NORWEGIAN UNIVERSITY OF TECHNOLOGY AND SCIENCE

Load Prediction using Hilbert-Huang Transformation

SMART GRID: HUGE DATA

By:
Jonas Eikli

Abstract

The new Smart Grid is the power grid for the next generation. It is based on the current power grid, but with improvements in every category. One of the main things to improve in the current power grid is the efficiency. With better prediction of power consumption, the efficiency can be increased. This paper tries to answer the question can the Hilbert-Huang Transformation be used for prediction, and specifically, for this kind of prediction. The results show that...

Preface

PREFACE

Contents

List of Figures

List of Tables

1 Introduction

The current power grids in use all over the world are inherently inefficient. The main reason for this is that only one third of the fuel is turned into electricity, and the overflow is wasted. The reason for this inefficiency is that it is designed to withstand maximum load at all times. 20% of the generation capacity exists to meet peak demands. If the electricity suppliers had more knowledge of how much power was needed, and when it was needed, the waste could be substantially reduced. As we spend more energy, and the resources are starting to dwindle, efficiency is becoming more and more important. This also holds true for areas with limited resources available.

Several other methods exists for predicting, or forecasting, electricity demand.

The Hilbert-Huang Transform is a method to decompose a signal to get the instantaneous frequency.

Research questions:

1. Can Hilbert-Huang Transformation be used for prediction?

2. Can Hilbert-Huang Transformation be used to predict the load of a power grid?

2 Power Grid

The power grid, or electrical grid, is a vital part of the infrastructure in any country. It serves to transport electricity over great distances, from suppliers to consumers, or from one country to another. While today's power grids certainly are impressive, they are far from perfect. It turns one-third of the fuel into electricity, without recovering the waste. Almost 8% of its output is lost along the transmission lines, while 20% of its generation capacity exists to meet peak demand (i.e., it is only in use 5% of the time) [2].

The power grid is made up from three main components. The *power stations* that produce the electricity, the *transmission lines* that transports the electricity from the suppliers to demand centers, and the *transformers* that reduce voltage so *distribution lines* can deliver it to consumers.

The power stations are usually located away from heavily populated areas. Because of this, a network of power lines are necessary to transport the electricity to where it is needed. This is done first over the transmission lines, then over the distribution lines. On the transmission lines the electricity is transmitted using at high voltages (110kV or above) to reduce the energy lost over long distances. The high

voltage electricity is delivered to substations, or transformers, that reduce the voltage and sends the electricity out on the distribution lines. When the power finally arrives at a service location, the voltage is stepped down again, from distribution voltage to the required service voltage. The electricity is then ready for the typical household.

2.1   Evolution of the Smart grid

The existing power grid is essentially a one-way pipeline where the source have no real-time information about the end points. Because of this, the grid is over-engineered to withstand maximum anticipated peak demand across its aggregated load. Since this is an infrequent occurrence, this system is inherently inefficient. The suppliers need to know, or at least predict, how much power is needed in order to increase efficiency.

The earliest attempts from the power industry at solving this was the automated meter reading (AMR). AMR let utilities read the consumptions records, alarms and status from customers' premises remotely. While this looked promising at first, it did not solve the major problem, that is, demand-side management. It is limited to reading data, so it can not take corrective action based on the received information. AMR systems do not allow the transition to the smart grid, where pervasive control at all levels is a basic premise [2].

As a consequence of this, AMR was short lived. Instead, the advanced metering infrastructure (AMI) was introduced. Through AMI, utilities get a two-way communication system to the meter, as well as the ability to modify customers' service-level parameters. AMI paved the path for the new smart grid, which is currently under development.

The Smart Grid is a developing network of new technologies, equipment, and controls working together to respond immediately to our 21st century demand for electricity [8]. Is is the full range of current and proposed solutions to the challenges of electricity supply. Smart Grid spans many different fields and technologies, including intelligent agents, intelligent applications, network management, distributed control, smart sensors and two-way communication. Some of these are already done by AMI, but the rest are still under development.

The goals of the Smart Grid is to get a more efficient, more robust, self-healing power grid, that can cope with the increased demand for electricity in years to come.

3   Hilbert-Huang Transformation

The Hilbert-Huang Transformation (HHT) is a way to decompose a signal into in-trinsic mode functions (IMF), and obtain instantaneous frequency data. Unlike the Fourier transformation, it is designed to work well for data that are nonstationary and nonlinear. The HHT consists of the empirical mode decomposition (EMD) and the Hilbert Spectral analysis (HSA). The EMD is used to make IMFs and the HSA is used to obtain the instantaneous frequency data.

