This is a repository copy of *Finite impulse response filter design using a forward orthogonal least squares algorithm*.

White Rose Research Online URL for this paper:
https://eprints.whiterose.ac.uk/74505/

Monograph:
Wu, X., Lang, Z.Q. and Billings, S.A. (2005) Finite impulse response filter design using a forward orthogonal least squares algorithm. Research Report. ACSE Research Report no.904 . Automatic Control and Systems Engineering, University of Sheffield

Finite Impulse Response Filter Design Using A Forward Orthogonal Least Squares Algorithm

Xiaofeng Wu, Z Q Lang and S.A.Billings
Department of Automatic Control and Systems Engineering
The University of Sheffield
Sheffield, S1 3JD, UK
Email: x.wu@sheffield.ac.uk

Abstract— **This paper is concerned with the application of forward Orthogonal Least Squares (OLS) algorithm to the design of Finite Impulse Response (FIR) filters. The focus of this study is a new FIR filter design procedure and to compare this with traditional methods known as the fir2() routine, provided by MATLAB.**

Keywords–Orthogonal Least Squares; Finite Impulse Response Filter; Parameter Estimation; Structure Selection

I. INTRODUCTION

Linear Finite Impulse Response (FIR) filter coefficients design has been extensively studied. Classical methods involve designing a FIR filter based on Fourier Series theory and Inverse-DFT transform. This is also the primary principle of the FIR design in MATLAB. However, researchers still continue to study improved methods for FIR filter design. Lertniphonphun developed an algorithm which designs a FIR filter using a weighted Chebyshev norm based optimization approach [1]. FIR designs subject to upper and lower bounds on the frequency response magnitude were studied by Wu [2]. Yong [3] investigated the design of FIR filters using a cluster of workstations as computing platform. Given the benefits of FIR filter applications to the digital signal processing field, it is reasonable to believe the feasibility of further innovation in this specific area.

There are many well developed FIR filter design methods. The fir2() routine is one method embedded in the MATLAB signal processing toolbox, which can be used to design frequency sampling-based FIR filters with arbitrarily shaped frequency responses. In the present study, fir2() is used as a representative of traditional FIR filter design and is compared with a new design method based on the forward orthogonal least squares algorithm.

The Orthogonal Least Squares (OLS) algorithm was derived as an effective solution for structure selection and parameter estimation in nonlinear system identification [4]. In this paper, a new OLS based procedure is introduced to provide an alternative method for FIR filter design. The new method employs the OLS algorithm to determine the terms and parameters of a FIR filter in order to meet a specified frequency response requirement. Compared with the traditional fir2() procedure, the new method not only provides a different way of designing FIR filters but also offers several advantages. More importantly, the OLS based design can easily be extended to deal with nonlinear filter designs which are issues currently under study and will be discussed in a future publication.

The paper is organized as follows. First, the fir2() routine and associated theory are introduced. Then, the new OLS based FIR design is proposed, and a comparison of the new method with the fir2() routine is conducted to show the advantages and potential of the new approach. Finally, results of case studies are presented and discussed in detail.

II. THE FIR2() ROUTINE FOR FIR FILTER DESIGN

Let $H(e^{j\omega})$ denote the frequency response of the digital transfer function $H(z)$ to be designed to approximate the desired (ideal) response $H_d(e^{j\omega})$. The basic idea behind the fir2() routine in MATLAB is to determine the transfer function coefficients so that the difference between $H(e^{j\omega})$ and $H_d(e^{j\omega})$ for all values of ω in the range $0 \leq \omega \leq \pi$ is minimized. The MATLAB routine fir2() satisfies the least-square criterion [6].

