

Received December 1, 2021, accepted December 13, 2021, date of publication January 14, 2022, date of current version January 26, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3143106

Maximum Visibility: A Novel Approach for Time Series Forecasting Based on Complex Network Theory

FILIFE RODRIGUES DE SOUZA MOREIRA¹, FILIFE ALVES NETO VERRI²,
AND TAKASHI YONEYAMA³

¹Fundamental Sciences Division (IEF), Aeronautics Institute of Technology (ITA), São José dos Campos 12228-900, Brazil

²Computer Science Division (IEC), Aeronautics Institute of Technology (ITA), São José dos Campos 12228-900, Brazil

³Electronic Engineering Division (IEE), Aeronautics Institute of Technology (ITA), São José dos Campos 12228-900, Brazil

Corresponding author: Filipe Rodrigues De Souza Moreira (filipe@ita.br)

This research was sponsored by The Brazilian Air Force through the Graduate Program on Operational Application (PPGAO), at the Technological Institute of Aeronautics (ITA).

ABSTRACT This article presents Maximum Visibility Approach (MVA), a new time series forecasting method based on the Complex Network theory. MVA initially maps time series data into a complex network using the visibility graph method. Then, based on the similarity measures between the nodes in the network, MVA calculates the one-step-ahead forecasts. MVA does not use all past terms in the forecasting process, but only the most significant observations, which are indicated as a result of the autocorrelation function. This method was applied to five different groups of data, most of them showing trend characteristics, seasonal variations and/or non-stationary behavior. We calculated error measures to evaluate the performance of MVA. The results of statistical tests and error measures revealed that MVA has a good performance compared to the accuracy obtained by the benchmarks considered in this work. In all cases, MVA surpassed other forecasting methods in Literature, which confirms that this work will contribute to the field of time series forecasting not only in the theoretical aspect, but also in practice.

INDEX TERMS Time series forecasting, complex networks, visibility graph, forecasting model.

I. INTRODUCTION

Among the strands in the field of Time Series Analysis is the creation of methods to study time-correlated data and make predictions [1]. This is a dynamic research area that has attracted the attention of the scientific community over the past few decades [2]. Applications of time series modeling and analysis encompass several fields of science, including: medicine [3], robotics [4], cyber defense [5], defense strategy [6], army mission analysis [7], finance [8], [9], social sciences [10], economics [11], seismology [12] and criminology [13]. In order to make estimates of the future terms of a time series, it is necessary to make the hypothesis that each observed data is somehow correlated with past data. Thus, it is natural to establish mathematical models that try to

The associate editor coordinating the review of this manuscript and approving it for publication was Dost Muhammad Khan.

explain such correlations and describe the observed data as a function of time.

Autoregressive integrated moving average (ARIMA) is one of the most popular forecasting methods in the literature [2], [14], [15]. This model is attributed to George Box and Gwilym Jenkins [15] and is essentially a linear regression model that describes movements of a stationary and univariate time series, that is, time series free from trends [16]–[18]. ARIMA is also a combination of other models, such as the autoregressive (AR) [16], moving average (MA) [16], [17], [19] and the autoregressive moving average (ARMA) [16], [20]. In the case of a time series with seasonality, SARIMA, a variation of ARIMA, can be applied [21]. There are also some extensions that are applicable to nonlinear [22] or non-Gaussian [20] time series.

Data mining is defined as the “process of automatic extraction of knowledge from large databases, using machine

learning, pattern recognition, database techniques, and statistics” [23]. According to [24] the six pure Machine Learning methods submitted in the M4-competition performed poorly, none of them outperforming the combination approaches and only one achieved accuracy better than Naive2. Under the influence of forecasting competitions, several combined methods have been created in the quest to achieve better accuracy [24]. See these examples: in [25] a hybrid technique based on bagging approach is proposed to achieve more accurate forecasts; [26] shows a combination between ARIMA and artificial neural network (ANN) to predict internet traffic data; [27] calibrated an ensemble structure based on ARIMA, multilayer perceptron (MLP) and long short-term memory (LSTM) with the objective of predict healthcare expenditures; and [28] which presents some hybrid approaches, including mixing the exponential smoothing model and ANN in order to forecast the price of agricultural products.

A wealth of content can be found in the literature on time series data mining, some of which are based on complex networks [29]–[32], [33]. In [30], one can find an approach for time-series forecasting based on complex network analysis. Specifically, this method is based on node similarities [34] extracted from a network constructed using the visibility graph method [35]. In this article, it is referred to as the Mao-Xiao approach (MXA). Its accuracy is better than other methods for a variety of applications including methods using fuzzy logic [36], [37], bandwidth interval-based forecasting methods [38], and hybrid fuzzy logic and visibility graph approach [39].

This study presents Maximum Visibility Approach (MVA) which is a new time series forecasting technique based on the complex network theory. This presentation follows the subsequent steps. Initially, some basic concepts, necessary to understand the method proposed here, are revisited and then MVA is described. The results of the tests carried out with five different datasets are then presented, with the objective of comparing the performances obtained by the MVA and the benchmarks: a state-of-the-art Mao-Xiao approach (MXA) [30], dynamic auto-ARIMA, support vector machine (SVM) [40], long short-term memory (LSTM) [41], multilayer perceptron (MLP) [27], hybrid additive ARIMA-ANN [26], hybrid additive ETS-ANN [42] and naive estimation.

II. BASIC THEORY

This section provides a brief presentation of the basic concepts and techniques required to describe the MVA.

A. VISIBILITY GRAPH

According to [35], the visibility graph transforms a time series into a complex network. Each time-series observation is associated with a vertex of a complex network. Let (t_i, y_i) and (t_k, y_k) , be two distinct points of a time series. Take any other point (t_u, y_u) between them. The points (t_i, y_i) and (t_k, y_k) are considered visually related if the following rule is

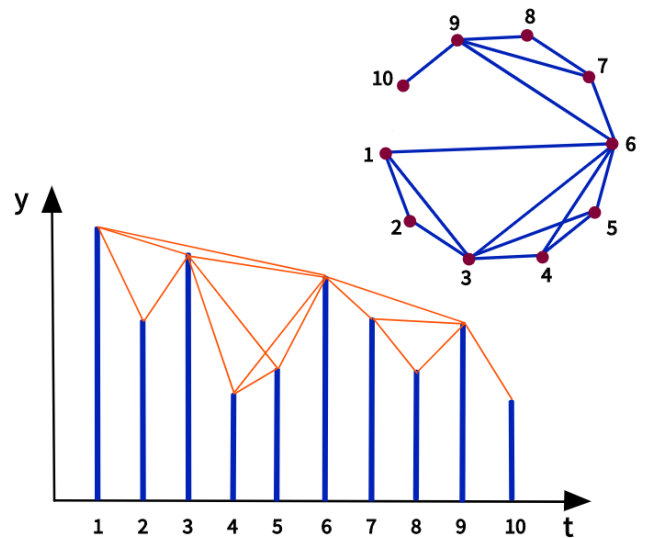


FIGURE 1. Scheme for visibility graph approach: A 10-elements time series is mapped into a graph where each node represents a bar, and each edge represents the connection between two bars. Source: Drawn by the authors.

satisfied:

$$y_u < y_k + \left(\frac{t_u - t_k}{t_k - t_i} \right) (y_k - y_i). \quad (1)$$

In Fig.1, the i th bar relates to the i th network node. The rule described by equation (1) is illustrated in Fig.1. It can be seen that each connection between two bars corresponds to a link between the two respective nodes in the complex network.

The visibility graph theory has established itself as an efficient approach for probing the dynamics underlying real complex systems from time series [31]. According to [35], the network obtained from this particular mapping inherits several properties of the time series, and its study reveals nontrivial information about the series itself. This mapping makes the structure of the time series be conserved in the graph topology: periodic series convert into regular graphs, random series into random graphs, and fractal series into scale-free graphs [35].

B. PREVIOUS WORK: THE MAO-XIAO APPROACH

The forecasting method presented by [30], here in this paper, called the Mao-Xiao Approach, is an innovative methodology to provide time series forecasting based on complex networks. Essentially, a time series with n observations is transformed into a network through the visibility graph method [35]. Second, the Node Similarity Matrix (NSM), that is, the matrix that contains the similarity measures between each two pairs of nodes, is calculated by the method explained in [34]. The last line of the NSM gives the similarity between the last node and each of the first $n - 1$ nodes, then the node k , with the highest similarity value, is taken and a linear equation is calculated considering both nodes: $N_n(t_n, y_n)$ and $N_k(t_k, y_k)$. The first estimation of the $(n + 1)$ -th point is

calculated by (2):

$$\tilde{y}_{n+1} = y_n + \frac{(y_n - y_k)}{(t_n - t_k)}(t_{n+1} - t_n). \quad (2)$$

Next, a refining is proposed according to (3):

$$\hat{y}_{n+1} = w_n \cdot y_n + w_{n+1} \cdot \tilde{y}_{n+1}. \quad (3)$$

The weights w_n and w_{n+1} are, respectively, calculated by:

$$w_{n+1} = \frac{d_{k \rightarrow n}}{d_{k \rightarrow n+1}} \quad (4)$$

and

$$w_n = \frac{d_{n \rightarrow n+1}}{d_{k \rightarrow n+1}}. \quad (5)$$

In (5) and (4), the notation $d_{k \rightarrow n}$ refers to the horizontal distance between observations n and k [39], therefore, considering the points (t_k, y_k) and (t_n, y_n) , hence $d_{k \rightarrow n} = t_n - t_k$.

