International Journal of Wavelets, Multiresolution and Information Processing

ADAPTIVE FOURIER DECOMPOSITION AND RATIONAL APPROXIMATION–PART II: SOFTWARE SYSTEM DESIGN AND DEVELOPMENT

LIMING ZHANG, WEI HONG, WEIXIONG MAI, TAO QIAN

Faculty of Science and Technology
University of Macau
Av.Padre Tomas Pereira, Taipa, Macau
lmzhang@umac.mo
hongwei7901@126.com
maiweixiong@gmail.com
fsttq@umac.mo

Received (Day Month Year)
Revised (Day Month Year)
Communicated by (xxxxxxxxxx)

This paper proposes a new software system which provides algorithms for three variations of adaptive Fourier decomposition (AFD), including Core AFD, Cyclic AFD and Unwending AFD. The related time frequency distributions are also provided. Remarks are made for the algorithms design and development. The software system can be used to analyze any signal with finite energy.

Keywords: Core AFD; Cyclic AFD; time-frequency distribution; Unwending AFD

AMS Subject Classification: 22E46, 53C35, 57S20

1. Introduction

Adaptive Fourier decomposition, called Core AFD, and its variations, called AFDs, offer decompositions of signals into basic pieces with positive frequencies that is what is desired by signal analysts. Not just because of the physical meaning of the concept, positivity of frequencies allows application-related mathematical analysis of signals. In fact, many quantitative concepts involving frequencies declare themselves to be of proper sense only when frequencies are restricted to be positive. Below we provide some detailed information in relation to applications.

First we would like to mention time-frequency distribution (TFD) of signals. For a given signal one wants to know at a certain time moment how many different frequencies present and how much energy is attributed at each of the existing frequency. This naturally introduces the concept TFD. It presents as the graph of a function of two real variables. Since Fourier frequencies are constant frequencies, they do not provide much information. Many efforts have been devoted to establish various kinds of TFDs for signal analysis purposes. Since none of the existing ones are satisfied, including Shot-Time Fourier Transform and Wagner distribution, etc., there are still on going attempts to build up new ones. The frequency-decomposition methods implied by various AFDs offer theoretically coherent TFDs with high resolution.

It is well known that Fourier analysis has direct applications to signal and image processing. As long as a practical problem can be approached by Fourier expansion, the problem can also also be approached by AFDs. That is because AFDs are generalizations and optimizations of Fourier decomposition. Our experiments show that AFDs offer fast converging positive-frequency decompositions into smooth functions with best analytical properties. AFDs are effective in denoising and compressing signals. In some practical problems, such as in image processing, the tasks are to extract singularities. Applications of AFDs in such aspect is on going. AFDs belong to the research trend of sparse representation, and in particular, by parameterized Szegö kernels.

AFDs result rational functions that offer rational approximations to unknown functions, the latter being required by many types of engineering problems, including system identification. In this direction we successfully formulated our approach that is well accepted by scholars in control theory[4],[5],[6],. In higher dimensions the concept rational approximation has not been well established. Our approach in higher dimension, however, offers an approach of rational approximation, and in the precise language, is approximation by reproducing kernels as generalization of Cauchy kernels. It is predictable that such approximations can find significant applications in practice.

2. The Design Principle of AFD Software System

In one-dimension we introduce three types of AFD algorithms. Besides the basic AFD, or Core AFD, there are two main variations, viz., Cyclic AFD (or n-Best AFD), and Unwending AFD. For each decomposition method, we provide examples in which signal input (can be function or data) and output (data) are provided.

The flowchart of the whole system is illustrated in Fig.1. The software system and program code can be found in the following website: http://www.fst.umac.mo/en/staff/fsttq.html.

2.1. *The design principle of Core AFD algorithm*

Core AFD is also called AFD (adaptive Fourier decomposition). The reason of adding the prefix "Core" is that it is the fundamental construction part of the other AFD related algorithms, including Cyclic AFD[7], Unwending AFD [8], Double Sequence AFD [11], Higher Order Szegö Kernel AFD[13], etc. AFD can also be related to sparse representations of Szegö kernels, in the frame work of compressed sensing and learning theory. The philosophy and method can be generalized to sparse

