Digital Signal Processing 77 (2018) 5–12

Contents lists available at ScienceDirect

Digital Signal Processing

www.elsevier.com/locate/dsp

Fast wavelet transform assisted predictors of streaming time series

Marco Stocchi [a,*], Michele Marchesi [b]

[a] Dept. of Electronic Engineering, University of Cagliari, v. Marengo 9, Italy
[b] Dept. of Mathematics and Computer Science, University of Cagliari, v. Porcell 4, Italy

ARTICLE INFO

Article history:
Available online 5 October 2017

Keywords:
Streaming datasets
Time series forecast
Fast Wavelet Transform
Shift variance theorem

ABSTRACT

We explore the shift variance of the decimated, convolutional Discrete Wavelet Transform, also known as Fast Wavelet Transform. We prove a novel theorem improving the FWT algorithm and implement a new prediction method suitable to the multiresolution analysis of streaming univariate datasets using compactly supported Daubechies Wavelets. An effective real value forecast is obtained synthesizing the one step ahead crystal and performing its inverse DWT, using an integrated group of estimating machines. We call Wa.R.P. (Wavelet transform Reduced Predictor) the new prediction method. A case study, testing a cryptocurrency exchange price series, shows that the proposed system can outperform the benchmark methods in terms of forecasting accuracy achieved. This result is confirmed by further tests performed on other time series. Developed in C++, Standard 2014 conformant, the code implementing the FWT and the novel Shift Variance Theorem is available to research purposes and to build efficient industrial applications.

1. Introduction

The problem of forecasting streaming datasets has been largely explored in the past, especially in the financial field; the main reason being the availability of large price time series, which are better suited to test any newly contrived predictor system. We also believe that such research field will become a core issue with the advancement of technologies such as the Internet of Things, giving the emerging need of forecasting the data generated by sensors and surveillance devices.

Recently, many researchers [15,10,7] are heading towards novel multiscale analysis approaches, motivated by recent findings about the powerful methods of wavelets, the latter being applied either alone, or in conjunction with other prediction models. Wang et al. [19] studied the possibility to improve the forecast of financial time series integrating a wavelet analysis denoising preprocessing module, and back propagation neural networks to perform regression. Their results show that such approach is promising, and motivated us to develop an implementation of the wavelet denoising neural network (WDNN) in order to benchmark the herein proposed inference engine's performance.

More recently, hybrid wavelet-based machine learning systems have been proposed. In Huang et al. [9] the wavelet analysis is combined with Support Vector Regression to forecast prices. The authors report improvements in the precision of the results, comparing them with the outputs of other standard SVR methods. Fang et al. [6] propose a prediction system based on genetic algorithms and wavelet neural networks, achieving a better accuracy than other benchmarking methods. A wavelet analysis is also used by Andrieş et al. [1] to investigate the behavior of exchange rates of several national currencies of eastern european countries.

Quite recently, many discrete wavelet transform (DWT algorithms) where proposed; the majority of them are classified as undecimated, shift invariant transforms, hence they are immediately applicable to the analysis of streaming datasets. Examples are the Stationary DWT [16], the "à trous" [17,12], the Maximum Overlap DWT [8]. These methods are alternative to the decimated, convolutional Discrete Wavelet Transform (also know as the Fast Wavelet Transform – FWT), which is implemented by iteratively filtering and downsampling a source series using two quadrature mirror filters. However, the FWT is known to be a *non* shift-invariant algorithm. Such feature causes major difficulties when performing the DWT of shifting time series. In particular the lack of shift invariancy, in signals processed using the FWT, impairs the possibility to compare directly two DWT crystals, calculated before and after a shift-insertion operation. Moreover, in order to extrapolate the *n*th step ahead of a time series, there must be full availability of the past data, hence the possibility to contrive a predictor based on the FWT analysis is difficult as well. It is then logical that the lack of any law describing the FWT coefficients transposition in the shift

* Corresponding author.
E-mail address: marco.stocchi@diee.unica.it (M. Stocchi).

https://doi.org/10.1016/j.dsp.2017.09.014

domain implies the need of endowing a forecasting system with a large number of regressor machines (one for each DWT coefficient). Such redundant configuration would cause lengthy training operations and it would inevitably degrade the accuracy achievable. It is our opinion that the challenge of shifting time series prediction, assisted by the wavelet analysis, would be better approached if the number of machines (deputed to perform real value forecasts) could be reduced to a minimum. Also, retaining the FWT algorithm, instead of adopting a shift invariant wavelet transform, would allow to implement a much more efficient and parsimonious system.

To this purpose, we explore the shift variance properties of the FWT of streaming input datasets, proving a novel Shift Variance Theorem – SVT. We integrate the SVT in the implementation of our predictor, greatly reducing the number of the required estimator machines. We name such predictor system a Wa.R.P. – Wavelet transform Reduced Predictor. Results show that the suggested method can outperform the benchmark models when applied to the Bitcoin-US Dollar hourly exchange rates. In order to evaluate the degree of generalization of the predictor, we also test the Wa.R.P. using three currency exchange pair datasets and three statistical time series. The predictor confirms its accuracy when compared with artificial neural networks and support vector regression.

The novel SVT is not only useful to devise highly efficient predictor systems, but it also allows to speed up the computation of the DWT of streaming univariate datasets. We studied the computational complexity of the reduced FWT, showing that an asymptotic reduction of 50% of the operations needed is achieved. We also tested a CPU implementation of the reduced FWT, observing an effective, significant improvement in computation time, proportional to the size of the sample used.

The software, available for download from a public repository (see Appendix A), is currently used for research purposes; the projected pathway to achieve an industrial impact has been considered at the Agile Group of the Department of Mathematics and Computer Science of the University of Cagliari. As such, an efficient SaaS application, suitable to provide DWT calculation services, is presently under development. Our ambition is to endow it with the highest feasible number of external univariate data sources.