C. SIMILARITY BETWEEN TWO NODES IN A GRAPH

Two nodes in a graph are said to be similar if they share the same predefined characteristics. Some techniques produce measures of similarities between vertices of a network and according to [34], [43], [44] such measures are the basis of the link prediction theory. For instance, can be cited similarity based on random walk process [30], [33], [34], the cosine similarity [45], the Jaccard similarity [46], and the inverse log-weighted similarity [47].

In this work we use the Dice similarity [48]. In order to present its definition, consider $n(\Theta)$ as the cardinal of the set Θ . In the Complex Network context, given any two vertices of a network, the Dice similarity [48] is calculated taking twice the number of common neighbors over the sum of the degrees of both vertices,

$$s_{ij}^{DICE} = \frac{2 \cdot n(N(v_i) \cap N(v_j))}{\rho(v_i) + \rho(v_j)}, \quad (6)$$

where $N(v_i)$ relates to the neighbor set of the vertex v_i and $\rho(v_i)$ is the vertex degree of the vertex v_i . It is not difficult to conclude that the Dice index range is from zero to 1. The result is zero if both vertices does not share any neighbor and it is 1 if both neighbors sets are equal.

D. SLIDING WINDOW

Both the autocorrelation function (ACF) and the partial autocorrelation function (PACF) quantify the influence of past observations on the current value of the time series. Additionally, ACF or PACF can indicate whether previous values cause, or not, a significant impact on the current observation. Considering that some past values may not be statistically significant to predict the next term of the time series, it is reasonable to disregard such data in the forecasting process. The set of past observations considered in the forecast is defined as a window.

Given the set $Y = \{y_1, y_2, \dots, y_n\}$, a sliding window, Z_i , over Y is a subset with fixed number of elements, k , which,

during the simulation, changes at each iteration i , according to the following rule:

$$\begin{aligned} Z_1 &= \{y_1, y_2, \dots, y_k\}, \\ Z_2 &= \{y_2, y_3, \dots, y_{k+1}\}, \\ &\dots \\ Z_{n-k+1} &= \{y_{n-k+1}, y_{n-k+2}, \dots, y_n\}. \end{aligned} \quad (7)$$

The sliding window process is based on the methodology walk-forward validation presented by [49].

E. STATISTICAL HYPOTHESIS TESTING

In order to analyze the quality of the prediction, it is necessary to use hypothesis tests. Initially they are applied to investigate the characteristics of the original time series and finally to evaluate the characteristics of the forecast residuals.

With the purpose of testing whether the data is level or trend stationary we apply the Augmented Dickey-Fuller (ADF) test [50] and confirm the obtained results with the Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test [51]. To check whether the data is independently distributed we use the Ljung-Box test [52]. This test is essential to verify whether ARIMA residuals have white noise signal characteristics. With the objective of identifying the data seasonality we perform the Webel-Ollech [53], which is basically a combination of the methods modified QS [54] and Friedmann [55] tests.

F. FORECASTING PERFORMANCE INDICATORS

Given the set $\{y_1, y_2, \dots, y_m\}$ and the respective predicted values $\{\hat{y}_1, \hat{y}_2, \dots, \hat{y}_m\}$, the performance indicators are

- 1) Mean Absolute Error (MAE),

$$MAE = \frac{1}{m} \sum_{t=1}^m |\hat{y}_t - y_t|, \quad (8)$$

- 2) Mean Absolute Percentage Error (MAPE),

$$MAPE = \frac{1}{m} \sum_{t=1}^m \left| \frac{\hat{y}_t - y_t}{y_t} \right|, \text{ and} \quad (9)$$

- 3) Root Mean Squared Error (RMSE),

$$RMSE = \sqrt{\frac{1}{m} \sum_{t=1}^m [\hat{y}_t - y_t]^2}. \quad (10)$$

The performance indicators were used to compare the simulation results obtained in this work.

III. MVA: NEW FORECASTING METHOD

The Maximum Visibility approach (MVA) is a new method based on the complex network theory, inspired by the technique proposed in [30], which, in this work, is called the Mao-Xiao approach (MXA). The first step of the MVA considers the autocorrelation function (ACF) related to the given time series. The ACF gives the correlation between the last observation and any k th observation, hence this step

guarantees that only the statistically significant elements will be used in the forecasting task.

The next step is to transform the given time series into a complex network. The literature presents some methods which are able to map a given time series into a complex network: visibility graph [35], horizontal visibility graph [56], multiscale limited penetrable horizontal visibility graph [31] and phase space reconstruction [57]. In this work, we chose the visibility graph approach due to the technical reasons described in the section II-A, but mostly due its ease of implementation and low computational cost.

After constructing the network, we calculated the node similarity matrix (NSM). In this matrix, each $s_{i,j}$ represents the similarity calculated between nodes N_i and N_j .

Let $\hat{y}_{(n+1)/k}$ be the pre-estimator for the time-series next term, that is, $y_{(n+1)}$, considering only the influence of y_k . This pre-estimator is calculated using a linear equation, based on the connection between both observations: y_k and y_n . We introduced a multiplicative parameter, α , to the original angular coefficient, as shown in (11). Consider the points: (t_k, y_k) and (t_n, y_n) . The expression that gives $\hat{y}_{(n+1)/k}$ is:

$$\hat{y}_{(n+1)/k} = y_n + \alpha \left(\frac{y_n - y_k}{t_n - t_k} \right) \cdot (t_{(n+1)} - t_n), \quad (11)$$

where α is divided into two parts: a constant $K \geq 0$ and $\gamma_{n,k}$, which are the autocorrelation between y_k and y_n .

$$\alpha = \gamma_{n,k} - K. \quad (12)$$

Observing the equation (11), the closer α is to zero, the greater is the effect of y_n on $\hat{y}_{(n+1)}$, approximating the forecasting result to that obtained from the naive estimation.

We define $Y_{(n+1)}^m$ as the vector that contains all pre-estimators for $y_{(n+1)}$, calculated through the influence of the first m lags. $Y_{(n+1)}^m$ is written as:

$$Y_{(n+1)}^m = (\hat{y}_{(n+1)/(n-m+1)}, \dots, \hat{y}_{(n+1)/(n-1)}). \quad (13)$$

Let $s_{i,n}$ be the similarity value between nodes N_i and N_n , associated to the time-series terms y_i and y_n , respectively and calculated based on the Dice similarity, according to described in the equation (6). If the last m nodes, prior to y_n are considered in order to estimate $y_{(n+1)}$, then the NSM has dimension $m + 1$ and the last row of the NSM contains the similarities between all last m observations and y_n . We define S_n^m as the vector formed by the first m elements of the last row of the NSM. Therefore, S_n^m is seen as:

$$S_n^m = (s_{n-m+1,n}, s_{n-m+2,n}, \dots, s_{n-1,n}). \quad (14)$$

It is possible to find $s_{k,n} = s_{i,n}$ with $k \neq i$, therefore more than one node can share the highest level of similarity with y_n . Define $I_{S_n^m}$ as the vector composed by the indexes of the elements contained in S_n^m which present the highest level of similarity to y_n . The one-step-ahead MVA estimator, $\hat{y}_{(n+1)}$ is calculated as:

$$\hat{y}_{(n+1)} = \text{MAX} \left\{ \hat{y}_{(n+1)/k} \in Y_{(n+1)}^m \right\} + h(E_n), \forall k \in I_{S_n^m}, \quad (15)$$

where $h(E_n)$ is a non-linear function of the error obtained in the estimation of y_n , that is $E_n = \hat{y}_{(n)} - y_n$. The non-linear function $h(E_n)$ is given by:

$$h(E_n) = -E_n e^{-K \cdot J |E_n|}, \quad (16)$$

where J is positive constant. For some time series, $h(E_n)$ is considered zero. The decision of not include $h(E_n)$ in (15) is taken by observing the results achieved by MVA in the validation phase.

A. THE MODEL PARAMETERS

According to presented in the equations (12) and (16) there are two parameters, K and J , respectively that must be defined before the forecasting process. In order to find the best values for K and J capable of providing the best fit in the data we proposed a partitioning of the original time series into three sets: training, validation and testing sets. The simulation has to be performed on the training and validation sets many times as necessary and after that, the best values of K and J are applied to the testing set to ascertain the method's ability to make predictions.

In order to generate the results presented in the section IV we kept J constant and variate K , until find the best sum of squared error (SSE) considering the validation set. For example, initially take K from 1 to 6, using step of 1. Suppose $K = 3$ gives the best SSE in the validation set. Secondly refine the search considering K from 2 to 4, in steps of 0.1. Suppose $K = 2.6$ gives the smallest SSE in the validation set. Now refine the search from 2.5 and 2.7, using steps of 0.01. Suppose $K = 2.66$ gives the smallest SSE. Refine the pursuance one more time, searching from 2.65 to 2.67, with steps of 0.001. Register the value of K associated to the smallest SSE. This procedure is repeated considering other values of J .

