Reconstruction of equidistant time series using neural networks

Ludwik Liszka
Swedish Institute of Space Physics
Sörfors 634
S-905 88 Umeå
Sweden

Introduction

Under certain circumstances it is impossible to perform equidistant data sampling of a time series. An example may be astronomical observations from the Earth surface, where the meteorological conditions are the limiting factor. In order to perform frequency analysis of such data, it has been necessary to use special algorithms for non-equidistant data (c.f. e.g. Wilcox & Wilcox, 1995, Breedon, J.L. and Packard N.H., 1992). Such algorithms are derived under the assumption of stationarity of the monitored process. If the observed process is non-stationary it is practically impossible to perform a frequency analysis of the data. For that reason it is necessary to convert the measured data into equidistant data. A possible method to convert non-equidistant data to equidistant data, using neural networks, is shown here.

Problem description

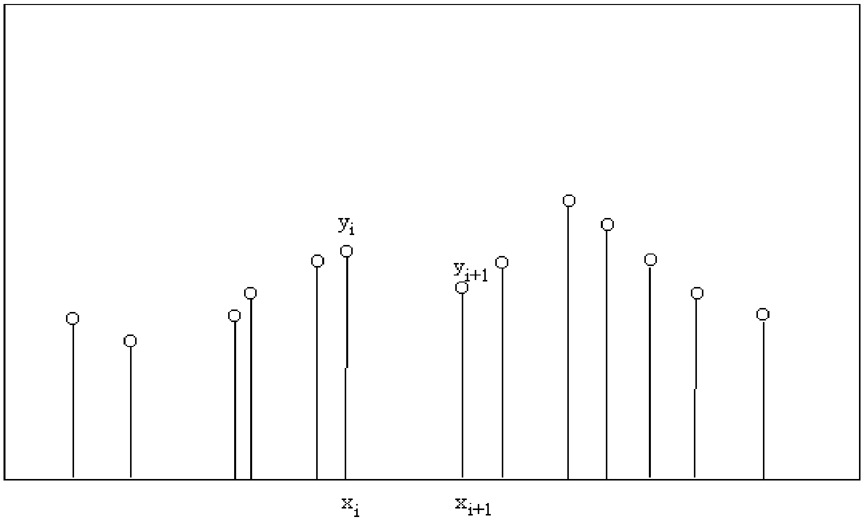


Fig. 1. A time series with non-equidistant points.

Assume a measured variable y_i sampled at non-equidistant points x_i (see Fig. 1). The non-linear interpolation between two points (y_i, x_i) and (y_{i+1}, x_{i+1}) may be performed in two steps:

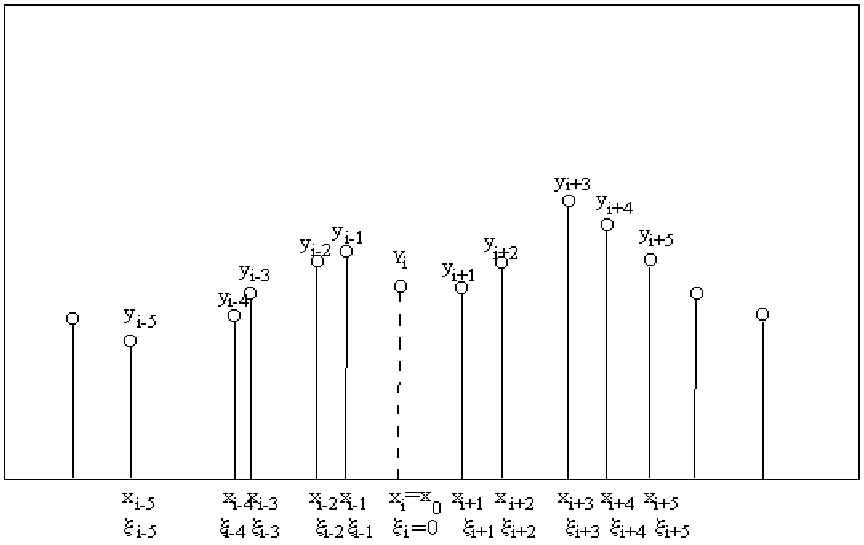


Fig. 2. Data points used to prepare a training file.

1. In the first step a window consisting of 2k+1 points of the time series is used to train a neural network model of the time series (see Fig. 2 for k=5). The y-value corresponding to the point $x_i = x_0$, in the window, y_i, is used as a desired output in a back-propagation network (Rumelhart, 1986); see Fig 3. The corresponding x-values are transformed into ξ-values with the x_0-coordinate of the point y_i as origo:

$$\xi_i = x_i - x_0 = 0 \qquad (1)$$

The input vector consists of following 4k values:

$$\xi_{i-k},\ y_{i-k},\ \xi_{i-(k-1)},\ y_{i-(k-1)},\ \ldots.\ \xi_{i-1},\ y_{i-1},\ \xi_{i+1},\ y_{i+1},\ldots\ldots\xi_{i+(k-1)},\ y_{i+(k-1)},\ \xi_{i+k},\ y_{i+k} \qquad (2)$$

Since the ξ_i value is always equal to 0, it may be omitted from the input data fed into the network (see Fig. 2).

