Contents lists available at ScienceDirect

Expert Systems With Applications

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

Exploiting Wavelet Recurrent Neural Networks for satellite telemetry data modeling, prediction and control

Christian Napoli [a,*], Giorgio De Magistris [a], Carlo Ciancarelli [b], Francesco Corallo [b], Francesco Russo [b], Daniele Nardi [a]

[a] *Department of Computer, Automation and Management Engineering, Sapienza University of Rome, via Ariosto 25 Roma 00185, IT, Italy*
[b] *Thales Alenia Space, via Saccomuro 24 Roma 00131, Italy*

ARTICLE INFO

Keywords:
Time series forecast
Multivariate time series
Wavelet analysis
LSTM
wavelet recurrent neural networks

ABSTRACT

Multidimensional times series prediction is a challenging task. Only recently the increased data availability has made it possible to tackle with such problems. In this work we devised a novel method to exploit the multiple correlated features in the time series. The recurrent neural networks and the wavelet transform have been important innovations in the fields of signal processing and time series prediction. This paper proposes a Wavelet Recurrent Network for multi-steps ahead prediction of multidimensional time series. The proposed model combines these two elements into a neural network that predicts multiple samples in the future that are multiple time steps ahead with respect to the input samples. This Wavelet Recurrent Network carries out a multiresolution decomposition of the input signal through the wavelet transform, predicts the future wavelet coefficients with the recurrent neural network and transforms the output back in the time domain. The proposed model is applied to the prediction of satellite telemetry data, that is composed of readings from multiple sensors which are highly correlated. The prediction of such telemetries can help the engineers to detect anomalies in the system, that, in the context of space missions, are particularly dangerous since they can compromise the entire mission if not handled properly. The results show that the proposed model outperforms the recurrent network without wavelet transform both in terms of accuracy and in the width of the forecast horizon.

1. Introduction

Time series prediction is a well studied task in many scientific fields and it is essential in many business decision processes. The high availability of data that characterizes the present time demands algorithms with the capability of processing huge quantities of data (Duan et al., 2017; Wang et al., 2020a; Zhang et al., 2020). In the same direction, also the computational power has increased leading the way to Neural Networks, that with their data driven approach and their nonlinear nature have demonstrated ability to learn complex dependencies in high-dimensional data (Hung et al., 2019; Pu et al., 2021). Neural networks have been widely applied to multidimensional time series prediction, and they outperformed many model based approaches used in the past, especially for multidimensional data with nonlinear patterns (Chakraborty et al., 1992; Wang et al., 2020b; Yang et al., 2020).

Wavelet Transform was an important advance in signal processing and its applications cover many different fields (Amaratunga et al., 1994; Lewis & Knowles, 1992; Luisier et al., 2007; Ravikanth et al., 2020). It was proven that their ability to decompose the signal in different scales can be exploited by predictive algorithms to improve their performances. In Jarrah and Salim (2019) the Haar scaling function is used as a feature extractor for a Recurrent Neural Network that predicts the future stock closing prices from the previous ones. In Gürsoy and Engin (2019) it is shown that, in the prediction of the future river discharge from the discharge collected over the year, a Feedforward neural network that receives as input a multiresolution decomposition of the original signal performs better than if the same architecture were applied to the raw input. In both these works the Wavelet Decomposition is used as a feature extractor, and the neural network is trained to predict the future signal directly from this set of features. The authors of Capizzi et al. (2012) created a neural network that can execute both decomposition and reconstruction of the predicted signal simulating the wavelet function that best fits the specific prediction problem using second generation wavelets. They implemented both the update U and

[*] Corresponding author.
E-mail addresses: cnapoli@diag.uniroma1.it (C. Napoli), demagistris@diag.uniroma1.it (G. De Magistris), Carlo.Ciancarelli@thalesaleniaspace.com (C. Ciancarelli), Francesco.Corallo@thalesaleniaspace.com (F. Corallo), Francesco.Russo@thalesaleniaspace.com (F. Russo), nardi@diag.uniroma1.it (D. Nardi).

https://doi.org/10.1016/j.eswa.2022.117831
Received 15 December 2021; Received in revised form 26 May 2022; Accepted 8 June 2022
Available online 18 June 2022
0957-4174/

Symbols Table

$L^2(R)$	The Vector space of square-integrable functions
$\emptyset$	The empty subspace of $L^2(R)$
V, W	Subspaces of $L^2(R)$
$\phi_{j,k}(t)$	A scaling function, where j and k are respectively the dilation and translation indexes
$\psi_{j,k}(t)$	A wavelet function, where j and k are respectively the dilation and translation indexes
$c_j(k)$	Approximation coefficients at approximation level j
$d_j(k)$	Detail coefficients at approximation level j
$\mathrm{PROJ}_{\vert V\vert}f(t)$	Orthogonal projection of function $f(t)$ onto the subspace V

predict P operators of the lifting scheme with neural networks, such that the wavelet filter can be learnt by the network and adapted to the specific task. With respect to Jarrah and Salim (2019) and Gürsoy and Engin (2019), that do not explicitly compute the reconstruction, our model reconstructs the signal in the time domain before computing the loss function. However, unlike the model presented in Capizzi et al. (2012) the wavelet decomposition and reconstruction are computed explicitly using the Pyramidal algorithm (Teukolsky et al., 1992) hence reducing drastically the complexity of the network and the training time. The idea of decomposing the signal to model specifically its different behaviors is employed also in classic model based approaches like ARIMA. ARIMA models however require the time series to be stationary, hence the signal is pre-processed by removing the trend and seasonal components respectively with differencing and seasonal differencing (Hyndman & Athanasopoulos, 2018). The seasonal differencing technique assumes that the period is fixed and known. The Wavelet decomposition instead is much more powerful since it allows to analyze the signal at different scales decomposing it into multiple resolution bands. In this work we will show how a neural network, thanks to its nonlinear nature, can be used to model the complex patterns at different scales, and consequently to predict the future signal directly in the Wavelet domain. To the best of our knowledge this is the first attempt to combine the Pyramidal algorithm with an Long Short-Time Memory (LSTM) (Hochreiter & Schmidhuber, 1997) based neural network in order to decompose the input signal and to predict the future signal directly in the Wavelet domain before reconstructing the output in the time domain. In particular our model exploits the separation of the signal into wavelet bands to predict the future signal in parallel in the wavelet domain. This is done without changing the loss function that is still the standard Mean Squared Error and it is computed in the time domain. Many models proposed in literature address the problem of single-step ahead prediction, meaning that the predicted samples are consecutive to the ones received as input, and the multi-steps ahead prediction is obtained running the model on the previous predictions. With this approach however the errors tend to accumulate and consequentially the performances drop when the forecast horizon increases. The proposed method, on the other hand, is designed specifically to predict samples that are multiple steps ahead in the future without relying on the intermediate predictions.

The first section gives some basic elements of the Wavelet theory, Multiresolution approximation and Recurrent Neural Networks. Section 3 defines the prediction problem and describes the data and the architecture of the proposed Wavelet Recurrent Network. Section 4.1 introduces a famous architecture based on LSTM that can be adapted to the task of multi-steps ahead prediction of a multivariate time series, which results will be compared with those obtained by the proposed model, while Section 4.2 explains how the proposed architecture can be used to detect anomalies in the telemetry data. In Section 5 conclusions are drawn.