We started with $J \in \{1, 10, 100, 500\}$. For most cases, higher values of J produces the effect of not considering the non-linear term. After testing some values of J we create a list of the obtained SSE's. Given the randomness of the real time series, there is no guarantee that the pair (J, K) related to the smallest SSE achieved in the training and validation phase will produce the smallest SSE in the testing set, however, we kept this procedure to find the pair (J, K) to be used in the testing phase.

B. ALGORITHM DESCRIPTION

We present two algorithms: (1) Steps of the MVA forecasting task, where the only interest is to calculate the MVA one-step-ahead estimator when the parameters K, J are known; and, (2) Steps to provide accuracy comparison against other methods.

Consider a n-element time-series $\{y_1, y_2, \dots, y_n\}$. Firstly we define the window size w . This is done by taking the first $(w - 1)$ lags, which are in the statistical relevance region provided by the ACF graph, see section II-D.

See the chart illustrating the steps presented in the algorithm 1.

Algorithm 1 Steps of the MVA Forecasting Task

Input: (K, J, w)
Output: $\hat{y}_{MV,n+1}$
Data: $Y = \{y_{n-w+1}, \dots, y_{n-1}, y_n\}$

- 1) Read the data.
- 2) Normalize the data. See (17).
- 3) Transform the window into a network using the visibility graph approach. See section II-A.
- 4) Calculate the Node Similarity Matrix (NSM) using the Dice Similarity approach. See (6).
- 5) Create $S_n^{(w-1)}$, the last row of the NSM. See (14).
- 6) Find and register the nodes associated to the the greater value in $S_n^{(w-1)}$.
- 7) Calculate the pre-estimators related to the nodes registered in the previous step. See (11).
- 8) Calculate $\hat{y}_{MV,(n+1)}$, taking the maximum of the pre-estimators calculated in the previous step. See (15).

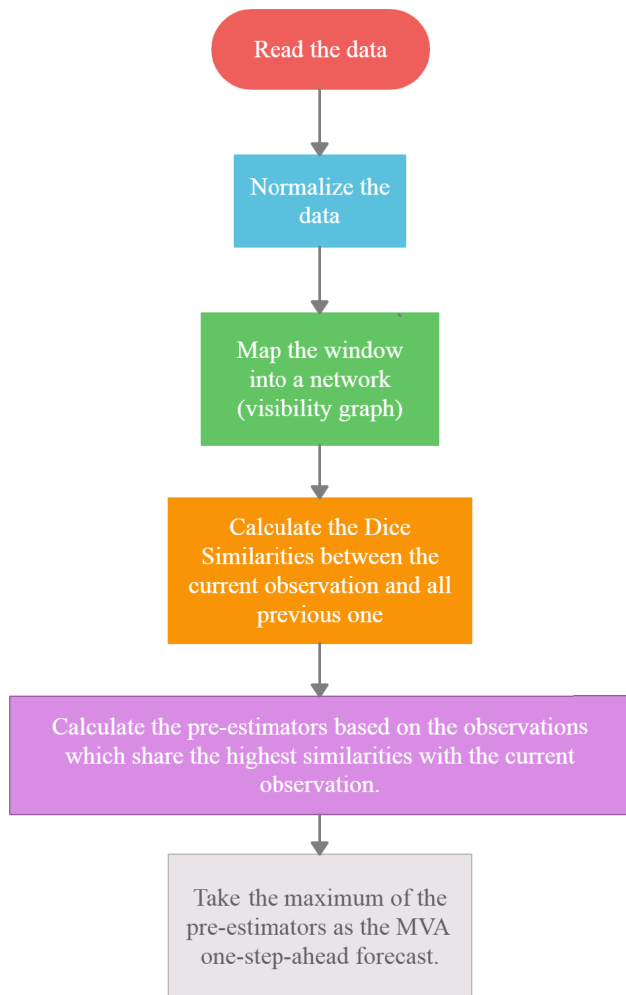


FIGURE 2. Steps of the MVA forecasting task. Source: Drawn by the authors.

With the purpose of compare MVA accuracy against other forecasting methods, we recommend using the steps

presented in the algorithm 2. The sets \mathbf{K} and \mathbf{J} contains the values of k and J , respectively, which will be tested in order to find the best MVA parameters to be used in the test set. Also, if one is only interested in estimate the next term of the time series using MVA but wants to search for the best parameters, then the algorithm 2 applies, but the validation and training sets must merge to contain $(n - w - 1)$ elements and the test set has to be composed by the last element y_n . The search for the best parameters can be done based on the methodology described in the section III-A.

Algorithm 2 Steps for Accuracy Comparison Against Other Methods

Input: $(\mathbf{K}, \mathbf{J}, w)$
Output: $\hat{y}_{MV,n+1}$
Data: $TrSet = \{y_{w+1}, y_{w+2}, \dots, y_{w+t}\}$ // Training Set
Data: $ValSet = \{y_{w+t+1}, y_{w+t+2}, \dots, y_{w+t+v}\}$ // Validation Set
Data: $TeSet = \{y_{w+t+v+1}, y_{w+t+v+2}, \dots, y_n\}$ // Test Set

```

for  $(\forall K \in \mathbf{K}) \wedge (\forall J \in \mathbf{J})$  do
  for  $i \in \{1, 2, \dots, t + v\}$  do
    Read the  $i$ -th window:
     $\{z_1 = y_i, z_2 = y_{i+1}, \dots, z_w = y_{w+i-1}\}$ .
    Calculate  $\hat{y}_{w+i}$  using the algorithm 1.
  Calculate and register
   $SSE(v) = \sum_{k=t+1}^{t+v-1} (\hat{y}_{w+k} - y_{w+k})^2$ .
  
```

Take the pair (J^*, K^*) associated to the $\min \{SSE\}$.

```

for  $i \in \{1, 2, \dots, n - w - t - v\}$  do
  Read the  $i$ -th window:
   $\{z_1 = y_{t+v+i}, \dots, z_w = y_{t+v+w+i-1}\}$ .
  Calculate  $\hat{y}_{w+t+v+i}$  using the algorithm 1.
  Register  $\hat{y}_{w+t+v+i}$ .
  
```

Calculate MAE, MAPE and RMSE according to (8),(9) and (10), respectively.

C. WHY MVA WORKS

At the beginning of Section III, MVA is defined as a forecasting technique based on the complex network theory, more specifically, MVA seeks for relations between nodes of a complex network to provide forecasts. In a complex network, two nodes tend to be linked if they have a higher similarity index [30]. Therefore, taking into account that MVA provides forecasts based on the previous time-series observations which present the greatest similarity index with y_n , we can expect that MVA will succeed in capturing the patterns contained along the time series. Why MVA is called *maximum*? Because the final estimator is obtained taking the maximum of the pre-estimators. This means that the method is creating a new term that, based on the visibility graph approach, will be linked to y_n and to all other y_k which has the same highest level of similarity with y_n .

The natural visibility graph maps a time series into a complex network based on the direct visibility criteria, that is, two points in the time series becomes a linked pair of vertices in the network, if and only if, there is an unobstructed straight line connecting both points in the time series plot.

Given the definition of the visibility graph mapping, all points y_k which has the maximum level of similarity with y_n have an unobstructed link to \hat{y}_{n+1} . Therefore, if we take as one-step-ahead the maximum of the pre-estimators, so we are forcing that all points, which originated the pre-estimators, be connected to the next-term estimator in the complex network domain.

IV. RESULTS AND ANALYSIS

In this section, we present the performance comparison among the forecasting approaches: MVA, MXA [30], ARIMA, SVM, LSTM, MLP, Hybrid additive ARIMA-ANN (HAAA), Hybrid additive ETS-ANN (HAEA) and naive estimation. To measure the accuracy quantitatively, we calculated the error measurements defined in the equations (8), (9), and (10). It is important to point out that all forecasting processes are **one-step-ahead** type.

MXA was chosen as a comparative method because, according to [30], it is considered a successful complex-network-based forecasting method with a good prediction performance. The ARIMA model was chosen because it is widely used as a benchmark in the literature, so if MVA is more accurate than ARIMA, then MVA can be taken as a benchmark against other new prediction methods. The methods SVM, LSTM and MLP are also being considered benchmarks given to the wide amount of works who use these methods to provide time-series forecasts. Inspired on the results of the M4-competition, presented by [24], combination methods tend to perform better than the pure one, therefore, we considered important compare the MVA forecasting results to other succeeded hybrid methods. We choose the Hybrid additive ARIMA-ANN (HAAA) and Hybrid additive ETS-ANN (HAEA). Lastly, we also considered the naive estimation because, despite its simplicity, for some kinds of time series, it has beaten ARIMA and even some hybrid models in terms of performance.

Before starting each forecasting process, we characterized the respective time series. First, by running the ADF and/or KPSS tests, we determine if each time series is level, trend or non-stationary. Next, we tested seasonal behavior using the Webel-Ollech test. Finally, through the Ljung-Box test, we check if there is any dependence between the pairs of the time-series observations.

Here, the only pre-processing job was: (a) to delete or properly replace all “NA”, non available, data; and (b) normalizing the data according to de following rule:

$$z_i = \frac{y_i - \min(Y)}{\max(Y) - \min(Y)}, \quad (17)$$

where Y is the original dataset. Actually, the normalization process, or any other is not necessary. We did that with the purpose of having comparable results, free of any scale. In the case of ARIMA, we used the R function `auto.arima`, which automatically performs all necessary preprocessing to obtain the best ARIMA model. The size of the window used in each forecasting process was chosen considering the results of the autocorrelation function, so only the most significant terms, statistically speaking, were taken into account. It is important to emphasize that the mathematical derivations to obtain the best K and J to be considered on MVA is beyond the scope of this work. Hence for each forecasting simulation, we found the pair (J^*, K^*) using simulation according to explained in section III-A.