This paper is organized as follows: Sec. 2 provides a background of the discrete wavelet transform, introducing the notation used; it addresses the shift variance problem and proofs the associated theorem. It also proposes a framework suitable to employ regression machines, using the DWT crystals of a sampled series. Sec. 3 describes the case-study application (the Bitcoin – US Dollar hourly exchange rates prediction), benchmarking the results obtained by the Wa.R.P. engine with other forecasting systems. Sec. 4 reports the prediction results obtained using further sets of source time series. Sec. 5 addresses several aspects of the calibration of the system, useful to achieve improvements in the forecasting performance. A final discussion and the conclusions are reported in Sec. 6.

2. Method

Let us denote $f(t):\mathbb{R}^+ \to \mathbb{R}$ a source signal, to be sampled at constant time intervals. At each new sampled data insertion, a multiresolution analysis (MRA) of a fixed size window of the sampled 1D series is performed, using the Fast Wavelet Transform algorithm; the resulting output, called *crystal*, is stored as a row entry of a matrix of appropriate size. The prediction system performs a forecast of the one-step ahead crystal, using dedicated regressor machines. In the present work, the Wa.R.P. engine is endowed with a group of multilayer perceptrons, each trained to perform the forecast of a single coefficient of the DWT. Eventually, the prediction of the input series is obtained by inverting the FWT of such estimated crystal.

In order to give a brief review of the discrete wavelet transform theory and, contextually, to establish notation, let us recall that the following families of functions $h_{m,n}(t)$, generated by shifting and scaling an appropriately chosen mother wavelet h:

$$h_{m,n}(t) = \left\{ 2^{-m/2}\, h(2^{-m}t - n) : m, n \in \mathbb{Z} \right\}, \tag{1}$$

constitute an orthonormal basis of the vector space of measurable square integrable 1D functions $L^2(\mathbb{R})$ [3]. They have been applied, in the past, to the analysis of signals pertaining to many scientific disciplines, and recently they were also applied to the study of financial time series. Their elements have good localization properties in both the spatial and Fourier domains. If the source time series is continuous, its DWT is defined as:

$$T_{m,n}(f) = \left\langle h_{m,n}, f \right\rangle = 2^{-m/2} \int_{\mathbb{R}} f(t)\, h(2^{-m}t - n)\, \mathrm{d}t. \tag{2}$$

Mallat [13] proved that the functions $f(t) \in L^2(\mathbb{R})$ can be considered as a limit of successive approximations (smoothed versions of $f(t)$), and that it is possible to find the wavelet coefficients $T_{m,n}$ as the difference of two approximations of $f(t)$ at consecutive different scales. To obtain this, it is necessary to define two families of scaling and wavelet functions, $\phi_{m_0,n}(t)$ and $\psi_{m,n}(t)$, respectively, which enable to express $f(t)$ by means of the following series:

$$f(t) = \sum_n c_{m_0,n} \phi_{m_0,n}(t) + \sum_m \sum_n d_{m,n} \psi_{m,n}(t). \tag{3}$$

Eq. (3) is the key to signal reconstruction (also called inverse DWT – IDWT), whereas the forward DWT allows to retrieve the $c_{m_0,n}$ and $d_{m,n}$ sets of coefficients. Moreover, a multiresolution decomposition of a sampled series can be performed efficiently in a recursive way, by means of filtering and downsampling operations, hence the algorithm name of *fast* wavelet transform (FWT). Usually, these filters are denoted by h and g. Because of their properties, in signal analysis they are referred to as quadrature mirror filters. The interested reader can rely on the original description of the Multiresolution Analysis, introduced in Mallat [13]. A detailed review of the wavelet based decomposition and reconstruction algorithms can be found in Daubechies [3].

Let us denote a source digital signal X of size n_X, composed of the last consecutive sampled values of $f(t)$:

$$X = \left\{ x_{n_X - 1}, \ldots, x_1, x_0 \right\}. \tag{4}$$

In eq. (4) the x_0 element is a newly inserted source element. The size n_X is kept constant by popping (removing) the first and older element from vector X (first in last out). Let us also denote n_h the size of both h and g filters, m the recursion depth of the procedure, 2^{-m} being the resolution $\mathcal{R}_m$ to which the signal is analyzed at depth m. If $m=0$ ($\mathcal{R}_0 = 2^0 = 1$), the input series is of course the source series itself. The forward FWT starts by convolving the source series with both filters h and g, and retaining (separately) one sample out of two. This allows to obtain, respectively, two sets of coefficients, herein denoted by c_1 and d_1, of the next level $m=1$ (lower resolution $\mathcal{R}_1 = 2^{-1}$), respectively representing a smoothed version of X and the difference of information between the two adjacent series $c_0 \triangleq X$ and c_1. The resulting vector of c_1 coefficients is used as input to the operation of the next level $m=2$, and the process can be repeated up to the maximum depth M. The last set of coefficients c_M is retained. The maximum depth M depends on both n_X and n_h:

$$M = \log_2 n_X - \lceil \log_2 n_h \rceil. \tag{5}$$

Eq. (5) expresses the maximum number of downsampling operations that can be performed, according to the sizes of the source input and the filters. Consequently, performing a FWT to the maximum depth M constrains n_{c_M} to be:

$$n_{c_M} = \frac{n_X}{2^M}, \tag{6}$$

and n_{d_m} (the size of the vector containing the difference of information between two adjacent c_{m-1}, c_m series) to be:

$$n_{d_m} = \frac{n_X}{2^m}. \tag{7}$$

At the recursion end (completion of the operation), the resulting vector T of forward transform coefficients – also referred to as a *crystal*, of the series X is populated in the following order:

$$\begin{aligned} T = \{ & c_{M,0}, c_{M,1}, \dots, c_{M,n_{c_M}}, \\ & d_{M,0}, d_{M,1}, \dots, d_{M,n_{c_M}}, \\ & d_{M-1,0}, d_{M-1,1}, \dots, d_{M-1,\, n_{d_{M-1}}}, \\ & d_{M-2,0}, d_{M-2,1}, \dots, d_{M-2,\, n_{d_{M-2}}}, \\ & \dots \\ & d_{1,0}, d_{1,1}, \dots, d_{1,n_{d_1}} \}. \end{aligned} \tag{8}$$

Note that the size of T equals the size of the source series n_X. Hereafter, the subsets of coefficients allocated at the same recursion depth m are also referred to as *subbands* B_m.

