

Application of Digital Signal Processing to the Foreign Currency Exchange Market

Kaichang Wang

University of Michigan, Ann Arbor, department of EESC

Abstract—This document is an introduction of DSP tools in matlab and their applications in forex market. By using these tools, a method for predicting the market is introduced.

Keywords—DSP; FOREX

I. INTRODUCTION

The worldwide forex market contains sub-markets in different countries. There are about 30 sub-markets in the world. London, New York, Paris, Tokyo, these are the most important markets. As the most violet market in the world of finance, the research becomes significant: 1. For individual investor, potential high profit of forex market is the essential motion. However, high profit also means high risk at the same time. So how to get higher profit and avoid more risks becomes a question that everyone concerns. Indicators (DSP tools) can reflect the evulsion of market. They give reference to investors and help investors to make decisions. 2. For government, forex market is a part of financial system. Nation economy affects it while it can affect the economy as a feedback. A huge fluctuation in forex market may cause a financial crisis or even an economy crisis. So government also needs to understand forex market and makes a proper currency policy. Previous researches in forex market are mostly based on statistics. Statistics methods can show accurate numerical data, but they are not intuitive enough. In this report, DSP tools like moving averaging would be used to analyze forex market.

II. MOVING AVERAGE FILTER

A. introduction of moving average filters

The moving average is the unweight mean of a series data points. It is the most common filer in DSP. Usually people use moving average filter to reduce noise in a signal. In another word, it makes a signal smoother. Here is the equation of moving average,

$$y[i] = \frac{1}{M} \sum_{j=0}^{M-1} x[i+j] \quad (1.1)$$

In matlab the command for moving average is `tsmovavg()`.

B. Plots and average period

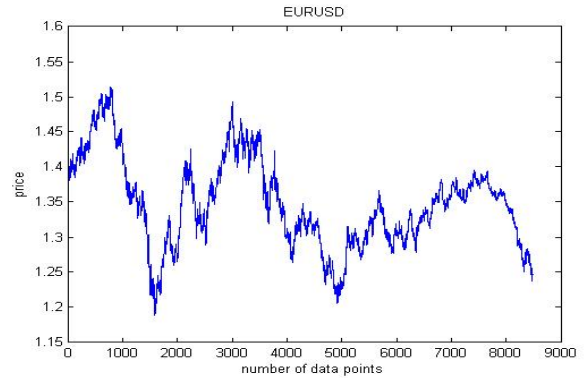


Fig. 1

Fig. 1 shows the original plot of euro versus us dollar. The data is sampled every 4 hours. The date of the first data point is June 11, 2009. And the last is November 9, 2014. Vertical axis shows the price which is the value of euro divided by the value of us dollar. The horizontal axis shows the number of data points.

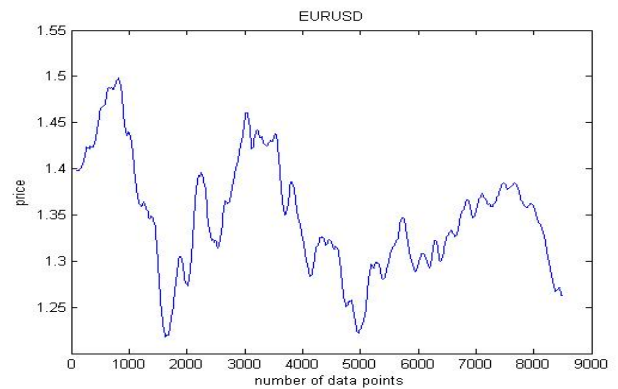


Fig. 2

Fig. 2 shows the moving average with a window size of 100. This is done by matlab function `tsmovavg`. It has four inputs. They are original data vector, mode, window size and dimension. `tsmovavg(m,'s',100,1)` where `m` is data vector, `s` means simple mode, 100 is the window size and 1 is the dimension.