Because $H_d(e^{j\omega})$ is a periodic function, it can be expressed as a Fourier series:

$$H_d(e^{j\omega}) = \sum_{n=-\infty}^{\infty} h_d[n]e^{-jn\omega} \quad (1)$$

with Fourier coefficients given by

$$h_d[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_d(e^{j\omega})e^{jn\omega}d\omega \quad -\infty < n < \infty \quad (2)$$

The fir2() routine is a frequency sampling approach. The desired frequency response $H_d(e^{j\omega})$ is first uniformly sampled at N equally spaced points $\omega_k = 2\pi k/N, k = 0, 1, ..., N-1$, providing N frequency samples. These samples compose of an N-point DFT $H[k]$ whose N-point inverse-DFT yields the impulse response coefficients $h[n]$ of the FIR filter of length N. From equation (1),

$$H[k] = H_d(e^{j\omega_k}) = H_d(e^{j(2\pi k/N)}) = \sum_{l=-\infty}^{\infty} h_d[l]W_N^{kl} \quad (3)$$

where $W_N = e^{-j(2\pi/N)}$. An inverse-DFT of $H[k]$ yields

$$h[n] = \frac{1}{N} \sum_{k=0}^{N-1} H[k]W_N^{-kn} \quad (4)$$

The coefficients $h[n]$ often produce an oscillatory magnitude response which is called Gibbs phenomenon. In order to reduce the ripples, a window function is finally used to yield the filter coefficients.

The MATLAB syntax formulation

$$b = fir2(N, f, m) \tag{5}$$

designs a N-th order FIR filter and returns the filter coefficients in vector b of length $N + 1$. Vectors f and m specify the frequency and corresponding magnitude sample points respectively. In practice, f is the normalized frequency point vector ranging from 0 to 1, where 1 represents the Nyquist frequency (corresponding to half the sample rate). m is a vector containing the desired magnitude response at the points specified in f. By default, fir2() uses a Hamming window. Usually, b is real, symmetric. Without loss of generality, It is assumed that even symmetric coefficients obey $b[k] = b[N + 2 - k], \quad k = 1, 2, ..., N + 1$ [7].

III. THE ORTHOGONAL LEAST SQUARES ALGORITHM

Model identification can generally be formulated as a standard least squares problem. Compared with simple least squares algorithms, the orthogonal least squares method has been demonstrated to be a powerful means to achieve this objective. Basically, the orthogonal algorithm was developed as an approach to combine parameter estimation and model structure detection. The principal idea of the algorithm is to decouple the candidate terms by introducing an orthogonal transform so that selected terms will not be affected when a new term is selected. For most system representations, the orthogonal decomposition approach of the regressor matrix avoids possible ill-conditioning and presents more accurate results. The forward OLS algorithm is based on the classical Gram-Schmidt method [4] [5].

A. Parameter Estimation

Consider a linear regression model

$$y(k) = \sum_{i=1}^{n} \phi_i(k)\theta_i + e(k) \qquad k = 1, ..., N \tag{6}$$

where $y(k)$ represents the k-th measurement, N is the data length, n is the number of column vectors, $\phi_i(k)$ and θ_i denote the regressors and parameters respectively, and $e(k)$ is the modelling error, assumed to be a zero mean white noise sequence. Using the orthogonal algorithm, the parameters θ_i are estimated by transforming model (6) into an equivalent auxiliary model

$$y(k) = \sum_{i=1}^{n} g_i w_i(k) + e(k) \tag{7}$$

where $w_i(k)$ are constructed to be orthogonal and g_i are constant coefficients.

First, orthogonal vectors can be constructed over the given data record as

$$w_1(k) = \phi_1(k) \tag{8}$$

$$w_j(k) = \phi_j(k) - \sum_{i=1}^{j-1} \alpha_{ij} w_i(k) \tag{9}$$

where

$$\alpha_{ij} = \frac{\sum_{k=1}^{N} w_i(k)\phi_j(k)}{\sum_{k=1}^{N} w_i^2(k)} \qquad \left\{ \begin{array}{l} j = 1, ..., n \\ i = 1, ..., j - 1, j \end{array} \right. \tag{10}$$

and from the orthogonal property, there is

$$\overline{w_i(k)w_j(k)} = 0 \qquad i \neq j \tag{11}$$

where the overline denotes the time average over the data length.

The second step consists of estimating the coefficients g_i, which is given by

$$\hat{g}_i = \frac{\sum_{k=1}^{N} w_i(k)y(k)}{\sum_{k=1}^{N} w_i^2(k)} \tag{12}$$

Finally, the unknown system parameters can be calculated from $\hat{g}_i$ according to

$$\hat{\theta}_n = \hat{g}_n \tag{13}$$

$$\hat{\theta}_i = \hat{g}_i - \sum_{j=i+1}^{n} \alpha_{ij}\hat{\theta}_j \qquad i = n - 1, n - 2, ..., 1 \tag{14}$$

The auxiliary regressor $w_i(k)$ are orthogonal so that additional terms can be added to the model without the need to recompute all the previous $\hat{g}_j$, $j < i$. The orthogonal least squares parameter estimation algorithm is therefore very simple and easy to implement.