The aforesaid recursive forward FWT procedure suffers of a main drawback: in case of streaming datasets, the computation on subsequent sampled windows is affected by a *shift variance* problem, since at each insertion of a new element in the source series, many DWT coefficients (located in different subbands) are affected. As a direct consequence of this problem, several shift invariant DWT algorithms were proposed (such as the Stationary, à trous, and Maximum Overlap DWT, cited in Sec. 1). Their implementations are expensive in terms of memory and computational resources needed because of the intrinsic redundancy of the undecimated DWT representations. Also, if multiple undecimated MRA series must be stored (i.e. for machine learning purposes), this implies working with tensors of coefficients.

Our interest is to explore the possibility to avoid the redundancy of the above mentioned undecimated DWTs, and research the FWT shifting properties suitable to retain the adoption of such algorithm, at least in the particular case of streaming datasets analysis. Such research effort is motivated by the higher efficiency and lower resource need of the FWT, which are very important in many application fields.

Note that, considering a single series of size n_X, and wavelet filters of size n_h, its multiresolution analysis performed using the FWT implies a number of operations of order $O = 2n_X(2n_h - 1)$, as outlined next, in the proposed Theorem 2.5. Moreover, since no splitting operations are performed on the source series, a CPU implementation of such algorithm is nearly as efficient as the one factoring wavelet transforms into lifting steps, despite the latter's theoretical higher efficiency [4].

Let us denote Q_{n_T,n_X} the matrix containing a set of n_T successive forward DWTs, also called crystals, calculated at each shift-insert operation to account for an incoming value of a streaming dataset:

$$Q_{n_T,n_X} = \begin{pmatrix} T_{1,1} & T_{1,2} & \cdots & T_{1,n_X} \\ T_{2,1} & T_{2,2} & \cdots & T_{2,n_X} \\ \vdots & \vdots & \ddots & \vdots \\ T_{n_T,1} & T_{n_T,2} & \cdots & T_{n_T,n_X} \end{pmatrix}. \tag{9}$$

Each row of Q_{n_T,n_X} contains coefficients in the same order, as specified by eq. (8). Our goal is to forecast the $(1 + n_T)$th row of the matrix, using the information contained in the previous rows. If such operation is performed accurately, since a perfect reconstruction of the original signal (after the DWT analysis making use of Daubechies orthonormal wavelet filters) is proved, it is also possible to forecast the one step ahead of the input series simply inverting the DWT of such estimated row. To this purpose, it is necessary to deploy a set of specialized predictors, each one trained to perform the one step ahead forecast of a single coefficient of the crystal. It turns out that not all the coefficients must be estimated; the majority of them can be deterministically calculated, as described next in sec. 2.1. The remaining coefficients must be estimated, for example using neural regressors. Since the MRA performs a time-frequency analysis of the input series from more detailed to coarser resolutions, the latter is characterized by a higher degree of regularity. This allows to rely on the well known good fitting capability, intrinsic to the multilayer perceptrons, hence the choice to develop a prediction system that makes use of such type of neural networks, each having input patterns sampled from the respective Q_{n_T,n_X} column, as described in sec. 2.2.

2.1. Theorem on coefficients transposition

A shift insertion operation in X causes the successive forward wavelet transform to possess some shifting features, implying that many coefficients of the DWT are automatically determined. This property greatly enhances the FWT algorithm's efficiency and opens a new pathway to the research on the use of decimated DWT for prediction purposes.

Lemma 2.1. *Given the choice of wavelet filters h, g, having size n_h, a source vector X with size n_X, and denoting as $M \in \mathbb{N}$ the maximum FWT recursion depth, the size n_{B_m} of each subband B_m at recursion depth $m > 0$, $(m \in \mathbb{Z})$ is expressed by the following:*

$$n_{B_m} = \frac{n_X}{2^m}. \tag{10}$$

Proof. At each recursion depth m, the size of the input series is halved by 2. Also, the size of the full depth MRA equals the size of the input series. □

Lemma 2.2. *Given a sampled pattern X on which the FWT is calculated at each new value shift-insertion, and given the choice of wavelet filters h, g, of size n_h, the number of varying (affected) coefficients (with respect to the previous crystal) at recursion depth m, $(1 < m \leqslant M)$ are the last v_m coefficients of the subband B_m, obtained evaluating the following system:*

$$\begin{cases} v_1 = {}^{n_h}/_2\,; \\ v_m = (1 + v_{m-1}) \text{ div } 2 + (1 + v_{m-1}) \bmod 2 + v_1 - 1. \end{cases} \tag{11}$$

Proof. After 2 shift insertions on the source series, the subband B_1 can be obtained shifting the first crystal by 1, except for the last coefficients, affected by the double insertion. Let us denote as v_1 the number of such affected B_1 coefficients: it equals the number of inner products of the filters h, g sliding the source series with a translation step $i = 2$. The first inner product affected by the double insertion is the one performed with the filters applied on the last portion of the source series (i.e. before hitting the source series borders). To complete the convolution the filters must advance (with the same translation step $i = 2$) over the end of the source series, meaning that $\frac{n_h - 2}{2}$ inner products are left. The total inner products performed over the last 2 newly inserted values of the source series are then $v_1 = 1 + \frac{n_h - 2}{2} = \frac{n_h}{2}$.