The EMD consists of a sifting process that is repeated a number of times. The sifting process is as follows:

1. Find all local extrema, maxima and minima, in the data.

2. Connect all the maxima using a cubic spline line as an upped envelope.

3. Connect all the minima the same way.

4. Calculate a mean between the upper and lower envelope.

5. Subtract the mean from the data.

6. The result of this is used as input in the next sifting.

A segment of the data from the power grid along with maxima- and minima-spline, as well as the mean of those are showed in Figure 1. The corresponding IMF is also showed.

There are two possible stoppage criteria that can determine how many times the sifting process is repeated. The first is proposed by Huang et al. (1998) and is similar to the Cauchy convergence test. First, a threshold is given, and when the sum of the difference is smaller that the threshold, the sifting process stop. The sum of the difference is defined as:

$$SD_k = \frac{\sum_{t=0}^{T} |h_{k-1}(t) - h_k(t)|^2}{\sum_{t=0}^{T} h_{k-1}^2(t)}. \tag{1}$$

The second possible stoppage criteria is based on the so-called S-number, which is defined as the number of consecutive siftings when the numbers of zero-crossings and extrema are equal or differ at the most by one. The sifting process with only stop when the number of successive siftings where this is true reaches the pre-selected S-number. Huang et al. (2003) suggests that the optimal range for S lies between 3 and 8.
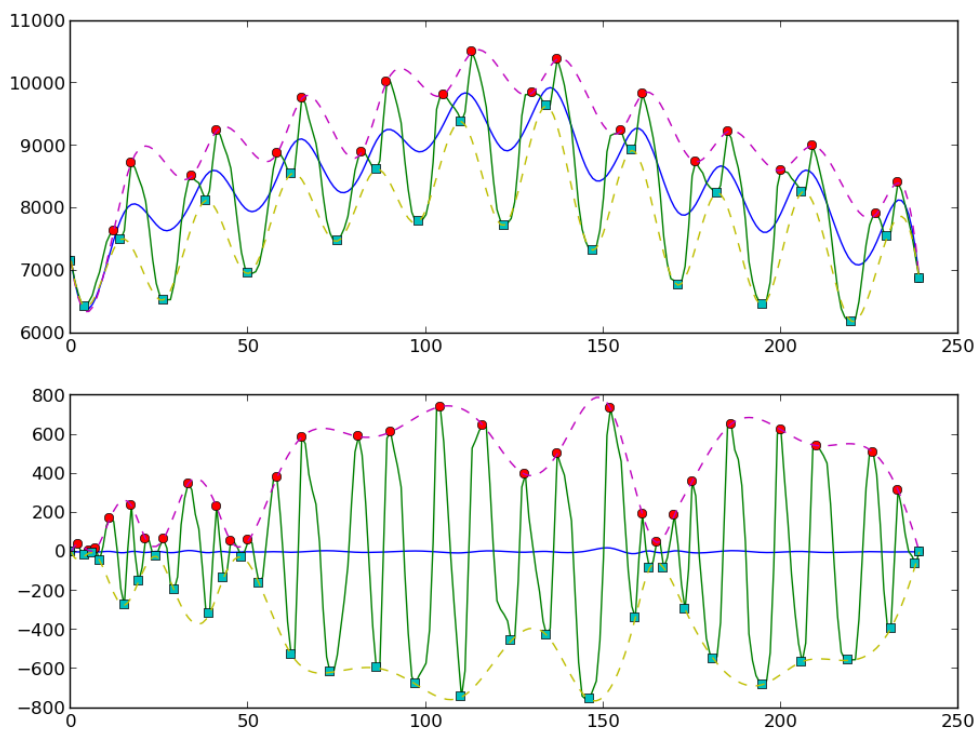
Figure 1: The first 10 days of data, with the corresponding IMF.

When the stoppage criterion is met, the first IMF is created. This IMF is removed from the original data by subtracting it, and the result is the residue. The first IMF contains the finest scale, or the shortest period component of the signal, and the residue is left with longer period variations in the data.

The whole procedure is repeated with the residue as input. This is done until the residue becomes a monotonic function from which no IMF can be extracted.