In order to compare the forecasting performance achieved from MVA, ARIMA, MXA, MLP, LSTM, SVM, HAAA, HAEA and naive estimation we chose the following time series: three classic time-series from different fields, provided by the R `datasets` package; a Brazilian econometric time series composed of daily closing prices of the Bovespa index; and one recursive time series that was artificially generated.

It is important to point out that the accuracies obtained by each method, considering each time series, were taken from the same test set. We are emphasizing that this measure was taken to ensure that comparisons between methods were made fairly.

A. CHARACTERIZATION OF THE FIVE TIME SERIES

This section is dedicated to showing the particularities of each time series used in this research. The appropriate hypothesis tests, presented in the section II-E, were chosen to characterize the series as stationary, seasonal or independent.

- Consider the monthly number of passengers for international airlines, in thousands of people, from 1949 to 1960. This a 144-element time series, which in the R `astsa` package is called `AirPassengers`, but which is also presented by [58]. ACP shows that the first 40 past observations impact the current value, so this set was chosen to compose the first window. In this forecasting process MVA had $K = 0.886$, $J = 6$ and the nonlinear term, expressed by equation (16) is considered in the model. The sizes of the training, validation and testing sets are 77, 36 and 31, respectively.
- In agreement with [59], ‘lynx’ is a time series of annual numbers of lynx trapped in Canada from 1821 to 1934. There are 114 observations in this time series. The ACF indicates that the first 77 past terms may contain significant information with respect to the current element. Additionally, the most significant values of autocorrelation between terms occur from time to time, suggesting that the series has a seasonal behavior, which was confirmed through the application of the WO test. In order to provide the one-step-ahead forecasting, we chose $w = 77$, and we settled MVA with $K = 0.633$, $J = 3$ and the presence of the nonlinear term in the

TABLE 1. The characteristics of the five time series.

Dataset	Level	Trend	Seasonal	Indep
Air Pass	No	Yes	Yes	No
Lynxs	No	Yes	Yes	No
Ibovespa	No	No	Yes	No
nhtemp	No	Yes	Yes	No
RBTS	No	No	Yes	No

model. The sizes of the training, validation and testing sets are 90, 13 and 11, respectively.

- Ibovespa is the most important performance indicator of stocks traded on the Brazilian financial market [60]. It was created in 1968 and has since served as a reference for investors worldwide [60]. The third time series is made up of daily closing prices of Ibovespa from January 1, 2019 to July 28, 2020. The 389 data used in this simulation were taken from the Yahoo Finance website and the ACF indicates window of length 200. According to the proper statistical tests this is a non-stationary dataset, with no trend, seasonal and with dependency among its observations. In this case, we calibrated MVA with $K = 0.844$, $J = 150$ and the presence of the nonlinear term in the model. The sizes of the training, validation and testing sets are 245, 50 and 94, respectively.
- The fourth dataset is the 60-element time series named “nhtem” on the R *datasets* package which regards the mean annual temperature, in degrees Fahrenheit, in New Haven, Connecticut, from 1912 to 1971 [61]. In this forecasting process MVA had $K = 1.087$, $J = 500$ and the presence of the nonlinear term in the model. The sizes of the window, training, validation and testing sets are 10, 30, 20 and 10, respectively.
- The last dataset comprises a recurrence-based time series with 100 observations. Each observation is given by the following law:

$$y_t = 0.54y_{t-1} + 0.17y_{t-2} + 0.3y_{t-3} + 0.4w_t, \quad (18)$$

where w_t is a random variable normally distributed with zero mean and variance 1. Additionally, $y_0 = 10$, $y_1 = 9.5$, $y_2 = 9$ and $t \in \{0, 1, 2, \dots, 99\}$. In this forecasting process MVA had $w = 85$, $K = 2.263$, $J = 600$ and the presence of the nonlinear term in the model. The sizes of the window, training, validation and testing sets are 85, 89, 4 and 7, respectively.

The Table 1 summarizes the characteristics of the five time series presented in this work, showing all their features, confirmed by proper statistical tests.

B. ADJUSTING THE MACHINE LEARNING MODELS

One of the most determining steps in creating a suitable forecasting model based on Machine Learning approaches such as ANN and SVM models is the choice of the input variables.

TABLE 2. Hyperparameters considered in each SVM model.

Time Series	C	γ	ϵ
Air Pass	30	38	0.1
Lynx	3	107	0.1
Ibovespa	15	40	0.1
nhtemp	1	20	0.1
RBTS	10	0.3	0.1

TABLE 3. Hyperparameters considered in each LSTM model.

Time Series	epochs	hl	lr
Air Pass	1000	512	0.1
Lynx	200	240	0.1
Ibovespa	500	128	0.1
nhtemp	200	64	0.1
RBTS	40	600	0.1

1) THE SVM MODELS

In this work we started the search for the best SVM hyperparameters according to was considered in [40]: $C = 10$ and $\epsilon = 0.1$. The kernel choice is for the radial basis function (RBF) starts with $\gamma = 0.1$ since, according to [62] and [63], this kernel has good performance in time series forecasting problem, it is able to capture nonlinear characteristics of the time series and presents less numerical problems.

We applied the SVM models in the training set, for multiples combinations of (C, ϵ, γ) . The set which provided the smallest MAE was applied to the testing set. Some forecasting strategies were tested to perform the SVM. Firstly we compared the 10-fold cross validation technique [40] to the “leave one out cross validation” (LOOCV) [64]. The second one gives better results in the training sets. However, the walk-forward validation technique, presented by [49], and used in all other methods, gave the smallest MAE considering the training sets. The table 2 shows the parameters considered in the SVM models.

2) THE LSTM MODELS

An LSTM model is an recurrent network model with memory, that is, this model can remember information from earlier points in the network [27]. The architecture of an LSTM consists of units called memory cells, self-connections and gates which are special multiplicative units [41]. According to [27] the connections keeps in memory the temporal state of the memory cells and the gates control the flow of information.

We considered a LSTM model according to proposed by [27]. The hyperparameters used to reach the best accuracy in the validation set, for each time series, are described in the table 3. The column called ‘hl’ gives the number of neurons in the hidden layer.

TABLE 4. Hyperparameters considered in each MLP model.

Time Series	epochs	h=nodes	lags
Air Pass	15	5	15
Lynx	10	10	29
Ibovespa	500	128	0.1
nhtemp	18	5	5
RBTS	25	8	50

3) THE MLP MODELS

According to [27], a Multilayer Perceptron (MLP) is a variant of the original model proposed by Rosenblatt in [65]. The MLP mathematical and structural model is presented by [27]. The training assignment is done based on the backpropagation approach, which is a supervised learning method [66].

The neuron functioning occurs according to described in equation (19):

$$O_k = f \left(\sum_{i=1}^w y_{n-i} \beta_{ki} + b_k \right), \quad (19)$$

where $f(\tau)$ is the activation function, y_{n-i} is the i -th input, w is the number of inputs, and β_{ki} and b_k are, respectively, the weight and the bias associated to the input y_{n-i} [27].

We considered a MLP structure composed by three layers of nodes: an input layer, a hidden layer and an output layer. All neurons were processed by the sigmoidal activation function, except for the output layer for which a linear activation function was used. The forecasting tasks were produced by the *mlp* function included in the *nnfor* R package. For each time series, the testing set is the same considered to all other forecasting methods and the forecasts were combined using the median operator. The table 4 shows the hyperparameters which provide the best accuracy in the training and validation sets.

To enable reproducibility, we set the seed at 2, so all randomly generated initial weights will always be the same.

C. ADJUSTING THE HYBRID MODELS

The last two time series forecasting methods are hybrid approaches. These are additive combinations of ANN with ARIMA and ANN with ETS. We consider the number of nodes in the ANN model’s input layer to be equal to the window size considered for all other forecasting methods. We tried other values of w , but in all cases, the obtained accuracy, relative to the validation sets, only got worse.

According to [40] and [67], a three-layer feed-forward ANN model is commonly used for time series forecasting. We normalized the training and testing data to fit in the interval $[0, 1]$ according to equation (17). The hyperbolic tangent function was used as the transfer function between the input and the hidden layers because it resulted in the best accuracy. We chose the linear function as the transfer function between the hidden and the output layers due to its robustness for continuous output variables [40]. The optimal

TABLE 5. Parameters considered in each ANN-ARIMA model.

Time Series	Number of hidden nodes
Air Pass	5
Lynx	3
Ibovespa	5
nhtemp	10
RBTS	20

TABLE 6. Parameters considered in each ANN-ETS model.

Time Series	Number of hidden nodes
Air Pass	5
Lynx	20
Ibovespa	20
nhtemp	14
RBTS	42

number of neurons in the hidden layer, h , was obtained by trial and error, but initially we used previous practical results [68]: $h = 0.5i$, where i is the number of input data. The initial model weights were chosen considering a uniform distribution in the interval $[-0.1, 0.1]$, with seed 2. Each training assignment was performed minimizing the mean absolute error.