The FWT decimates the input series at each recursion step. Due to this decimation, a double shift at level m implies that the series at $m+1$ is shifted once. Hence, in order to have one-shifted subbands B_2, 4 shift insertion at source level (level $m=0$) must have been performed. In this case, the filter h creates a decimated averaging series c_1, (at level $m=1$) which differs by $1+n_h/2$ elements from the one calculated 4 data insertions before. c_1 is the input series of the convolution that will be calculated at the higher recursion step $m=2$, and the number of varying coefficients v_2 of B_2 will depend both on the size of the filter and on the number of changed coefficients in c_1.

This algorithmic behavior, intrinsic to the fast wavelet transform, is extended to any filter size n_h. Hence the number of the varying coefficients v_m at depth m, created by filter g, depends on the maximum translations of the filter g on the c_m coefficients that have changed. Generally, denoting as v_c the number of such changed coefficients of the c series (which can be an even or odd integer quantity), the maximum number of translations of a filter of size n_h over v_c coefficients, (given the same translation step $i=2$) is trivially $n_h/2+\lceil v_c/2\rceil-1$. At depth $m=2$ this means $n_h/2+\lceil\frac{1+v_1}{2}\rceil-1=v1+\lceil\frac{1+v_1}{2}\rceil-1$. Note that the sum $(1+v_1)$ div $2+(1+v_1)$ mod 2 is nothing but the definition of ceiling of $\frac{1+v_1}{2}$. Finally, at depth $m=3$, the number of translations of the filters over v_2+1 coefficients is $n_h/2+\lceil\frac{1+v_2}{2}\rceil-1=v1+\lceil\frac{1+v_2}{2}\rceil-1$, and since the equivalence of the ceiling approximation to the sum of the first two terms of the second equation of system (11), the proof is obtained by induction. □

Lemma 2.3. *Having a matrix Q_{n_T,n_X} of DWT crystals, each calculated at a new shift-insertion in the source series, the non-varying coefficients $d_{h,i}$ at ith column of the B_m subband of the hth DWT remain the same, and can be retrieved from the respective rightish $i+1$th column of a previous crystal, as expressed below:*

$$d_{h,i}=d_{h-2^m,\ i+1},\quad 0\leqslant h<n_T,\ \ n_T>2^m\ . \tag{12}$$

Proof. Since at depth m the series has been subsampled m times, 2^m new insertions are needed to shift by 1 the wavelet coefficients at that depth. □

Theorem 2.4 *(On the shift variance of the 1D fast wavelet transform). Given an appropriate choice of wavelet filters h, g, having size n_h, a matrix Q_{n_T,n_X} of DWT crystals calculated using the FWT, performed at a new shift-insertion in the source series, only the last $v_m=(1+v_{m-1})$ div $2+(1+v_{m-1})$ mod $2+v_1-1$ coefficients of subband B_m are affected by one shift-insertion, while the others are previously determined and can be retrieved transposing the respective rightish column coefficients contained in the DWT performed 2^m insertions before, $d_{h,i}=d_{h-2^m,\ i+1}$. We refer to those coefficients as Shift Variance Theorem – SVT coefficients.*

Proof. Relies on the proofs of Lemmas 2.1, 2.2, 2.3. □

Remarks. Table 1 contains the number of varying coefficients v_m – also called non SVT coefficients – for each subband B_m, and for different Daubechies filter sizes. Implementation details of Theorem 2.4 is described in Stocchi and Marchesi [18] technical manuscript (the interested reader can refer to Sec. 2.2, Software functionalities).

Theorem 2.5 *(On the computational efficiency of the fast wavelet transform applied to 1D streaming datasets). Given an appropriate choice of wavelet filters h, g, and a matrix Q_{n_T,n_X}, the following equation expresses the number of operations O_r needed to calculate the reduced FWT:*

$$O_r=(2n_h-1)\ (n_X+\sum_{m=1}^{M}v_m). \tag{13}$$

Table 1
Number of varying (non SVT) coefficients at each depth m in a decimated DWT, performed using Daubechies orthogonal wavelet filters (filter size $n_H=2N$) on a streaming source series. The maximum reachable recursion depth m depends also on the size of the input series, which is not considered here. From recursion depth m onward, the number of varying coefficients is stable.

m \ N	2	3	4	5	6	7	8	9	10
1	2	3	4	5	6	7	8	9	10
2	3	4	6	7	9	10	12	13	15
3	3	5	7	8	10	12	14	15	17
4	3	5	7	9	11	13	15	16	18
5	3	5	7	9	11	13	15	17	19
6	3	5	7	9	11	13	15	17	19
7	...								

Proof. The maximum MRA recursion depth M is given by eq. (5). At each recursion step m and for each filter, $n_{B_m}(p+d)$ operations are performed, where p denotes the number of products, d the number of additions. Both p and d depend on the filter size. Each inner product needs a number of products $p=n_h$, and a number of additions $d=p-1$. Hence the total recursion step operations is $2n_{B_m}(n_h+n_h-1)=2n_{B_m}(2n_h-1)$, whereas the total computation order is $O=(2n_h-1)\ 2\sum_{m=1}^{M}n_{B_m}$. In case the shift variance theorem can be usefully applied (streaming input dataset), this can be rewritten as $O_r=(2n_h-1)\ (\sum_{m=1}^{M}n_{B_m}+\sum_{m=1}^{M}\not{v}_m+v_m)=(2n_h-1)\ (n_X+\sum_{m=1}^{M}v_m)$. In the latter equation, non-varying coefficients $\not{v}_m$ are discarded, because the theorem allows to retrieve them, hence no operations are performed. □