4   Artificial Neural Networks

An artificial neural network (ANN) is a mathematical model inspired by biological neural networks. The ANN consists of several layers; the input layer, one or more hidden layers, and the output layer. Each layer has one or more artificial neurons. The neurons in the different layers are connected by weighted edges. An example of an ANN, with 4 input neurons, 5 hidden neurons and 2 output neruons, can be seen in Figure 2. Each neuron also has an activation function that determines if it is activated or not. The activation level is based on the input and determines the output of the neuron. An ANN is typically defined by three types of parameters:
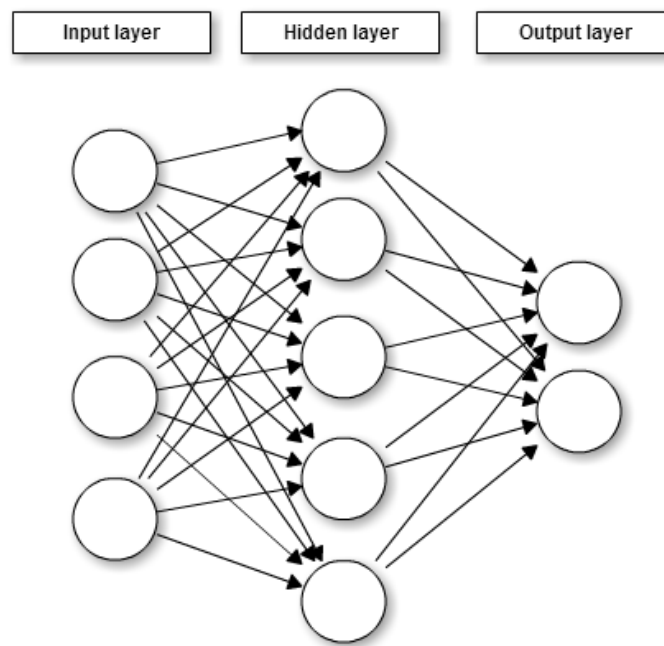

Figure 2: A typical feedforward ANN

1. The interconnection pattern between different layers of neurons.

2. The learning process for updating the weights of the interconnection.

3. The activation function that converts a neuron's weighted input to its output activation.

There are three different paradigms for training ANNs, these are:

Supervised Learning

When using supervised learning, the network is provided with both sample inputs, as well as the desired outputs for each case. The well-known *back-propagation* algorithm can be used for training. Typical tasks solved by this paradigm is pattern recognition and regression.

Unsupervised Learning

In unsupervised learning, the desired output is not given, so the goal is typically to minimize the cost function for the input data. Typical tasks here are general estimation problems.

Reinforcement Learning

In reinforcement learning, the input data is typically not given. The aim is to discover a policy for selection actions that minimize some measure of long-term cost. Typical applications for reinforcement learning are control problems like for example games.

For the task in this project, supervised learning is the obvious paradigm to use.

Based on the structured literature search in Appendix A, the neural network that will be used in this project is the General Regression Neural Network, discussed in Section 4.1.

4.1   General Regression Neural Network and the Radial Basis Function

The General Regression Neural Network (GRNN)[11] is a type of Radial Basis Function (RBF) network. An RBF is a network that uses radial basis functions, described below, as activation functions. The special property of GRNN is that it has one input neuron for each point in the training set, whereas RBF networks have a variable number of neurons that are usually lower than the number of training points.

GRNN are conceptually similar to K-Nearest Neighbour models. It is based on the theory of non-linear (kernel) regression. That is, estimation of the expected value of output given a set of inputs. Eq. 2 summarizes the GRNN logic in an equivalent non-linear regression formula. y is the output, X. $E[y|X]$ is the expected value of the output y given vector X. $f(X, y)$ is the joint probability density function of X and y.

$$E[y|X] = \frac{\int_{-\infty}^{\infty} y f(X, y) \, dy}{\int_{-\infty}^{\infty} f(X, y) \, dy} \tag{2}$$

The radial basis function is used in the second (pattern) layer of the GRNN. It is a multivariate Gaussian function shown in Eq. 3. This is an extended Gaussian function, put forward by Cacoullus[1] and adopted by Specht[11] in his GRNN design[6]. This function computes an output, θ_i, from the neurons, where X is the input vector, or the predictor variables to the GRNN, the U_i is the specific training vector represented by pattern neuron i.

$$\theta_i = e^{-(X-U_i)'(X-U_i)/2\sigma^2} \tag{3}$$

σ is the smoothing parameter. A large σ will result in a large spread Gaussian and the sample points will cover a wide range of inputs, while a small value will create a limited spread Gaussian and the sample points will cover a small range of inputs. It is important to find a good value for σ, so that it will cover up any noise, while at the same time not smoothing over important features. With a higher number of test cases, a lower value for σ can be used.

The output of the pattern layer is forwarded to the third layers, the summation neurons. Eqs. 4a and 4b show the operations performed by the weighted and 'simple' summation neurons.

$$S_s = \sum_i \theta_i \tag{4a}$$

$$S_w = \sum_i w_i \theta_i \tag{4b}$$

The sums calculated in the summation neurons are sent onwards to the fourth layer, the output neuron. The mathematical operation done in the output neuron is shown in Eq. 5. y is the final output from the GRNN.

$$y = \frac{S_w}{S_s} \tag{5}$$

GRNN has several advantages over other non-linear regression techniques[11]:

1. GRNN can provide a multivariate vector of outputs without losing generality.

2. The network is trained by one pass through the data and can generalize from examples as soon as they are stored.

3. The estimate converges to the conditional mean regression surface as more and more examples are observer, yet it can form a very reasonable regression surface based on only few examples.