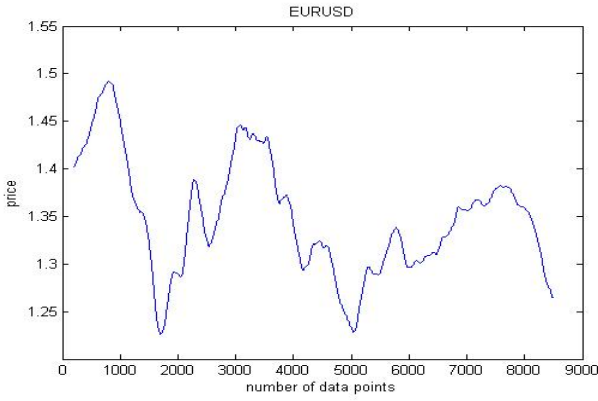


Fig. 3

Fig.3 shows the moving average with a window size of 200. The matlab command is `tsmovavg(m,'s',200,1)`.

The only difference between fig 2 and fig 3 is the window size. The window size of fig2 is 100 and the window size of fig 3 is 200. We can tell that the larger window size is, the smoother the signal is. What if we set the window size big enough and then subtract the original signal?

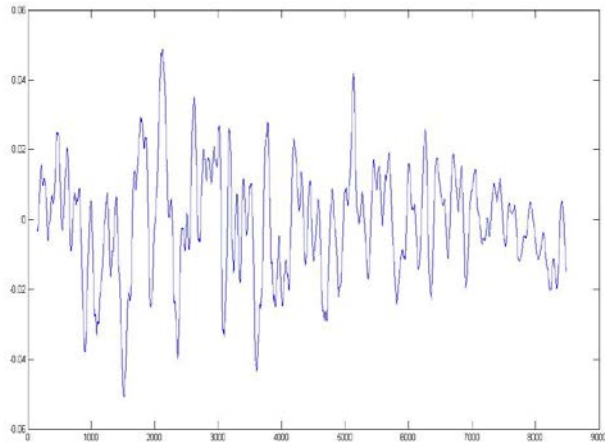


Fig. 4

Fig 4 shows `[m - tsmovavg(m,'s',200,1)]`. Actually it is the high frequency component of the signal. We want to calculate the average period of it. One way to calculate it is using fft in matlab. Since fft is complex, we need to take the absolute value. The index is associate with the normalized frequency which has the maximum energy. The matlab command is,

$$[\text{maxValue}, \text{indexMax}] = \text{max}(\text{abs}(\text{fft}(m - \text{mean}(m)))); \quad (1.2)$$

Then the frequency is,

$$\text{frequency} = \text{indexMax} * F_s / L; \quad (1.3)$$

$$\text{period} = 1 / \text{frequency}; \quad (1.4)$$

where F_s is the sampling frequency and L is the length of the signal. The average period we calculated is about 34 trading days. So about every month, there is a cycle in the market. When a trend is longer than 15 days, we can assume there will be a turning point in the near future. This the mid-term period of the market. If we want to get a long term market period, we can use the same method but different data. This time we load the data sampled weekly.

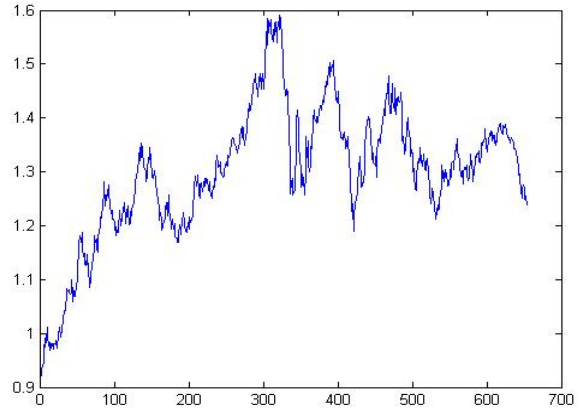


Fig. 5

Figure 5 shows euro vs us dollar weekly.