Remarks. Let us denote $K=2n_h-1$, the number of arithmetic operations performed at each translation step of the convolution of the FWT. The total forward FWT computational complexity can then be rewritten as $O=2K\sum_{m=1}^{M}n_{B_m}=2Kn_X$. When performing the reduced FWT, the number of operations O_c, performed to obtain a c_m set of coefficients, is retained the same as the original FWT, in order to avoid expensive extra copy operations of temporary data; hence $O_c=K\sum_{m=1}^{M}n_{B_m}=Kn_X$. The number of operations needed to calculate the d_m sets of coefficients is instead greatly reduced, since only the varying coefficients must be explicitly calculated. The number of operations performed to find the varying coefficients is $O_d=K\ \sum_{m=1}^{M}v_m$. The non varying coefficients are directly transposed from crystals previously calculated and stored as described by eq. (9). Hence the computational complexity of the reduced FWT is $O_r=O_c+O_d=Kn_X+K\ \sum_{m=1}^{M}v_m$. In the next Corollary 2.6 we are going to prove that the computational complexity of the reduced FWT of an arbitrarily large sample series tends to half the complexity of the full forward FWT.

Corollary 2.6 *(Asymptotic computational efficiency of the fast wavelet transform of streaming 1D datasets). Using of the shift variance theorem of the FWT (thus skipping the unnecessary calculations of SVT coefficients), the computational complexity of the forward FWT of 1D streaming series is reduced by 50% as n_X tends to infinity:*

$$\lim_{n_X\to+\infty}{}^{O_r}/_{O}={}^{1}/_{2},\quad n_h\ll n_X. \tag{14}$$

Proof.

$$\frac{O_r}{O}=\frac{(2n_h-1)\ (n_X+\sum_{m=1}^{M}v_m)}{2(2n_h-1)n_X}={}^{1}/_{2}+\frac{\sum_{m=1}^{M}v_m}{2n_X}\ ;$$

$$\lim_{n_X\to+\infty}{}^{O_r}/_{O}=\lim_{n_X\to+\infty}{}^{1}/_{2}+\frac{\sum_{m=1}^{M}v_m}{2n_X}$$

$$={}^{1}/_{2}+\lim_{n_X\to+\infty}\frac{\sum_{m=1}^{M}v_m}{2n_X}$$

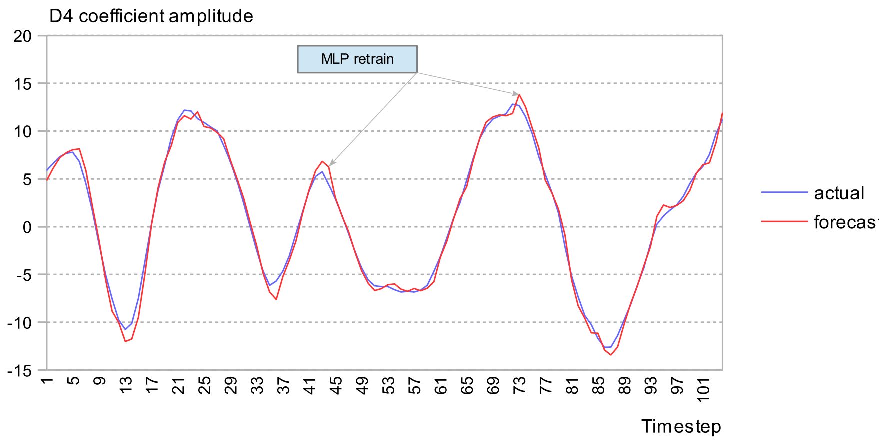

Fig. 1. Prediction session of one of the Q coefficients. Red line: forecasted series, blue line: real coefficients amplitude. Despite the non linearities of the shown source series, the forecasting accuracy can be considered quite satisfactory. The need for retraining the MLPs can often be explained with the presence of outliers in the data source series, which currently are not filtered by the Wa.R.P. engine; reducing the negative impact of isolated singularities on the machine learning quality can greatly improve the prediction accuracy and, for such reason, improvements to the proposed model are under research.

$$= {}^1/_2 + \lim_{n_X \to +\infty} \frac{\frac{n_h}{2} + \sum_{m=2}^{\log_2 n_X - \lceil \log_2 n_h \rceil} v_m}{2n_X}$$
$$= {}^1/_2. \quad \square$$

Remarks. The implementation of an efficiency test, simulating the asymptotic computational efficiency of the reduced FWT is described in Stocchi and Marchesi [18] technical manuscript (the interested reader can refer to Sec. 1, motivation and significance).

2.2. Wa.R.P. system core predictors: multilayered perceptrons

For each column of the matrix Q_{n_T, n_X} we train, validate and test a single hidden layer backpropagation multilayered perceptron (BPMLP), with a single output, to forecast the corresponding signed real input coefficient. The neural activation function of the perceptrons is the hyperbolic tangent, that provides a signed output. Denoting n_L the size of each network's input layer, the last $1 + n_L$ elements of the ith matrix column are preprocessed, in order to create the effective input vector I_i, as follows:

$$I_i = \left\{ T_{h,\,i} - T_{h-1,\,i} \ : \ n_T - n_L \leqslant h \leqslant n_T \right\}. \tag{15}$$

In this way, we generate the input vectors to the BPMLPs with the first differences of subsequent coefficients belonging to the same column of the matrix.

2.3. Inverse transform for prediction

The last operation needed to extract the forecasted value is to perform an inverse transform on the forecasted forward DWT. The forecasted one-step ahead value is obviously the last element of the inverse forecast:

$$\widehat{T} = \left\{ w_i : 0 \leqslant i \leqslant n_X - 1 \right\}, \tag{16}$$
$$\widehat{X} = \widetilde{T}\,\widehat{T}\,, \tag{17}$$
$$\hat{x} = \widehat{X}_{n_X - 1}, \tag{18}$$

where $\widetilde{T}$ denotes the inverted DWT.

2.4. Wa.R.P. update and retrain operations

The incoming data is shift-inserted in the source series, and a new DWT is immediately performed and stored in a history queue. This is useful to determine the predictors' fitness values at the last forecasting step. Also, the new calculated crystal is used to retrain the estimator machines.