4. The estimate is bounded by the minimum and maximum of the observations.

5. The estimate does not converge to poor solutions corresponding to local minima or maxima.

6. Its easy to use.

7. The network can provide a mapping from one set of sample points to another. If this is one-to-one, it can be easily reversed.

The neural network in a GRNN consist of 4 layers; the input layer, the hidden layer, the pattern layer/summation layer and the decision layer. The input layer has one neuron for each predictor variable. The hidden layer has one neuron for each case in the training set. In the pattern layer, there are two neurons, one is the denominator summation unit and the other is the numerator summation unit. The decision layer has one node which divides the value accumulated in the numerator summation unit by the value in the denominator summation unit and uses the result as the predicted output value.

Let us consider a case where there is one input value and N training cases. This network would have 1 neuron in the input layer, N neurons in the hidden layer, two in the pattern layer and one in the decision layer. This GRNN can be seen in Figure 3.
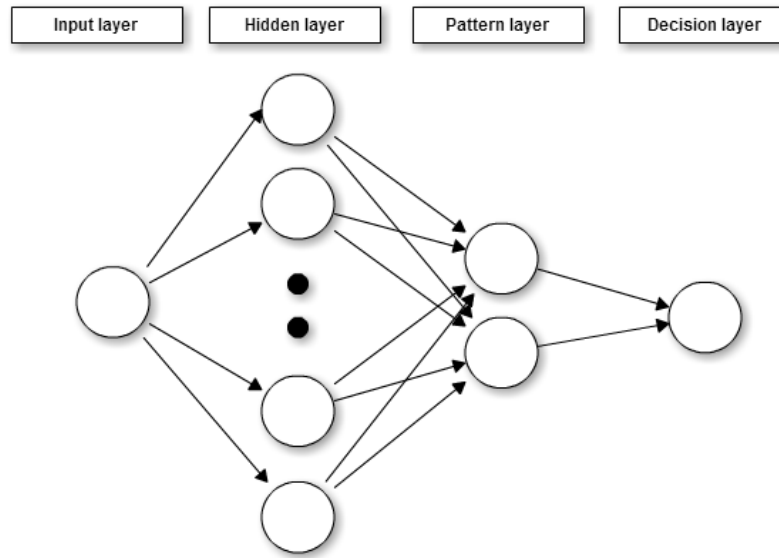
Figure 3: A general regression neural network with one input value and N training cases.

5 Implementation

This section will describe the implementation of the different parts of the project. Decisions made during the implementation will be reasoned for and explained. The power grid, which is the raw data used in this project, is discussed in Section 5.1. Section 5.2 discusses why Python selected as the programming language to use, as well as which libraries were used. The implementation of the Hilbert-Huang Transform is discussed in Section 5.3. Section 5.4 discusses the method used for prediction, Artificial Neural Network, and why that was chosen.

The source code for the entire project can be found in Appendix B.

5.1 Power Grid

The data used in this project is from a database provided by the advisor. The data is collected over a period of 6 years from a distribution network in Canada. Because it is from a distribution network, there is very little noise in the data, as the distribution network provides power to a lot end users. This results in a very good signal, that clearly shows each day. A section of this signal can be seen in Figure 4. This figure shows 10 days. It is easy to see the days, with peaks in the morning and evenings, and low points at night and during the day when people are at work. A lot more

power is consumed during the days in the middle of the graph. Why this is, can be hard to say. The most likely reason is that there was cold weather during this period.
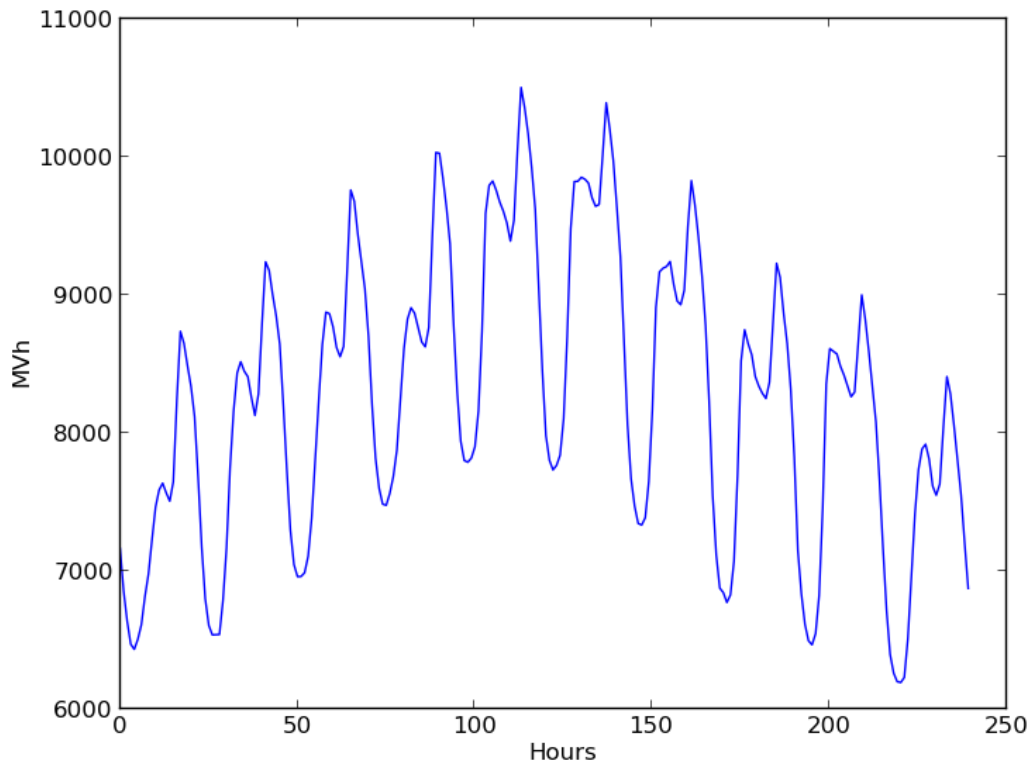

