two-Warehouse Inventory Model Using Particle Swarm Optimization. *Proceedings of Congress on Intelligent Systems.* 2021. Vol. 2. P. 215-226

34. Sharma A., Sharma H., Khandelwal A., Sharma N. Designing Controller Parameter of Wind Turbine Emulator Using Artificial Bee Colony Algorithm. *Proceedings of Congress on Intelligent Systems*. 2020, P. 143-151

Eugene Fedorov **Євген Федоров**	Doctor of Technical Science, Professor of the Statistics and Applied Mathematics Department, Cherkasy State Technological University, Ukraine. e-mail: fedorovee75@ukr.net https://orcid.org/0000-0003-3841-7373 Scopus Author ID: 47161155900, https://scholar.google.com/citations?user=hbzfMWcAAAAJ	Доктор технічних наук, професор кафедри статистики та прикладної математики Черкаського державного технологічного університету, Україна.
Maryna Leshchenko **Марина Лещенко**	PhD, Associate Professor of the International Economics and business Department, Cherkasy State Technological University, Ukraine. e-mail: mari.leshchenko@gmail.com https://orcid.org/0000-0002-0210-9582, Scopus Author ID: 57210600995	Кандидат економічних наук, доцент кафедри міжнародної економіки та бізнесу Черкаського державного технологічного університету, Україна.
Liubov Kibalnyk **Любов Кібальник**	PhD, Professor of the Economics and Business Modelling Department, Cherkasy Bohdan Khmelnytskyi National University, Ukraine. e-mail: liubovkibalnyk@gmail.com	Кандидат економічних наук, професор кафедри економіки та бізнес-моделювання Черкаського національного університету імені Богдана Хмельницького, Україна.

	https://orcid.org/0000-0001-7659-5627, Scopus Author ID: 57210337198, https://scholar.google.com/citations?user=tPw46YYAAAAJ&hl=uk	
Hanna Danylchuk **Ганна Данильчук**	PhD, Associate Professor of Economics and Business Modeling Department, Cherkasy Bohdan Khmelnytsky National University, Ukraine. e-mail: abdanilchuk@gmail.com https://orcid.org/0000-0002-9909-2165, Scopus Author ID: 57209541581	Кандидат економічних наук, доцент кафедри економіки та бізнес-моделювання Черкаського національного університету імені Богдана Хмельницького, Україна.