1) THE ARIMA-ANN MODELS

The hybrid additive approach combining ARIMA and ANN was implemented according to described in [26] using the Khashei-Bijari approach [67]. The table 5 shows the best number or nodes in the hidden layer for each time series. These parameters were defined from the accuracy obtained on the training and validation sets.

2) THE ETS-ANN MODELS

The ETS models are composed by a family of time series models with a basal state space model containing an error term (E) and a level, trend (T) or seasonal (S) components. The hybrid additive approach combining ETS and ANN was implemented according to described in [28]. The ETS-models were defined based on the R function “ets” from the package forecast. This function is able to calculate the best ETS model which fits with the given time series. The table 5 shows the best number or nodes in the hidden layer for each time series. These parameters were defined from the accuracy obtained on the training and validation sets.

D. COMPARING THE ERROR MEASUREMENTS

In the last three sections we described the five datasets used in this work and we presented all eight forecasting techniques and their respective set of parameters considered in each forecasting process. In this section we show the results obtained in terms of MAE, MAPE and RMSE, according described in the equations (8), (9) and (10), respectively. The accuracy results are presented in the table 7.

According to what is presented for the Air Passengers dataset in table 7, MVA outperformed all other methods

TABLE 7. Comparative performance indicators obtained from the forecasting process of the five time series considering the nine techniques.

Time Series: AirPassengers									
	MVA	MXA	ARIMA	Naive	HAAA	HAEA	LSTM	MLP	SVM
MAE	0.0696	0.0894	0.0897	0.0887	0.0923	0.0972	0.0800	0.0995	0.0791
MAPE	10.63	14.26	13.74	13.93	14.24	15.35	26.52	14.98	12.39
RMSE	0.0804	0.1090	0.1095	0.1030	0.1071	0.1144	0.0941	0.1151	0.0927
Time Series: lynx									
	MVA	MXA	ARIMA	Naive	HAAA	HAEA	LSTM	MLP	SVM
MAE	0.0714	0.1393	0.0877	0.1104	0.1115	0.0725	0.1053	0.1267	0.0828
MAPE	46.72	64.92	32.88	52.91	54.18	43.07	155.94	60.78	39.48
RMSE	0.1016	0.1514	0.1125	0.1268	0.1241	0.0970	0.1226	0.1525	0.0964
Time Series: IBOV									
	MVA	MXA	ARIMA	Naive	HAAA	HAEA	LSTM	MLP	SVM
MAE	0.0272	0.0308	0.0470	0.0276	0.0287	0.0429	0.0815	0.0361	0.0386
MAPE	23.53	24.71	45.59	23.85	22.97	42.63	146.3	25.13	27.52
RMSE	0.0372	0.0448	0.0881	0.0379	0.0382	0.0643	0.1362	0.0525	0.0485
Time Series: nhtemp									
	MVA	MXA	ARIMA	Naive	HAAA	HAEA	LSTM	MLP	SVM
MAE	0.0906	0.1139	0.1094	0.0970	0.1131	0.1125	0.1105	0.1222	0.0982
MAPE	17.16	22.74	21.53	18.98	20.52	21.53	16.89	24.49	19.29
RMSE	0.1060	0.1392	0.1324	0.1174	0.1406	0.1309	0.1290	0.1543	0.1233
Time Series: RBTS									
	MVA	MXA	ARIMA	Naive	HAAA	HAEA	LSTM	MLP	SVM
MAE	0.0172	0.0329	0.0266	0.0257	0.02824	0.0415	0.0268	0.0256	0.0234
MAPE	1.77	3.44	2.79	2.67	2.92	4.29	2.54	2.68	2.43
RMSE	0.0281	0.0384	0.0322	0.0327	0.0357	0.0487	0.0294	0.0328	0.0289

based on the MAE, MAPE and RMSE values. For this time series, MVA had its superiority confirmed by the Diebold-Mariano(DM) test when compared to all methods except for LSTM and SVM for which the quality of the forecasts are statistically equivalent. Since the DM test did not find any method to be superior to MVA and considering that its errors are the smallest, we can conclude that MVA is the most accurate method, among all those who participated in this study, to make predictions for this time series.

Considering the results of tables 7 and 9 for the Lynx time series, MXA, naive, Additive-ARIMA-ANN (HAAA), LSTM and MLP are less accurate than MVA, quantitative and qualitatively given that their values of MAE, MAPE and RMSE are higher than MVA's, and the DM test indicates MVA's superiority. However, MVA, ARIMA, Additive-ETS-ANN (HAEA) and SVM showed similar results considering error measures and statistical comparison. In order to illustrate the closeness of these methods to the real values in the test set, we have plotted the comparison graph for lynx time series and presented it in Fig. 3.

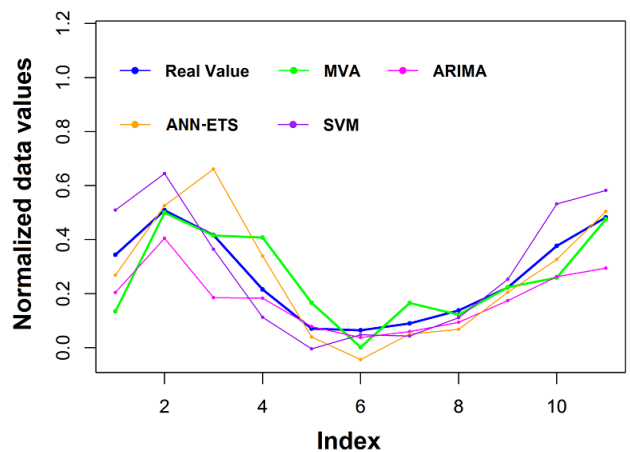


FIGURE 3. Forecasts for lynx time series considering MVA, ARIMA, Additive-ETS-ANN and SVM. Source: Drawn by the authors.

Comparing the accuracy results obtained by each method for the IBOVESPA time series, MVA outperformed all other

TABLE 8. Comparative MVA's RMSE when calculated from Jaccard, Dice and Inverse Log-Weighted similarities considering the five time series.

	RMSE - MVA		
	Jaccard	Dice	InvLogWeighted
Air Passengers	0.0804	0.0804	0.0954
Linx	0.1016	0.1016	0.0980
IBOVESPA	0.0372	0.0372	0.0384
NHtemp	0.1060	0.1060	0.1158
RBTS	0.0281	0.0281	0.0297

methods. Naive is the only method which provided similar results, statistically speaking, confirmed by the DM test. Definitely, in order to produce forecasts for the IBOVESPA time series, MVA is the most indicated method among all methods considered in this study.

The fourth time series, was chosen precisely because it is a dataset with characteristics that are difficult to define. How would all methods respond in the forecasting process for this time series? It is classified as trend stationary, seasonal and not-independent dataset, however, the p-values obtained from each hypothesis test is very close to the 10% type I error. It was expected that most methods would present statistically similar results, therefore the most accurate method would have to be defined based on the error measures. According to the results presented in the table 7 MVA outperforms all other methods in terms of MAE, MAPE and RMSE, thus, it is considered the most accurate method in the forecasting assignment for this time series.

Lastly, we present the results for a recurrence based time series artificially created. This dataset was created with the intention of favoring the ARIMA method, since the sequence is based on an AR(3) model. ARIMA was expected to achieve the best results, however, MVA had the best accuracy. The conclusion is similar to that obtained in the case of the air passengers time series, since MVA obtained the smallest errors, but tied with other methods in the statistical comparison of the quality of the estimation. None of the considered methods was classified by the DM test as superior to MVA.

We consider important to emphasize that all MVA forecasts, presented in this work, were obtained based on using the Dice similarity. We considered using other methods, such as Jaccard [46] and the Inverse Log-weighted [47], both cited in the section II-C. In order to compare the results, the obtained RMSE was taken into account, for each time series.

The Jaccard method led to the same results obtained from the Dice similarity index, considering the five time series presented here. The Inverse Log-weighted similarity produced similar results to those presented here. Better if compared to that obtained from Dice, for the Lynx time series and worse in the other cases. The comparative analysis of the MVA accuracy based on the different measures of similarities is beyond the scope of this article, but we ratify that this is an

important topic to be considered since improved versions of MVA can be developed.

E. TIME COMPLEXITY OF MVA

As we commented that one of the advantages of MVA is its low computational cost, it is important to present a brief discussion about the time complexity of the method. Defining n as the size of the window to be mapped in a graph, consider the following worst-case time complexity analysis for each step of the MVA algorithm.

- 1) Read the time series data - Complexity: $O(n)$.
- 2) Normalize the data - Complexity: $O(n)$.
- 3) Map the time series into a network (visibility graph) - Complexity: $O(n^2)$. See [69].
- 4) Calculate the Dice similarity - Complexity: $O(dn^2)$, where d is the maximum degree of the vertices in the graph. See [70].
- 5) Use of the function which.max to find the nodes which share the greatest similarity with y_n - Complexity: $O(n)$.
- 6) Calculate the pre-estimators - Complexity: $O(n)$.
- 7) Take the maximum among the pre-estimators - Complexity: $O(n)$.