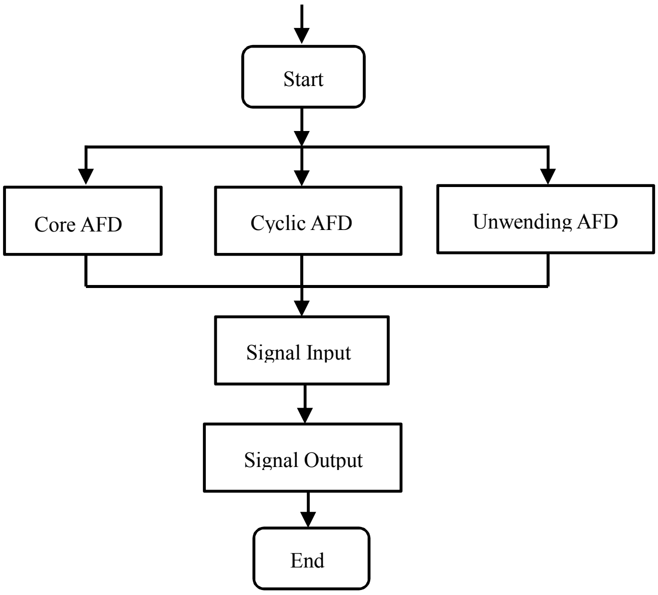

Fig. 1. The flowchart of AFD based stock index movement forecasting model

representation in reproducing kernel Hilbert spaces. It is also naturally related to rational approximation.

A standard Core AFD starts selecting a parameter a_1 inside the unit disc based on the *Maximal Selection Principle* (maximally choosing a_1). Then, similarly, select $a_2, a_3, ...$, also in the unit disc, applied to the recursively defined *reduced remainder* f_k, until the objective function reaches a prescribed threshold. In such case each component, B_k, a member of the obtained TM system, is a *pre-mono-component*. In particular, in our case, for each k, $zB_k(z)$ is a mono-component. A Core AFD can start from the fixed selection $a_1 = 0$ (prescribed a_1). The rest $a_2, a_3, ...$, are selected based on the Maximal Selection principle. In such way, all $B_k, k \geq 1$, are automatically mono-components. The algorithm published in this paper is for such non-standard Core AFD (with the first parameter $a_1 = 0$)[15], [16]. The *nth* objective function to be examined against the given threshold $\epsilon > 0$ for prescribed or maximally chosen $a_1, ...a_n$ in the unit disc is

$$A(f, n) = \|f\|^2 - \sum_{k=1}^{n}(1 - |a_k|^2)|f_k(a_k)|^2, \qquad (2.1)$$

where

$$(1 - |a_k|^2)|f_k(a_k)|^2 = |\langle f_k, e_{a_k}\rangle|^2 = |\langle f, B_k\rangle|^2,$$

and f_k is the *kth* reduced remainder.

The following relations can greatly simplify the algorithm: If F is a given real-valued signal and f is its associated analytic signal, i.e., $f = (1/2)(F + iHF)$, where

H is the Hilbert transformation in the context, then with $f_1 = f$,

$$\langle f_k, e_{a_k}\rangle = \langle F_k, e_{a_k}\rangle,$$

where

$$F_k(e^{it}) = \frac{F_{k-1}(e^{it}) - \langle F_{k-1}, e_{a_{k-1}}\rangle e_{a_{k-1}}(e^{it})}{\frac{e^{it}-a_{k-1}}{1-\overline{a}_{k-1}e^{it}}}.$$

Based on this one does not have to work out the associated analytic signal f through computing Hilbert transformation but only use the original real-valued signal F. In such way one still gets the series expansion $\sum$ for f. After that one just takes $F = 2\mathrm{Re}f - c_0$ as the series expansion for F^3.

For Core AFD there are following remarks to be noted.

1. At each step the decomposition grasps as large as possible energy portion from the given signal. This, however, does not mean that after the nth step the accumulated result is equivalent to the energy portion for all the n parameters being maximally selected at one time: The consecutive selection of chances coming one by one is surely not as good as comparatively if all the chances come together and to be selected at one time. Nevertheless, the convergence is fast in general, and in many examples, after 7 to 8 sifting steps we can grasp around more than 95 percent of the total energy.

2. Performing AFD the result may not be unique. At each step of selecting a parameter a_k by using the Maximal Selection Principle the solutions for $a_k = \arg\max|\langle f_k, a_k\rangle|$ may be multiple. However, if, for instance, the algorithm is arranged to select the a_k that comes up first, then the standard Core AFD or non-standard Core AFD will give stable results, or the uniqueness of the algorithm. We regard this as stability of AFD.