2.5. Wa.R.P. system's implementation

In Stocchi and Marchesi [18] technical manuscript (Sec. 3, illustrative examples) we provide the details of an implementation of Wa.R.P. system, integrating a reduced FWT preprocessing module.

The goal of such implementation is to present an initial framework on which implementors, motivated by different operational requirements, can rely to build a specific predictor system using the decimated DWT shifting properties described above.

3. Case study: Bitcoin–US Dollar hourly exchange rates prediction

Since the beginning of the work we were interested in developing a system whose results could be directly compared using one of our previous works as a benchmark model. To this purposes, we selected the Bitcoin-USD hourly series (close price) spanning from January 1st 2013 to June 30th 2015. Such series shows an overall appreciation and volatility spark during the last quarter of 2013, the price reaching a top on December 2013. During 2014, instead, Bitcoin showed a well defined downtrending correction and slowly decaying volatility, as the price tested and then lowered below the psychological support level of 300 USD. Data inspection reveals an overall good quality of the whole set, since only 13 gaps can be found (103 missing bars, 21760/21863 observations, i.e. 0.47%).

The whole dataset is partitioned into an in-sample period, used for training and validation of the system, (from January 1st 2013 to December 31st 2014), and an out-of-sample period (from January 1st 2015 to June 30th 2015) to test the Wa.R.P. method on real data.

We tested the prediction system, developed as described in the previous sections, after having trained it on the exchange rate series. One of the estimator's performance is plotted in Fig. 1.

Price forecast session results are plotted in Fig. 2 (sorted absolute errors for the Bitcoin-USD currency pair hourly price prediction). The Wa.R.P. performance is benchmarked against the following methods: a naive forecast, a SVM based prediction, and a Wavelet Denoising Neural Network – WDNN, the latter being specialized for financial time series forecasts [2]. The first benchmark (naive approach), despite its simplicity, is generally recognized hard to beat in terms of price forecasting accuracy when us-

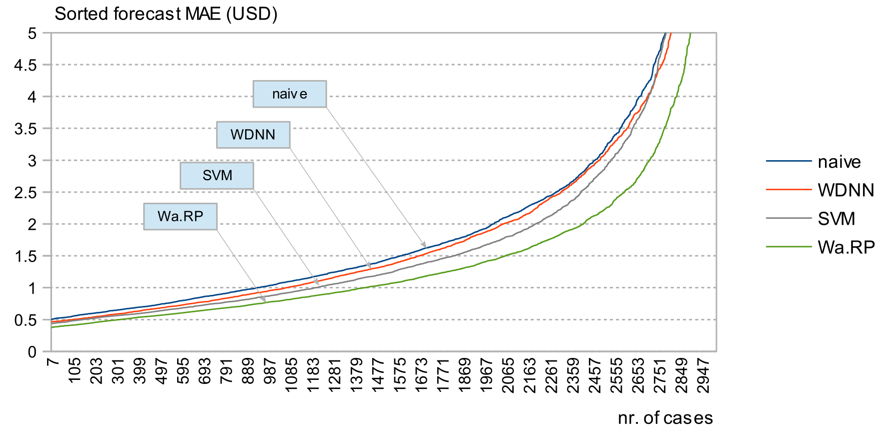

Fig. 2. Sorted absolute errors of the Bitcoin hourly price forecast session. Blue line: naive prediction, the worst performer among the benchmark methods; red line: Wavelet Denoising Neural Network – WDNN; black like: performance of a SVM prediction session; green line: performance of the fast wavelet transform assisted predictor, herein proposed. The results show that Wa.R.P. outperforms all the benchmark methods. It is worth noting that the WDNN achieves lower maximum errors when compared to the naive and SVM methods.

ing exclusively technical analysis indicators, or when compared to quantitative finance models – see for instance Dunis and Williams [5]. Let us recall that naive methods are generally designed to forecast a time series using the most recent observed trend change. If the last occurred price variation is taken into consideration (as we did in the present work), the naive model is expressed by the relation $\hat{x}_t = x_{t-1} + \Delta x_{t-1}$, where $\Delta x_{t-1} \triangleq x_{t-1} - x_{t-2}$. The SVM based predictor's performance outperforms the naive approach in terms of magnitude of forecasting errors. Moreover, the WDNN generally approximates the precision achieved by the SVM, obtaining better performances in terms of maximum error. We also tested a MLP predictor, feeding such network with unpreprocessed raw data. However, its forecasting performance improvement, compared to the naive model's output, has been found to be negligible, so we do not report it here. The Wa.R.P. engine substantially outperforms all the benchmark models: occurrence of errors less than 50 cents is 51%, while the second best performer achieves only 43%. Wa.R.P. cumulative absolute error is 3259 USD, while the best performer benchmark's one is 3662 USD.

Let us remark that the use of the new Theorem 2.4 on the shift variance of the FWT allows to greatly reduce the extensive testing times of the proposed system, and provides the predictor a very high absorption rate (meaning that the frequency of shift-insertion operations that the system is able to process is inversely proportional to the wavelet transform calculation time). In fact, since very few coefficients must be estimated, only a small number of predictors is required. For example, if the input series size is $n_X = 256$, using $N = 4$ (8 taps) Daubechies filters, then, $M = \log_2 256 - \lceil \log_2 8 \rceil = 5$ (see eq. (5)), the total number of non SVT coefficients per single DWT (hence those to be estimated) is just 31. Also, the correct retrieval of SVT coefficients allows to obtain a much more accurate estimate of the one-step ahead crystal.

Let us also emphasize that input data preprocessing before training the neural predictors (multilayered perceptrons in the present implementation of the system) is key to achieve a satisfactory forecasting accuracy. In the case of price series prediction, the creation of first differences input patterns, as illustrated by eq. (15), greatly enhances the forecasting results, reducing the number of MLP retraining operations to be performed during the update phase.