All these steps are performed sequentially, so the complexity of the whole algorithm is given by the maximum among the individual complexities, that is, $O(dn^2)$. Obviously, the maximum value for d is $(n - 1)$, so in the worst case scenario, our code is $O(n^3)$. However, this is very unlikely, given the inputs are real time series, so, rarely, is one observation connected to all other observations. In summary, the complexity for MVA depends on the maximum degree of the vertices in the graph, but based on the randomness of the real time series, this maximum degree is relatively low, for example, considering the time series used in this article, the maximum degree were always less than 5% of the time series length. Therefore, MVA can be considered approximately $O(n^2)$ in most applications.

Just for an example, we ran the simulation of the MVA with fixed K and J , considering 95% (also very unlikely) of the data to compose the window. So the algorithm just read the data, normalized it, constructed the complex network, calculated the dice similarities of all pairs of vertices, found the greater value among all values, calculated the pre-estimators considering only the points which presented this greater level of similarity with the current observation, and finally took as the MVA one-step-ahead estimator the maximum value among all pre-estimators. We ran this simulation for an artificial time series AR(1) composed by: 100, 500, 1000, 5000 and 10000 points, which means networks with 95, 475, 950, 4750 and 9500 vertices. The execution times were, respectively, 0.007, 0.183, 0.340, 6.827 and 27.927 seconds, which is in consonance with quadratic time complexity. The simulations were performed in a Intel(R) Core(TM) i7-2630QM CPU @ 2.00GHz with 8.00GB (RAM) processor based on x64 under Windows 8.1.

TABLE 9. P-values obtained from the application of the Diebold-Mariano test in the forecasting residuals of the five datasets.

Time Series: AirPassengers (Type I error = 10%)								
	MXA	ARIMA	Naïve	HAAA	HAEA	LSTM	MLP	SVM
MVA	0.0470	0.0170	0.0076	0.0055	0.0130	0.2033	0.0012	0.1483
Time Series: Lynx (Type I error = 10%)								
	MXA	ARIMA	Naïve	HAAA	HAEA	LSTM	MLP	SVM
MVA	0.0114	0.3263	0.0396	0.0491	0.4847	0.0904	0.0875	0.2937
Time Series: IBOV (Type I error = 10%)								
	MXA	ARIMA	Naïve	HAAA	HAEA	LSTM	MLP	SVM
MVA	0.0321	< 0.001	0.2587	0.0563	< 0.001	< 0.001	0.0012	< 0.001
Time Series: nhtemp (Type I error = 10%)								
	MXA	ARIMA	Naïve	HAAA	HAEA	LSTM	MLP	SVM
MVA	0.1558	0.1637	0.3462	0.1830	0.1237	< 0.001	0.0560	0.3486
Recurrence-based Time Series (Type I error = 10%)								
	MXA	ARIMA	Naïve	HAAA	HAEA	LSTM	MLP	SVM
MVA	0.0235	0.1070	0.0409	0.1263	0.0055	0.1594	0.1324	0.1854

F. MVA: MAIN ADVANTAGES AND DISADVANTAGES

Among the advantages observed with the use of MVA in the forecasting task are the ease of implementation, low computational cost and adaptability to different types of time series, considering characteristics of stationarity, seasonality and dependence between observations. This study showed that the only pre-processing necessary is to delete or properly replace the 'NA' data (not available) if any, no further pre-processing is necessary, thus, the prediction process can be carried out considering the original data. Given the inclusion of the non-linear term, MVA is able to quickly correct the error made in the previous prediction. We can point out as a disadvantage the lower capacity to predict the reversal of the trend of the time series. The biggest errors produced by MVA were mainly at trend reversal points.

V. CONCLUSION

In summary, this study presents the Maximum Visibility Approach (MVA) as new time series forecasting method. To calculate the forecast with MVA, we used the visibility graph technique to map a given time series into a complex network, and then calculate the node similarity matrix using the Dice approach. All nodes with the higher similarity index with the node associated to the last time-series observation are eligible to impact the calculation of the MVA one-step-ahead estimator, $\hat{y}_{MV,(n+1)}$.

Just in order to compare the performances, we chose the comparative methods ARIMA, the Mao-Xiao approach [30], naive estimation, support vector machine (SVM), artificial neural networks (ANN), long-short term memory (LSTM),

multilayer perceptron (MLP) and the hybrid additive models ARIMA-ANN and ETS-ANN. The nine methods discussed in this work were applied to five different time series. Each dataset has some or none of the characteristics: seasonal behavior, level or trend stationarity, and dependence between pairs of observations. It is important to note that the time series used in this work also have different time horizons: daily, monthly, annual, and weekly data. The performance results were measured using the mean absolute error (MAE), mean absolute percentage error (MAPE), and root mean squared error (RMSE). Additionally, the Diebold-Mariano test [71] was considered to evaluate the statistical comparison between accuracy achieved by MVA and each benchmark.

Considering the MAE, MAPE and RMSE values, MVA outperformed all methods for all time series, except for the Lynx time series, in which the methods ARIMA, additive ARIMA-ETS and SVM reached the same level of accuracy as that obtained by MVA. This set of results reveals that our proposed method is capable of presenting at least similar accuracy when compared to other well-established methods in the literature, considering varied applications and datasets with different time horizons. Therefore, our method can contribute not only in the academic field, but also in practical aspects.

According to discussed in the section IV-F, we can point as advantages of MVA the ease of implementation, low computational cost, in most applications $O(n^2)$, and the adaptability to forecast multiple types of time series, considering characteristics of stationarity, seasonality, and dependence between its observations. Another advantage is

the fact that, except for excluding or replacing the non-available data (NA), the forecasting process can be carried out considering the original data. In the case of performing sequential forecasts MVA is able to quickly correct the error made in the previous forecasting given the inclusion of the nonlinear term in the model. We consider the disadvantage of MVA its low ability to predict trend reversal in time series. The biggest errors produced by the MVA were mainly in the reversal points.

Despite the good forecasting performance of MVA shown in this work, this method can be improved. As future works, we can mention: the investigation of the impact on accuracy obtained by using different time series mapping methods in complex networks; the creation of a hybrid methodology to combine MVA with MLP, LSTM, ANN or other prediction methods based on machine learning, such as the Elman Recurrent Neural Networks (ERNN), similarly to what was proposed by [72]; develop a mathematical formulation that calculates the best parameters J and K, based on the observations which compose the training set; propose a version of MVA which provides *h-step-ahead* forecasts, with $h > 1$; study adequate data preprocessing to increase MVA's forecasting accuracy; provide the comparative analysis of the MVA accuracy based on the different measures of similarities; and develop a multidimensional version of MVA, in which more than one time series can provide information in order to improve the accuracy in the forecasting process of one or more time series at the same time.

We will end this article by making it clear that the results presented here do not mean that MVA will be able to outperform all forecasting methods, considering all time series. However, such results certainly indicate that MVA is an excellent option of approach to be considered in time series forecasting processes, both for research purposes and for practical use.

ACKNOWLEDGMENT

The authors greatly appreciate the reviewer's important suggestions and the editor's positive incentives.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