3. A Core AFD algorithm is essentially an approximation algorithm to functions in the Hardy space that are basically complex-valued. In our approximation problems, however, the data are real-valued, and what we see are approximations to real-valued functions. What happens is that the algorithm itself first conveys a real-valued signal to its associated analytic signal, one in the Hardy space, and finally, using the relation $F = 2\mathrm{Re}f - c_0$ to recover the original signal. This is the same as in the following Cyclic AFD and Unwending AFD.

4. In the algorithm, there are two default parameters. One is the initial value of $a_0 = 0$, the other is the parameter used to discretize the complex unit disc. The latter is chosen to be 0.02 in the algorithm. The parameter chosen through maximal principle is comparatively time consuming. It, however, is balanced by promising resolutions. It is suitable for decomposing arbitrary signals. For some specific practices, the above mentioned two figures can be changed in the program code by the user.

2.2. *The design principle of Cyclic AFD algorithm*

Below we give an explanation for Cyclic AFD. It is an algorithm to solve the n-best rational approximation problem. Cyclic AFD, like the existing RARL2 algorithm by the French institute INRIA, offers a conditional solution[1]. That is, when the objective function has only one critical point, then the algorithm gives rise to the solution. The related publications include 18, 12, 7.

For any fixed $n \geq 1$, the objective function to be minimized for simultaneously chosen $a_1, ...a_n$ in the unit disc is

$$A(f, n) = \|f\|^2 - \sum_{k=1}^{n} (1 - |a_k|^2)|f_k(a_k)|^2. \tag{2.2}$$

We have the following remarks.

1. A Core AFD algorithm is to expand a signal into an infinite series, although practically it stops somewhere according to the given threshold. Cyclic AFD, however, is an expansion to the nth partial sum while n is previously given. The latter requires "the best" for the fixed n among all possible selections of the n parameters at one time.

2. The Cyclic AFD starts from an initial n-tuple. This n-tuple can be formulated in various ways. For instance, one can start from any n-tuple such as $(0, ..., 0)$. Or, one can perform the Core AFD algorithm to obtain an initial n-tuple. Or, one can depend on a computer program choosing an n tuple randomly.

3. Practically, if several initial n-tuples give rise to the same limit n-tuple, then this may be the evidence that the objective function has a single critical point. In the case one gets the solution. If several initial n-tuples correspond to several different solutions n-tuples, then one should choose the best one, or one of those giving the best approximation. This latter case shows that the critical point is not unique: There should be several local minima for the objective function.

4. Cyclic AFD repeats the step from $n-1$ to n in the Core AFD algorithm. This means that for any fixed $n-1$ parameters, one formulates the corresponding $(n-1)$-TM system, and, based on that, chooses a new nth parameter. Every time one fixes a different $(n-1)$ tuple.

5. In Cyclic AFD the order of the finally obtained n parameters $a_1, ..., a_n$, does not really matter. What is obtained is a hyper-plane and the solution to the n-best problem is just the projection of the given signal to the hyper-plane. For the n-best approximation related time-frequency distribution the order does matter, because under any order we can formulate a corresponding n-TM system giving rise to frequency-decomposition of the signal. The order issue associated with Cyclic AFD is to be further investigated.

6. In the algorithm, there is one default parameter. It is the parameter used to discretize the complex unit disc. Same as in Core AFD algorithm, it is chosen as 0.02 in the algorithm. It is noted that this default parameter has more influence in Cyclic (or n-Best) AFD than that in Core AFD. Smaller value would give rise to

better resolution but with higher computation cost. The algorithm allows the user to choose the decomposition number n. In the provided algorithm we made $n = 33$ for general purposes. The initial n-tuple $(a_1^0, ..., a_n^0)$ is also selected by default.