2. Wavelet transform and recurrent neural networks theory

2.1. Multiresolution approximation

A multiresolution analysis of the space of square integrable functions is a sequence of subspaces with the following properties:

$$f(t) \in V_j \rightarrow f(2t) \in V_{j+1} \tag{1}$$

$$f(t) \in V_j \rightarrow f(t-2^{-j}k) \in V_j \qquad k \in Z \tag{2}$$

$$\emptyset = V_{-\infty} \subset \ldots \subset V_0 \subset \ldots \subset V_{\infty} = L^2(R) \tag{3}$$

The idea is that a function in $L^2(R)$ can be represented at different resolution levels and the approximation at level j is obtain projecting the function on the subspace V_j (Eq. (4))

$$\mathrm{PROJ}_{|V_j|}f(t) = \sum_k c_j(k)\phi_{j,k}(t) \tag{4}$$

The functions $\phi_{j,k}(t) = 2^{j/2}\phi(2^j x - k)$ in (4) are translated and dilated version of the same function $\phi(t)$ that is called "scaling function" and the coefficients c_j are referred as "approximation coefficients" (or sometimes "scaling coefficients"). Let W be the orthogonal complement of V_j in V_{j+1}, then a function can be represented at approximation level $j+1$ as its projection onto V_{j+1} or as its projection onto V_j plus its projection onto W_j (Eq. (5)).

$$\mathrm{PROJ}_{|V_{j+1}|}f(t) = \sum_k c_j(k)\phi_{j,k}(t) + d_j(k)\psi_{j,k}(t) \tag{5}$$

The functions $\psi_{j,k}(t) = 2^{j/2}\psi(2^j x - k)$ in (5) are translated and dilated version of the same function $\psi(t)$ that is called "wavelet function" and the coefficients d_j are referred as "detail coefficients" (or sometimes "wavelet coefficients").

2.2. Filter banks implementation

Wavelet and scaling functions can be defined recursively through a finite sets of coefficients ($g(n)$ and $h(n)$ in Eqs. (6) and (7)) called respectively wavelet filter and scaling filter.

$$\phi(t) = \sum_n h(n)\sqrt{2}\phi(2t-n) \tag{6}$$

$$\psi(t) = \sum_n g(n)\sqrt{2}\psi(2t-n) \tag{7}$$

These filters can be used to derive the approximation and detail coefficients from the approximation coefficients of the higher resolution level (Eqs. (8) and (9)) and to derive the approximation coefficients from the detail and approximation coefficients from the lower resolution level (Eq. (10)) where the highest resolution level approximation coefficients are the samples of the function itself (more information on how Eqs. (8)–(10) are derived can be found here Burrus, 2015).

$$c_j(k) = \sum_m h(m-2k)c_{j+1}(m) \tag{8}$$

$$d_j(k) = \sum_m g(m-2k)c_{j+1}(m) \tag{9}$$

$$c_{j+1}(k) = \sum_m c_j(m)h(k-2\ m) + \sum_m d_j(m)g(k-2\ m) \tag{10}$$

The signal decomposition consists in the repeated application of Eqs. (8) and (9) to obtain the lower resolution scaling and wavelet coefficients from the scaling coefficients at higher resolution. The signal reconstruction is the inverse process and it consists in the repeated

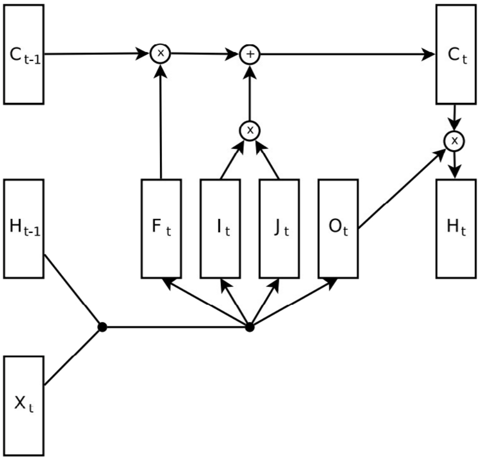


Fig. 1. Long Short-Term Memory diagram, where C is the memory cell and it contains the state. F,I,O are respectively the forget,input and output gates. H is the hidden vector (the output of the network) that depends both on the input X and state C.
Source: The image was taken from Jozefowicz et al. (2015).

application of Eq. (10) to obtain the higher resolution scaling coefficients from the scaling and wavelet coefficients from the lower level. In the case of discrete input signal with finite length, the decomposition (and reconstruction) can be computed through repeated matrix vector multiplications (more information about the transformation matrix is available in Teukolsky et al., 1992). The result of the decomposition is a vector with the same length of the input that contains the coefficients of the lower resolution level followed by the detail coefficients from the higher levels. The maximum number of levels of decomposition is bounded by the length of the input signal through Eq. (11).

$$\text{max level} = \lfloor \log_2 \left(\frac{\text{input length}}{\text{filter length} - 1} \right) \rfloor \tag{11}$$

2.3. Long short-term memory recurrent network

Recurrent Neural Networks were an important advance in neural computing for sequential data processing. The main limitation of Feedforward Neural Networks (FFNN), in the context of time series prediction, consists in the fact that the output depends only on the current input and, in order to train these networks to predict the future values of a time series, the signal is split into blocks with fixed sizes, where the size of each block is the number of input neurons. For this reason FFNN are not able to capture temporal dependencies that are longer than the current input sequence. Moreover in FFNN each output depends equally on any input, hence they inherently do not account for temporal dependencies or any ordering in the data. Recurrent Neural Networks were introduced to add "memory" of the past inputs such that the output depends not only on the current input but also on the previous ones. This mechanism can be implemented adding to a FFNN a feedback loop that connects the output back to the hidden layer (Kubat, 1999). This kind of architecture can learn in theory temporal dependencies with arbitrary time lag, but in practice they do not for the problem known as Vanishing or Exploding Gradient (Pascanu et al., 2013). Long Short-Term Memory (LSTM) recurrent networks (Hochreiter & Schmidhuber, 1997) overcome this limitation using a different architecture that enforces constant error propagation. This architecture is based on a memory cell and a set of gates that control the amount of information that flows from the input to the memory cell and from the memory cell to the output (Fig. 1).

3. Proposed wavelet recurrent network

3.1. Data

We have at our disposal simulated data that emulate the operation of a LEO (Low Earth Orbit) satellite. In particular we studied the sensors monitoring a *Reaction Wheel* (RW), which is a type of flywheel used primarily by spacecrafts for three-axis attitude control. The RW has a high pointing accuracy and it is particularly useful when the spacecraft must be rotated by very small amounts, for example for keeping a telescope pointed at a star. The Reaction Wheel was equipped with four sensors monitoring: the current absorbed, the temperature, the velocity and the commanded torque (see Fig. 2). Each sensor sends its data to Earth in fixed sampling intervals that differ from sensor to sensor, hence the four sequences must be downsampled in order to be aligned. Moreover, the satellite does not operate exclusively in a nominal mode, and can have non-nominal behaviors such as flights maneuvers. The data collected under these circumstances are marked with a special flag and are filtered by the system in order to mitigate the false positive rate. TAS-Italia industry[1] provided us the simulation of four months of observation data, one of those with an anomalous behavior.

