Neural Networks ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Hybrid intelligent methodology to design translation invariant morphological operators for Brazilian stock market prediction

Ricardo de A. Araújo

Information Technology Department, [gm][2] Intelligent Systems, Campinas, SP, Brazil

ARTICLE INFO

Article history:
Received 11 August 2009
Received in revised form 4 June 2010
Accepted 17 June 2010

Keywords:
Hybrid intelligent systems
Mathematical morphology
Morphological neural networks
Translation invariant morphological operators
Quantum-inspired evolutionary algorithms
Stock market prediction
Financial time series prediction

ABSTRACT

This paper presents a hybrid intelligent methodology to design increasing translation invariant morphological operators applied to Brazilian stock market prediction (overcoming the random walk dilemma). The proposed Translation Invariant Morphological Robust Automatic phase-Adjustment (TIMRAA) method consists of a hybrid intelligent model composed of a Modular Morphological Neural Network (MMNN) with a Quantum-Inspired Evolutionary Algorithm (QIEA), which searches for the best time lags to reconstruct the phase space of the time series generator phenomenon and determines the initial (sub-optimal) parameters of the MMNN. Each individual of the QIEA population is further trained by the Back Propagation (BP) algorithm to improve the MMNN parameters supplied by the QIEA. Also, for each prediction model generated, it uses a behavioral statistical test and a phase fix procedure to adjust time phase distortions observed in stock market time series. Furthermore, an experimental analysis is conducted with the proposed method through four Brazilian stock market time series, and the achieved results are discussed and compared to results found with random walk models and the previously introduced Time-delay Added Evolutionary Forecasting (TAEF) and Morphological-Rank-Linear Time-lag Added Evolutionary Forecasting (MRLTAEF) methods.

1. Introduction

The stock market prediction problem is considered a rather difficult problem due to the many complex features frequently present in the stock market (irregularities, volatility, trends and noise). Several approaches have been studied for the development of predictive models able to predict the stock market, based on its past and present data.

In order to solve such a problem, a wide number of linear statistical models have been proposed (Box, Jenkins, & Reinsel, 1994). Among them, the popular linear statistical approach based on Auto Regressive Integrated Moving Average (ARIMA) models (Box et al., 1994) is one of the most common choices of financial traders. However, since the ARIMA models are linear and most real world applications involve nonlinear problems, this can introduce an accuracy limitation of the generated forecasting models.

In this way, nonlinear statistical approaches have been presented in the literature to overcome linear statistical model limitations, such as the bilinear models (Rao & Gabr, 1984), the threshold autoregressive models (Ozaki, 1985), the exponential autoregressive models (Priestley, 1988), and the general state dependent models (Rumelhart & McCleland, 1987), among others. The drawbacks of those nonlinear statistical models are the high mathematical complexity associated with them (resulting in many situations in similar performances to the linear models) and the need, most of the time, of a problem dependent specialist to validate the predictions generated by the model, limiting the development of an automatic prediction system (Clements, Franses, & Swanson, 2004).

Alternatively, Artificial Intelligence (AI) approaches, in particular Artificial Neural Networks (ANNs), have been applied for nonlinear modeling of time series (Crottel, Girard, Girard, Mangeas, & Muller, 1995; Hocevar, Širok, & Blagojevic, 2005; Khotanzad, Elragal, & Lu, 2000; Myhre, 1992; Preminger & Franck, 2007; Sitte & Sitte, 2002; Zhang & Kline, 2007; Zhang, Patuwo, & Hu, 1998). However, ANNs require the definition of a wide number of system parameters, which are not easy to determine, such as the topology, the number of hidden processing units, the training algorithm (and its corresponding parameters), among others. It is worth mentioning that in the particular case of the time series prediction problem, another crucial element necessary to determine is the relevant time lags to represent the time series (Araújo & Ferreira, 2009).

In this context, hybrid intelligent prediction models have been proposed for such (Araújo & Ferreira, 2009; Binner, Kendall, & Gazely, 2004; Ferreira, Vasconcelos, & Adeodato, 2008; Islan, Yao, & Muraz, 2003; Leung, Lam, Ling, & Tam, 2003; Matilla-García & Argüello, 2005; Miller, Arquello, & Greenwood, 2004; Stanley & Miikkulainen, 2002; Yao, 1999). However, these models use classical Evolutionary Algorithms (EAs), which present a

E-mail addresses: ricardo@gm2.com.br, araujora@gmail.com.

0893-6080/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.neunet.2010.06.007

slower performance due to time-consuming fitness functions. In order to improve the EAs performance, da Cruz, Vellasco, and Pacheco (2006) presented a new Quantum-Inspired Evolutionary Algorithm (QIEA), which uses a novel real-valued representation, showing superior performance when compared to other well-established classical and quantum-inspired EAs.

However, a dilemma arises from all these models regarding stock market time series, known as the random walk dilemma (Sitte & Sitte, 2002), where the predictions generated by such models show a characteristic one step delay with regard to the original time series data. This behavior has been seen as a dilemma regarding the time series representation, where it has been posed that the time series follow a random walk like model and cannot, therefore, be predicted (Malkiel, 2003).

On the other hand, nonlinear operators (based on the framework of Mathematical Morphology (MM) (Maragos, 1989; Serra, 1982)) have been widely applied to signal processing. Many works have focused on the design of morphological operators (Coyle & Lin, 1988; Davidson & Hummer, 1993; Herwing & Shalkoff, 1994; Loce & Dougherty, 1992; Maragos, 1989; Wilson, 1989; Yang & Maragos, 1992). In the branch of filters and Artificial Intelligence integration, Pessoa (1997) also proposed a neural network architecture involving Morphological–Rank–Linear (MRL) operators at every processing node.

It is worth mentioning that the Morphological neural networks (MNNs) represent an important class of nonlinear systems and differ from classical ANNs in the sense that the computation in each node is carried out by simple morphological operators in the context of Image Algebra (Ritter, 1991). MNNs are typically used in image and signal processing applications. Based on the fundamental result of minimal representations of translation invariant set mappings via Mathematical Morphology, proposed by Banon and Barrera (1991), as well as based on the Matheron decomposition (Matheron, 1975) of increasing translation invariant operators, Sousa (2000) presented a general morphological neural network architecture, referred to as Modular Morphological Neural Network (MMNN). The MMNN training is via the Back Propagation (BP) algorithm, which relies on the methodology of Pessoa (1997) for overcoming the problem of nondifferentiability of rank functions in the training equations.

In this work a hybrid intelligent methodology, referred to as Translation Invariant Morphological Robust Automatic phase-Adjustment (TIMRAA), to design increasing translation invariant morphological operators is presented to overcome the random walk dilemma for Brazilian stock market prediction. An experimental analysis is conducted with the proposed approach using four Brazilian stock market time series, and five well-known performance measurements are used to assess the performance of the proposed method. The proposed method is compared to random walk models and the previously introduced Time-delay Added Evolutionary Forecasting (TAEF) (Ferreira et al., 2008) and Morphological–Rank–Linear Time-lag Added Evolutionary Forecasting (MRLTAEF) (Araújo & Ferreira, 2009) methods, which showed boosted performance when compared to classical prediction models (Araújo, Madeiro, Sousa, Pessoa, & Ferreira, 2006; Araújo, Vasconcelos, & Ferreira, 2007; Binner et al., 2004; Box et al., 1994; Matilla-García & Argüello, 2005; Ozaki, 1985; Priestley, 1988; Rao & Gabr, 1984; Rumelhart & McCleland, 1987).

This work is organized as follows. In Section 2 we present the fundamentals of the time series prediction problem, the random walk dilemma for stock market prediction and the QIEA. Section 3 presents the concepts of mathematical morphology, the MMNN fundamentals and its training algorithm. Section 4 describes the proposed TIMRAA method. Section 5 shows the performance measures used to assess the proposed TIMRAA method. In Section 6 we present simulations and experimental results with the proposed TIMRAA method, as well as a comparison between the obtained results and those given by the TAEF and MRLTAEF methods for four relevant Brazilian stock market time series. Finally, in Section 7, we present the conclusions of this work.

2. Background

In this section the fundamentals and theoretical concepts necessary for comprehension of the proposed TIMRAA method will be presented.

2.1. The time series prediction problem

A time series is a sequence of observations of a given phenomenon, which is observed in a discrete or continuous space. In this work all time series will be considered time discrete and equidistant.

Commonly, a time series can be defined by

$$X_t = \{x_t \in \mathbb{R} \mid t = 1, 2, \ldots, N\}, \tag{1}$$

where t is the temporal index and N is the number of observations. The term X_t will be seen as a set of temporal observations of a given phenomenon, orderly sequenced and equally spaced.

The goal of prediction models applied to a given time series (X_t) is to provide a mechanism that allows, with a certain accuracy, the prediction of the future values of X_t, given by X_{t+k}, $k = 1, 2, \ldots$, where k represents the prediction horizon. These mechanisms will try to identify certain regular patterns present in the data set, creating a model capable of generating the next temporal patterns, where, in this context, the most relevant factor for an accurate prediction of performance is the correct choice of the past window, or the time lags, considered for the time series representation.

Box et al. (1994) showed that when there is a clear linear relationship among the historical data of a given time series, the auto-correlation and partial auto-correlation functions are able to identify the relevant time lags to represent a time series, and such procedure is widely applied in linear models. However, when it uses a real world time series, or more specifically, a complex time series with all their dependencies on exogenous and uncontrollable variables, the relationship that involves the time series historical data is generally nonlinear, which makes the Box and Jenkins' analysis procedure of the time lags only a crude estimate.

In the mathematical sense, such a relationship involving time series historical data defines a d-dimensional phase space, where d is the minimum dimension capable of representing such a relationship. Therefore, a d-dimensional phase space can be built so that it is possible to unfold its corresponding time series. Takens (1980) proved that if d is sufficiently large, such a phase space is homeomorphic to the phase space that generated the time series. Takens' Theorem (Takens, 1980) is the theoretical justification that it is possible to build a state space using the correct time lags, and if this space is correctly rebuilt, Takens' Theorem (Takens, 1980) also guarantees that the dynamics of this space is topologically identical to the dynamics of the real system state space.