2.3. *The design principle of Unwending AFD algorithm*

Unwending AFD is most suitable for signals of high frequency. It also converges much faster than other AFD algorithms. The shortcoming of Unwending AFD is that its computation depends on that of Hilbert transform. Hilbert transform is a singular integral whose computation is to be made more efficient. What does it mean by high frequency? The primary idea is that if a signal in the Hardy space is of the form $s(z) = z^N g(z)$, while N is large and $g(z)$ remains in the Hardy space, then s is said to be of high frequency. Based on this idea we define what is meant by high frequency as follows. Let $s(z)$ be a function in the hardy H^2 space and $s(z) = I(z)O(z)$ is its Nevanlinna decomposition into the product of its inner factor part and its outer function part[9]. Since $I(z)$ is an inner function, we have $I(e^{it}) = e^{i\theta(t)}$. By a result of Qian[9], if θ is non-trivial, that is, if the inner function part is not a constant, then $\theta'(t) > 0$. We define the the *carriage instantaneous frequency* of s as $\theta'(t)$, and the *average carriage frequency* of s as

$$\frac{1}{2\pi}\int_0^{2\pi} \theta'(t)dt.$$

By these definitions, if the average carriage frequency is large, then we say that the *signal is of high frequency.* Due to the fact that the same quantity for any outer function is zero[9], the concept of average carriage frequency depends only on the inner function part. Unwending AFD is one that at each step performs the above factorization process to the reduced remainder and does the parameter selection based on the Maximal Selection Principle only to the outer function part[2], [8]. Comparisons on Core AFD, Cyclic AFD and Unwending AFD may be found in 10.

Unwending seems to be theoretically the most effective decomposition process that gives rise to instantaneous frequency decomposition of a signal. For the concern of algorithm, Unwending AFD may need large computer space to record inner functions, and the computation of the factorization, in particular, of the outer functions, depends on the computation of Hilbert transforms. The latter is a hard problem in computational mathematics. Nevertheless, Unwending AFD algorithm is performable with concrete examples.

2.4. *The design principle of signal inputs*

The system provides two categories for input signals. One is Signal Generation and the other is Input Data. Under Signal Generation, users can generate commonly used signals by signal classifications, including Sine Wave, Square Wave, Triangular

Wave, and Samtooth Wave. The characteristic parameters include sampling frequency, sampling number, frequency, amplitude, and phase. In this category users can also do signal decomposition for signals with explicit formulas. Under Input Data category, users can input the signal data through a .txt file or .mat file generated by Matlab. Besides, users can also choose to add the white noise to the generated signal.

2.5. *The design principle of signal outputs*

After input the signal, users can choose one of the three decomposition models to process the data, including Core AFD, Cyclic AFD and Unwending AFD. Under Core AFD, there are two parameters to be chosen. One is the decomposition number, the other is the energy difference between the original signal and reconstructed signal (the threshold). Default values are provided by the system. For both cyclic AFD and Unwending AFD, the decomposition number is the sole parameter. The signal outputs include the Reconstructed signal, Time Frequency Distribution (TFD), and Mean Instantaneous Frequency (IF).

3. Experiment Results

A sample speech word "water" is used in the experiment. It is chosen from TIMIT Acoustic-Phonetic Continuous Speech Corpus [17]. The TIMIT corpus of read speech is designed to provide speech data for acoustic-phonetic studies and for the development and evaluation of automatic speech recognition systems. It includes a 16-bit, 16kHz speech waveform file for each utterance. The word "water" is used to illustrate the experiment results. The "water" signal has total 4096 sampling points. The third 1024 sampling points are chosen in the experiment. The original signal is shown in Fig.2.

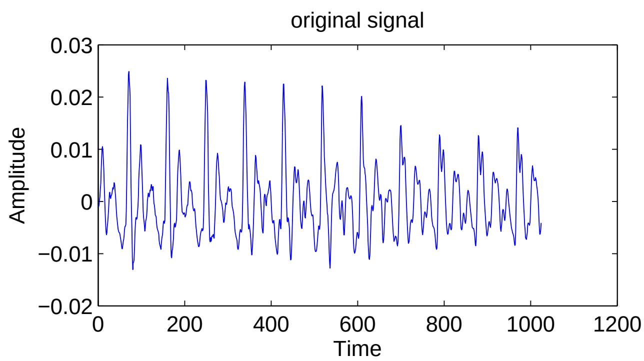

Fig. 2. The original signal of a piece of sound file

3.1. *Experiment Results of Core AFD*

Core AFD is used to decompose the signal. The comparison between the original signal and the reconstructed signal is illustrated in Fig.3. The time frequency distribution is shown in Fig.4.