3.2. Input pipeline

The Satellite telemetries are collected from four sensors that measure different pieces of information of the same physical component. A reliable predictive model should consider these correlations, hence the problem is formulated as a multivariate time series prediction task where each data point has four features, corresponding to the four sensors. The Wavelet Recurrent Neural Network (WRNN), that will be described in the next section, receives as input the multiresolution decomposition of the signal. For this reason the signal is split into blocks which length must be chosen carefully since it determines the decomposition level according to Eq. (11). Specifically, it was chosen such that it contains one period of the lower frequency periodic component that has a period of about two hours, as shown in Fig. 3.

The final length was rounded up to 2048 samples per channel such that it was a power of two (this is necessary because at each decomposition level the number of coefficients is halved).

For the wavelet decomposition, the Daubechies wavelet and scaling filters (Daubechies, 1992) with four vanishing moments have been selected. The Daubechies family is an orthonormal wavelet system which filters have the minimum length given the number of vanishing moments. In particular the decomposition uses the filters with four vanishing moments because they can approximate the trend of the signal at the lower resolution scale. Fig. 4 compares the Daubechies filters varying the number of vanishing moments.

Once the input signal is converted in the wavelet domain two filtering techniques are applied to simplify the task of the recurrent network. The first technique is hard thresholding (Donoho & Johnstone, 1994) (Eq. (12)), that is used in signal processing for denoising and it consists in replacing with zero the coefficients which absolute value is below a fixed threshold.

$$d'_j(k) = \begin{cases} d_j(k) & if \quad d_j(k) > \tau \\ 0 & otherwise \end{cases} \tag{12}$$

The second one consists in replacing the higher resolution detail coefficients with zeros. This approach allows to reduce drastically the load of the network, since the number of coefficients is halved. However if the wavelet system is chosen properly (especially the number of vanishing moments) these coefficients should carry few information and the reconstruction should not be affected by this substitution. Fig. 5 shows the reconstruction of the signal from the coefficients where the higher resolution detail coefficients are replaced with zeros.

[1] https://www.thalesgroup.com/it/global/activities/space

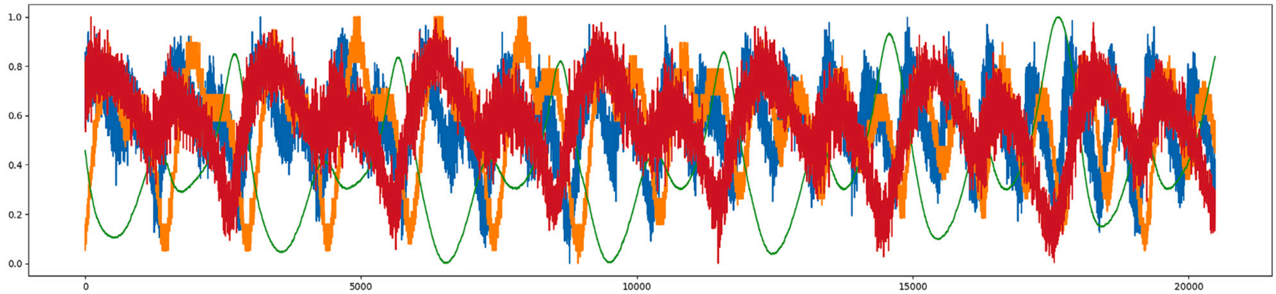


Fig. 2. A sample of telemetry data. Each line corresponds to a sensor: motor current (blue line), temperature (orange line) and angular speed (green line).

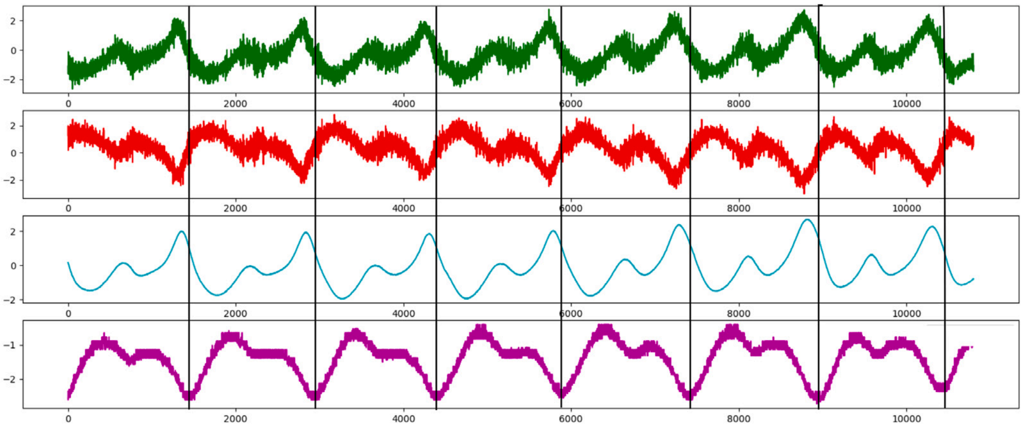
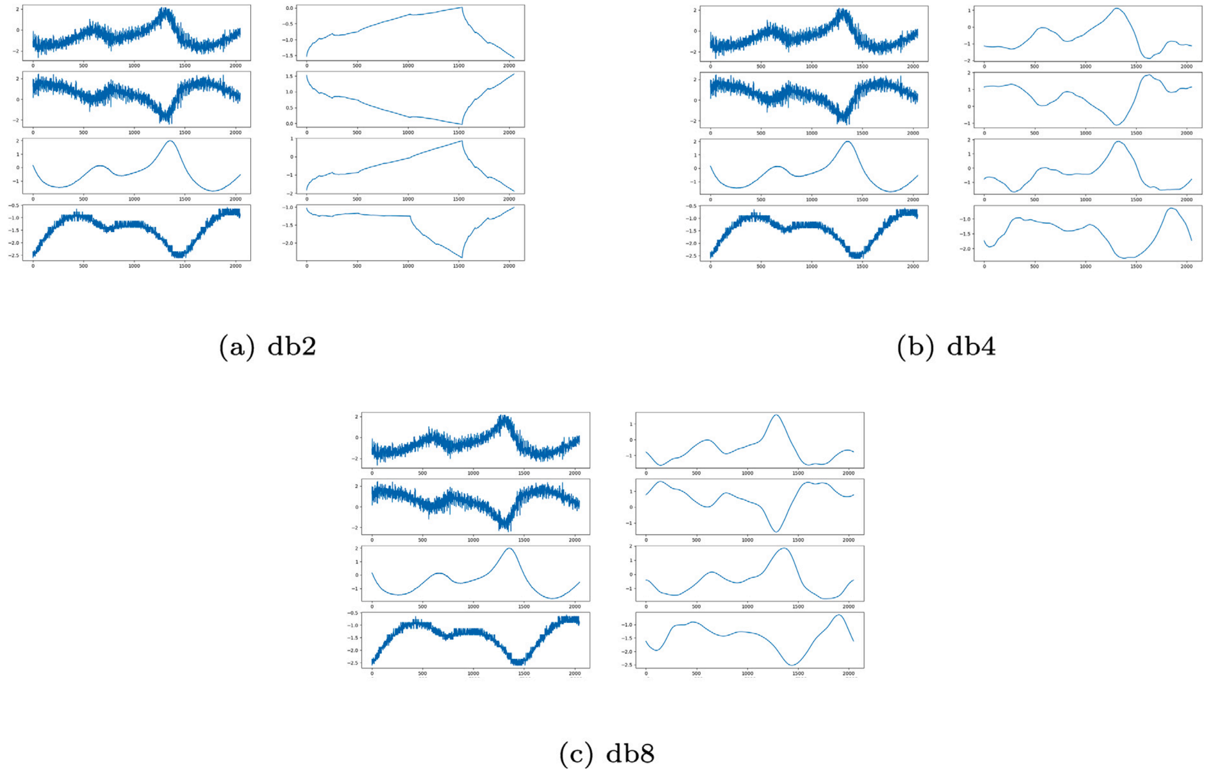