Figure 4: 10 days of power consumption

The data is stored in a *SQLite*[1] database, as Pandas Time-Series objects. Pandas time-series objects is a way to store time series data, and it provides a lot of functionality for working with such data. The database and python methods for accessing it was provided by the advisor at the beginning of the project.

---

[1]http://www.sqlite.org/

5.2   Python

Python[2] is a general purpose, interpreted high-level programming language. The reasons for choosing Python as programming language in this project are:

- Pre-existing code for retrieving data from the database containing power consumption data.

- Python is easy to use, and easy to get started with. It's ideal for systems of this scale.

- Libraries containing useful code, as discussed below.

- Using Python was recommended from the advisor.

- Python, as well as most of the libraries, are free.

One of the main reasons for using Python is that there are many libraries, or packages or modules, available to make things easier. The, not already included in Python, libraries used in this project is described below:

NumPy[3]

NumPy is the fundamental package for scientific computing with Python. It provides Python with a host of both simple and advanced mathematical functionality. It is required for using SciPy.

SciPy[4]

SciPy is another package for scientific computing. It provides modules for, among other things, statistics, optimization, numerical integration, signal processing and Fourier transforms. In this project, it is used for finding the spline functions used when finding maxima and minima in HHT, as well as in the Hilbert transform.

Pandas[5]

Pandas is a library that provides Python with high-performance, easy-to-use data structures and data analysis tools. The power consumption data from the database was stored in Pandas time series.

---

[2]http://www.python.org/
[3]http://numpy.scipy.org/
[4]http://www.scipy.org/
[5]http://pandas.pydata.org/

MatPlotLib[6]

MatPlotLib is a powerful library used for plotting 2D graphs. It is used to plot all the graphs in this report.

Peach[7]

Peach is a module based on SciPy and NumPy used to implement algorithms for computational intelligence and machine learning. The ANN used in this project is implemented using Peach. The ANN is discussed further in Section 5.4.

5.3   Hilbert-Huang Transform

HHT is used in this project to find input variables for the prediction method. Why this is done should be obvious, as this is the very reason for the project. However, as HHT generates many IMFs as output, and each IMF has a instantaneous frequency and amplitude, there is the question of which of these to choose as input to the ANN. From the literature, it is clear that there are many ways to do this, but a mix of IMFs, frequencies and amplitudes tends to perform very well. This is something that needs to be experimented on, to see what gives the best results.

The HHT was implemented from scratch during this project. There are several reasons for this; to be sure it was done correctly, to learn how it was done and to become better at writing Python. The exception to this is the method for finding the instantaneous frequency, which was taken from a python-hht project called *pyhht*[8,9]. Using this code was decided to be safe after testing it and seeing that the results looked reasonable, as well as looking on the background of the author.

5.4   General Regression Neural Network

The decision the use an artificial neural network (ANN) for prediction is based on the results of the structured literary review, Appendix A. From the articles found there, it was clear that ANN is a widely used method of prediction when dealing with time series, and it also gives the best results.

Of the ANNs used in the literature, the general regression neural network (GRNN) was selected as the one to be used in this project. The other ANNs discussed was Multilayer Perceptron (MLP) and Radial Basis Function Network. Because of time

---

[6]http://matplotlib.org/

[7]http://pypi.python.org/pypi/Peach/0.3.1

[8]http://brocabrain.blogspot.no/2011/11/python-for-hilbert-huang-transform.html

[9]https://github.com/jaidevd/pyhht/blob/master/pyhht.py

constraints, only one was implemented. The reason for choosing GRNN was that, according to the literature, it gave very good results. Additionally, the Peach library for Python has methods for implementing a GRNN, so that saves a lot of time.