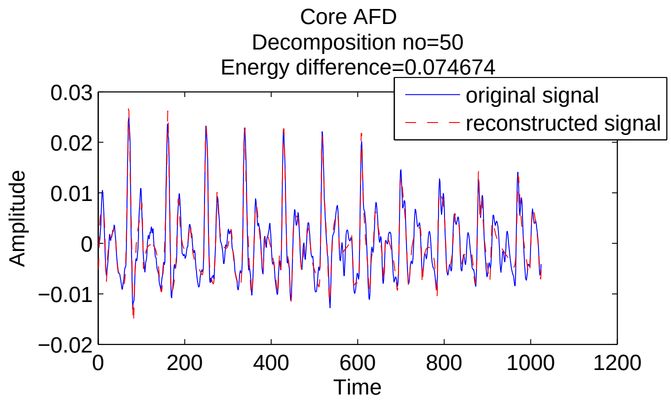

Fig. 3. Comparison between the original and reconstructed signals based on Core AFD.

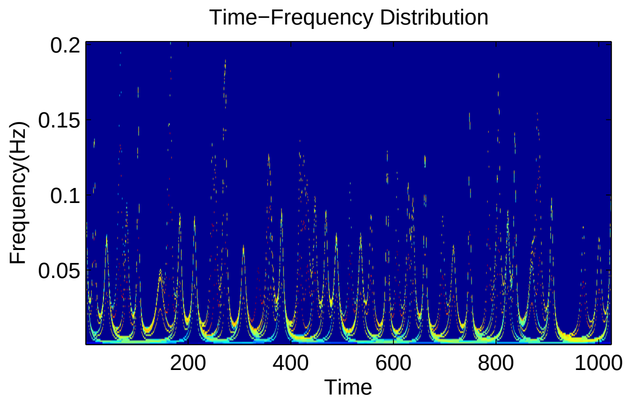

Fig. 4. Core AFD based TFD.

3.2. *Experiment Results of Cyclic AFD*

Cyclic AFD is used to decompose the signal. The comparison between the original signal and the reconstructed signal is illustrated in Fig.5. The time frequency distribution is shown in Fig.6.

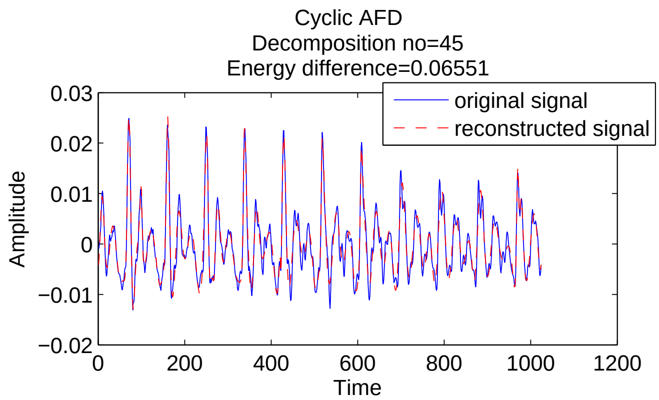

Fig. 5. Comparison between the original and reconstructed signals based on Cyclic AFD.

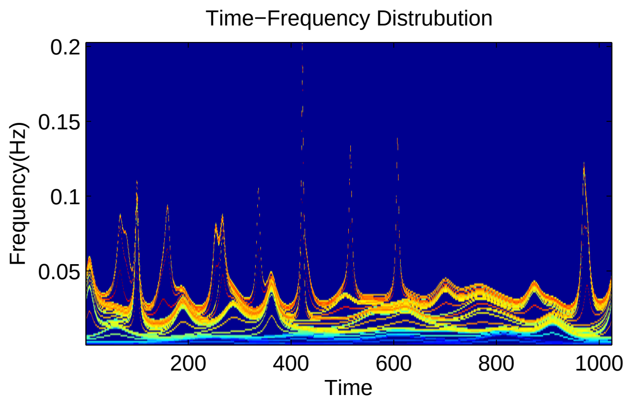

Fig. 6. Cyclic AFD based TFD.

3.3. *Experiment Results of Unwending AFD*

Cyclic AFD is used to decompose the signal. The comparison between the original signal and the reconstructed signal is illustrated in Fig.7.

4. Conclusion

After providing some general information on the nature and possible applications of three types of AFDs, namely, Core AFD, Cyclic AFD and Unwending AFD, their algorithm systems design and development are described in detail. Examples are provided to assist the users implementation of the algorithm.