Fig. 3. Input signal over a time period of 12 h. The vertical lines highlight the lower frequency periodic component, that has a period of 1 h 43 m.



(a) db2

(b) db4

(c) db8

Fig. 4. Comparison between db2,db4 and db8 Daubechies filters. The plots on the left show the original signal while the ones on the right are the reconstructions using only the lower approximation level scaling coefficients. While db4 and db8 provide a good approximation of the signal db2 does not. This experiment shows that four vanishing moment are enough to represent the signal at a low level of approximation.

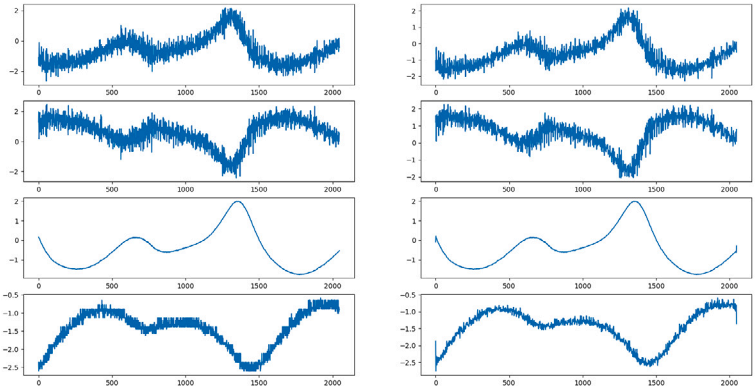


Fig. 5. On the left the original signal, on the right the signal reconstructed from the wavelet coefficients where the high detail coefficients cd8 are set to zero.

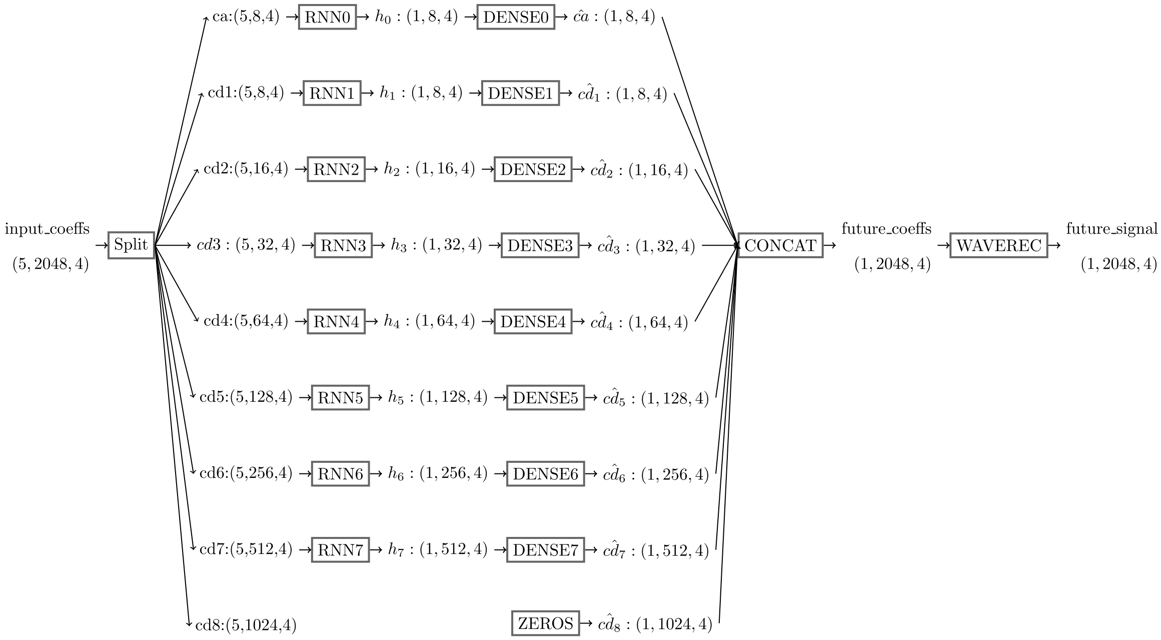


Fig. 6. Wavelet Recurrent Neural Network architecture. The input wavelet coefficients are split according to their resolution band. For each resolution band there is a specific recurrent network that produces the encoding of the past coefficients and a dense layer that uses the encoding to predict the future coefficients (with the exception of the higher resolution detail coefficients that are all zero). Then the predicted coefficients (and the zero vector for the higher resolution detail coefficients) are concatenated and passed to the reconstruction algorithm that outputs the future signal.

3.3. Model architecture

The model architecture is illustrated in Fig. 6, it is composed of multiple LSTM encoders and dense layers (one for each band of the multiresolution decomposition). The proposed WRNN, as described in the previous section, processes the input in blocks of 2048 samples per channel. Hence the time unit, referred as "step", corresponds to about two hours of signal. The LSTM encoders process a sequence of 5 steps and each of them receives the coefficients from a specific resolution band. For each LSTM there is a dense layer that uses the encoding to predict the coefficients at that resolution level for a single step that is 10 steps ahead in the future (about 20 h ahead in the future). The output coefficients from each band are the input of the reconstruction algorithm (described in Section 2.2) that returns a vector with the same shape of the input containing the samples of the predicted signal. In conclusion, the proposed Wavelet Recurrent Network uses about ten hours of signal (10240 samples per channel) to predict the two hours (2048 samples per channel) that are about twenty hours ahead in the future.