The main problem in reconstructing the original state space is naturally the correct choice of the variable d, or more specifically, the correct choice of the important time lags necessary for the characterization of the system dynamics. Many proposed methods can be found in the literature for the definition of the lags (Pi & Peterson, 1994; Savit & Green, 1991; Tanaka, Okamoto, & Naito, 2001). Such methods are usually based on measures of conditional probabilities, which consider,

$$X_t = f(x_{t-1}, x_{t-2}, \ldots, x_{t-d}) + r_t, \tag{2}$$

Please cite this article in press as: Araújo, R. A. Hybrid intelligent methodology to design translation invariant morphological operators for Brazilian stock market prediction. Neural Networks (2010), doi:10.1016/j.neunet.2010.06.007

where $f(x_{t-1}, x_{t-2}, \ldots, x_{t-d})$ is a possible mapping of the past values to the facts of the future (where x_{t-1} is the lag 1, x_{t-2} is the lag $2, \ldots, x_{t-d}$ is the lag d) and r_t is a noise term.

However, in general, these tests found in the literature are based on the primary dependence among the variables and do not consider any possible induced dependencies. For example, if

$$f(x_{t-1}) = f(f(x_{t-2})), \tag{3}$$

it is said that x_{t-1} is the primary dependence, and the dependence induced on x_{t-2} is not considered (any variable that is not a primary dependence is denoted as irrelevant).

The method proposed in this paper, conversely, does not make any prior assumption about the dependencies between the variables. In other words, it does not discard any possible correlation that can exist among the time series parameters, even higher order correlations, since it carries out an iterative automatic search in solving the problem of finding the relevant time lags.

2.2. The random walk dilemma

A naive prediction strategy is to define the last observation of a time series as the best prediction of its next future value ($x_{t+1} = x_t$). This kind of model is known as the Random Walk (RW) model (Mills, 2003), which is defined by

$$x_t = x_{t-1} + r_t, \tag{4}$$

or

$$\Delta x_t = x_t - x_{t-1} = r_t, \tag{5}$$

where x_t is the current observation, x_{t-1} is the immediate observation before x_t, and r_t is a noise term with a Gaussian distribution of zero mean and standard deviation σ $(r_t \approx N(0, \sigma))$. In other words, the rate of time series change (Δx_t) is a white noise.

The model above clearly implies that, as the information set consists of past time series data, the future data are unpredictable. On average, the value x_t is indeed the best prediction of value x_{t-1}. This behavior is common in the finance market and in economic theory, and is known as the random walk dilemma or random walk hypothesis (Mills, 2003).

The computational cost for time series forecasting using the random walk dilemma is extremely low. Therefore, any other prediction method more costly than a random walk model should have a very superior performance than a random walk model, otherwise its use is not interesting in practice.

However, if the time series phenomenon is driven by a law with a strong similarity to a random walk model, any model applied to this time series phenomenon will tend to have the same performance as a random walk model.

Assuming that an accurate prediction model is used to build an estimated value of x_t, denoted by $\widehat{x_t}$, the expected value ($E[\cdot]$) of the difference between $\widehat{x_t}$ and x_t must tend to zero,

$$E[\widehat{x_t} - x_t] \to 0. \tag{6}$$

If the time series generator phenomenon is supposed to have a strong random walk linear component and a very weak nonlinear component (denoted by $g(t)$), and assuming that $E[r_t] = 0$ and $E[r_t r_k] = 0$ $(\forall k \neq t)$, the expected value of the difference between $\widehat{x_t}$ and x_t will be

$$E[\widehat{x_t} - (x_{t-1} + g(t) + r_t)] \to 0$$
$$E[\widehat{x_t}] - E[x_{t-1}] - E[g(t)] - E[r_t] \to 0$$
$$E[\widehat{x_t}] - E[x_{t-1}] - E[g(t)] \to 0$$
$$E[\widehat{x_t}] \to E[x_{t-1}] + E[g(t)].$$

But $E[g(t)] \to 0$, then $E[x_{t-1}] + E[g(t)] \simeq E[x_{t-1}]$ and

$$E[\widehat{x_t}] \to E[x_{t-1}]. \tag{7}$$

Therefore, in these conditions, to escape the random walk dilemma is a hard task. Indications of this behavior (strong linear random walk component and a weak nonlinear component) can be observed from time series lagplot graphics. For example, lagplot graphics where strong linear structures are dominant with respect to nonlinear structures (Kantz & Schreiber, 2003), generally observed in the financial and economical time series.

2.3. The quantum-inspired evolutionary algorithm

Quantum-Inspired Evolutionary Algorithms (QIEAs) are based on quantum bits concepts, referred to as Qubits, and also on the superposition of states of quantum computing (Han & Kim, 2002), which can be expressed as a linear combination between states $|0\rangle$ and $|1\rangle$. A Qubit can be defined (Han and Kim (2002)) by

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \tag{8}$$

where $\alpha, \beta \in \mathbb{C}$ represent the probability amplitudes of states $|0\rangle$ and $|1\rangle$, respectively.

In this way, $|\alpha|^2$ and $|\beta|^2$ gives the probability that the Qubit will be found in states $|0\rangle$ and $|1\rangle$, respectively, where the normalization of the state to unity guarantees (Han & Kim, 2002)

$$|\alpha|^2 + |\beta|^2 = 1. \tag{9}$$

Inspired by concept of quantum computing and based on (Han & Kim, 2002), da Cruz et al. (2006) proposed a new QIEA using a novel real-valued representation, which is described in Fig. 1.

2.3.1. Quantum population generation

The QIEA quantum population at generation G is defined by $\underline{QP}_G$, which is given by a superposition of states that are observed to generate classical individuals (possible solutions of the problem), defined by,

$$\underline{QP}_G = (\underline{QP}_{1;G}, \underline{QP}_{2;G}, \ldots, \underline{QP}_{S;G}), \tag{10}$$

with

$$\underline{QP}_{i;G} = (QP_{i1}, QP_{i2}, \ldots, QP_{iN}), \tag{11}$$

and

$$QP_{ij} = (\mu_{ij}, \sigma_{ij}), \tag{12}$$

in which $\underline{QP}_G$ denotes the quantum population at generation G; $\underline{QP}_{i;G}$ denotes i-th quantum individual of the population $\underline{QP}_G$; QP_{ij} denotes the j-th parameter (gene) of the i-th individual of the population; S denotes the quantum population size (number of quantum individuals); N denotes the number of individual parameters (genes or variables of the problem). Terms μ_{ij} and $\sigma_{ij} \in \mathbb{R}$ represent the center and the width of a square pulse, which is used in order to build the set of possible observable values over the problem domain da Cruz et al. (2006). The height (h_{ij}) of each pulse is defined using the quantum gene width (σ_{ij}) and the maximum number of quantum individuals (N) in the quantum population, and is given by da Cruz et al. (2006)

$$h_{ij} = \frac{1/\sigma_{ij}}{N}. \tag{13}$$

2.3.2. Classical individuals generating

In this step the interference process among quantum individuals is performed to generate a Probability Density Function (PDF) (da Cruz et al., 2006). The PDF consists of summing up the quantum individuals genes. In other words, the first genes of all quantum individuals are summed. The same is done for all the other genes of a given quantum individual. The PDF is defined by da Cruz et al. (2006) as

$$PDF_j = \sum_{i=1}^{N} QP_{ij}, \tag{14}$$

Please cite this article in press as: Araújo, R. A. Hybrid intelligent methodology to design translation invariant morphological operators for Brazilian stock market prediction. Neural Networks (2010), doi:10.1016/j.neunet.2010.06.007

```
begin QIEA
    t = 0;                // t: actual iteration
    create quantum population;
    while not termination condition do
        t = t + 1;
        create the PDF's using quantum individuals;
        generate temporary classical population observing quantum population and using CDF's;
        if t=1 then
            classical population ⟵ temporary classical population;
        else
            temporary classical population ⟵ crossover operator between temporary classical
            population and classical population;
            evaluate temporary classical population;
            classical population ⟵ K best individuals from temporary classical population;
        end
        begin with the individuals of classical population
            quantum population ⟵ apply translate operation to classical population;
            quantum population ⟵ apply resize operation to quantum population;
        end
    end
end
```

Fig. 1. Quantum-inspired evolutionary algorithm procedure.

where QP_{ij} represents a square pulse with width $_{ij}$ and center $_{ij}$ of the j-th gene of the i-th quantum individual.

According to da Cruz et al. (2006) these PDFs are able to generate the classical individuals, which is a real-valued vector with same number of quantum individuals' genes, where such values are randomly selected using the PDFs as a probability function. In order to perform such a random selection, da Cruz et al. (2006) defined a Cumulative Distribution Function (CDF), which is given by

$$CDF_j.x/ = \int_l^u PDF_j.x/dx; \tag{15}$$

where u and l represent the upper and lower bounds of the PDF_j function.

After, as all PDFs are made by a sum of square pulses, it is possible to calculate the PDF area by dividing the function curve into rectangles and by summing up its corresponding area, and then it is possible to verify that CDFs can be calculated using such PDFs based on these rectangles (da Cruz et al., 2006).

Through these CDFs, according to da Cruz et al. (2006), it is possible to generate a set of classical individuals by using such curves. Such a classical population is created by an uniform choice of an aleatory number between 0 and 1 and then by identifying this point in the CDF. This operation is mathematically defined by da Cruz et al. (2006) as

$$x = CDF^{-1}.r/; \tag{16}$$

where r is an aleatory number in the range $[0; 1]$.

This procedure allows the building of the Temporary Classical Population (TCP), which is responsible for the classical individuals observed in the quantum population. At the first QIEA generation, the Classical Population (CP), which is the best observation (in terms of fitness function) of the quantum population—a clone of the TCP. For the next generations, the multi-point crossover operator is applied to the classical population of the QIEA in order to generate better classical individuals, hence improving the quantum population.