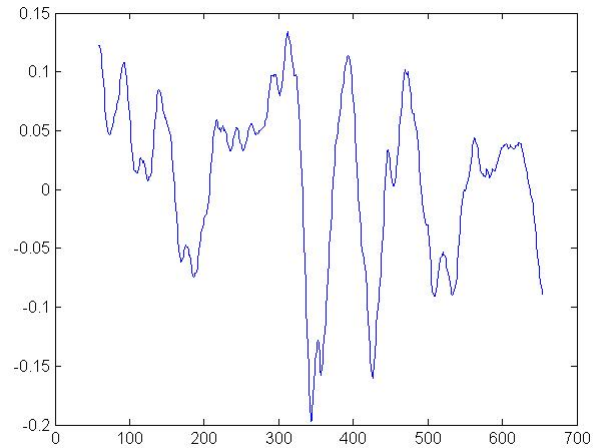


Fig. 6

Figure 6 shows the higher frequency component of the original signal. With same method above, we get the long term period of the market is about 350 days. It is almost one year.

III. HILBERT HUANG TRANSFORM

A. introduction of Hilbert-huang transform

In 1998, Norden E. Huang [1] introduced a new method to analyze digital signal. The method is HHT. It has two main parts: empirical mode decomposition (EMD) and Hilbert transform. Essentially, EMD is used to decompose a signal into several intrinsic mode functions (IMF). Usually each IMF has a unique frequency. Arranging IMFs by their frequency, people can find characters of the original signal. EMD was used to research wave at the first time. Then it was applied in medicine, audio processing and etc.. After EMD, people can do the Hilbert transform to get the Hilbert spectrum. HHT can process non-linear and non-stationary data.

000000

B. Process of EMD

First, each IMF has two properties: 1. The number of maximum and minimum is equal to the number of zero crossing point or differ by one; 2. The mean of upper envelope and lower envelope is equal to zero. Upper envelope and lower envelope are calculated by spline interpolation. Let's define original signal as $X(t)$, the first upper envelope and lower envelope as $U1(t)$ and $L1(t)$. Then the first mean is defined as,

$$M1(t) = [U1(t)+L1(t)]/2 \quad (2.1)$$

After the shifting process, the remaining part $H1(t)$ is defined as,

$$H1(t) = X(t) - M1(t) \quad (2.2)$$

Ideally we expect $H1(t)$ to be an IMF. However, in real situation, a small error on a envelope can generate a new maximum or minimum, so usually $H1(t)$ is not a IMF. Thus we need to repeat the process. Replace $X(t)$ by $H1(t)$ and recalculate upper and lower envelope based on $H1(t)$,

$$M2(t) = [U2(t)+L2(t)]/2$$

$$H2(t) = H1(t) - M2(t)$$

.....

$$Mk(t) = [Uk(t)+Lk(t)]/2 \quad (2.3)$$

$$Hk(t) = H_{k-1}(t) - Mk(t) \quad (2.4)$$

Until $Hk(t)$ satisfies two conditions mentioned above. Now we get the first IMF, $C1(t)$,

$$C1(t) = Hk(t) \quad (2.5)$$

The remaining part of the signal $R1(t)$ is defined as,

$$R1(t) = X(t) - C1(t) \quad (2.6)$$

Then we EMD to decompose $R1(t)$. Again repeat all the steps above we can get,

$$R2(t) = R1(t) - C2(t)$$

.....

$$Rn(t) = R_{n-1}(t) - Cn(t) \quad (2.7)$$

The original signal can be represent as,

$$X(t) = \sum_{i=1}^n C_i + R_n \quad (2.8)$$

C. Hilbert transform

Then we can use Hilbert transform to get Hilbert spectrum of each IMF,

$$Y_1(t) = \frac{1}{\pi} P \int_{-\infty}^{\infty} \frac{C_1(t)}{t-\tau} d\tau \quad (9)$$

D. Matlab code

The matlab code we use to implement HHT is written by Alan Tan [2] in 2008. The package has three matlab files: plot_hht.m, emd.m and findpeaks.m.