3.4. Training

For the training phase, according to the common practice, we split the dataset into training, validation and test splits, using the three observation periods without anomalies for training (80% training and 20% validation) and the fourth observation period for testing. The input of the network is the signal that was split into blocks and transformed in the Wavelet domain. Then the network splits the coefficients according to their resolution level and each split is processed by a parallel branch composed of an LSTM layer and a final dense layer. The parallel branches are trained together because the loss is computed once on the reconstructed signal in the time domain. The output of each branch is the vector of wavelet coefficients of the future signal for a specific resolution level. Before computing the loss, the output coefficients are concatenated and the signal is reconstructed with the reconstruction algorithm. The decomposition and reconstruction algorithms were implemented in TensorFlow (Abadi et al., 2015). In particular we implemented the Pyramidal algorithm using only differentiable operations (multiplications by circulant matrices and permutation matrices), such that the wavelet decomposition and reconstruction can be inserted

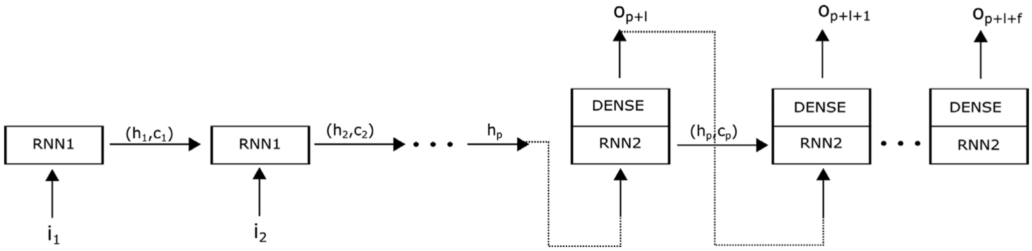


Fig. 7. Illustration of an RNN encoder–decoder for multi-steps ahead prediction. The first recurrent unit (RNN1) loops on the input steps to produce the encoding h_p. The second recurrent unit (RNN2) receives as its first input the encoding h_p, then it loops on its previous predictions to produce the output sequence.

into a differentiable computational graph. Hence, even though the parameters of the decomposition and reconstruction algorithms are fixed, the network can be still trained end to end with Gradient Descent. We trained the network for about 70 epochs using Mean Squared Error loss and the Adam optimizer with default parameters and we adopted early stopping to prevent overfitting.

4. Results

In Section 4.1 the proposed model is compared with an LSTM encoder–decoder architecture. After the evaluation of the performances for the prediction task we will explain how the proposed architecture can be used to detect anomalies in spacecraft telemetry data (in Section 4.2). In particular we will describe the experiment and the results that we obtained.

4.1. Prediction

Recurrent Networks have been widely used for time series prediction, in particular LSTM have demonstrated the ability to learn complex temporal dependencies with a time lag greater than 1000 steps (Hochreiter & Schmidhuber, 1997). However training directly a simple LSTM network to predict 22528 points (2048 points that are 20480 points ahead in the future) from the 10240 points in the past was not feasible. Even an iterative approach, in which the model is trained to predict a smaller number of samples and, at validation time, it is fed back with the previous predictions to increase the forecast horizon, was not sufficient, because, due to the error propagation (Boné & Crucianu, 2002), the model deviated from the correct prediction before it could reach the desired point in the future. For this reason the Wavelet Recurrent Network was compared with an LSTM encoder–decoder (Sutskever et al., 2014) that is a more general model that learns a mapping between the input and output sequence without any previous assumptions on those sequences, and can be trained directly to predict the samples that are multiple steps ahead in the future from the past samples. This architecture (illustrated in Fig. 7) was introduced for natural language translation, but it was successfully adopted for complex time series prediction (Park et al., 2018). In particular, to enforce fairness, we compared both architecture on the same temporal horizon. However, to make this comparison feasible we had to downsample the signal by a factor 16 for the LSTM encoder–decoder, because the training was not feasible with the available hardware. This fact highlights another important advantage of our method, namely the ability to predict huge amounts of data with much less training time than the baseline. In particular the proposed architecture exploits the separation of the signal into wavelet bands to predict the future signal in parallel directly in the wavelet domain. The decomposition of the input data allows to predict longer sequences because the temporal correlation is considered only within the same wavelet band. This assumption is supported by the intuition that the signal can be decomposed into different frequency bands that are connected to different phenomena, and it is also supported by the empirical results. Indeed, despite the different sampling rate (the WRNN processes 16 times more data), the training time is halved with respect to the baseline. In this way it was possible to compare the two models on the same temporal horizon. In particular the LSTM encoder–decoder was trained to predict directly 128 points that are 1280 points ahead in the future from the 640 points in the past received as input.

Table 1
Percentage error for the Wavelet Recurrent Network and the LSTM encoder–decoder for the task of multi-steps ahead prediction. The two models are compared on the same temporal horizon, however the input of the LSTM encoder–decoder has been downsampled by a factor 16.

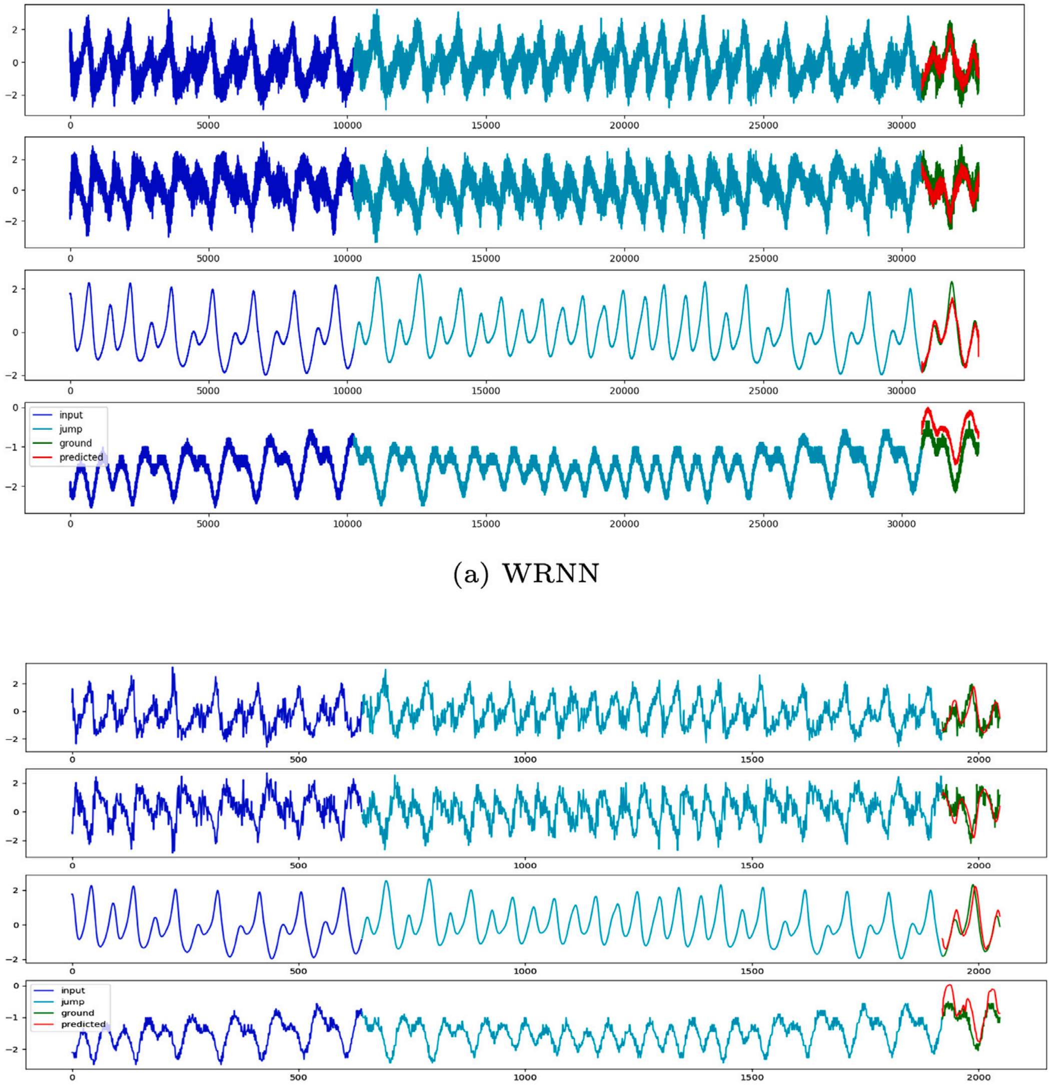
MAAPE (%)		
	WRNN	LSTM encoder–decoder
CH0	5.55	6.44
CH1	5.73	6.55
CH2	3.54	5.15
CH3	0.78	1.62