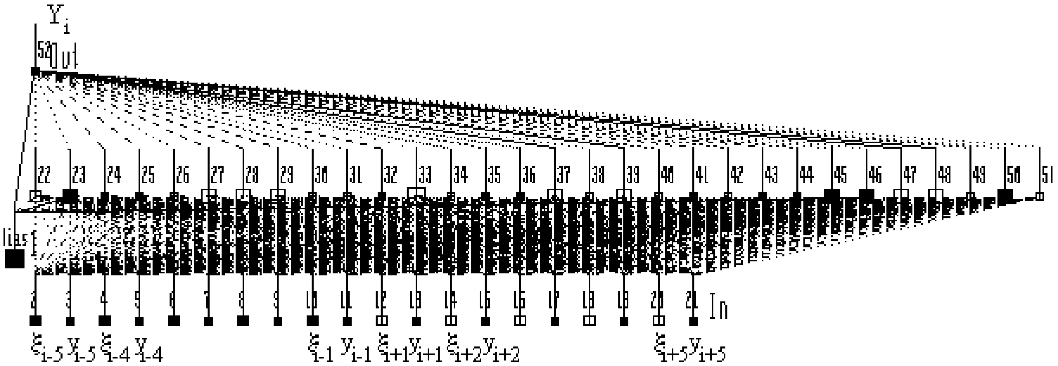


Fig. 3. An example of the neural network of the back-propagation type used for modeling of the observed non-equidistant time series; k=5.

. The window is moved through all the data, and a training file is constructed. The training file is then used to construct the model of the time series. If the time series is non-stationary, a hybrid model (Liszka, 1993), consisting of a back-propagation network and a self-organizing map network (SOM) (cf Kohonen, 1989) may be used. The model will, at the recall, reproduce the y-value corresponding to a point between x_{i-1} and x_{i+1}.

2. In the second step a recall file is constructed for the measured data. The interpolation may start after the point no. k of the time series, and it may be carried on until the point no. N-k, where N is the number of observed points of the time series. The x_i-coordinate of the interpolated point is then moved at constant steps between the points no. i and i+1. For each interpolated point in that interval there will be a vector consisting of the same set of y-values and a new set of ξ-values, depending on the location of the interpolated point with respect to the points i and i+1.

A three-step model

It has been found that the best results will be obtained using a three-step model. The principle of the model is shown in Fig. 4. The model consists of two back-propagation networks, BP1 and BP2 and one SOM network. The procedure starts at the bottom of the diagram where the BP1 is trained with the training file consisting of input vectors, as shown in (2), and of the desired outputs y_i. The BP1 is then used to estimate the first approximation of the interpolated values, Y_i.

The same training file is then used to train the SOM-network. In the present study, networks with 10 x 10 processing elements self-organizing maps have been used. The recall is made twice.

The first time, the recall is made with the training file in order to obtain the Kohonen coordinates r_i, c_i (row & column no. on the map). These will be added to the original training input vector $\mathbf{T}1_i$ in the second back-propagation network, BP2. Using the SOM-network in the interpolation process is equivalent to the categorization of the shape of the data contained in the 2k point's window. Adding the Kohonen coordinates to the input vector is equivalent to adding a category membership index to the original input data.

The second recall is made using the recall file consisting of vectors $\mathbf{R}1_i$ and of the interpolated values Y_i, estimated by the first back-propagation network BP1. This step will produce a set of Kohonen coordinates corresponding to the recall file.

In the last step of the interpolation process the BP2 network is trained with the input vectors supported by the Kohonen coordinates. The last recall generates the final interpolated data, Y_{fi}.