B. Structure Selection

The determination of a parsimonious representation of a system is one of the most important tasks in system identification. A great advantage of the orthogonal estimator is the possibility of selecting the relevant terms using the Error Reduction Ratio (ERR). The basic principle is given below.

Multiplying equation (7) by itself gives

$$\overline{y^2(k)} = \sum_{i=1}^{n} g_i^2 \overline{w_i^2(k)} + \overline{e^2(k)} \tag{15}$$

The above equation shows that the contribution of each term $g_i^2 \overline{w_i^2(k)}$ to the output energy $\overline{y^2(k)}$. Expressing this quantity as a fraction of i-th term as

$$ERR_i = \frac{\hat{g}_i^2 \overline{w_i^2(k)}}{\overline{y^2(k)}} = \frac{\hat{g}_i^2 \sum_{k=1}^{N} w_i^2(k)}{\sum_{k=1}^{N} y^2(k)} \times 100\% \qquad 1 \leq i \leq n \tag{16}$$

The forward regression procedure is implemented as:

1) Consider all regressors $\phi_i(k) \quad i = 1, 2, ..., n$ as possible candidates for $\omega_1(k)$, calculating through equation (8), (12) and (16), find the maximum of ERR_1 for example $ERR_1^{(j)} = max\{ERR_1^{(i)}, 1 \leq i \leq n\}$. Then the first term selected should be the jth term. i.e. $\omega_1(k) = \phi_j(k)$.

2) Consider all the $\phi_i(k)$ $i = 1, 2, ..., n, i \neq j$ as possible candidates for $\omega_2(k)$ calculating through equation (9), (12) and (16), find the maximum of ERR_2 for example $ERR_2^{(l)} = max\{ERR_2^{(i)}, 1 \leq i \leq n, i \neq j\}$. Then the second term selected should be the lth term. i.e. $\omega_2(k) = \phi_l(k) - \alpha_{12}\omega_l(k)$.

3) Continue the process and in each loop, the term with the maximum error reduction ratio is then selected to produce the term $\omega_i(k)$.

The ERR value could be computed together with the parameter estimate to indicate the significance of each candidate term. Insignificant terms will be discarded from the model by defining threshold value of ERR summation. The forward Orthogonal Least Squares algorithm is terminated when the summation of ERR values is close to 100%.

IV. OLS-BASED FIR FILTER DESIGN

In the following, the FIR filter design is formulated as a difference equation with $2N + 1$ coefficients [8],

$$y(n) = \sum_{i=-N}^{N} b_i x(n - i) \tag{17}$$

Applying the z transform to both sides of equation (17) yields

$$H(z) = \frac{Y(z)}{X(z)} = \sum_{i=-N}^{N} b_i z^{-i} \tag{18}$$

Replacing z in equation (18) with $e^{j\omega}$ yields the filter frequency response

$$H(e^{j\omega}) = \sum_{n=-N}^{N} b_n e^{-jn\omega} \tag{19}$$

Since the filter coefficients can only be real valued, the frequency response must be conjugate symmetric. This premise is the same as for the fir2() routine in MATLAB. From equation (19), a noncausal filter is designed. Then by shifting the filter coefficients, the filter can be made to be causal and practical.

A causal FIR filter with impulse response $b_c[n]$ can be derived from $b[n]$ by shifting the sequence by N samples. i.e.

$$b_c[n] = b[n - N] \tag{20}$$

Accordingly the corresponding transfer function is

$$H(z) = \sum_{i=0}^{2N} b_c[i] z^{-i} \tag{21}$$

The key step in using the OLS algorithm to design the FIR filter coefficients θ_i involves the choice of arguments, ϕ_i and $y(k)$ in equation (6). Comparing equation (6) and (19), it can be concluded that the regressor ϕ_i corresponds to each frequency component $e^{-jn\omega}$, $y(k)$ corresponds to $H(e^{j\omega})$, and the index k in equation (6) should be replaced by frequency sampling points ω in equation (19). Using all these relationships, the filter coefficients $\{b_n\}$ can be easily determined using the forward OLS algorithm.