The Mean Arctangent Absolute Percentage Error (MAAPE) (Kim & Kim, 2016) was used to compare those models. It is a variation of the Mean Absolute Percentage Error that, instead of computing the mean of the absolute percentage errors (APE), computes the arctangent of the absolute percentage errors (see Eq. (13), where y is the true value and $\hat{y}$ is the predicted value). In this way it avoids the "outliers" problem (Makridakis, 1993) according to which the APE is indefinite when the true value is close to zero, because the Arctangent Absolute Percentage Error (AAPE) is always bounded in $[0, \frac{\pi}{2}]$.

$$MAAPE = 100 \times \frac{1}{N} \sum_{n} \arctan(|\frac{y_n - \hat{y}_n}{y_n}|) \tag{13}$$

According to Table 1, that measures the performances of both models on the validation set, the Wavelet Recurrent Network has the smallest percentage error in each channel, despite it processes 16 times more data. Fig. 8 shows the results of the two models on the same window.

The results show that our WRNN performs better than the LSTM encoder–decoder both in terms of training time and prediction error. This can be explained by the fact that our WRNN considers the temporal correlations only within each wavelet bands. In particular it assumes that the information carried by the coarse signal is independent from the one carried by the details. This makes sense in the domain of the satellite telemetries, where low frequency components may depend on the orbit period, while high frequency oscillations may depend on the noise introduced by the sensors and other components may be related to different causes. We took advantage of this assumption to devise an efficient architecture that is able to process very long sequences with respect to a simple LSTM, where the input is processed one sample at a time. The wavelet recurrent network introduced in Capizzi et al. (2012) is not designed for multi-steps ahead prediction. Hence a direct comparison of the results would require a re-implementation of the model from scratch, which would be too costly. However we can still affirm that the proposed WRNN converges much quicker still obtaining good results. This because in our method the wavelet filters, that are used in the Pyramidal algorithm for the decomposition and reconstruction, are



(a) WRNN

(b) LSTM encoder-decoder

Fig. 8. The Wavelet Recurrent Neural Network (WRNN) and the LSTM encoder–decoder are compared on the same input for the task of multi-steps ahead prediction.

fixed and consequently the complexity of the network is reduced. In particular the proposed WRNN took about 70 epochs of training for convergence, while the Wavelet recurrent network by Capizzi et al. (2012) was trained for hundreds of epochs.

4.2. Anomaly detection

Another important task, that is the primarily purpose of telemetry data, is anomaly detection. It consists in the identification of patterns in the data that deviate from the normal behavior. The reason why this task is so hard is the problematic definition of "normal behavior". A good predictor, trained on "normal" data, implicitly defines a model of "normality" that can be used to detect anomalies as deviations between the predicted and the real values (as shown in Fig. 9). In our case the dataset contained just a single known anomaly, that was left in the test split. However we were able to identify and perfectly localize it through the analysis of the residuals of the forecast produced by our model, where a residual is defined as:

$$e(t) = |y(t) - y'(t)| \tag{14}$$

where $y(t)$ and $y'(t)$ are respectively the true and predicted values at time t. In particular, we first analyzed the distributions of the residuals over long observation periods. These distributions allow to notice immediately whether or not an anomaly has occurred, even though they do not allow to localize the anomaly in time. Fig. 10 shows two distributions corresponding to two observation periods, where the first contains the known anomaly. We can see that the distribution of the residuals corresponding to the period with no anomalies is almost a Gaussian with zero mean, while the second is a bimodal distribution, where the principal mode (the one with higher probability) corresponds to the normal behavior while the second one corresponds to the abnormal behavior. Once the anomalous period is identified it is possible to localize the anomaly by plotting the evolution of the residuals (as shown in Fig. 11). An important characteristic of an anomaly detection system is its ability to distinguish noise from outliers (or anomalies) (Salgado et al., 2016). In our case, and even in most real scenarios, the data is acquired from sensors that introduce a certain amount of noise in the measures. For this reason, it is very important not to confuse the aforementioned noise with anomalies. At this purpose, the ability of the proposed method of decomposing the signal comes at hand since it

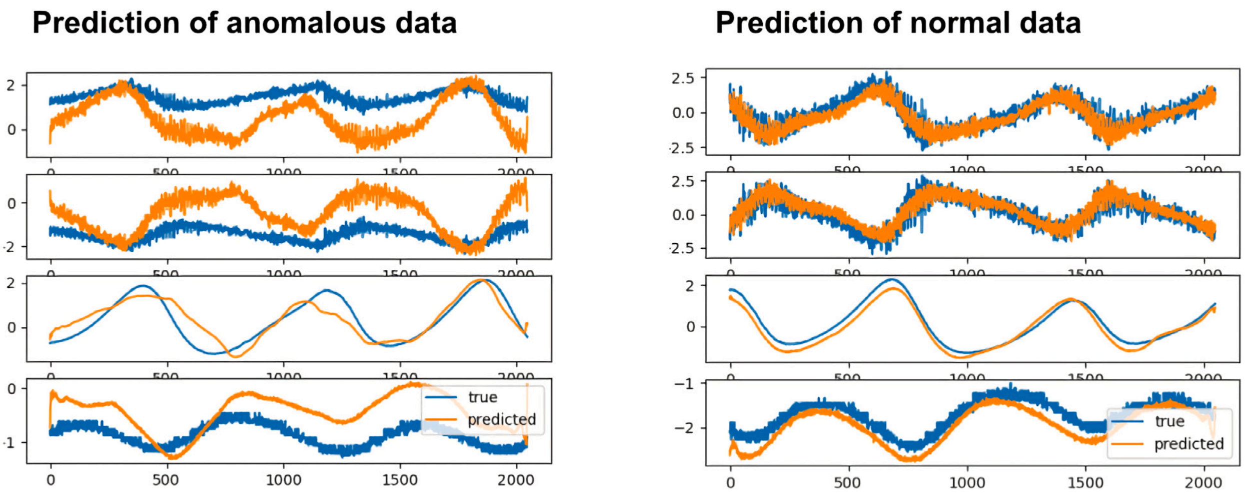


Fig. 9. This figure shows a single step of predicted signal (in orange) against the true signal (in blue). In the left figure the true signal is taken from a normal period while in the right figure the true signal is anomalous. We can observe that in the first picture the prediction is quite accurate, while in the second one it deviates significantly from the ground truth.