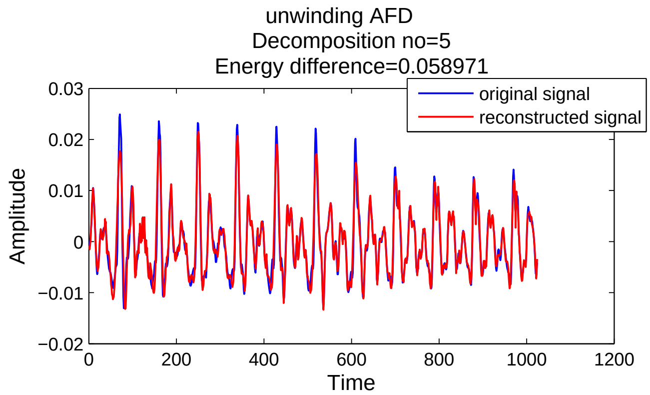

Fig. 7. Comparison between the original and reconstructed signals based on Unwending AFD.

Acknowledgments

We wish to acknowledge the research grant supports with the numbers MYRG 116(Y1-L3)-FST13-QT; MYRG104(Y1-L3)-FST13-ZLM.

References

1. L. Baratchart, M. Cardelli, M. Olivi, *Identification and rational L^2 approximation a gradient algorithm*, Automatica, 27(1991), 413-418.
2. P. Dang and T. Qian, *Analytic Phase Derivatives, All-Pass Filters and Signals of Minimum Phase,* IEEE Transactions on Signal Processing, Issue Date: Oct. 2011, Volume: 59 Issue:10, 4708-4718.
3. Z.X. Li, *Tips of Computation of AFD,* Master thesis at University of Macau, 2010.
4. W. Mi and T. Qian, *Frequency Domain Identification: An Algorithm Based On Adaptive Rational Orthogonal System,* Automatica, 48(6), 1154-1162.
5. W. Mi and T. Qian, *On backward shift algorithm for estimating poles of systems,* accepted to appear in Automatica.
6. W. Mi, T. Qian, F. Wan, A Fast Adaptive Model Reduction Method Based on Takenaka-Malmquist Systems, by W. Mi, T. Qian and F. Wan, Systems and Control Letters, Volume 61, Issue 1, January 2012, 223-230.
7. T. Qian, *Cyclic AFD Algorithm for Best Approximation by Rational Functions of Given Order,* accepted by Mathematical Methods in the Applied Sciences.
8. T. Qian, *Intrinsic mono-component decomposition of functions: An advance of Fourier theory,* Mathematical Methods for the Applied Sciences, 2010, 33, 880-891, DOI: 10.1002/mma.1214.
9. T. Qian, *Boundary derivatives of the phases of inner and outer functions and applications,* Mathematical Methods for the Applied Sciences, 2009; **32**:253-263.
10. T. Qian, H. Li, M. Stessin, *Comparison of Adaptive Mono-component Decompositions*, Nonlinear Analysis, Volume 14, Issue 2, April 2013, 10551074.
11. T. Qian, L.H. Tan and Y.B., *Wang,Adaptive Decomposition by Weighted Inner Functions: A Generalization of Fourier Serie,* J. Fourier Anal. Appl. 17 (2011), no. 2, 175V190.
12. T. Qian and E. Wegert, *Optimal Approximation by Blaschke Forms,* Complex Vari-

ables and Elliptic Equations, Volume 58, Issue 1, 2013, page 123-133.

13. T. Qian and J.X. Wang, *Adaptive Decomposition of Functions by Higher Order Szegö Kernels I. A Method for Mono-component Decomposition,* preprint.
14. T. Qian, Y.B. Wang, *Adaptive Fourier Series-A Variation of Greedy Algorithm*, Advances in Computational Mathematics, 34(2011), no.3, 279-293.
15. T. Qian, Y.B.Wang, *Adaptive Fourier Series-A Variation of Greedy Algorithm*, Advances in Computational Mathematics, 34(2011), no.3, 279-293.
16. T. Qian, L.M. Zhang and Zh.X. Li, *Algorithm of Adaptive Fourier Decomposition,* IEEE Transaction on Signal Processing, Dec., 2011, Volume 59, Issue 12, 5899-5902.
17. TIMIT Acoustic-Phonetic Continuous Speech Corpus, http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC93S1 (Retrieved on 27/11/2013)
18. J.L. Walsh, *Interpolation and Approximation by Rational Functions in the Complex Domain,* American Mathematical Society, 1935.