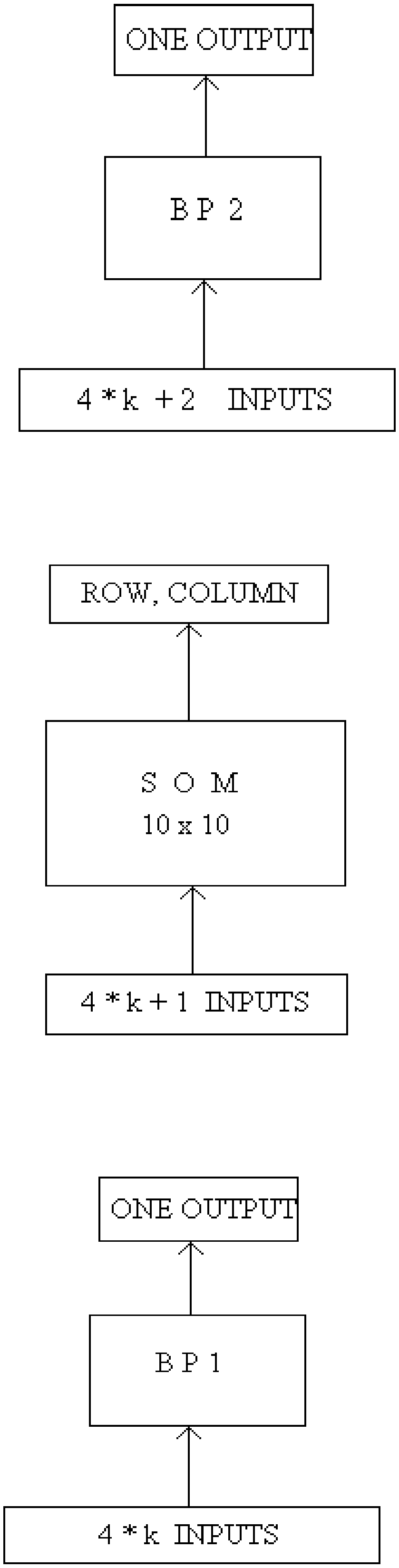
The principle of the model may be described as follows: The first network BP1 finds a value of Y_i, for which the conditional probability:

$$P(Y_i \mid \xi_{i-k}, y_{i-k}, \xi_{i-(k-1)}, y_{i-(k-1)}, \dots \xi_{i-1}, y_{i-1}, \xi_{i+1} y_{i+1}, \dots\dots \xi_{i+(k-1)}, y_{i+(k-1)}, \xi_{i+k}, y_{i+k})$$

is a maximum. The SOM network is categorizing the actual window together with the estimated Y_i value. The Kohonen coordinates generated by the SOM network are a kind of two-dimensional category membership index of the interpolated data. The Kohonen coordinates facilitate the function of the last network BP2 which finds the final value of Y_{fi}, for which the conditional probability:

$$P(Y_{fi} \mid \xi_{i-k}, y_{i-k}, , \dots \xi_{i-1}, y_{i-1}, \xi_{i+1} y_{i+1}, \dots\dots, \xi_{i+k}, y_{i+k}, R_i, C_i)$$

is a maximum.



Training: y_i

Recall: Y_{fi} - Final interpolated data

Back-Propagation Network

Training: { $\mathbf{T}1_i$, r_i, c_i }

Recall : { $\mathbf{R}1_i$, R_i, C_i }

Kohonen Coordinates

Recall with the training file: r_i, c_i

Recall with the recall file: R_i, C_i

Self-Organizing Map Network

Training: { $\mathbf{T}1_i$, y_i }

Recall: { $\mathbf{R}1_i$, Y_i }

Training: y_i (observed values)

Recall: Y_i (interpolated values)

Back-Propagation Network

Training: $\mathbf{T}1_i$ (one vector for each observed value)

Recall: $\mathbf{R}1_i$ (one vector for each interpolated value)

Fig. 4. The model used for interpolation of non-equidistant observations.

It must be emphasized that the value reconstructed by the method is not necessarily identical with the true one, i. e. the value which would be measured at the actual instant.

An important question is how to select a proper value of k, which will give the best results of interpolation. In general, the proper value of k will depend on the statistical properties of the measured variable y and on the distribution of time intervals between the measurements.

The present interpolation method will be illustrated by two examples.

Example 1: A stationary, deterministic variable being a sum of two harmonic components

A time series consisting of 8000 points has been calculated using an expression:

$$Y(x_i) = A_1 (1+n_1) \cos(2\pi x_i/T_1) + A_2 (1+n_2) \cos(2\pi x_i/T_2)$$

where n_1 and n_2 are random numbers between 0 and 0.1 acting as noise factors. In order to simulate realistic data, a noise with a maximum amplitude of 10% is added to the signal. A fraction of the time series is shown in Fig. 5. A non-equidistant time series consisting of 2000 points was then constructed by selecting (in random) 25% of points from the original time series.

The resulting non-equidistant time series is shown in Fig. 6.