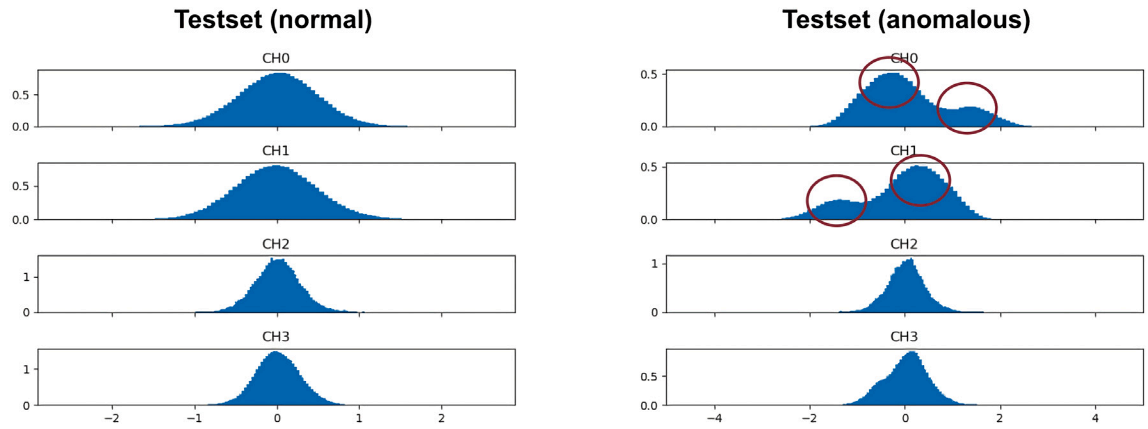


Fig. 10. Distribution of the residuals of two distinct periods from the trainingset. The one on the left refers to a period with no anomalies, while the one on the right refers to the period with the known anomaly.

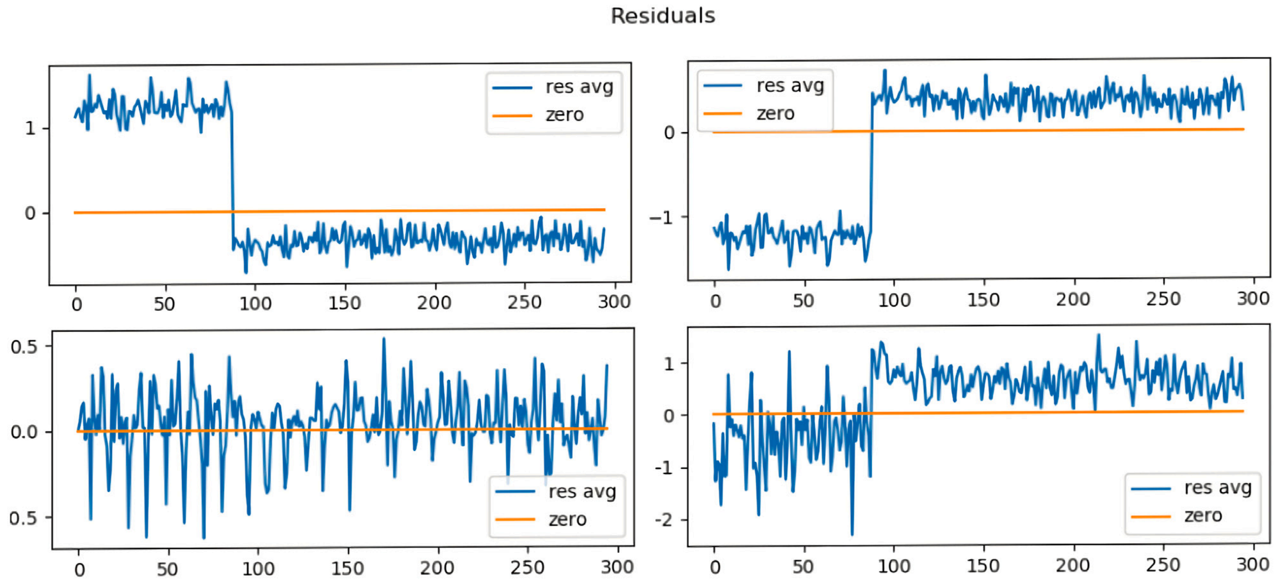


Fig. 11. In this plot we see the evolution of the residuals in the anomalous period. From the first two channels (first row) we can clearly identify the anomaly, that can be localized in the period that goes from the first block, to the block number 80.

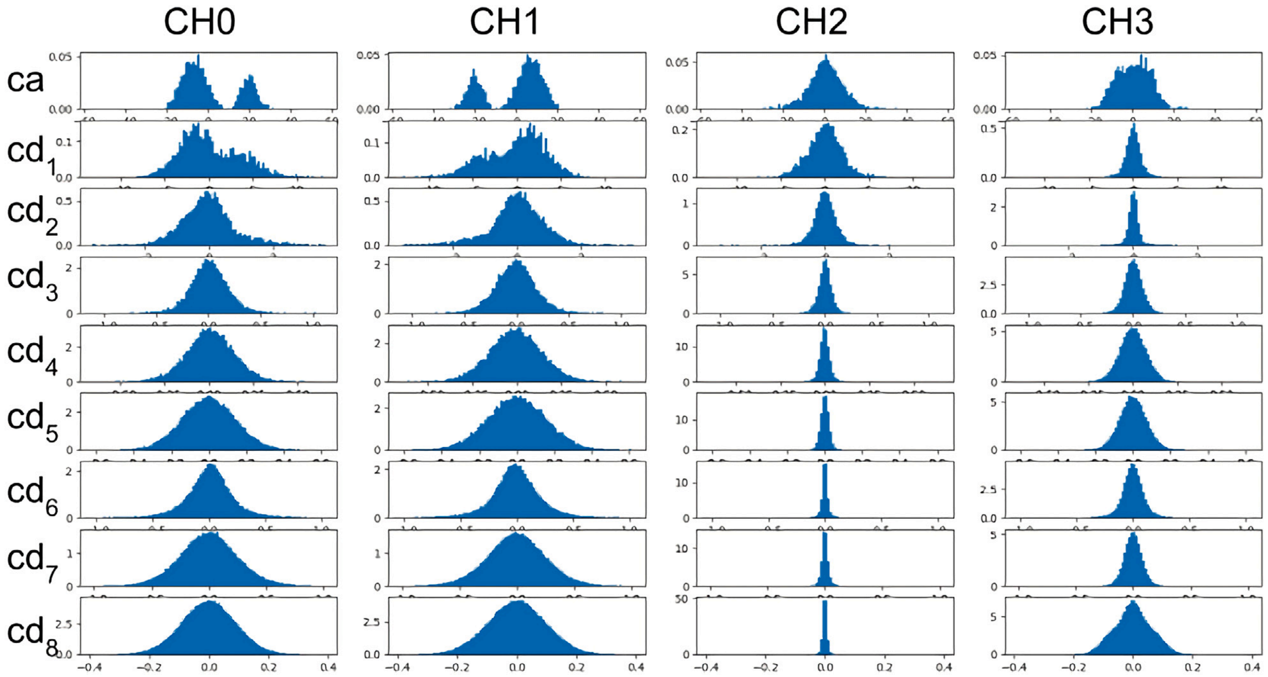


Fig. 12. The figure shows the distribution of the residuals for each wavelet band. In particular we can observe that the anomaly is localized in the low frequency components of the signal.

can easily separate the noise from the anomalies under the assumption that they live in different wavelet bands. With the proposed approach, we can also compute the error directly on the wavelet coefficients and analyze the residuals independently for each wavelet band (as shown in Fig. 12). The same principle can also be used to inject prior knowledge in the method, since if we know that anomalies cannot appear in some wavelet bands we can safely ignore them.