REFERENCES

- [1] R. Shumway and D. Stoffer, *Time Series Analysis and Its Applications With R Examples*, 3rd ed. New York, NY, USA: Springer, Jan. 2011.
- [2] R. Adhikari and R. K. Agrawal, "An introductory study on time series modeling and forecasting," 2013, *arXiv:1302.6613*.
- [3] M. Imhoff, M. Bauer, U. Gather, and D. Löhlein, "Time series analysis in intensive care medicine," Univ. Dortmund, Dortmund, Germany, Tech. Rep. 1998,01, 1998. [Online]. Available: <http://hdl.handle.net/10419/77286>
- [4] G. Radhakrishnan, D. Gupta, S. Sindhuula, S. Khokhawati, and T. S. B. Sudarshan, "Experimentation and analysis of time series data from multi-path robotic environment," in *Proc. IEEE Int. Conf. Electron., Comput. Commun. Technol. (CONECCT)*, Jul. 2015, pp. 1–6.
- [5] A. B. Abdullah, T. R. Pillai, and L. Z. Cai, "Intrusion detection forecasting using time series for improving cyber defence," *Int. J. Intell. Syst. Appl. Eng.*, vol. 3, no. 1, p. 28, Mar. 2015.
- [6] K. Emmanouilidis and C. Karpetis, "The defense-growth nexus: A review of time series methods and empirical results," *Defence Peace Econ.*, vol. 31, no. 1, pp. 86–104, Jan. 2020, doi: [10.1080/10242694.2018.1428261](https://doi.org/10.1080/10242694.2018.1428261).
- [7] L. R. Marusich and N. Buchler, "Time series modeling of army mission command communication networks: An event-driven analysis," *Comput. Math. Org. Theory*, vol. 22, no. 4, pp. 467–486, Dec. 2016, doi: [10.1007/s10588-016-9211-7](https://doi.org/10.1007/s10588-016-9211-7).
- [8] Q. Wen, "Asset growth and stock market returns: A time-series analysis," *Rev. Finance*, vol. 23, pp. 599–628, Jun. 2018, doi: [10.1093/rof/rfy018](https://doi.org/10.1093/rof/rfy018).
- [9] T. Zhan and F. Xiao, "A fast evidential approach for stock forecasting," *Int. J. Intell. Syst.*, vol. 36, no. 12, pp. 7544–7562, Aug. 2021, doi: [10.1002/int.22598](https://doi.org/10.1002/int.22598).
- [10] A. Rehman, L. Jingdong, A. A. Chandio, I. Hussain, S. A. Wagan, and Q. U. A. Memon, "Economic perspectives of cotton crop in Pakistan: A time series analysis (1970–2015) (Part 1)," *J. Saudi Soc. Agricult. Sci.*, vol. 18, no. 1, pp. 49–54, Jan. 2019. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1658077X1630162X>
- [11] R. M. Kunst and M. Wagner, "Economic forecasting: Editors' introduction," *Empirical Econ.*, vol. 58, no. 1, pp. 1–5, Jan. 2020, doi: [10.1007/s00181-019-01820-3](https://doi.org/10.1007/s00181-019-01820-3).
- [12] D. M. Filatov and A. A. Lyubushin, "Precursory analysis of GPS time series for seismic hazard assessment," *Pure Appl. Geophys.*, vol. 177, no. 1, pp. 509–530, Jan. 2020, doi: [10.1007/s00024-018-2079-3](https://doi.org/10.1007/s00024-018-2079-3).
- [13] D. F. Greenberg, "Time series analysis of crime rates," *J. Quant. Criminol.*, vol. 17, no. 4, pp. 291–327, 2001, doi: [10.1023/A:1012507119569](https://doi.org/10.1023/A:1012507119569).
- [14] G. Box, G. Jenkins, and C. Reinsel, *Time Series Analysis: Forecasting and Control*, 4th ed. Hoboken, NJ, USA: Wiley, May 2013.
- [15] G. E. P. Box, *Time Series Analysis: Forecasting and Control* (Wiley Series in Probability and Statistics). Hoboken, NJ, USA: Wiley, Jun. 2015. [Online]. Available: <https://www.xarg.org/ref/a/1118675029/>
- [16] G. E. P. Box, J. A. Jenkins, and M. Gwilym, *Time Series Analysis: Forecasting and Control*. San Francisco, CA, USA: Holden-Day, 1970. [Online]. Available: <http://www.gbv.de/dms/hbz/toc/ht000495926.pdf>
- [17] Y. Hu, "Iterative and recursive least squares estimation algorithms for moving average systems," *Simul. Model. Pract. Theory*, vol. 34, pp. 12–19, May 2013. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1569190X1300004X>
- [18] J. H. Cochrane, *Time Series for Macroeconomics and Finance*. Graduate School of Business, University of Chicago, 1997. [Online]. Available: <http://econ.lse.ac.uk/staff/wdenhaan/teach/cochrane.pdf>
- [19] K. Hipel and A. McLeod, *Time Series Modelling of Water Resources and Environmental Systems*, vol. 45, 1st ed. Amsterdam, The Netherlands: Elsevier, 1997.
- [20] M. A. Benjamin, R. A. Rigby, and D. M. Stasinopoulos, "Generalized autoregressive moving average models," *J. Amer. Stat. Assoc.*, vol. 98, no. 461, pp. 214–223, 2003, doi: [10.1198/016214503388619238](https://doi.org/10.1198/016214503388619238).
- [21] R. B. Carmona-Benítez and M. R. Nieto, "SARIMA damp trend grey forecasting model for airline industry," *J. Air Transp. Manage.*, vol. 82, Jan. 2020, Art. no. 101736. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S096969719301711>
- [22] Y.-T. Bai, X.-Y. Wang, X.-B. Jin, Z.-Y. Zhao, and B.-H. Zhang, "A neuron-based Kalman filter with nonlinear autoregressive model," *Sensors*, vol. 20, no. 1, p. 299, Jan. 2020. [Online]. Available: <https://www.mdpi.com/1424-8220/20/1/299>
- [23] M. J. Zaki, *Data Mining and Analysis: Fundamental Concepts and Algorithms*. Cambridge, U.K.: Cambridge Univ. Press, May 2014. [Online]. Available: <https://www.xarg.org/ref/a/0521766338/>
- [24] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, "Statistical and machine learning forecasting methods: Concerns and ways forward," *PLoS ONE*, vol. 13, no. 3, Mar. 2018, Art. no. e0194889. [Online]. Available: <https://pubmed.ncbi.nlm.nih.gov/29584784>
- [25] E. Eğrioğlu and R. Fildes, "A new bootstrapped hybrid artificial neural network approach for time series forecasting," *Comput. Econ.*, pp. 1–29, Nov. 2020, doi: [10.1007/s10614-020-10073-7](https://doi.org/10.1007/s10614-020-10073-7).
- [26] N. B. Chindanur and B. Eswara, "Performance comparison of four new ARIMA-ANN prediction models on internet traffic data," *J. Telecommun. Inf. Technol.*, vol. 2015, pp. 67–75, Jan. 2015.
- [27] S. Kaushik, A. Choudhury, P. K. Sheron, N. Dasgupta, S. Natarajan, L. A. Pickett, and V. Dutt, "AI in healthcare: Time-series forecasting using statistical, neural, and ensemble architectures," *Frontiers Big Data*, vol. 3, p. 4, Mar. 2020. [Online]. Available: <https://www.frontiersin.org/article/10.3389/fdata.2020.00004>