The GRNN was implemented using the Peach package. This package contains methods for making the network, training it as well as finding predictions based on an input and the trained network.

6 Results

This section will present the results from performing the Hilbert-Huang Transform on the dataset, as well as the predictions generated by the artificial neural network. Which parameters was used to get the result will also be described.

6.1 Results from HHT

6.2 Results from ANN

7   Discussion

8   Conclusion

9   Future Work

end point optimization in HHT
    radial basis function
    multilayer perceptron
    more work on which parameters to use as input
    more robust
    how will it do with a lot of noise
    how weather impacts things

10   Project evaluation

References

[1] Theophilos Cacoullos. Estimation of a multivariate density. *Annals of the Institute of Statistical Mathematics*, 1964.

[2] Hassan Farhangi. The path of the smart grid. *IEEE power & energy magazine*, pages 18–28, 2010.

[3] V. G. Kurbatskii, D. N. Sidorov, V. A. Spiryaev, and N. V. Tomin. On the neural network approach for forecasting of nonstationary time series on the basis of the hilbert-huang transform. *Automation and Remote Control*, 72(7):1405–1414, 2011.

[4] V. Kurbatsky, N. Tomin, D. Sidorov, and V. Spiryaev. Electricity prices neural networks forecast using the hilbert-huang transform. *2010 9th Conference on Environment and Electrical Engineering, EEEIC 2010*, pages 381–383, 2010. cited By (since 1996) 1.

[5] V.G. Kurbatsky, D.N. Sidorov, V.A. Spiryaev, and N.V. Tomin. The hybrid model based on hilbert-huang transform and neural networks for forecasting of short-term operation conditions of power system. *PowerTech, 2011 IEEE Trondheim*, pages 1 –7, june 2011.

[6] Mark T. Leung, An-Sing Chen, and Hazem Daouk. Forecasting exchange rates using general regression neural networks. *Computers & Operations Research 27*, 2000.

[7] Zhigang Liu, Weili Bai, and Gang Chen. A new short-term load forecasting model of power system based on hht and ann. *Advances in Neural Networks - ISNN 2010*, pages 448–454, 2010.

[8] U.S Department of Energy. Smartgrid.gov, July 2012. [Online] `http://www.smartgrid.gov/`.

[9] Jie Shi, Wei-Jen Lee, Yongqian Liu, Yongping Yang, and Peng Wang. Short term wind power forecasting using hilbert-huang transform and artificial neural network. *Electric Utility Deregulation and Restructuring and Power Technologies (DRPT), 2011 4th International Conference on*, pages 162 –167, july 2011.

[10] T. Shifu, M. Hong, and S. Liyun. Filter principle of hilbert-huang transform and its application in time series analysis. *International Conference on Signal Processing Proceedings, ICSP*, 4, 2007.

[11] Donald F. Specht. A general regression neural network. *IEEE Transactions on Neural Networks vol. 2*, pages 568–576, 1991.