2.3.3. Quantum population update

After the CP generation, it is necessary to update the quantum individuals in the quantum population. The first step, referred to as the translate operation, is responsible for updating the center ( ) of each quantum gene. In this way, da Cruz et al. (2006) showed a simple procedure to do this, which is done by making the mean of each gene value equal to the value of the genes from the classical individuals. This step is formally defined by,

$$_{ij} = CP_{ij}; \tag{17}$$

where $_{ij}$ represents the center of the j-th gene of the i-th quantum individual from the quantum population, and CP_{ij} denotes the j-th gene of the i-th classical individual from CP.

The second step, referred to as the resize operation, is responsible for reducing or enlarging the width ($_{ij}$) of the quantum gene (da Cruz et al., 2006). This change should be made homogeneously for all quantum genes and for all quantum individuals (da Cruz et al., 2006). da Cruz et al. (2006) used the 1=5th rule to determine if such a width should be enlarged or reduced, which is given by da Cruz et al. (2006) as

$$_{ij} = \begin{cases} _{ij}\cdot & \text{if } ' < 1=5 \\ _{ij}= & \text{if } ' > 1=5 \\ _{ij} & \text{otherwise;} \end{cases} \tag{18}$$

where $_{ij}$ represents the width of the j-th gene of the i-th quantum individual from the quantum population, denotes an aleatory number in the interval $[0; 1]$, and $'$ is the rate giving how many classical individuals generated in new generation have their overall evaluation improved.

3. Mathematical morphology

The Mathematical Morphology (MM) method is based on two basic operations, the sum and subtraction of Minkowski (1911), which are respectively given by Sousa (2000)

$$X \oplus B = \bigcup_{b \in B} X_b; \tag{19}$$

$$X \ominus B = \bigcap_{b \in B^r} X_b; \tag{20}$$

where $X_b = \{x + b : x \in X\}$ represents the input signal and $B^r = \{-b : b \in B\}$ is the reflected structuring element B.

All of MM transformations are based on combinations of four basic operations, which are defined by Sousa (2000)

$$\text{Dilation: } {}_B.X/ = X \oplus B; \tag{21}$$

$$\text{Erosion: } {}_B.X/ = X \ominus B; \tag{22}$$

$$\text{Anti-Dilation: } {}^a_B.X/ = .X \oplus B^{rc}/^{rc}; \tag{23}$$

$$\text{Anti-Erosion: } {}^a_B.X/ = .X \ominus B^{rc}/^{rc}; \tag{24}$$

where $B^{rc} = \{-b : b \notin B\}$ represents the reflected complement of the structuring element B.

Please cite this article in press as: Araújo, R. A. Hybrid intelligent methodology to design translation invariant morphological operators for Brazilian stock market prediction. Neural Networks (2010), doi:10.1016/j.neunet.2010.06.007

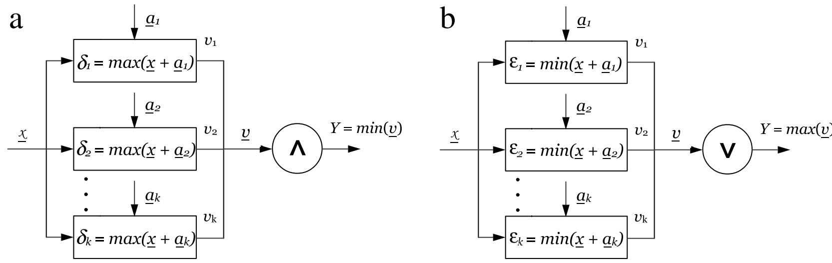

Fig. 2. MMNN architectures for the Matheron Decomposition.

According to Sousa (2000), an operator of kind $: P(E) \to P(E)$, where $P(E)$ represents all subsets of $E = \mathbb{R}^N$, may be a translation invariant (Eq. (25)), increasing (Eq. (26)), decreasing (Eq. (27)) or window (Eq. (28)).

$$(X_{\underline{h}}) = (X)_{\underline{h}}; \tag{25}$$

where $X_{\underline{h}} = \{x + \underline{h} : x \in X\}$ represents the translation of $X \in P(E)$ by vector $\underline{h} \in E$.

$$X \subset Y \Rightarrow (X) \subset (Y), \quad \forall X, Y \in P(E), \tag{26}$$

$$X \supset Y \Rightarrow (X) \supset (Y), \quad \forall X, Y \in P(E), \tag{27}$$

$$\forall x \in E, \quad x \in (X) \Leftrightarrow x \in (X \cap L_x), \tag{28}$$

where L_x is the translation of $L \in E$ finite.

The Matheron Decomposition Theorem (MDT) (Matheron, 1975) guarantees that any increasing translation invariant operator may be decomposed by a union of erosion operators or an intersection of dilation operators. Below, the formal definition of the MDT is presented.

Let $: P(E) \to P(E)$ be a window operator (L), increasing and translation invariant, there is a family of sets $A_k \subset P(E)$, $k = 1, 2, \ldots, ND$, in which the decomposition of by dilations or erosions is defined respectively by Sousa (2000)

$$= \bigcap_{k}^{ND} {}^{L}_{A_k}(X), \tag{29}$$

$$= \bigcup_{k}^{ND} {}^{L}_{A_k}(X), \tag{30}$$

where ND represents the number of decompositions.

3.1. Modular Morphological Neural Network (MMNN) preliminaries

Definition 1 (*Rank Function*)**.** The r-th rank function of the vector $\underline{t} = (t_1, t_2, \ldots, t_n) \in \mathbb{R}^n$ is the r-th element of the vector $\underline{t}$ sorted in decreasing order $(t_{(1)} \geq t_{(2)} \geq \cdots \geq t_{(n)})$. It is denoted by Pessoa and Maragos (1998) as

$$\mathcal{R}_r(\underline{t}) = t_{(r)}, \quad r = 1, 2, \ldots, n. \tag{31}$$

For example, given the vector $\underline{t} = (3, 0, 5, 7, 2, 1, 3)$, its 4-th rank function is $\mathcal{R}_4(\underline{t}) = 3$.

Definition 2 (*Unit Sample Function*)**.** The unit sample function is given by Pessoa and Maragos (1998) as

$$q(v) = \begin{cases} 1, & \text{if } v = 0, \\ 0, & \text{otherwise} \end{cases} \tag{32}$$

where $v \in \mathbb{R}$.

Applying the unit sample function to a vector $\underline{v} = (v_1, v_2, \ldots, v_n) \in \mathbb{R}^n$, yields a vector unit sample function $(Q(\underline{v}))$, given by Pessoa and Maragos (1998) as

$$Q(\underline{v}) = [q(v_1), q(v_2), \ldots, q(v_n)]. \tag{33}$$

Definition 3 (*Rank Indicator Vector*)**.** The r-th rank indicator vector $\underline{c}$ of $\underline{t}$ is given by Pessoa and Maragos (1998) as

$$\underline{c}(\underline{t}, r) = \frac{Q\left(z \cdot \underline{1} - \underline{t}\right)}{Q\left(z \cdot \underline{1} - \underline{t}\right) \cdot \underline{1}^T}, \tag{34}$$

where $z = \mathcal{R}_r(\underline{t})$, $\underline{1} = (1, 1, \ldots, 1)$, "$\cdot$" represents a scalar product and symbol T denotes transposition.

For example, given the vector $\underline{t} = (3, 0, 5, 7, 2, 1, 3)$, its 4-th rank indicator function is $\underline{c}(\underline{t}, 4) = \frac{1}{2}(1, 0, 0, 0, 0, 0, 1)$.

Definition 4 (*Smoothed Rank Function*)**.** The smoothed r-th rank function is given by Pessoa and Maragos (1998) as

$$\mathcal{R}_{r,}(\underline{t}) = \underline{c}(\underline{t}, r) \cdot \underline{t}^T, \tag{35}$$

with

$$\underline{c}(\underline{t}, r) = \frac{Q(z \cdot \underline{1} - \underline{t})}{Q(z \cdot \underline{1} - \underline{t}) \cdot \underline{1}^T}, \tag{36}$$

where c is an approximation for the rank function $\underline{c}$ and $Q(\underline{v}) = [q(v_1), q(v_2), \ldots, q(v_n)]$ is a smoothed impulse function (where $q(v)$ is like $\text{sech}^2(v/\ )$) (where sech is the hyperbolic secant), ≥ 0 is a scale parameter and "$\cdot$" represents the scalar product.

Thus, $\underline{c}$ is an approximation for the rank indicator vector $\underline{v}$. Using ideas from fuzzy set theory, $\underline{c}$ can also be interpreted as a membership function vector (Pessoa & Maragos, 1998). For example, if the vector $\underline{t} = (3, 0, 5, 7, 2, 1, 3)$, $q(v) = \text{sech}^2\left(\frac{v}{}\right)$ and $= 0.5$ then its smoothed 4-th rank indicator function is

$$\underline{c}(\underline{t}, 4) = \frac{1}{2}(0.9646, 0, 0.0013, 0, 0.0682, 0.0013, 0.9646),$$

where $\underline{c}(\underline{t}, 4) = \frac{1}{2}(1, 0, 0, 0, 0, 0, 1)$.

3.2. MMNN definition

Sousa (2000) defined the MMNN for designing translation invariant operators that satisfy the MDT (Matheron, 1975) for dilations as well as for erosions. Fig. 2(a) presents the MMNN architecture for the Matheron Decomposition (Matheron, 1975) by dilations.

The following equations define the MMNN architecture for the Matheron Decomposition (Matheron, 1975) via dilations according to this approach.

$$v_k = {}_k = \max\left(\underline{x} + \underline{a}_k\right), \tag{37}$$

where $\underline{x}$ represents the MMNN input signal.

$$\text{MMNN Output: } Y = \min(\underline{v}), \tag{38}$$

in which

$$\underline{v} = (v_1, v_2, \ldots, v_k). \tag{39}$$

Please cite this article in press as: Araújo, R. A. Hybrid intelligent methodology to design translation invariant morphological operators for Brazilian stock market prediction. Neural Networks (2010), doi:10.1016/j.neunet.2010.06.007

The MMNN weights matrix, A, is defined by

$$A = (\underline{a}_1, \underline{a}_2, \ldots, \underline{a}_k), \tag{40}$$

in which $\underline{a}_k \in \mathcal{R}^k, k = 1, 2, \ldots,$ ND represents the MMNN weights (i.e., matrix rows composed by structuring elements $\underline{a}_k$). The Symbol $\wedge$ represents the minimum operator.