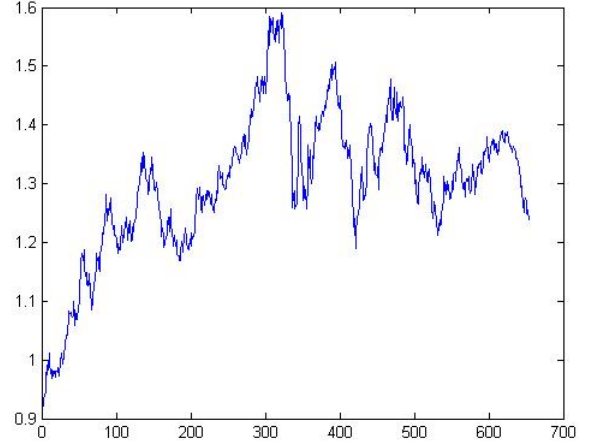


Fig. 7

Figure 7 shows the original signal, euro vs usd weekly price.

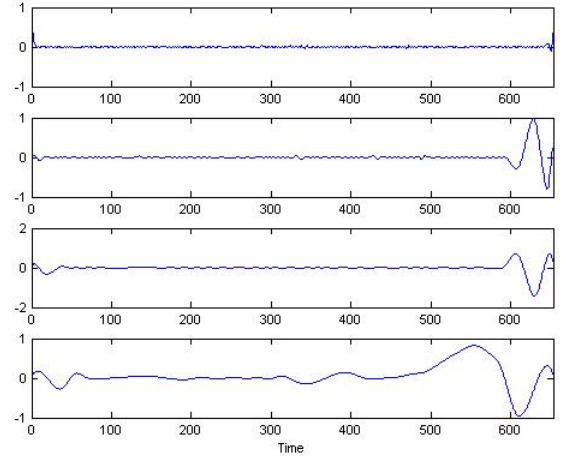


Fig. 8

Figure 8 shows the first, second, third and fourth IMF.

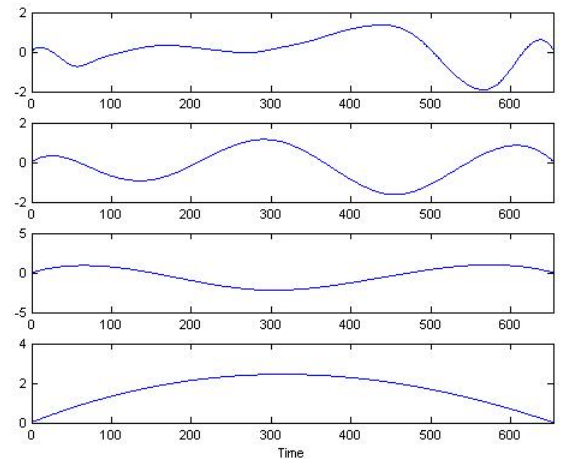


Fig.9

Figure 9 shows the fifth, sixth, seventh, eighth IMF.

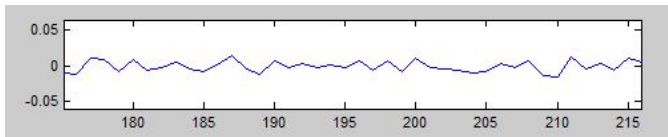


Fig.10

Figure 10 zooms in the first IMF.

As we can see, the first IMF is a high frequency component. There are many maximum and minimum in the plot. The fourth IMF is a median frequency component, without zoom in, we can still see some extremes. The last IMF is a low frequency component, the frequency is less than $1/1200$.

So what can we get from all these IMFs? High frequency IMF can show the oscillation of market perfectly. Every small ripple can be mapped back to the original signal. So people can use high frequency IMF to research the detail of oscillation. Median frequency IMF can be used to research long-term period of the market. Low frequency IMF show the overall trend of the market.

IV. SUMMARY

1 Both moving average filter and Hilbert Huang transform can decompose a signal into higher and lower frequency components. In the forex market, we can use them to calculate different periods in terms of long term, mid-term or short term.

2 FFT and moving average are in class DSP tools.

3 Hilbert Huang Transform is out of class DSP tool.

V. REFERENCE

[1] Norden E. Huang and Samuel S. P. Shen, editors (2005) Hilbert-Huang Transform and Its Applications. World Scientific

[2]<http://www.mathworks.com/matlabcentral/fileexchange/19681-hilbert-huang-transform>