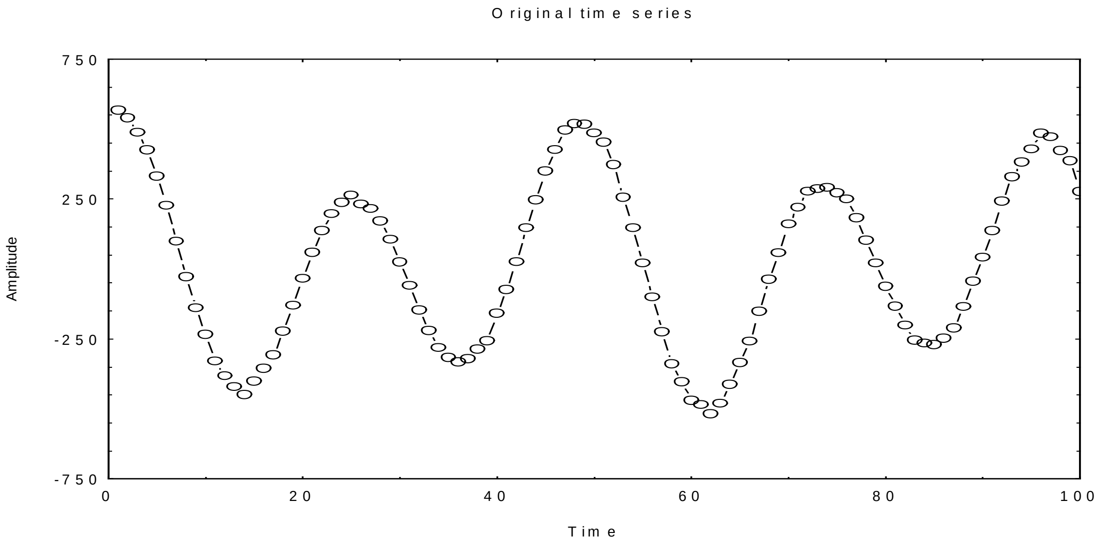


Fig. 5. An initial part of the original time series

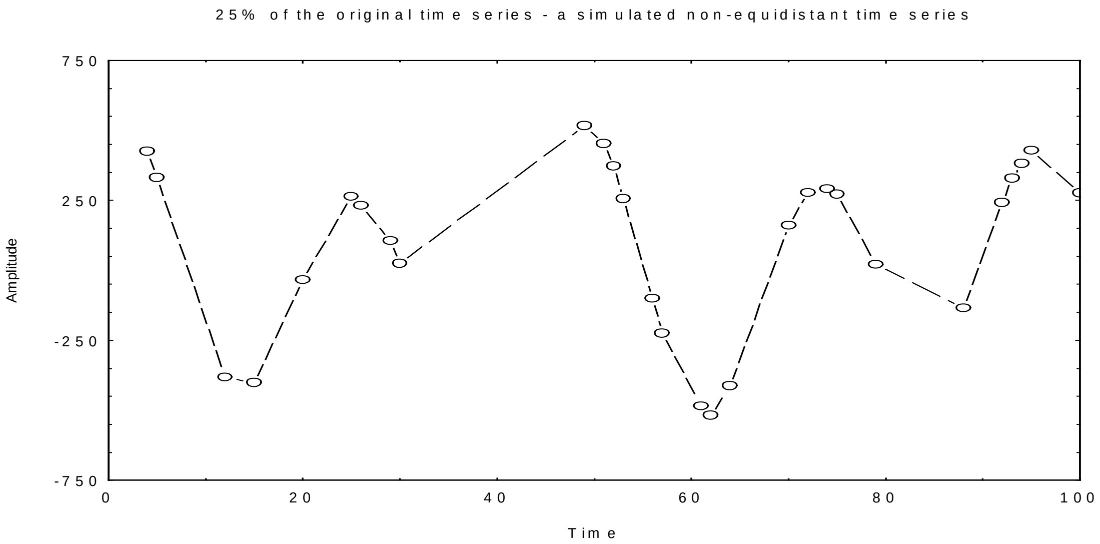


Fig. 6. 25% randomly selected points from the original time series - a simulation of a non-equidistant time series

A comparison of the original data, the non-equidistant data, being 25% of the original data, and finally the equidistant data computed from the non-equidistant data, using a k=2 model, is shown in Fig. 7 for a small section of the time series.