In a dual manner, the MMNN architecture for the Matheron Decomposition (Matheron, 1975) via erosions is defined by substituting dilations by erosions and symbol $\wedge$ by $\vee$, where $\vee$ represents the maximum operation. Fig. 2(b) presents the MMNN architecture for the Matheron Decomposition (Matheron, 1975) by erosions.

3.3. MMNN training algorithm

Based on the Back Propagation (BP) algorithm. Sousa (2000) defined the MMNN training for Matheron Decomposition (Matheron, 1975), which is formally defined by the following equations (Sousa, 2000):

$$A(n+1) = A(n) - \nabla_A J(A), \quad n = 0, 1, \ldots \tag{41}$$

in which A is the weight matrix, is the learning rate and $\nabla_A J(A)$ is the gradient matrix of a cost function $J(A)$ (to be minimized with respect to the weight matrix A). For a given training set,

$$\left\{(\underline{x}_m, d_m), \ m = 1, 2, \ldots, M\right\}, \tag{42}$$

where d_m is the desired output of a given input $\underline{x}_m$ and M is the number of patterns of a training set, $J(A)$ is defined by

$$J(A) = \frac{1}{2} e_m^2, \tag{43}$$

where $e_m = d_m - y_m$ is the difference between the desired output and the actual output for the input $\underline{x}_m, m = 1, 2, \ldots, M$. The gradient presented in Eq. (41) is given by Sousa (2000)

$$\frac{\partial J}{\partial \underline{a}_k} = -e \frac{\partial y}{\partial v_k} \frac{\partial v_k}{\partial \underline{a}_k}, \quad k = 1, 2, \ldots, \text{ND}. \tag{44}$$

According to Sousa (2000), the partial derivatives in Eq. (44) are estimated by the methodology of Pessoa and Maragos (1998) via rank indication vectors $\underline{c}$ and smooth impulse functions Q. In matrix terms, the gradient may be defined by Sousa (2000) as

$$\nabla_A J(A) = -e \cdot \text{diag}(\underline{c}) \cdot C, \tag{45}$$

in which $C = (\underline{c}_1, \underline{c}_2, \ldots, \underline{c}_k)$. Term "$\cdot$" represents the scalar product. Terms $\underline{c}$ and $\underline{c}_k$ are defined by Matheron Decomposition (Matheron, 1975) via dilations by Sousa (2000) as

$$\underline{c} = \frac{Q(\min(\underline{v}) \cdot \underline{1} - \underline{v})}{Q(\min(\underline{v}) \cdot \underline{1} - \underline{v}) \cdot \underline{1}^T}, \tag{46}$$

$$\underline{c}_k = \frac{Q(\max(\underline{x} + \underline{a}_k) \cdot \underline{1} - \underline{x} - \underline{a}_k)}{Q(\max(\underline{x} + \underline{a}_k) \cdot \underline{1} - \underline{x} - \underline{a}_k) \cdot \underline{1}^T}, \tag{47}$$

where T denotes transposition and "$\cdot$" represents a scalar product.

In a similar way, terms $\underline{c}$ and $\underline{c}_k$ are defined by Matheron Decomposition (Matheron, 1975) via erosions by Sousa (2000) as

$$\underline{c} = \frac{Q(\max(\underline{v}) \cdot \underline{1} - \underline{v})}{Q(\max(\underline{v}) \cdot \underline{1} - \underline{v}) \cdot \underline{1}^T}, \tag{48}$$

$$\underline{c}_k = -\frac{Q(\min(\underline{x} - \underline{a}_k) \cdot \underline{1} - \underline{x} + \underline{a}_k)}{Q(\min(\underline{x} - \underline{a}_k) \cdot \underline{1} - \underline{x} + \underline{a}_k) \cdot \underline{1}^T}. \tag{49}$$

4. The proposed translation invariant morphological robust automatic phase-adjustment method

The methodology proposed in this work uses a quantum-inspired evolutionary search mechanism in order to design increasing translation morphological operators (to train and to adjust the Modular Morphological Neural Network (MMNN)), overcoming in an optimal way the random walk dilemma for stock market time series prediction, as well as to optimally define the time lag dimensionality of a given time series in order to better characterize its generator phenomenon. It is based on the definition of the three main elements necessary for building an accurate prediction system (Araújo & Ferreira, 2009): (i) The underlying information necessary to predict the time series (the minimum number of time lags adequate for representing the time series), (ii) the structure of the model capable of representing such underlying information for the purpose of prediction (the number of modules in the MMNN structure), and (iii) the appropriate algorithm for training the model. It is important to consider the minimum possible number of time lags in the correct time series representation because the model must to be as parsimonious as possible.

The proposed Translation Invariant Morphological Robust Automatic phase-Adjustment (TIMRAA) method consists of a hybrid intelligent model composed of a MMNN with a QIEA. It uses two QIEAs with distinct purposes. The first QIEA is responsible for the time lag dimensionality (MaxLags) definition (each individual represents a time lag dimensionality within the range [1, MaxDim]). For each individual generated of the classical population (represented by term MaxLags$_i$ with $i = 1, 2, \ldots, S$, where S represents the individuals amount) in the first QIEA, it performs a second QIEA to determine the following important parameters: (i) The particular time lags capable of a fine tuned time series characterization in the interval [1, MaxLags$_i$] for each individual of population, and (ii) the weights ($\underline{a}_k$), architecture (by dilations or by erosions—MMNNArch) and number of modules of the MMNN (NModules): initially, a maximum number of MMNN modules (MMNNModulesMax) is pre-defined and then the second QIEA chooses, for each individual candidate, the weights, the most adequate MMNN architecture and the number of MMNN modules in the range [1, MMNNModulesMax]. The Back Propagation (BP) (Sousa, 2000) is then used to train each individual of the second QIEA classical population, as it has been proved to be effective in speeding up the training while limiting its computational complexity. After BP training, for each candidate solution, a behavioral statistical test is employed to adjust time phase distortions that appear in stock market time series.

The idea proposed here is to use the first QIEA in order to search for a better time lag dimensionality according to Takens Theorem (Takens, 1980) and the second QIEA to determine, according to a given time lag dimensionality, better specific time lags to optimally describe the time series generator phenomenon, as well as to evolve the complete MMNN architecture and parameters. Also, the second QIEA idea is to conjugate a local search method (BP) to a global search method (QIEA). While the QIEA makes possible the testing of varied solutions in different areas of the solution space, the BP acts on the initial solution to produce a fine-tuned forecasting model.

Those processes offer the effective capacity to seek the most compact MMNN, reducing the computational cost and probability of model overfitting. Each individual of the first QIEA represents a time lag dimensionality and each individual of the second QIEA represents a MMNN, where its input is defined by the number of time lags and its output represents the prediction horizon of one step ahead. Fig. 3 shows the scheme of the proposed TIMRAA method.

Please cite this article in press as: Araújo, R. A. Hybrid intelligent methodology to design translation invariant morphological operators for Brazilian stock market prediction. Neural Networks (2010), doi:10.1016/j.neunet.2010.06.007

Fig. 3. The scheme of the proposed TIMRAA method.

Initially, it is necessary to define some parameters of the TIMRAA method, the maximum time lag dimensionality (MaxDim) and the maximum number of MMNN modules (MMNNModulesMax). These parameters are initially user predefined and they are adjusted when the method evolves through the search for the most fitted forecasting model and for the time lag dimensionality to describe the time series generator phenomenon.

Most works found in the literature have a fitness function (or objective function) based on just one performance measure, like Mean Square Error (MSE). However, Clements and Hendry (1993), in 1993, showed that the MSE measure has some limitations compared to available predictive model performance. The information as the absolute percentage error, the accuracy in the future direction prediction and the relative gain regarding naive predictive models (such as random walk models and mean prediction) are not well-described when it uses only MSE measure.

In order to provide a more robust forecasting model, Araújo and Ferreira (2009) used an alternative fitness function, which is a combination of five well-known performance measures: Prediction Of Change In Direction (POCID), Mean Square Error (MSE), Mean Absolute Percentage Error (MAPE), U of Theil Statistics (THEIL) and Average Relative Variance (ARV), where all these measures will be formally defined in Section 5. The fitness function used here is given by Araújo and Ferreira (2009)

$$\text{Fitness Function} = \frac{\text{POCID}}{1 + \text{MSE} + \text{MAPE} + \text{THEIL} + \text{ARV}}. \tag{50}$$

While there are linear and nonlinear metrics in the fitness function and each one of the metrics can contribute in different ways to the evolution process, Eq. (50) was built by Araújo and Ferreira (2009) empirically in order to have all the necessary information to describe the time series generator phenomenon as well as possible. It is worth mentioning that this fitness function is used by both QIEAs used in this work. The second QIEA uses the fitness function to search for the best forecasting model according to a given time lag dimensionality supplied by the first QIEA, where, for each i-th individual, its fitness function is calculated using the fitness function of the best individual generated by the second QIEA.

After parameter initialization, the TIMRAA method performs the first QIEA to search for the time lag dimensionality (MaxLags). Each individual of the first QIEA represents a time lag dimensionality (MaxLags_i). The idea of the first QIEA is to determine the time window size (time lag dimensionality) that optimally describes the time series generator phenomenon. Then, for each individual generated in the first QIEA, the method performs another QIEA in order to determine, within the time window, the best particular time lags to optimally represent the time series, as well as searches for the best parameters, topology and architecture of the forecasting model. Regarding time lags, the first QIEA is used to perform a wider search for the time lag dimensionality and the second QIEA is used to perform a local search within the dimensionality selected by the first QIEA.