4. Further tests of the Wa.R.P. system

In order to evaluate the degree of generalization of the Wa.R.P. engine, we performed further tests, using two different sets of time series.

Table 2
Prediction performance of currency exchange pairs – hourly close prices (H1). Accuracy is expressed in terms of mean absolute error (MAE) and root mean square error (RMSE). Values are multiplied by 10^3.

Source: Myfxbook.com

	MAE				RMSE			
	MLP	SVM	WDNN	WaRP	MLP	SVM	WDNN	WaRP
EURUSD[a]	1.175	1.042	1.039	**1.037**	1.759	1.533	1.610	**1.489**
GBPUSD[b]	1.770	1.670	1.600	**1.470**	2.498	2.201	2.320	**2.031**
XAUUSD[c]	2705	2358	2502	**1888**	3790	3379	3464	**2747**

[a] Euro – US Dollar, January 1st 2016–May 31st 2016.
[b] Great Britain Pound – US Dollar, January 1st 2016–May 31st 2016.
[c] Gold – US Dollar, January 1st 2016–May 31st 2016.

The first set is composed of three hourly currency exchange data rates, published on Myfxbook.com: Euro–US Dollar (EURUSD), 13944 sample observations; Great Britain Pound–US Dollar (GBPUSD), 13944 samples; Gold–US Dollar (XAUUSD), 13327 samples. All the series span from March 2014 to the end of May 2016. The tests are performed considering the subsets of each currency pair, spanning from January 1st 2016 to May 31st 2016, as the out of sample period, while the previous data is used to train the predictors.

The performance of the system is benchmarked against a triple hidden layer backpropagation Multilayer Perceptron predictor, a Support Vector Machine (Dlib-ml implementation by King [11]), and a Wavelet Denoising based Neural Network (WDNN). The forecasting error is evaluated in terms of mean absolute error and root mean square error (MAE, RMSE). The results, shown in Table 2, indicate that the Wa.R.P. engine outperforms the benchmark methods, suggesting that it can be effectively used for the analysis and forecast of the currency markets.

The second set is composed of three different statistical time series published by the Eurostat. They are monthly indicators of the Eurozone Producers Price Index. The industrial output series spans from 1981 to 2015 (420 samples), which are divided into an in-sample period used for training (the first 180 observations) and an out-of-sample period (the remaining 240 observations). Both intermediate goods and mining output series span from 1991 to 2015 (300 samples): the in-sample and out-of-sample periods are respectively the first 180, and the last 120 samples.

The performance of the system is benchmarked against a triple hidden layer backpropagation Multilayer Perceptron predictor, and a Support Vector Machine. The Wavelet Denoising based Neural Network has not been considered a suitable benchmark method

Table 3
Prediction performance of Producers Price Indexes – PPI time series Euro Area 19, Monthly data. Accuracy is expressed in terms of mean absolute error (MAE), root mean square error (RMSE), mean absolute percentage error (MAPE). Values are multiplied by 10.
Source: Eurostat

	MAE			RMSE			MAPE		
	MLP	SVM	WaRP	MLP	SVM	WaRP	MLP	SVM	WaRP
Industry[a]	3.4	3.2	**2.9**	5.7	4.4	**4.1**	3.6	3.4	**3.1**
Int. goods[b]	2.5	3.1	**2.5**	3.8	4.3	**3.4**	2.7	3.4	**2.5**
Mining[c]	4.8	4.4	**3.9**	7.3	5.7	**5.2**	4.8	4.4	**3.9**

[a] Industry, 1981–2015.
[b] Intermediate goods, 1991–2015.
[c] Mining, quarrying; manufacturing; electricity, gas, steam and air conditioning supply, 1991–2015.

in this case, since we believe that the efficacy of such approach requires larger source time series than the ones under test.

The forecasting error is evaluated in terms of mean absolute error, root mean square error and mean absolute percentage error (MAE, RMSE, MAPE). The results, shown in Table 3, indicate that the Wa.R.P. engine outperforms both the MLPs and SVMs; as such, its forecasting accuracy is confirmed to be quite satisfactory.

5. Optimizing the forecasting performance of the Wa.R.P. engine

The herein presented system has the ambition to achieve a high forecasting accuracy of input digital signals. This is obtained by decomposing the original series using the wavelet analysis, predicting of the one-step ahead DWT, and finally performing the inverted FWT of such estimated crystal. Obviously, since it is proved that the FWT provides perfect signal reconstruction [14], the precision of the final result depends only on the accuracy with which the crystal is estimated. In order to improve the output results, let us outline the following considerations.

First, the quality of the signal analysis is affected by the choice of the wavelet used. The published software allows to instantiate the Wa.R.P. engine choosing one of the Daubechies orthonormal filters. Whenever feasible, a tradeoff between the selected source sample length n_X and the wavelet filters size n_h should be identified. This can be accomplished either by performing benchmark testing on the same training subset, in order to evaluate the quality of the forecasting results; or automatically, by running a process suitably designed to choose the optimum wavelet filter.

Second, it is important to evaluate the prediction performance history of each single estimator machine integrated in the Wa.R.P. If the multilayer perceptrons exhibit poor performance, an implementor might derive different types of regressors, hence changing the behavior of the system's inference engine. We performed extensive tests, useful to evaluate the efficacy of the MLPs, noting that the best results are obtained forecasting the series of wavelet coefficients that possess some level of autocorrelation (this happens in the low resolution subbands of the DWT). This suggests that both the number and sizes of the MLPs' hidden layers should be set low, because the long memory features of MLPs are not required. However, higher resolution coefficients of the DWT are able to detect sharp variations of the source signal; as such, they are more difficult to be predicted using the same configuration set up for the lower subbands. Hence, depending on the statistical properties of the input series, it is important to carefully calibrate the parameters of the integrated predictors. This motivated us to develop a uniform interface for all the derived predictor classes: with minor modifications to the code, an implementor could instantiate the engine using even different types of estimator machines (which could be MLPs responsible to predict the lower subbands' coefficients of the DWT, and other types of statistical estimators aimed at forecasting the upper subbands).