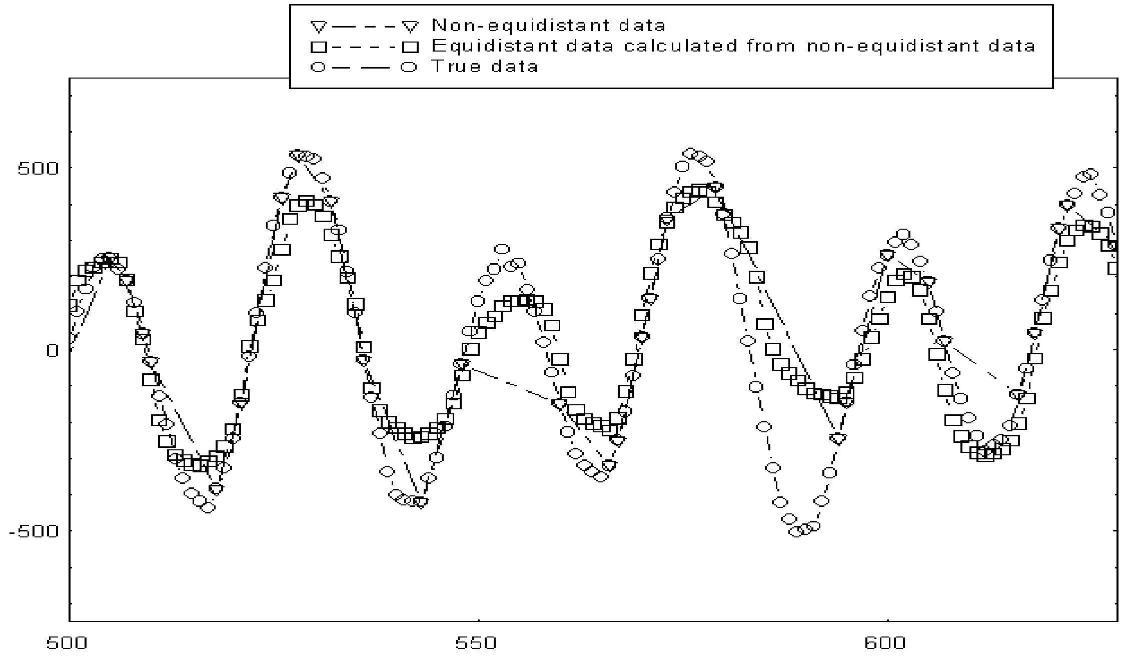


Fig. 7. A comparison of the original data (circles), the non-equidistant data, being 25% of the original data (triangles), and finally the equidistant data computed from the non-equidistant data (squares).

It may be interesting to compare the structure of the intermediate result (after BP1) with the final result (after BP2). A short section of the time series showing both the intermediate and the final, reconstructed time series is shown in Fig. 8.

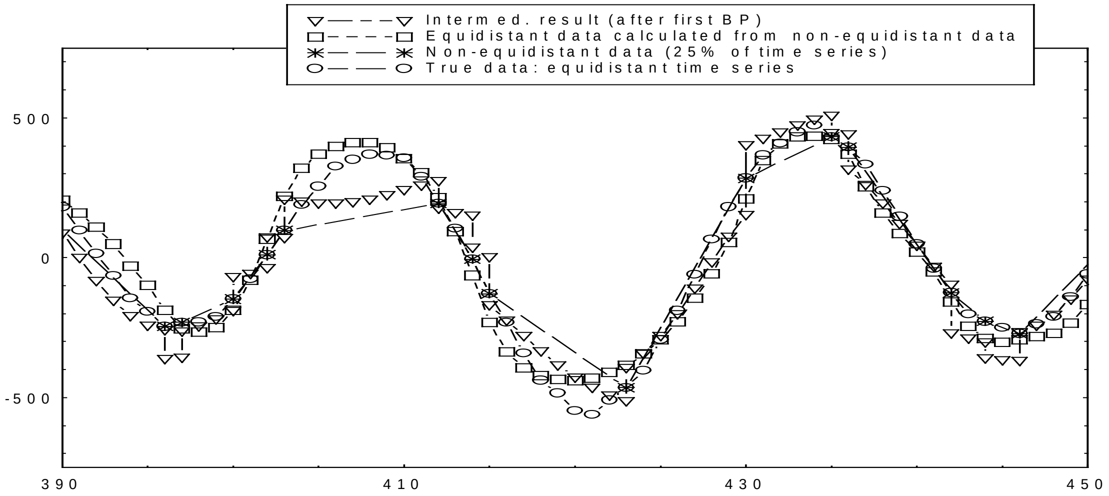


Fig. 8. A comparison of the original data (circles), the non-equidistant data, being 25% of the original data (asterisks), intermediate results after the BP1 network (triangles) and finally the equidistant data computed from the non-equidistant data (squares).

In connection with observations of time series, it is usually interesting to study the frequency spectrum. The reconstruction of the time series must be performed in such a way that the frequency spectrum of the reconstructed time series will be as close to the true frequency spectrum, as possible. For the present time series, the FFT was performed using a 128 points sliding constant window stepping 8 points at a time. Average linear frequency spectra for the original time series and for the reconstructed time series, both the final results and the intermediate results, are shown in Fig. 9. It may be seen that the frequency spectrum of the final reconstructed time series agrees well with the original spectrum. As it could be expected, the intermediate result of the reconstruction gives a spectrum, which is much further from the true one than the spectrum of the final reconstructed time series.

An interesting detail is that the reconstruction process does not seem to produce any significant high frequency components.