For each individual of the second QIEA, after model training (at the end of the MMNN training epochs), the proposed method uses the phase fix procedure, where a two step procedure is introduced to adjust the time phase distortions observed in stock market time series. It is possible to verify that the representations of some time series (natural phenomena) were developed by the model with a very close approximation between the actual and the predicted time series (referred to as "in-phase" matching), whereas the predictions of other time series (mostly stock market time series) were always presented with a one step delay regarding the original data (referred to as "out-of-phase" matching).

In this way, the proposed TIMRAA method uses the statistical test (t-test) to check if the MMNN representation (represented by an individual of the second QIEA) has reached an in-phase or out-of-phase matching. This is conducted by comparing the

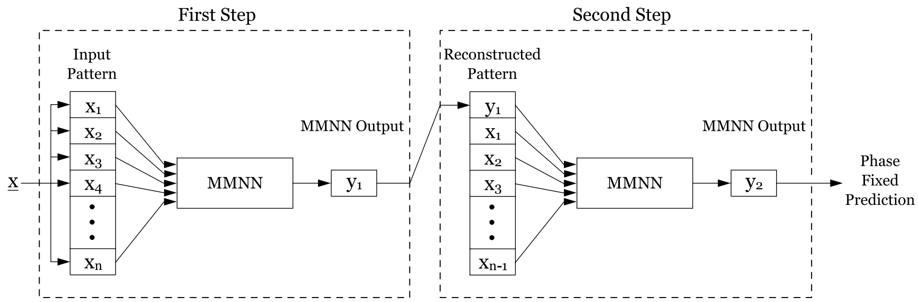

Fig. 4. Phase fix procedure.

outputs of the predictive model with the actual series, making use only of the validation data set. This comparison is a simple hypothesis test, where the null hypothesis is that the prediction corresponds to in-phase matching. The alternative hypothesis is that the prediction does not correspond to in-phase matching (or correspond to out-of-phase matching). If this test accepts the in-phase matching hypothesis, the elected model is ready for practical use. Otherwise, the TIMRAA method performs a new procedure to adjust the relative phase between the prediction and the actual time series. The phase fix procedure has two steps (described in the Fig. 4): (i) the validation patterns are presented to the MMNN and the output of these patterns are re-arranged to create new input patterns (reconstructed patterns), and (ii) these reconstructed patterns are re-presented to the same MMNN and the output set as the phase fixed prediction target.

It is worth mentioning that this procedure of phase adjustment considers that the MMNN does not overcome the random walk model, it just shows a behavior characteristic of a random walk model: the $t+1$ prediction is taken as the t value (Random Walk Dilemma—Section 2.2). If the MMNN was like a random walk model, the phase adjust procedure would not work.

The termination conditions for the first QIEA are:

(1) Number of iterations ($\text{QIEA1}_{\text{iter}} = 10$);
(2) The increase in the validation error or generalization loss (Gl) (Prechelt, 1994): Gl $> 5\%$;
(3) The decrease in the training error process training (Pt) (Prechelt, 1994): Pt $\leq 10^{-6}$.

The termination conditions for the second QIEA are:

(1) Number of iterations ($\text{QIEA2}_{\text{iter}} = 1000$)
(2) Minimum value of fitness function: fitness ≥ 40, where this value means the accuracy to predict direction around 80% (POCID $\gtrsim 80\%$) and the sum of the other errors around one ($\text{MSE} + \text{MAPE} + \text{THEIL} + \text{ARV} \cong 1$);
(3) Gl (Prechelt, 1994): Gl $> 5\%$;
(4) Pt (Prechelt, 1994): Pt $\leq 10^{-6}$.

Each individual of the first QIEA population is a time lag dimensionality. The individuals are represented by chromosomes that have the following gene:

- MaxLags: the time window size, that is, the maximum dimensionality that describe the time series generator phenomenon.

Each individual of the second QIEA population is a QuMLP. The individuals are represented by chromosomes that have the following genes (QuMLP parameters):

- $\underline{a}_k$: weights (structuring elements) of the MMNN;
- NModules: the number of modules in the MMNN structure (number of decompositions);
- MMNNArch: a real-valued variable, which is used to determine if the architecture is by dilations (MMNNArch > 0) or by erosions (MMNNArch ≤ 0);
- NLags: a vector, where each position has a real-valued codification, which is used to determine if a specific time lag will be used ($\text{NLags}_i > 0$) or not ($\text{NLags}_i \leq 0$).

It is important to mention that due to QIEA operators, the resulting chromosome gene values may exceed their valid boundary values. Whenever that happens, the corresponding genes are truncated (projected) to their upper/lower bounds.

5. Performance evaluation

Many performance evaluation criteria are found in the literature. However, most of the existing literature on time series prediction frequently employs only one performance criterion for prediction evaluation. The most widely used performance criterion is the Mean Squared Error (MSE), given by

$$\text{MSE} = \frac{1}{N}\sum_{j=1}^{N}\left(\text{target}_j - \text{output}_j\right)^2; \tag{51}$$

where N is the number of patterns, target_j is the desired output for pattern j and output_j is the predicted value for pattern j.

The MSE measure may be used to drive the prediction model in the training process, but it cannot be considered alone as a conclusive measure for comparison of different prediction models (Clements & Hendry, 1993). For this reason, other performance criteria should be considered to allow a more robust performance evaluation.

A measure that presents accurately identifying model deviations is the Mean Absolute Percentage Error (MAPE), given by

$$\text{MAPE} = \frac{1}{N}\sum_{j=1}^{N}\left|\frac{\text{target}_j - \text{output}_j}{\text{target}_j}\right|; \tag{52}$$

The random walk dilemma can be used as a naive predictor ($X_{t+1} = X_t$), commonly applied to stock market time series prediction. Thus, a way to evaluate the model regarding a random walk model is using the U of Theil Statistic (THEIL), which associates the model performance with a random walk model, and is given by

$$\text{THEIL} = \frac{\sum_{j}^{N}\left(\text{target}_j - \text{output}_j\right)^2}{\sum_{j}^{N}\left(\text{target}_j - \text{target}_{j-1}\right)^2}; \tag{53}$$

where, if the THEIL is equal to 1, the predictor has the same performance as a random model. If the THEIL is greater than 1, then the predictor has a worse performance than a random walk model, and if the THEIL is less than 1, the predictor is better than a random walk model. In the perfect model, the THEIL tends to zero.

Another interesting measure maps the accuracy in the future direction prediction of the time series or, more specifically, the

Please cite this article in press as: Araújo, R. A. Hybrid intelligent methodology to design translation invariant morphological operators for Brazilian stock market prediction. Neural Networks (2010), doi:10.1016/j.neunet.2010.06.007

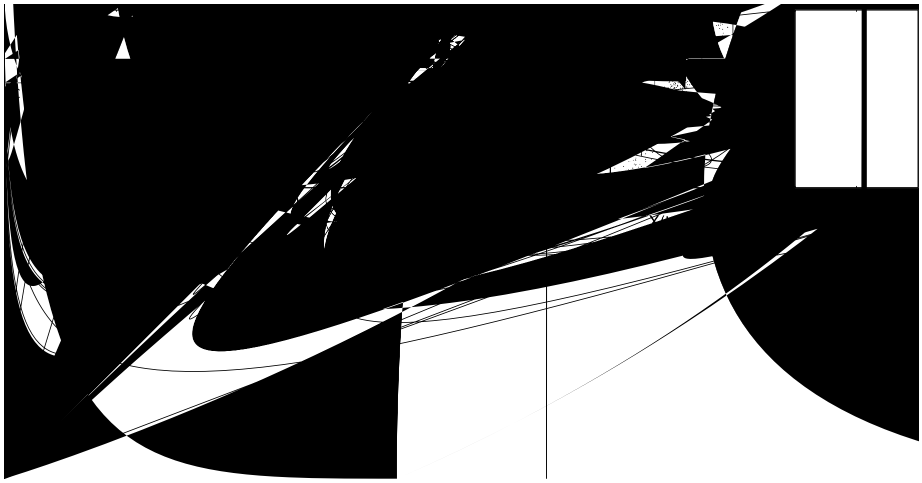

Fig. 5. A typical random walk lagplot.

ability of the method to predict if the future series value (prediction target) will increase or decrease with respect to the previous value. This metric is known as the Prediction Of Change In Direction (POCID), and is given by

$$\mathrm{POCID} = \frac{100}{N} \sum_{j=1}^{N} D_j, \tag{54}$$

where

$$D_j = \begin{cases} 1, & \text{if } (\mathrm{target}_j - \mathrm{target}_{j-1})(\mathrm{output}_j - \mathrm{output}_{j-1}) > 0 \\ 0, & \text{otherwise.} \end{cases} \tag{55}$$

The last measure associated the model performance with the mean of the time series. The measure is the Average Relative Variance (ARV), and given by

$$\mathrm{ARV} = \frac{\sum_{j=1}^{N} (\mathrm{target}_j - \mathrm{output}_j)^2}{\sum_{j=1}^{N} (\mathrm{output}_j - \overline{\mathrm{target}})^2}, \tag{56}$$

where $\overline{\mathrm{target}}$ is the mean of the time series. If the ARV is equal to 1, the predictor has the same performance as the time series average prediction. If the ARV is greater than 1, then the predictor has a performance worse than the time series average prediction, and if the ARV is less than 1, the predictor is better than the time series average prediction. In the ideal model, ARV tends to zero.

6. Simulations and experimental results

A set of four Brazilian stock market time series (Embraer (EMBR3) Stock Prices, Petrobras (PETR4) Stock Prices, Usiminas (USIM5) Stock Prices and Vale (VALE5) Stock Prices) were used as a test bed for evaluation of the proposed TIMRAA method. These times series were normalized to lie within the range $[0, 1]$ and divided in three sets according to Prechelt (1994): training set (50% of the data points), validation set (25% of the data points) and test set (25% of the data points).