[12] Wu Wang, Xueyao Li, and Rubo Zhang. Boundary processing of hht using support vector regression machines. *Computational Science – ICCS 2007*, 4489:174–177, 2007.
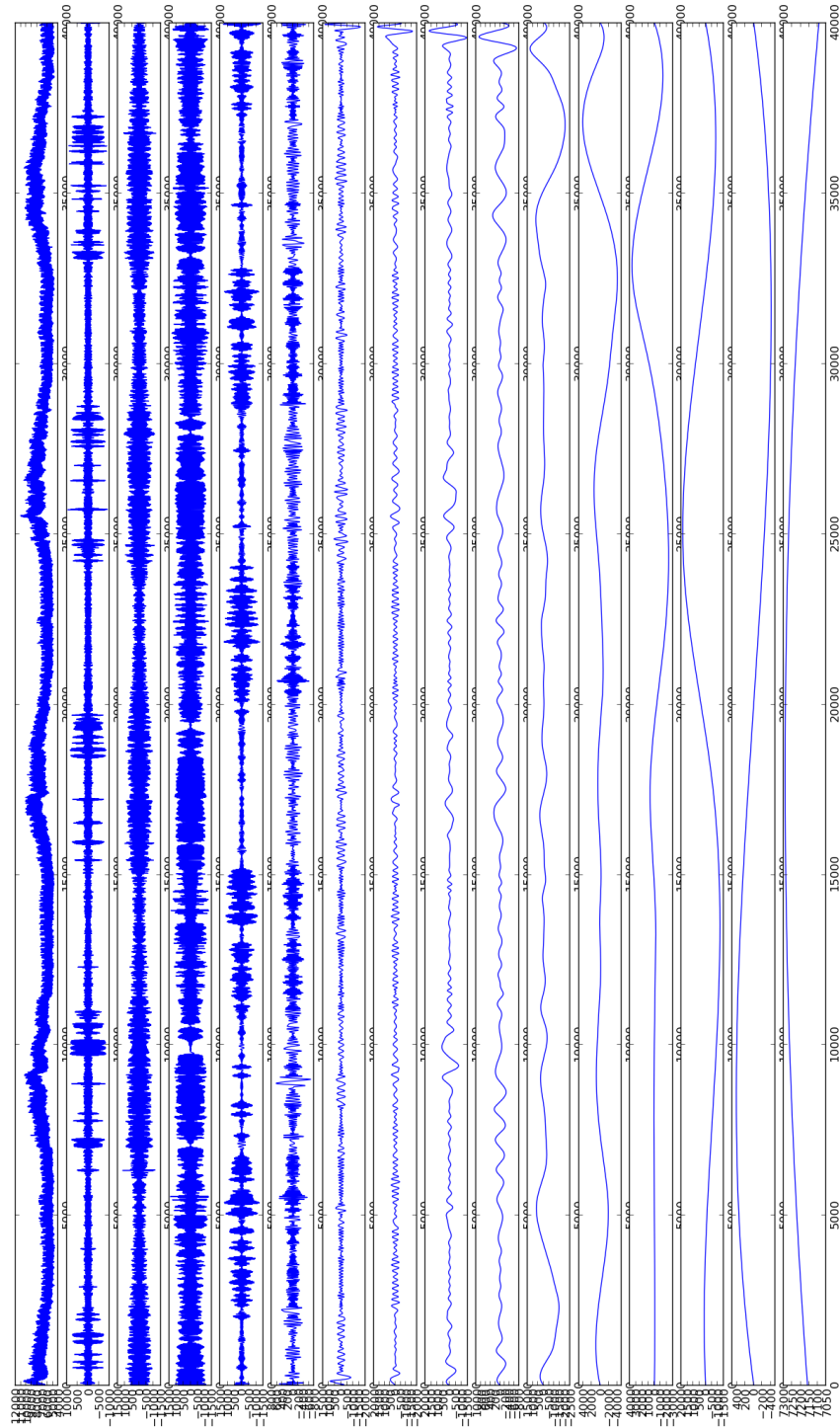
Figure 5: All IMFs generated from 40000 data points.

A   Systematic Literary Review

This is the full version of the systematic literary research. It describes how and where the sources used in this project can be found, as well as the rationale for selecting the sources that are used. The reason for including a systematic literary review is to help the reader understand why certain articles are included, while others are not.

A.1   Sources

Source
IEEE Xplore
SpringerLink
Scopus

Table 1: Databases used for literary search

Table 1 show the databases used for finding the articles cited in this project. The reasons for choosing these databases is that they are very popular and reliable, especially in the fields related to this project.

A.2   Search Terms

	Group 1	Group 2	Group 3	Group 4
Term 1	Hilbert Huang Transform	Forecast	Time series	Smart Grid
Term 2	HHT	Prediction		Power Grid
Term 3		Forecasting		Electrical Grid

Table 2: Phrases used when searching

When searching for literature related to this project, the search phrases in Table 2 was used. They were put together in a search string in an attempt to get only relevant articles. The phrases was combined like this (T being the term, and G being the group):

$$([G1, T1] \vee [G1, T2]) \wedge ([G2, T1] \vee [G2, T2] \vee [G2, T3]) \wedge ([G3, T1]) \wedge ([G4, T1] \vee [G4, T2] \vee [G4, T3])$$

However, this resulted in only one hit in the databases. This is far from enough, so the search was repeated without Group 4. This gave better results, while still being relevant.

A.3   Research Questions

RQ1	Can Hilbert-Huang Transform (HHT) be used for load forecasting?
RQ2	How can the HHT be use for prediction?
RQ3	Does using HHT for prediction offer any improvements over other methods of prediction?

Table 3: Research questions

The research questions in Table 3 are the questions that the report will try to answer. RQ1 is really the reason for doing the project, more than a research question. It is still included in the table, as it is something that hopefully will be answered by the project.

A.4   Inclusion Criteria

IC1	The article's main concern is using HHT for prediction.
IC2	The article describes how and why HHT was used
IC3	The article compares HHT to other methods for prediction
IC4	The article has empirical proof, or cites another article, when making statements

Table 4: Inclusion criteria

The articles that got selected from the search results was selected using the inclusion criteria found in Table 4. IC1 was of course the most important inclusion criteria. All articles selected had to, at least, fulfil that one, as well as one or more of the other criteria.