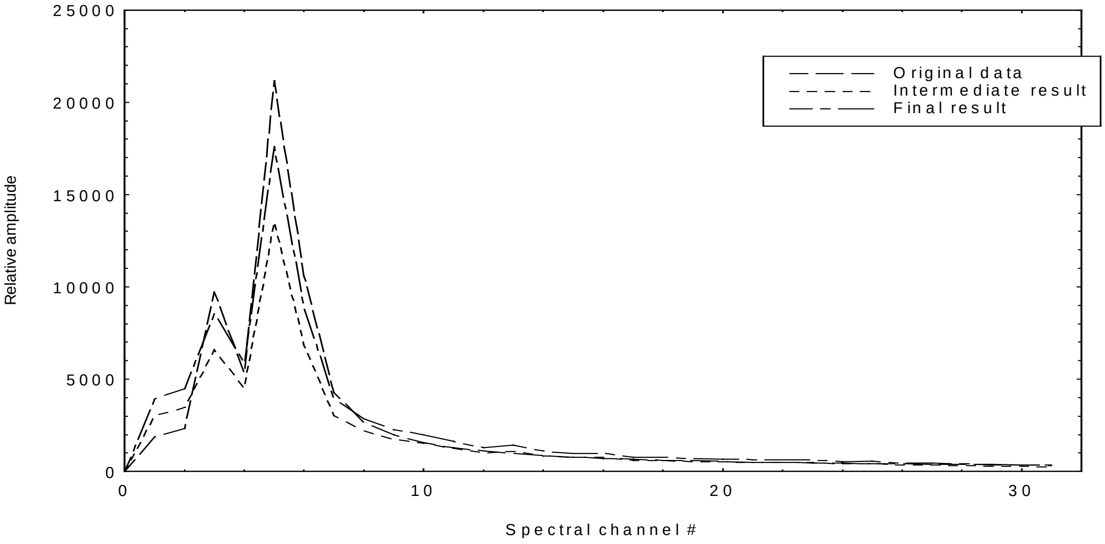


Fig. 9. Linear frequency spectra of the original time series, of the intermediate result after the BP1 network and of the final result after the BP2.

Example 2: Reconstruction of Wolf numbers

The present technique to reconstruct time series was also tested on 9 years of solar sunspot numbers (Wolf numbers), being a typical non-stationary, non-deterministic time series. Years 1980 - 89, including a sunspot minimum, were used in the present example. In the present example only 20% randomly chosen data points were used to simulate the non-equidistant data. A three stage neural network model with k=2 has been used for reconstruction of the data in the same way as in the previous example.

The entire original time series is shown in Fig 10.

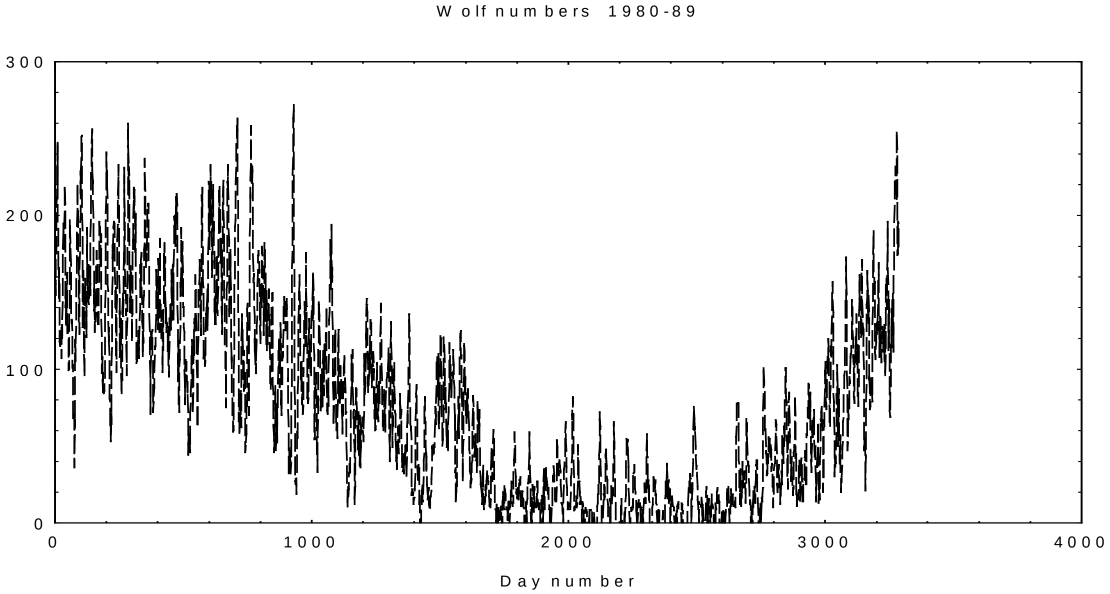


Fig. 10. Solar sunspot numbers (Wolf numbers) during 1980-89

It may be seen from the example of Fig. 12 that the method reconstructs, in some cases, missing structures in the data, as between day's no. 365 and 380. In other cases, as between day's no. 305 and 325, the reconstruction is not very accurate, but it shows the general shape of the curve when the data are missing.

The average frequency spectra obtained using a constant window consisting of 128 points, shifted 8 points of the time series at a time, for the whole period 1980-89 are shown in Fig. 13 for both the original and the reconstructed data. It may be seen that for frequencies below channel #4 (corresponding to a period of 32 days) the spectrum of the reconstructed data shows lower amplitudes than the original spectrum. The opposite is observed for higher frequencies. The differences are, however, small as the diagram displays the linear spectral amplitudes. The location of spectral peaks is essentially the same for both spectra.