5. Conclusion

The difficulty in the prediction of satellite telemetries resides both in the multi-dimensionality of the input data and in the high number of samples. The proposed wavelet recurrent network was able to capture long term dependencies building an internal model of the signal that allowed to predict a high number of samples with a large time lag between the input samples and the predicted ones leveraging the multiresolution decomposition performed by the wavelet transform. The proposed model outperformed the LSTM recurrent network both in terms of accuracy and in the number of predicted samples. The prediction of telemetry, especially when the predicted data is multiple steps ahead in the future, allows acting in advance in order to be prepared to handle the probable configuration that the monitored system could assume in the near future. We also showed how the proposed method can be applied to the task of anomaly detection, that is of paramount importance in the field of telemetry analysis. However we were not able to properly evaluate the model for this second task, since our dataset contains only a single example of anomaly. We therefore leave more comprehensive experimentation in this area for a future work.

CRediT authorship contribution statement

Christian Napoli: Investigation, Conceptualization. **Giorgio De Magistris:** Investigation, Methodology. **Carlo Ciancarelli:** Data curation. **Francesco Corallo:** Data curation. **Francesco Russo:** Data curation. **Daniele Nardi:** Supervision, Reviewing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., ... Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org URL https://www.tensorflow.org/.

Amaratunga, K., Williams, J. R., Qian, S., & Weiss, J. (1994). Wavelet–Galerkin solutions for one-dimensional partial differential equations. *International Journal for Numerical Methods in Engineering*, *37*(16), 2703–2716.

Boné, R., & Crucianu, M. (2002). Multi-step-ahead prediction with neural networks: a review. *9emes Rencontres Internationales: Approches Connexionnistes En Sciences*, *2*, 97–106.

Burrus, C. S. (2015). *Wavelets and wavelet transforms*. OpenStax CNX.

Capizzi, G., Napoli, C., & Bonanno, F. (2012). Innovative second-generation wavelets construction with recurrent neural networks for solar radiation forecasting. *IEEE Transactions on Neural Networks and Learning Systems*, *23*(11), 1805–1815.

Chakraborty, K., Mehrotra, K., Mohan, C. K., & Ranka, S. (1992). Forecasting the behavior of multivariate time series using neural networks. *Neural Networks*, *5*(6), 961–970.

Daubechies, I. (1992). *Ten lectures on wavelets*. SIAM.

Donoho, D. L., & Johnstone, I. M. (1994). Threshold selection for wavelet shrinkage of noisy data. In *Proceedings of 16th annual international conference of the IEEE engineering in medicine and biology society, Vol. 1* (pp. A24–A25). IEEE.

Duan, M., Li, K., Liao, X., & Li, K. (2017). A parallel multiclassification algorithm for big data using an extreme learning machine. *IEEE Transactions on Neural Networks and Learning Systems*, *29*(6), 2337–2351.

Gürsoy, O., & Engin, S. N. (2019). A wavelet neural network approach to predict daily river discharge using meteorological data. *Measurement and Control*, *52*(5–6), 599–607.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

Hung, C. W., Mao, W. L., & Huang, H. Y. (2019). Modified PSO algorithm on recurrent fuzzy neural network for system identification. *Intelligent Automation and Soft Computing*, *25*(2), 329–341.

Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: principles and practice*. OTexts.

Jarrah, M., & Salim, N. (2019). A recurrent neural network and a discrete wavelet transform to predict the saudi stock price trends. *International Journal of Advanced Computer Science and Applications*, *10*(4), 155–162.

Jozefowicz, R., Zaremba, W., & Sutskever, I. (2015). An empirical exploration of recurrent network architectures. In *International conference on machine learning* (pp. 2342–2350). PMLR.

Kim, S., & Kim, H. (2016). A new metric of absolute percentage error for intermittent demand forecasts. *International Journal of Forecasting*, *32*(3), 669–679.

Kubat, M. (1999). Neural networks: a comprehensive foundation by simon haykin, macmillan, 1994, ISBN 0-02-352781-7. *The Knowledge Engineering Review*, *13*(4), 409–412.

Lewis, A. S., & Knowles, G. (1992). Image compression using the 2-D wavelet transform. *IEEE Transactions on Image Processing*, *1*(2), 244–250.

Luisier, F., Blu, T., & Unser, M. (2007). A new SURE approach to image denoising: Interscale orthonormal wavelet thresholding. *IEEE Transactions on Image Processing*, *16*(3), 593–606.

Makridakis, S. (1993). Accuracy measures: theoretical and practical concerns. *International Journal of Forecasting*, *9*(4), 527–529.

Park, S. H., Kim, B., Kang, C. M., Chung, C. C., & Choi, J. W. (2018). Sequence-to-sequence prediction of vehicle trajectory via LSTM encoder-decoder architecture. In *2018 IEEE intelligent vehicles symposium* (pp. 1672–1678). IEEE.

Pascanu, R., Mikolov, T., & Bengio, Y. (2013). On the difficulty of training recurrent neural networks. In *International conference on machine learning* (pp. 1310–1318). PMLR.

Pu, B., Li, K., Li, S., & Zhu, N. (2021). Automatic fetal ultrasound standard plane recognition based on deep learning and iIoT. *IEEE Transactions on Industrial Informatics*, *17*(11), 7771–7780.

Ravikanth, G., Sunitha, K., & Reddy, B. E. (2020). Location related signals with satellite image fusion method using visual image integration method. *Computer Systems Science and Engineering*, *35*(5), 385–393.

Salgado, C. M., Azevedo, C., Proença, H., & Vieira, S. M. (2016). Noise versus outliers. *Secondary Analysis of Electronic Health Records*, 163–183.

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. arXiv preprint arXiv:1409.3215.

Teukolsky, S. A., Flannery, B. P., Press, W., & Vetterling, W. (1992). Numerical recipes in C. *SMR*, *693*(1), 59–70.

Wang, J., Yang, Y., Wang, T., Sherratt, R. S., & Zhang, J. (2020). Big data service architecture: a survey. *Journal of Internet Technology*, *21*(2), 393–405.

Wang, J., Zou, Y., Lei, P., Sherratt, R. S., & Wang, L. (2020). Research on recurrent neural network based crack opening prediction of concrete dam. *Journal of Internet Technology*, *21*(4), 1161–1169.

Yang, G., Zeng, J., Yang, M., Wei, Y., & Wang, X. (2020). Ott messages modeling and classification based on recurrent neural networks. *Computers, Materials & Continua*, *63*(2), 769–785.

Zhang, J., Zhong, S., Wang, T., Chao, H.-C., & Wang, J. (2020). Blockchain-based systems and applications: a survey. *Journal of Internet Technology*, *21*(1), 1–14.