A.5   Quality Assessment

The articles that got selected using the inclusion criteria described, are rated using the quality criteria from Table 5. Each article got a score (0, 0.5 or 1) for each quality criteria. The scores where summed up, and the ones with the highest scores

QC1	The study is put into context of other studies
QC2	The aim of the research is clearly stated
QC3	Decisions made are clearly justified
QC4	The results and procedures are thoroughly explained and analysed
QC5	The article is structured and uses good language

Table 5: Quality criteria

were considered the best articles, and were given the most weight when doing this project.

A.6   SLR Results

This section describes the results achieved from doing the structured literary review described in Section A. Section A.6.2 briefly describes all the selected articles from Section A.6.1. A summary is found in Section A.6.3.

A.6.1   Literature Found

A1	Short term wind power forecasting using Hilbert-Huang Transform and artificial neural network[9]
A2	The hybrid model based on Hilbert-Huang Transform and neural networks for forecasting of short-term operation conditions of power system[5]
A3	On the neural network approach for forecasting of non-stationary time series on the basis of the Hilbert-Huang transform[3]
A4	Electricity prices neural networks forecast using the Hilbert-Huang transform[4]
A5	Boundary processing of HHT using support vector regression machines[12]
A6	Filter principle of Hilbert-Huang transform and its application in time series analysis[10]
A7	A new short-term load forecasting model of power system based on HHT and ANN [7]

Table 6: The articles selected in the SLR, using the key words from Table 2 and inclusion criteria from Table 4

Table 6 shows the articles that were selected, using the inclusion criteria, from the larger set of articles return from doing the search. Addition information about the articles can be found in the bibliography.

A.6.2   Description of Articles

Article	QC1	QC2	QC3	QC4	QC5	SUM
A1	1	1	1	0.5	0.5	4
A2	1	0.5	1	1	1	4.5
A3	0.5	0	1	1	1	3.5
A4	1	1	0	0	1	3
A5	1	1	0.5	0.5	1	4
A6	0.5	1	1	0.5	1	4
A7	1	1	1	0.5	1	4.5

Table 7: Quality criteria scores for the selected articles

The scores each article got for each quality criteria can be seen in Table 7. Most articles scored quite good, with the exception of A4, which only got a total score of 3. A brief description of each article follows.

A1

This paper describes how HHT and Artificial Neural Networks (ANN) can be used for forecasting the power generated by wind power. The paper proposes an HHT-ANN model for wind power forecasting, and analyses a case study of a wind farm in Texas to find its results. The conclusion is that the proposed HHT-ANN model is suitable for short-term wind power output forecasting.

A2

This paper addresses the conventional approaches to the short-term forecasting of non-stationary processes in complex power systems using ANNs. It asks the question if preprocessing of the data can significantly improve the forecast. The focus of the article is HHT, and using that as a means for preprocessing. The conclusion is that using ANN for preprocessing gives better result that no preprocessing.

A3

This article proposes a two-stage adaptive approach for time series forecasting. The two stages are decomposition of the original time series using Hilbert

Transform, and the second is using the obtains functions, as well as instantaneous amplitudes as input in a neural network. The article concludes that the use of HHT permits increasing the accuracy of forecasting.

A4

The focus of this article is the problem of forecasting of electricity prices. The paper proposes an "intelligent" approach which supposes the use of neural network technology together with HHT. The conclusion is that the use of the "intelligent" approach increases the accuracy of forecasting, and that the error of short term forecasting is decreased. It is not clear what it is compared against.

A5

This article focuses on how to restrain the end effects in HHT. Support Vector Regression Machines (SVRM) that are adopted to extend the data at both ends of a signal, is proposed as a solution. Particle Swarm Optimization (PSO) is proposed used for optimizing parameters. The conclusion is that experiments show that the end effects in HHT can be restrained effectively by this method, and that SVRM is a suitable forecasting method for time series.

A6

This paper puts forward a smoothing filter theory for explaining Empirical Mode Decomposition (EMD) in HHT. HHT is promoted for providing higher resolution and concentration in time-frequency plane and for avoiding false high frequency and energy dispersion. The schemed method is validated through experiments.

A7

This article proposes a new short-term load forecasting model based on HHT and ANN. The article discuss disadvantages of using HHT for forecasting. The article concludes that, through using the new model, a higher accuracy of short term forecasting can be achieved.

A.6.3 Summary

After reading the articles in Table 6, it is pretty clear that HHT can indeed be used for prediction, or forecasting. At least together with some other form for training algorithm. With the exception of article A5 and A6, all the articles dealt with using HHT and ANN for forecasting. A5 mentions using PSO and SVRM, but considering

the overwhelming majority of articles that uses ANN, the latter will be considered the best option when doing the project.

According to several of the articles HHT is used to create the input to the ANNs. That is, both intrinsic mode functions (IMFs) and instantaneous frequencies and amplitudes are generated from HHT and used in an ANN.

Using this approach also gives increased performance over various other approaches, like using no preprocessing at all.

B   Source Code