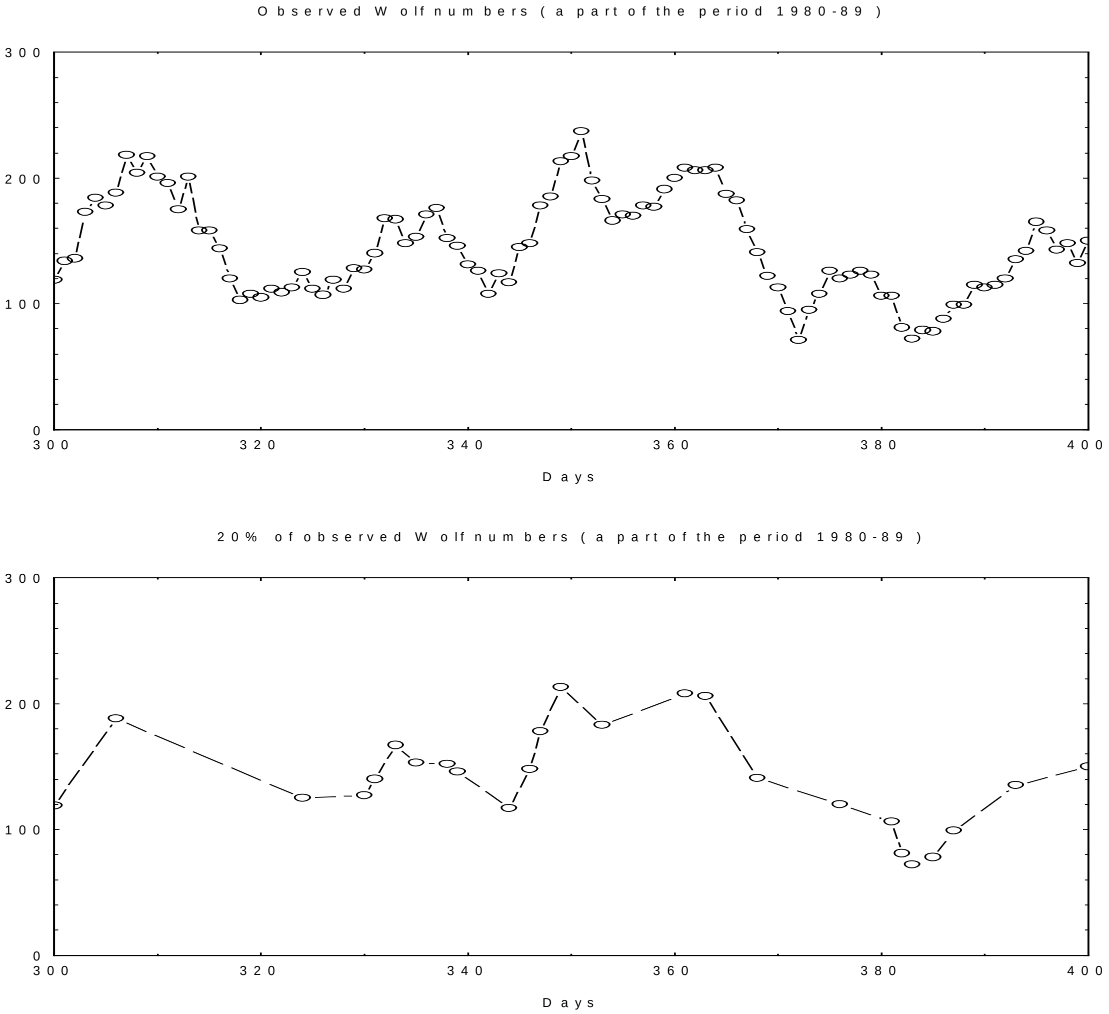


Fig. 11. Wolf numbers during 100 days at the end of 1980 (upper diagram) and 20% randomly selected Wolf numbers from the same period (lower diagram).