- [28] S. K. Purohit, S. Panigrahi, P. K. Sethy, and S. K. Behera, "Time series forecasting of price of agricultural products using hybrid methods," *J. Appl. Artif. Intell.*, pp. 1–19, Sep. 2021. [Online]. Available: https://www.researchgate.net/publication/354686351_Time_Series_Forecasting_of_Price_of_Agricultural_Products_Using_Hybrid_Methods
- [29] M. A. Kramer, U. T. Eden, S. S. Cash, and E. D. Kolaczyk, "Network inference with confidence from multivariate time series," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 79, no. 6, Jun. 2009, Art. no. 061916. [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevE.79.061916>
- [30] S. Mao and F. Xiao, "Time series forecasting based on complex network analysis," *IEEE Access*, vol. 7, pp. 40220–40229, 2019. [Online]. Available: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8669744>
- [31] Z.-K. Gao, Q. Cai, Y.-X. Yang, W.-D. Dang, and S.-S. Zhang, "Multiscale limited penetrable horizontal visibility graph for analyzing nonlinear time series," *Sci. Rep.*, vol. 6, no. 1, p. 35622, Dec. 2016, doi: [10.1038/srep35622](https://doi.org/10.1038/srep35622).
- [32] L. N. Ferreira. (2017). *Time Series Data Mining Using Complex Networks*. [Online]. Available: <https://teses.usp.br/teses/disponiveis/55/55134/tde-01022018-144118/en.php>
- [33] G. Liu, F. Xiao, C.-T. Lin, and Z. Cao, "A fuzzy interval time-series energy and financial forecasting model using network-based multiple time-frequency spaces and the induced-ordered weighted averaging aggregation operation," *IEEE Trans. Fuzzy Syst.*, vol. 28, no. 11, pp. 2677–2690, Nov. 2020.
- [34] W. Liu and L. Lü, "Link prediction based on local random walk," *Europhys. Lett.*, vol. 89, Mar. 2010, Art. no. 58007, doi: [10.1209/0295-5075/89/58007](https://doi.org/10.1209/0295-5075/89/58007).
- [35] L. Lacasa, B. Luque, F. Ballesteros, J. Luque, and J. C. Nuño, "From time series to complex networks: The visibility graph," *Proc. Nat. Acad. Sci. USA*, vol. 105, no. 13, pp. 4972–4975, 2008. [Online]. Available: <https://www.pnas.org/content/105/13/4972>
- [36] S.-M. Chen, "Forecasting enrollments based on fuzzy time series," *Fuzzy Sets Syst.*, vol. 81, no. 3, pp. 311–319, Aug. 1996. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/0165011495002200>
- [37] H.-K. Yu, "Weighted fuzzy time series models for TAIEX forecasting," *Phys. A, Statist. Mech. Appl.*, vol. 349, nos. 3–4, pp. 609–624, 2005. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0378437104014128>
- [38] H. K. Pathak and P. Singh, "A new bandwidth interval based forecasting method for enrollments using fuzzy time series," *Appl. Math.*, vol. 2, no. 4, pp. 504–507, 2011. [Online]. Available: <https://www.scirp.org/html/4515.html>
- [39] R. Zhang, B. Ashuri, and Y. Deng, "A novel method for forecasting time series based on fuzzy logic and visibility graph," *Adv. Data Anal. Classification*, vol. 11, no. 4, pp. 759–783, Dec. 2017, doi: [10.1007/s11634-017-0300-3](https://doi.org/10.1007/s11634-017-0300-3).
- [40] R. Samsudin, A. Shabri, and P. Saad, "A comparison of time series forecasting using support vector machine and artificial neural network model," *J. Appl. Sci.*, vol. 10, no. 11, pp. 950–958, May 2010. [Online]. Available: <https://scialert.net/fulltext/?doi=jas.2010.950.958>
- [41] S. Hochreiter and J. J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997. [Online]. Available: https://www.researchgate.net/publication/13853244_Long_Short-term_Memory
- [42] S. Panigrahi and H. S. Behera, "A hybrid ETS–ANN model for time series forecasting," *Eng. Appl. Artif. Intell.*, vol. 66, pp. 49–59, Nov. 2017.
- [43] L. Dong, Y. Li, H. Yin, H. Le, and M. Rui, "The algorithm of link prediction on social network," *Math. Problems Eng.*, vol. 2013, Sep. 2013, Art. no. 125123, doi: [10.1155/2013/125123](https://doi.org/10.1155/2013/125123).
- [44] L. Lu and T. Zhou, "Link prediction in complex networks: A survey," 2010, *arXiv:1010.0725*.
- [45] G. Salton, *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer* (Addison-Wesley Series in Computer Science). Reading, MA, USA: Addison-Wesley, 1989.
- [46] V. Arnaboldi, A. Passarella, M. Conti, and R. I. Dunbar, "Evolutionary dynamics in Twitter ego networks," in *Online Social Networks* (Computer Science Reviews and Trends), V. Arnaboldi, A. Passarella, M. Conti, and R. I. Dunbar, Eds. Boston, MA, USA: Elsevier, 2015, ch. 5, pp. 75–92. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/B9780128030233000059>
- [47] L. A. Adamic and E. Adar, "Friends and neighbors on the web," *Social Netw.*, vol. 25, no. 3, pp. 211–230, 2003. [Online]. Available: <http://social.cs.uiuc.edu/class/cs591kgk/friendsadamic.pdf>
- [48] S. Andrews and G. Hamarneh, "Multi-region probabilistic dice similarity coefficient using the Aitchison distance and bipartite graph matching," 2015, *arXiv:1509.07244*.
- [49] I. Kaastra and M. Boyd, "Designing a neural network for forecasting financial and economic time series," *Neurocomputing*, vol. 10, no. 3, pp. 215–236, 1996. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/0925231295000399>
- [50] W. A. Fuller, *Introduction to Statistical Time Series* (Wiley Series in Probability and Statistics), 2nd ed. Hoboken, NJ, USA: Wiley, 1995.
- [51] D. Kwiatkowski, P. C. B. Phillips, P. Schmidt, and Y. Shin, "Testing the null hypothesis of stationarity against the alternative of a unit root," *J. Econometrics*, vol. 54, nos. 1–3, pp. 159–178, Oct. 1992. [Online]. Available: <http://debis.deu.edu.tr/userweb/onder.hanedar/dosyalar/kpss.pdf>
- [52] G. M. Ljung and G. E. P. Box, "On a measure of lack of fit in time series models," *Biometrika*, vol. 65, pp. 297–303, Aug. 1978.
- [53] K. Webel and D. Ollech, "An overall seasonality test based on recursive feature elimination in conditional random forests," in *Proc. Int. Conf. Time Ser. Forecasting*, vol. 1, Sep. 2018, pp. 20–31.
- [54] D. Lytras, "On seasonality: Comparing X-13ARIMA-SEATs diagnostics for quarterly series," U.S. Census Bureau, Suitland-Silver Hill, MD, USA, Tech. Rep., 2015. [Online]. Available: <https://www.census.gov/library/working-papers/2015/adrm/lytras-01.html>
- [55] T. Pohlert. (2014). *The Pairwise Multiple Comparison of Mean Ranks Package (PMCMR)*, *R Package*. [Online]. Available: <http://CRAN.R-project.org/package=PMCMR>
- [56] B. Luque, L. Lacasa, F. Ballesteros, and J. Luque, "Horizontal visibility graphs: Exact results for random time series," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 80, no. 4, Oct. 2009, Art. no. 046103. [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevE.80.046103>
- [57] Z. Gao and N. Jin, "Complex network from time series based on phase space reconstruction," *Chaos, Interdiscipl. J. Nonlinear Sci.*, vol. 19, no. 3, Sep. 2009, Art. no. 033137, doi: [10.1063/1.3227736](https://doi.org/10.1063/1.3227736).
- [58] G. E. P. Box, J. A. Jenkins, and M. Gwilym, *Time Series Analysis: Forecasting and Control*, 3rd ed. San Francisco, CA, USA: Holden-Day, 1976.
- [59] M. J. Campbell and A. M. Walker, "A survey of statistical work on the Mackenzie river series of annual Canadian lynx trappings for the years 1821–1934 and a new analysis," *J. Roy. Stat. Soc. A, Gen.*, vol. 140, no. 4, pp. 411–431, 1977. [Online]. Available: <http://www.jstor.org/stable/2345277>
- [60] *Indice Bovespa (IBOVESPA)*. Accessed: Nov. 4, 2020. [Online]. Available: http://www.b3.com.br/pt_br/market-data-e-indices/indices/indices-amplos/ibovespa.htm
- [61] (2021). *R Core Team and Contributors Worldwide, Datasets-Package: The R Datasets Package, R Package Version 4.2.0*. [Online]. Available: <https://stat.ethz.ch/R-manual/R-devel/library/datasets/html/nhtemp.html>
- [62] S. S. Eslamian, S. A. Gohari, M. Biabanaki, and R. Malekian, "Estimation of monthly pan evaporation using artificial neural networks and support vector machines," *J. Appl. Sci.*, vol. 8, no. 19, pp. 3497–3502, Sep. 2008.
- [63] Y. Ding, X. Song, and Y. Zen, "Forecasting financial condition of Chinese listed companies based on support vector machine," *Expert Syst. Appl.*, vol. 34, no. 4, pp. 3081–3089, May 2008.
- [64] A. Vehtari, A. Gelman, and J. Gabry, "Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC," *Statist. Comput.*, vol. 27, no. 5, pp. 1413–1432, Sep. 2017, doi: [10.1007/s11222-016-9696-4](https://doi.org/10.1007/s11222-016-9696-4).
- [65] F. Rosenblatt, "Principles of neurodynamics. Perceptrons and the theory of brain mechanisms," Cornell Aeronaut. Lab., Buffalo, NY, USA, Tech. Rep., Mar. 1961. [Online]. Available: <https://apps.dtic.mil/sti/citations/AD0256582>
- [66] M. Shiblee, P. Kalra, and B. Chandra, "Time series prediction with multilayer perceptron (MLP): A new generalized error based approach," in *Proc. Int. Conf. Neural Inf. Process.*, Nov. 2008, pp. 37–44.
- [67] M. Khashei and M. Bijari, "A novel hybridization of artificial neural networks and ARIMA models for time series forecasting," *Appl. Soft Comput.*, vol. 11, no. 2, pp. 2664–2675, Mar. 2011. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1568494610002759>
- [68] S. Y. Kang, "An investigation of the use of feedforward neural networks for forecasting," M.S. thesis, Kent State Univ., Kent, OH, USA, 1992.

[69] X. Lan, H. Mo, S. Chen, Q. Liu, and Y. Deng, "Fast transformation from time series to visibility graphs," *Chaos, Interdiscipl. J. Nonlinear Sci.*, vol. 25, no. 8, Aug. 2015, Art. no. 083105, doi: [10.1063/1.4927835](https://doi.org/10.1063/1.4927835).

[70] G. Csardi and T. Nepusz, "The igraph software package for complex network research," *Int. J. Complex Syst.*, vol. 1695, no. 5, pp. 1–9, 2006. [Online]. Available: <http://cneurocv.s.rmk.kfki.hu/igraph/docs/0.5.1/igraph-docs.pdf>

[71] D. Francis and M. Roberto, "Comparing predictive accuracy," *J. Bus. Econ. Statist.*, vol. 13, pp. 253–263, Jul. 1995. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.454.4490&rep=rep1&type=pdf>

[72] C. H. Aladag, E. Egrioglu, and C. Kadilar, "Forecasting nonlinear time series with a hybrid methodology," *Appl. Math. Lett.*, vol. 22, no. 9, pp. 1467–1470, Sep. 2009. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0893965909001475>



FILIFE RODRIGUES DE SOUZA MOREIRA

received the B.Sc. degree in mechanical engineering from the Instituto Tecnológico de Aeronáutica (ITA), Brazil, the first M.Sc. degree in guided weapon systems from Cranfield University, U.K., in 2011, and the second M.Sc. degree in operational research from the ITA, Brazil, in 2012. He is currently a Lecturer in control theory, operational research, and statistics and probability of the aerial weapon engineering specialization course,

the operations analysis specialization course, and the engineering graduation at the Fundamental Sciences Department, ITA. In addition, he attends a Doctoral Program at ITA.



FILIFE ALVES NETO VERRI received the B.Sc. degree (Hons.) in computer science and the Ph.D. degree (Hons.) in computer science and computational mathematics from the University of São Paulo, in 2014 and 2018, respectively. He was a Visiting Scholar with the School of Electrical, Computer and Energy Engineering, Arizona State University, in 2016, under the supervision of Prof. Ying-Cheng Lai. He is currently an Assistant Professor with the Division of Computer Science,

Department of Computer Methods, Instituto Tecnológico de Aeronáutica (ITA). His research interests include data science, machine learning, complex networks, and complex systems.



TAKASHI YONEYAMA

received the B.Eng. and M.Sc. degrees in electronic engineering from the Instituto Tecnológico de Aeronáutica (ITA), Brazil, the M.D. degree in medicine from the Universidade de Taubaté, Brazil, and the Ph.D. degree in electrical engineering from Imperial College London, U.K., in 1983. He is currently a Professor of control theory at the Electronic Engineering Department, ITA. He has published more than 100 journal articles and 370 conference

papers, has written four books, and has supervised more than 95 thesis/dissertations. His research is concerned mainly with stochastic control theory. From 2004 to 2006, he worked as the President of the Brazilian Automatics Society.

...