For all the experiments, the following initialization system parameters were used: MaxDim $=$ 20 (maximum number of the time lag dimensionality) and MMNNModulesMax $=$ 25 (maximum number of MMNN modules). The parameters of first QIEA are the same for all experiments: the maximum number of generations is $\mathrm{QIEA1}_{\mathrm{iter}} = 10$, number of quantum individuals equals 50 and the crossover probability $p_{\mathrm{cross}} = 0.9$. The classical population of the first QIEA is composed of 50 individuals, where each individual represents a time lag dimensionality ($\mathrm{MaxLags}_i$) (time window size that optimally describe the time series generator phenomenon). The parameters of the second QIEA are the same for all experiments: the maximum number of generations is $\mathrm{QIEA2}_{\mathrm{iter}} = 1000$, number of quantum individuals equals 10 and the crossover probability $p_{\mathrm{cross}} = 0.9$. The classical population of the second QIEA is composed of 10 individuals, where each individual is a MMNN. After the MMNN weight initialization, each classical individual of the second QIEA is trained by the BP algorithm for 10^4 training epochs using a smoothing parameter $= 0.05$ and a convergence factor $= 0.01$. The termination conditions for the BP algorithm are the maximum number of epochs (10^4), the increase in the validation error or generalization loss (Gl $> 5\%$) and the decrease in the error of the process training (Pt $< 10^{-6}$).

The simulation experiments involving the proposed model were conducted with and without the phase fix procedure, referred to as TIMRAA model out-of-phase and TIMRAA model in-phase, respectively. These two procedures (in-phase and out-of-phase) were used to study the possible performance improvement, in terms of fitness function and of the phase fix procedure, applied to the proposed TIMRAA model. For each time series, ten experiments were executed and the instance with the larger validation fitness function was chosen to represent the prediction model.

In order to analyze time lag relations in the studied time series, the graphical methodology proposed by Kantz and Schreiber (2003) and Percival and Walden (1998), referred to as lagplot (Percival & Walden, 1998) or phase portrait (Kantz & Schreiber, 2003), was employed. This consists of a dispersion graph relating the different time lags of the time series (X_t vs X_{t-1}, X_t vs X_{t-2}, X_t vs X_{t-3}, …), and allows observations of possible relative strong relationships in any time lags (when a structure appearance is shown in the graph). Although such a technique is very limited, since it depends on human interpretation of the graphs, its simplicity is a strong argument for its utilization (Ferreira et al., 2008). For example, a time series like a random walk model will have a lagplot with just linear structures, as illustrated in the Fig. 5.

In order to establish a performance study, results previously published in the literature with the random walk model, the MRLTAEF method (Araújo & Ferreira, 2009) and TAEF method (Ferreira et al., 2008) under the same conditions were employed for comparison. Ten distinct experiments were carried out and the best prediction model (with the larger fitness function of the validation set) was chosen as the winning model.

Please cite this article in press as: Araújo, R. A. Hybrid intelligent methodology to design translation invariant morphological operators for Brazilian stock market prediction. Neural Networks (2010), doi:10.1016/j.neunet.2010.06.007

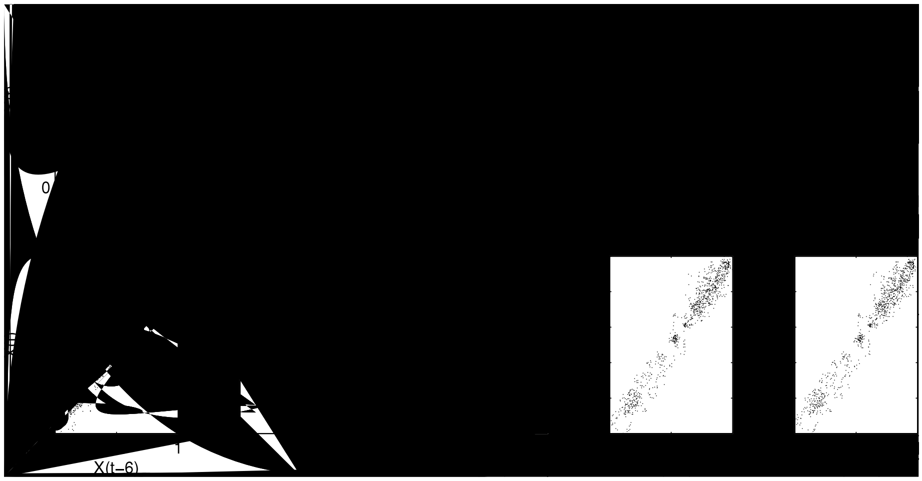

Fig. 6. Embraer (EMBR3) Stock Prices series lagplot.

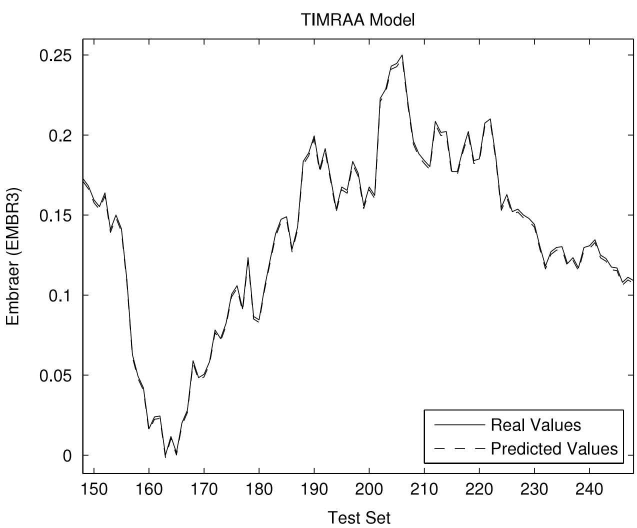

Fig. 7. Prediction results for the Embraer (EMBR3) Stock Prices series.

6.1. The Embraer (EMBR3) Stock Prices series

The Embraer (EMBR3) Stock Prices series correspond to the daily records of Embraer (EMBR3) from August 18 2005 to July 10 2009, constituting a database of 1000 points. Fig. 6 shows the Embraer (EMBR3) Stock Prices lagplot.

According to Fig. 6, it is verified for all time lags of the Embraer (EMBR3) Stock Prices series that there is a clear linear relationship among the lags. However, with the increase in the time lag degree, the appearance of the structure towards the center of the graph can indicate a nonlinear relationship among the lags.

The parameters automatically determined by the proposed model for the prediction of the Embraer (EMBR3) Stock Prices series were the lags 2, 7 and 10 as the most fitted lags for the time series representation, the MMNN model via erosions with 22 modules and the model as "out-of-phase" matching. Table 1 shows the results (with respect to the test set) for all the performance measures for the Random Walk, TAEF, MRLTAEF and the proposed TIMRAA model.

Fig. 7 shows the prediction results of Embraer (EMBR3) Stock Prices for the last hundred points of the test set.

6.2. The Petrobras (PETR4) Stock Prices series

The Petrobras (PETR4) Stock Prices series correspond to the daily records of Petrobras (PETR4) from August 19 2005 to July 10 2009, constituting a database of 1000 points. The lagplot of Petrobras (PETR4) Stock Prices is presented in Fig. 8.

Note in Fig. 8 that for all time lags of the Petrobras (PETR4) Stock Prices series there is a clear linear relationship among the lags. But, analyzing the increase in the time lag degree, it is possible to identify the appearance of a structure towards the upper corner on the right hand side of the graph that can indicate a nonlinear relationship among the lags.

The proposed method chose, for the prediction of the Petrobras (PETR4) Stock Prices series, the following parameters: (i) the lags 2, 3, 6, 7, 9 and 10 as the best time lags to optimally characterize the time series generator phenomenon, (ii) the MMNN model via dilations, (iii) the number of MMNN modules equal to 12, and (iv) the prediction model as "out-of-phase" matching. The obtained results (regarding the test set) for all the performance measures for the Random Walk, TAEF, MRLTAEF and the proposed TIMRAA model are presented in Table 2.

The prediction results of Petrobras (PETR4) Stock Prices for the last hundred points of the test set is presented in Fig. 9.

6.3. The Usiminas (USIM5) Stock Prices series

The Usiminas (USIM5) Stock Prices series correspond to the daily records of Usiminas (USIM5) from August 19 2005 to July 10 2009, constituting a database of 1000 points. Fig. 10 shows the Usiminas (USIM5) Stock Prices lagplot.

It is possible to verify in Fig. 10 that for all time lags of the Usiminas (USIM5) Stock Prices series there is a clear linear relationship among the lags. However, with the increase in the time lag degree, the appearance of the structure towards the center of the graph can indicate a nonlinear relationship among the lags.

The parameters chosen by the proposed method for the prediction of the Usiminas (USIM5) Stock Prices series were the lags 2 and 6 as the most fitted lags for the time series representation, the MMNN model via dilations with 25 modules and the model as "out-of-phase" matching. Table 3 shows the results (with respect to the test set) for all the performance measures for the Random Walk, TAEF, MRLTAEF and the proposed TIMRAA model.

Fig. 11 shows the prediction results of Usiminas (USIM5) Stock Prices for the last hundred points of the test set.

6.4. The Vale (VALE5) Stock Prices series

The Vale (VALE5) Stock Prices series correspond to the daily records of Vale (VALE5) from August 19 2005 to July 10 2009,

Please cite this article in press as: Araújo, R. A. Hybrid intelligent methodology to design translation invariant morphological operators for Brazilian stock market prediction. Neural Networks (2010), doi:10.1016/j.neunet.2010.06.007

Table 1

Table 4
Results for the Vale (VALE5) Stock Prices series.