6. Conclusions

The presented Wa.R.P. system was initially developed with the goal to serve as a general time series predictor and thus no *a priori* assumptions on the input series features were made during the code implementation. When used to forecast the Bitcoin-US Dollar exchange rates (hourly data), the results are considered quite satisfactory. Similarly, further Wa.R.P. tests, performed on fiat currency exchange pairs datasets and on statistical time series, suggest that the system has a good degree of generalization.

Considering the exponentially increasing number of devices connected to the Internet, and the extensive need for prediction in many fields (e.g. smart cities, transportation, power, etc.), it will not be a surprise if such issue will be a major research field in the next future. The present work proposes a general framework for the creation of inference engines suitable to effectively analyze streaming 1D datasets. Further research could be dedicated to extend it to the analysis of 2D streaming series, such as those captured by moving observers, probes and surveillance devices.

The system, whose core components are developed in C++ using some novel features introduced in the 2014 Standards, is coded fully using parametric polymorphism, in order to be flexible enough to be instantiated at compile-time with different types of wavelet filters, as well as with different types of neural networks.

Presently, the code runs in a single thread. However, we believe the system could benefit by the introduction of multiple threads, especially regarding neural networks training. This could be useful to increase the absorption rate potential of the system. However, by using the novel shift variance theorem proposed, the fast wavelet transform of streaming univariate datasets is already a highly efficient method.

Appendix A. Wa.R.P. engine source code

The Wa.R.P. engine source code, as well as all the supporting files listed in the in Stocchi and Marchesi [18] technical manuscript appendix, are hosted on a Github repository and available for download with name: "DSPX_Fast_wavelet_transform_assisted_predictors_of_streaming_time_series".

References

[1] A.M. Andrieş, I. Ihnatov, A.K. Tiwari, Comovement of exchange rates: a wavelet analysis, Emerg. Mark. Financ. Trade 52 (3) (2016) 574–588.
[2] R.C. Cavalcante, R.C. Brasileiro, V.L. Souza, J.P. Nobrega, A.L. Oliveira, Computational intelligence and financial markets: a survey and future directions, Expert Syst. Appl. 55 (2016) 194–211.
[3] I. Daubechies, Orthonormal bases of compactly supported wavelets, Commun. Pure Appl. Math. 41 (7) (1988) 909–996.
[4] I. Daubechies, W. Swelden, Factoring Wavelet Transforms Into Lifting Steps, Program for Applied and Computational Mathematics, Princeton University, 1997.
[5] C.L. Dunis, M. Williams, Application of advanced regression analysis for trading and investment, in: W.F. Series (Ed.), Applied Quantitative Methods for Trading and Investment, John Wiley & Sons Ltd, 2003, pp. 34–35.

[6] Y. Fang, K. Fataliyev, L. Wang, X. Fu, Y. Wang, Improving the genetic-algorithm-optimized wavelet neural network for stock market prediction, in: 2014 International Joint Conference on Neural Networks (IJCNN), IEEE, 2014, pp. 3038–3042.
[7] I. Francis, K. Sangbae, An Introduction to Wavelet Theory in Finance: A Wavelet Multiscale Approach, World Scientific, 2012.
[8] M. Gallegati, Wavelet analysis of stock returns and aggregate economic activity, Comput. Stat. Data Anal. 52 (6) (2008) 3061–3074.
[9] C. Huang, L.-l. Huang, T.-t. Han, Financial time series forecasting based on wavelet kernel support vector machine, in: 2012 Eighth International Conference on Natural Computation (ICNC), IEEE, 2012, pp. 79–83.
[10] R. Jammazi, Cross dynamics of oil-stock interactions: a redundant wavelet analysis, Energy 44 (2012) 750–777.
[11] D.E. King, Dlib-ml: a machine learning toolkit, J. Mach. Learn. Res. 10 (2009) 1755–1758.
[12] A. Kozakevicius, A. Schmidt, A trous transform using wavelet extrapolation, 2010.
[13] S. Mallat, A theory for multiresolution signal decomposition: the wavelet representation, IEEE Trans. Pattern Anal. Mach. Intell. 11 (7) (1989) 674–693.
[14] S. Mallat, A Wavelet Tour of Signal Processing: The Sparse Way, Academic Press, 2008.
[15] M. Masih, O.A.T. Alzahrani, Systematic risk and time scales: new evidence from an application of wavelet approach to the emerging gulf stock markets, Int. Rev. Financ. Anal. 19 (1) (2010) 10–18.
[16] G.P. Nason, B.W. Silverman, The stationary wavelet transform and some statistical applications, in: Wavelets and Statistics, Springer, 1995, pp. 281–299.
[17] M.J. Shensa, The discrete wavelet transform: wedding the a trous and Mallat algorithms, IEEE Trans. Signal Process. 40 (10) (1992) 2464–2482.
[18] M. Stocchi, M. Marchesi, Fast wavelet transform assisted predictors of streaming time series, Digit. Signal Process. SoftwareX (2017), http://dx.doi.org/10.1016/j.softx.2017.09.006.
[19] J.-Z. Wang, J.-J. Wang, Z.-G. Zhang, S.-P. Guo, Forecasting stock indices with back propagation neural network, Expert Syst. Appl. 38 (11) (2011) 14346–14355.

Marco Stocchi holds a Ph.D. degree in Electronics and Information Engineering from the University of Cagliari. His research interests are mainly the time series analysis and machine learning.

Michele Marchesi is professor of software engineering at the University of Cagliari. His current research interests include analysis of cryptocurrency markets and business, financial time series forecasting, and blockchain applications.