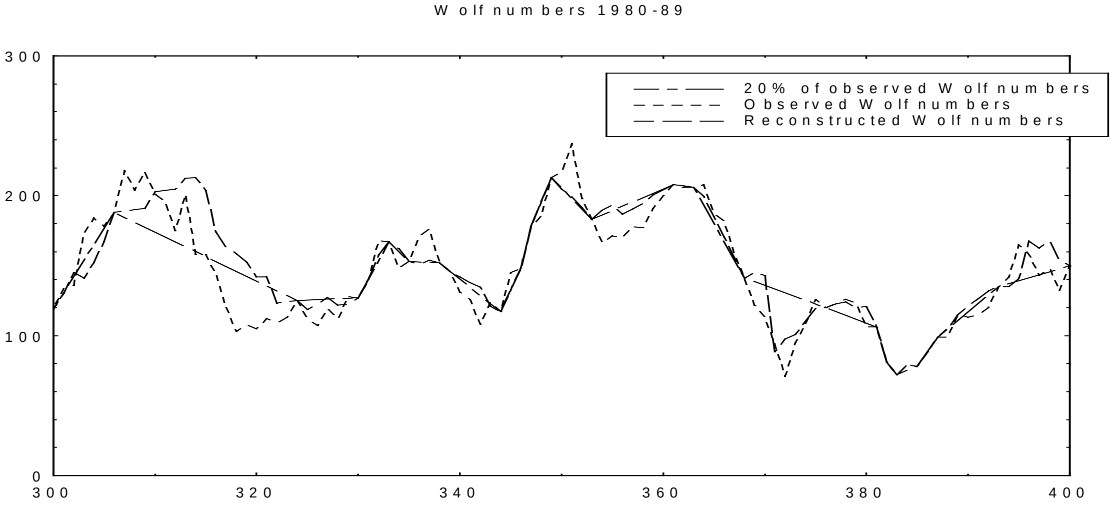


Fig. 12. A comparison of observed Wolf numbers at the end of 1980, 20% randomly selected Wolf numbers and reconstructed Wolf numbers for the same period

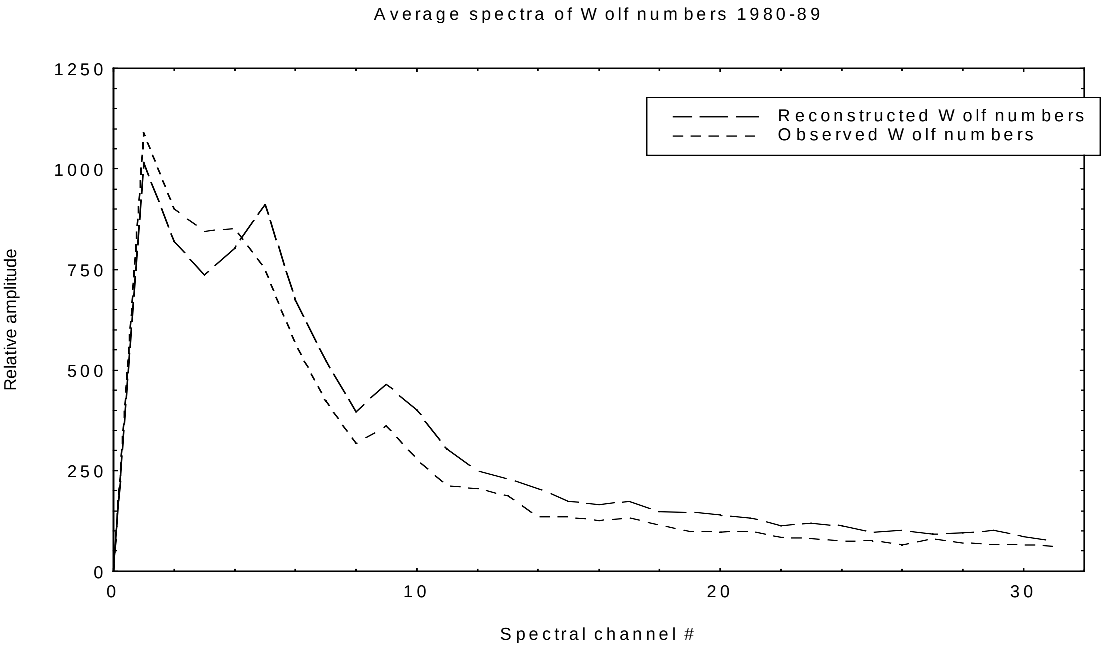


Fig. 13. Average spectra of observed and reconstructed Wolf numbers during the entire period 1980-89.

Conclusions

A three-step model of a time series, if properly designed, may be useful for conversion of non-equidistant measurements into an equidistant time series. The reconstructed time series shows essentially the same frequency spectrum as the original time series. The present method may be used for processing observations of both stationary and non-stationary processes.

References

Breedon, J. L., Packard, N.H.: Nonlinear analysis of data sampled nonuniformly in time. Physica D, 58, p.273, 1992.

Kohonen, T.: Self-Organization and Associative Memory, Springer-Verlag 1989.

Liszka, L.: Modelling of Pseudo-Indeterministic Processes Using Neural Networks. Invited Lecture at the International Workshop on Artificial Intelligence Applications in Solar-Terrestrial Physics. Lund, Sweden, 22-24 September 1993.

Rumelhart, D.E.: Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume I, Foundations. MIT Press, 1986.

Wilcox J. Z., Wilcox T. J.: Algorithm for extraction of periodic signals from sparse, irregularly sampled data. Astronomy and Astrophysics Supplement, V. 112, p.395, 1995.