Evaluation metrics	Random Walk	TAEF	MRLTAEF	TIMRAA
MSE	3.5966e−4	1.3239e−4	4.5137e−6	3.6650e−7
MAPE	0.0341	4.6639e−2	1.0047e−2	4.2798e−3
THEIL	1.0000	0.1603	1.5167e−2	4.4575e−3
ARV	0.0152	9.2433e−4	1.8548e−3	2.5596e−4
POCID	48.39	95.77	97.51	99.60
Fitness function	23.6080	79.2801	94.9397	98.7122

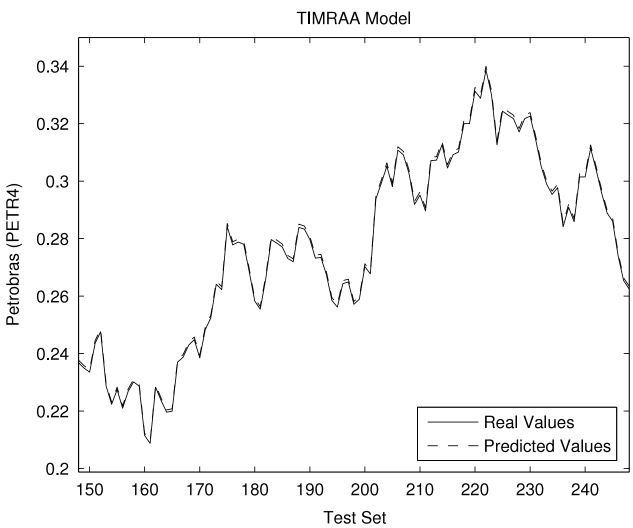

Fig. 9. Prediction results for the Petrobras (PETR4) Stock Prices series.

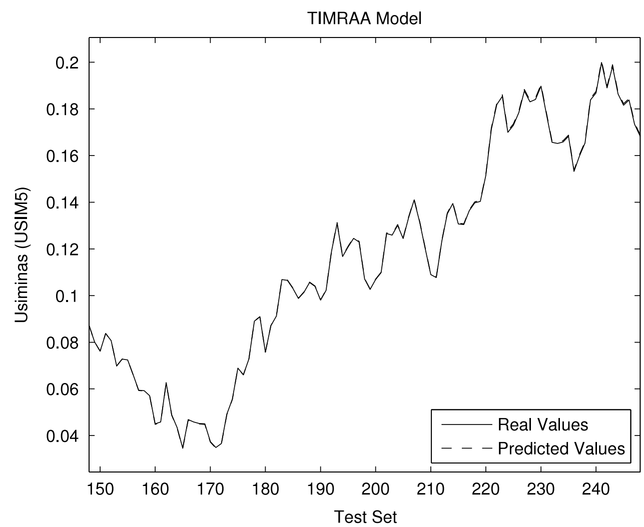

Fig. 11. Prediction results for the Usiminas (USIM5) Stock Prices series.

In this way, all generated prediction models using the phase fix procedure to adjust time phase distortions showed forecasting performance better than TAEF and MRLTAEF models. The TIMRAA method was able to adjust the time phase distortions of all analyzed time series (the prediction generated by the out-of-phase matching hypothesis is not delayed with respect to the original data). This corroborates the assumption made by Araújo and Ferreira (2009), where it is discussed that the success of the phase fix procedure is strongly dependent on an accurate adjustment of the prediction model parameters and on the model itself used for forecasting.

7. Conclusion

This paper presented a hybrid intelligent methodology to design increasing translation invariant morphological operators applied to Brazilian stock market prediction (overcoming the random walk dilemma). Five well-known performance measurements were used to assess performance of the proposed method, where an empirical fitness function was used in order to improve the description of the time series phenomenon as much as possible. However, a more sophisticated analysis has to be done to determine the optimal combination of such metrics in the fitness function.

The results were collected with four real world time series from the Brazilian stock market. The results obtained in the performed experiments showed a consistently better performance of the proposed model when compared to TAEF and MRLTAEF models. It was also observed that the TIMRAA model obtained a much better performance than a random walk model for the analyzed stock market time series, overcoming the random walk dilemma (where the predicted values are shifted one step ahead of the original values).

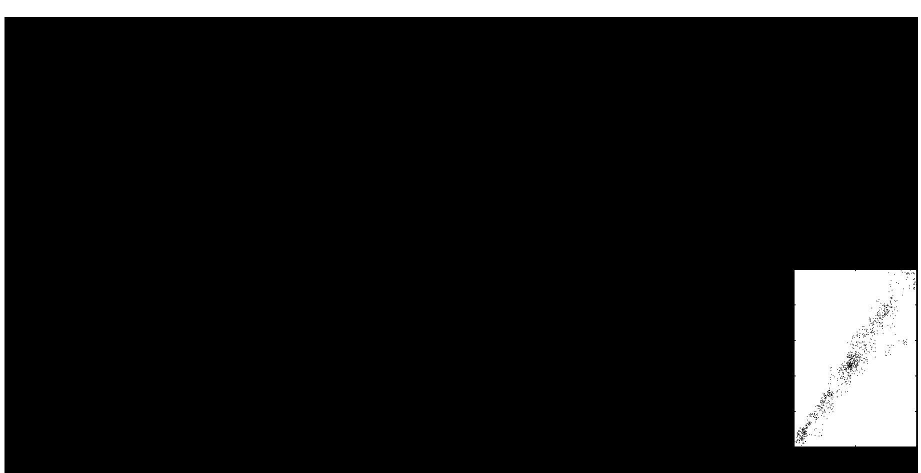

Fig. 10. Usiminas (USIM5) Stock Prices series lagplot.

Please cite this article in press as: Araújo, R. A. Hybrid intelligent methodology to design translation invariant morphological operators for Brazilian stock market prediction. Neural Networks (2010), doi:10.1016/j.neunet.2010.06.007

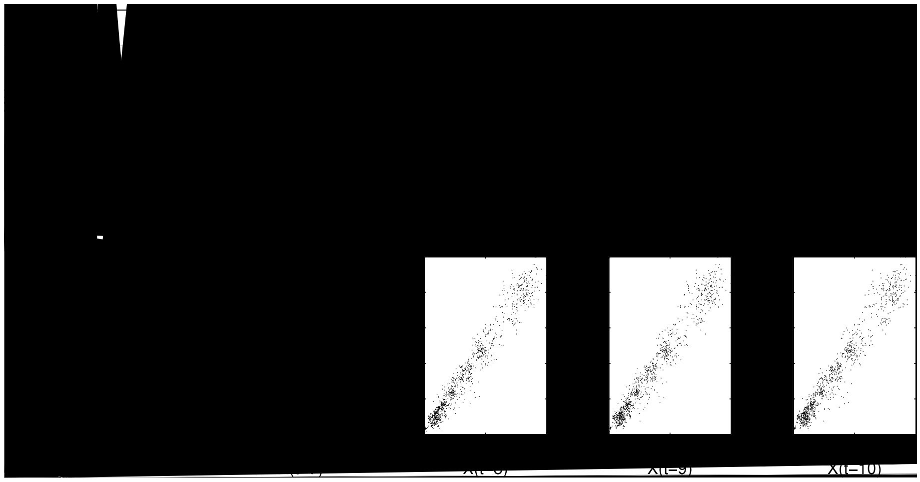

Fig. 12. Vale (VALE5) Stock Prices series lagplot.

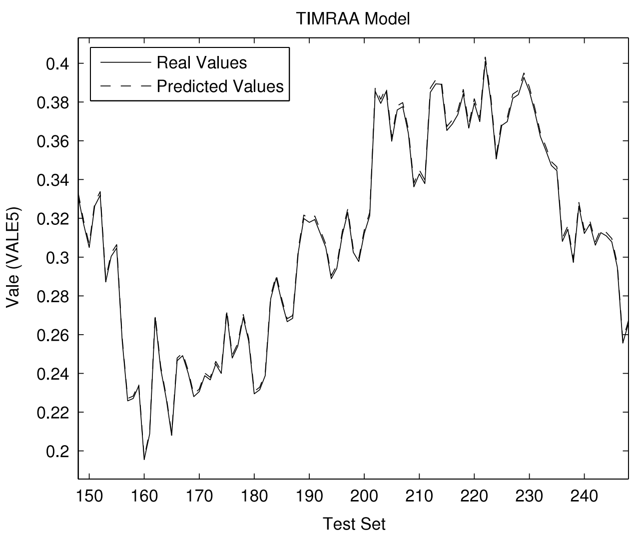

Fig. 13. Prediction results for the Vale (VALE5) Stock Prices series.

It is worth mentioning that the out-of-phase behavior found by the model proposed in this work appears often when the time series is from the stock market, as reported by Araújo and Ferreira. However, the out-of-phase behavior is characterized by a prediction delay (shift) with respect to the original time series data (this behavior is well-known in the literature, and is referred to as the random walk dilemma for stock market forecasting). In this way, the models generated by TIMRAA method were not real random walk models, they display similar characteristics of a behavior. This observation is supported by the fact that the phase fix procedure in the TIMRAA model was able to correct the one step delay distortion.

If the models generated by the TIMRAA method were real random walk models, the phase fix procedure would not be able to overcome the random walk dilemma, generating similar results of the original prediction, since in a real random walk model the $t+1$ prediction value is always the t prediction value. Such facts can be noted regarding relationship between linear and nonlinear structures which may be viewed and indicated by lagplot analysis.

It is possible to notice that for time series with clear nonlinear structure embedded in time lags, the proposed method can reach a very close estimation (prediction) of real time series values, being able to accurately overcome the random walk dilemma. However, if the nonlinear structures had a small amplitude when compared to the linear component (embedded in a linear structure), the proposed method could not reach the same improved performance, due to a slight decrease in the prediction performance results. However, even with this limiting factor, the TIMRAA method was still capable of adjusting the time phase distortion, being able to overcome the random walk dilemma.

A feasible explanation of such phenomenon is that the phase fix procedure depends on the information complexity contained in the time series data. When the time series generator phenomenon is so close to a random walk model the TIMRAA model has a decrease in prediction performance, although the phase fix procedure still works and overcomes the random walk dilemma. Also, it is important to mention that the success of the phase fix procedure is strongly dependent on an accurate adjustment of the prediction model parameters and on the model itself used for forecasting. Finally, the obtained results showed that the phase fix procedure was able to correct more efficiently the time phase distortion of stock market time series of the proposed TIMRAA model when compared to TAEF and MRLTAEF models.

Further studies are being developed to better formalize and explain the properties of the TIMRAA model and to determine possible limitations of the method with other stock market time series with components such as trends, seasonalities, impulses, steps and other nonlinearities. Also, further studies, in terms of risk and financial return, are being developed in order to determine the additional economic benefits, for an investor, with the use of the TIMRAA method.

Also, a particular study on the computing complexity of the proposed model with respect to the classic algorithms must be done in order to establish a complete cost-performance evaluation of the proposed model. According to this investigation, it will be possible to relate, in terms of cost, the necessary time to generate an optimal prediction model.

References

Araújo, R. A., & Ferreira, T. A. E. (2009). An intelligent hybrid morphological-rank-linear method for financial time series prediction. *Neurocomputing*, *72*(10–12), 2507–2524.

Araújo, R. A., Madeiro, F., Sousa, R. P., Pessoa, L. F. C., & Ferreira, T. A. E. (2006). An evolutionary morphological approach for financial time series forecasting. In *Proceedings of the IEEE congress on evolutionary computation. Vancouver, Canada*.

Araújo, R. A., Vasconcelos, G. C., & Ferreira, T. A. E. (2007). Hybrid differential evolutionary system for financial time series forecasting. In *Proceedings of the IEEE congress on evolutionary computation. Singapore*.

Banon, G. J. F., & Barrera, J. (1991). Minimal representation for translation invariant set mappings by mathematical morphology. *SIAM Journal on Applied Mathematics*, *51*(6), 1782–1798.

Binner, J., Kendall, G., & Gazely, A. (2004). Evolving neural networks with evolutionary strategies: a new application to divisia money. *Advances in Econometrics*, *19*, 127–143.

Box, G. E. P., Jenkins, G. M., & Reinsel, G. C. (1994). *Time series analysis: forecasting and control* (3rd ed.). New Jersey: Prentice Hall.

Clements, M. P., Franses, P. H., & Swanson, N. R. (2004). Forecasting economic and financial time-series with non-linear models. *International Journal of Forecasting*, *20*(2), 169–183.

Clements, M. P., & Hendry, D. F. (1993). On the limitations of comparing mean square forecast errors. *Journal of Forecasting*, *12*(8), 617–637.

Coyle, E. J., & Lin, J. H. (1988). Stack filters and the mean absolute error criterion. *IEEE Transactions on Acoustics, Speech, & Signal Processing*, *36*, 1244–1254.

Crottel, M., Girard, B., Girard, Y., Mangeas, M., & Muller, C. (1995). Neural modeling for time series: a statistical stepwise method for weight elimination. *IEEE Transactions on Neural Networks*, *6*(6), 1355–1364.

da Cruz, A. V. A., Vellasco, M. M. B. R., & Pacheco, M. A. C. (2006). Quantum-inspired evolutionary algorithm for numerical optimization. In *Proceedings of the IEEE congress on evolutionary computation. Vancouver, Canada*.

Davidson, J. L., & Hummer, F. (1993). Morphology neural networks: an introduction with applications. *Circuits, Systems, and Signal Processing*, *12*(2), 179–210.

Ferreira, T. A. E., Vasconcelos, G. C., & Adeodato, P. J. L. (2008). A new intelligent system methodology for time series forecasting with artificial neural networks. *Neural Processing Letters*, *28*, 113–129.

Han, K.-H., & Kim, J.-H. (2002). Quantum-inspired evolutionary algorithm for a class of combinatorial optimization. *IEEE Transactions on Evolutionary Computation*, *6*(6), 580–593.

Herwing, C. B., & Shalkoff, R. J. (1994). Morphological image processing using artificial neural networks. In C. T. Leondes (Ed.), *Control and dynamic systems*: *Vol. 67* (pp. 319–379). Academic Press.

Hocevar, M., Širok, B., & Blagojevic, B. (2005). rediction of cavitation vortex dynamics in the draft tube of a Francis turbine using radial basis neural networks. *Neural Computing & Applications*, *14*(3), 229–234.

Islan, M. M., Yao, X., & Muraz, K. (2003). A construtive algorithm for training cooperative neural networks ensembles. *IEEE Transactions on Neural Networks*, *14*(4), 820–834.

Kantz, H., & Schreiber, T. (2003). *Nonlinear time series analysis* (2nd ed.). New York, NY, USA: Cambridge University Press.

Khotanzad, A., Elragal, H., & Lu, T.-L. (2000). Combination of artificial neural-network forecasters for prediction of natural gas consumption. *IEEE Transactions on Neural Networks*, *11*(2), 464–473.

Leung, F. H. F., Lam, H. K., Ling, S. H., & Tam, P. K. S. (2003). Tuning of the structure and parameters of the neural network using an improved genetic algorithm. *IEEE Transactions on Neural Networks*, *14*(1), 79–88.

Loce, R. P., & Dougherty, E. R. (1992). Facilitation of optimal binary morphological filter design via structuring element libraries and design constraints. *Optical Engineering*, *31*, 1008–1025.

Malkiel, B. G. (2003). *A random walk down Wall Street, completely revised and updated edition*. W.W. Norton & Company.

Maragos, P. (1989). A representation theory for morphological image and signal processing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *11*, 586–599.

Matheron, G. (1975). *Random sets and integral geometry*. New York: Wiley.

Matilla-García, M., & Argüello, C. (2005). A hybrid approach based on neural networks and genetic algorithms to the study of profitability in the Spanish stock market. *Applied Economics Letters*, *12*(5), 303–308.

Miller, D. A., Arquello, R., & Greenwood, G. W. (2004). Evolving artificial neural network structures: experimental results for biologically inspired adaptive mutations. In *Proceedings of CEC2004—congress on evolutionary computation. Vol. 2* (pp. 2114–2119). IEEE.

Mills, T. C. (2003). *The econometric modeling of financial time series*. Cambridge: Cambridge University Press.

Minkowski, H. (1911). *Gesammelte abhandlungen*. Leipzig, Berlin: Teubner Verlag.

Myhre, T. C. (1992). Financial forecasting at Martin Marietta Energy Systems, inc. *The Journal of Business Forecasting Methods & Systems*, *11*(1), 28–30.

Ozaki, T. (1985). *Hand book of statistics*: *Vol. 5*. *Nonlinear time series models and dynamical systems*. Amsterdam: North-Holland.

Percival, D. B., & Walden, A. T. (1998). *Spectral analysis for physical applications—multitaper and conventional univariate techniques*. New York: Cambridge University Press.

Pessoa, L. F. C. (1997). Nonlinear systems and neural networks with hybrid morphological/rank/linear nodes: optimal design and applications to image and pattern recognition. *Ph.D. thesis*. Georgia Institute of Technology.

Pessoa, L. F. C., & Maragos, P. (1998). MRL-filters: a general class of nonlinear systems and their optimal design for image processing. *IEEE Transactions on Image Processing*, *7*, 966–978.

Pi, H., & Peterson, C. (1994). Finding the embedding dimension and variable dependences in time series. *Neural Computation*, *6*, 509–520.

Prechelt, L. (1994). Proben1: a set of neural network benchmark problems and benchmarking rules. *Tech. rep. 21/94*. URL: http://citeseer.ist.psu.edu/prechelt94proben.html.

Preminger, A., & Franck, R. (2007). Forecasting exchange rates: a robust regression approach. *International Journal of Forecasting*, *23*(1), 71–84.

Priestley, M. B. (1988). *Non-linear and non-stationary time series analysis*. Academic Press.

Rao, T. S., & Gabr, M. M. (1984). *Lecture notes in statistics*: *Vol. 24*. *Introduction to bispectral analysis and bilinear time series models*. Berlin: Springer.

Ritter, G. X. (1991). Recent developments in image algebra. In *Advances in electronics and eletron physics*: *Vol. 80*. Academic Press.

Rumelhart, D. E., & McCleland, J. L. (1987). *Parallel distributed processing, explorations in the microstructure of cognition*: *Vol. 1–2*. MIT Press.

Savit, R., & Green, M. (1991). Time series and dependent variables. *Physica D*, *50*, 95–116.

Serra, J. (1982). *Image analysis and mathematical morphology*. London: Academic Press.

Sitte, R., & Sitte, J. (2002). Neural networks approach to the random walk dilemma of financial time series. *Applied Intelligence*, *16*(3), 163–171.

Sousa, R. P. (2000). Design of translation invariant operators via neural network training. *Ph.D. thesis*. UFPB, Campina Grande, Brazil.

Stanley, K. O., & Miikkulainen, R. (2002). Evolving neral networks though augmenting topologies. *Evolutionary Computation*, *10*(2), 99–127.

Takens, F. (1980). Detecting strange attractor in turbulence. In A. Dold, & B. Eckmann (Eds.), *Lecture notes in mathematics*: *Vol. 898*. *Dynamical systems and turbulence* (pp. 366–381). New York: Springer-Verlag.

Tanaka, N., Okamoto, H., & Naito, M. (2001). Estimating the active dimension of the dynamics in a time series based on a information criterion. *Physica D*, *158*, 19–31.

Wilson, S. S. (1989). Morphological networks. In *Proceedings of the SPIE visual communication and image processing IV 1199* (pp. 483–493).

Yang, P., & Maragos, P. (1992). Character recognition using min–max classifiers designed using an LMS algorithm. *Visual Communications and Image Processing*, *92*(1818), 674–685.

Yao, X. (1999). Evolving artificial neural networks. *Proceedings of IEEE*, *87*(9), 1423–1447.

Zhang, G. P., & Kline, D. (2007). Quarterly time-series forecasting with neural networks. *IEEE Transactions on Neural Networks*, *18*(6), 1800–1814.

Zhang, G., Patuwo, B. E., & Hu, M. Y. (1998). Forecasting with artificial neural networks: the state of the art. *International Journal of Forecasting*, *14*, 35–62.

Please cite this article in press as: Araújo, R. A. Hybrid intelligent methodology to design translation invariant morphological operators for Brazilian stock market prediction. Neural Networks (2010), doi:10.1016/j.neunet.2010.06.007