The matrix expression of system parameter estimation on equation (6) is

$$Y = \Phi\Theta + \Xi \tag{22}$$

Consequently, the related FIR filter design is of the form

$$H = U\beta + \Xi \tag{23}$$

where H, a $(2m + 1) \times 1$ vector, represents the frequency response at the equally divided frequency sampling points as

$$H = \begin{pmatrix} H(e^{j\omega_{-m}}) \\ \vdots \\ H(e^{j\omega_m}) \end{pmatrix} \tag{24}$$

and U, a $(2m + 1) \times (2N + 1)$ complex matrix, is given by

$$U = \begin{pmatrix} e^{-j(-N)\omega_{-m}} & \cdots & e^{-j(N)\omega_{-m}} \\ \vdots & \ddots & \vdots \\ e^{-j(-N)\omega_m} & \cdots & e^{-j(N)\omega_m} \end{pmatrix} \tag{25}$$

In order to apply the OLS algorithm to perform FIR filter design, equation (23) is partitioned into real and imaginary parts to yield

$$[\frac{H_R}{H_I}] = [\frac{U_R}{U_I}]\beta + [\frac{\Xi_R}{\Xi_I}] \tag{26}$$

where the subscript R denotes the real part and I the imaginary part [9].

The OLS algorithm is employed to compute the parameters in the column vector β, a $(2N + 1) \times 1$ coefficients vector,

$$\beta = \begin{pmatrix} b_{-N} \\ \vdots \\ b_N \end{pmatrix} \tag{27}$$

which are the FIR filter coefficients.

V. CASE STUDY AND DISCUSSIONS

In order to verify the OLS approach to FIR filter design and compare the OLS-based approach with the fir2() routine in MATLAB, a lowpass FIR filter design is considered in this section.

A. Lowpass FIR Filter Designs

Consider an order-N (N=10) lowpass FIR filter that has frequency response defined as normalized frequency $f = \{0, 0.6, 0.6, 1\}$ and magnitude response $m = \{1, 1, 0, 0\}$, The duplicated frequency points at $f = 0.6$ indicates a step in frequency response. The desired frequency response is depicted in Figure 1.

The ideal frequency response can be analytically described as

$$H_d(\omega) = \begin{cases} 1 & |\omega| \leq 0.6\pi \\ 0 & 0.6\pi \leq |\omega| \leq \pi \end{cases} \tag{28}$$

The impulse response corresponding to the frequency response is given by

$$h_d(n) = \frac{0.6\pi}{\pi} \frac{\sin 0.6\pi n}{0.6\pi n} \quad -\infty < n < \infty \tag{29}$$
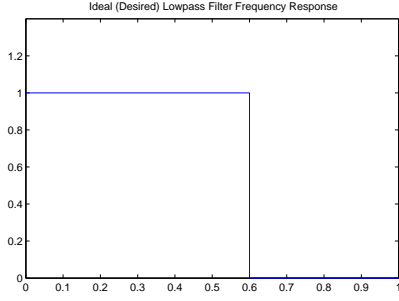
Fig. 1. Desired Normalized Frequency Response

TABLE I

COEFFICIENTS COMPARISON OF THE TWO ALGORITHMS AND THE IDEAL FILTER.(ORDER-10)

n	$h_d(n)$	$b_{fir2}(n)$	$b_{OLS}(n)$
-5	0.00000000000000	0.00038455366668	0.00002979805319
-4	0.07568267286407	0.07480846476052	0.07561109368412
-3	-0.06236595225258	-0.06232992055387	-0.06235323232102
-2	-0.09354892837886	-0.09299995683258	-0.09359782644052
-1	0.30273069145626	0.30266212174209	0.30275145161048
0	0.60000000000000	0.59960937500000	0.59991711964520
1	0.30273069145626	0.30266212174209	0.30275145161048
2	-0.09354892837886	-0.09299995683258	-0.09359782644052
3	-0.06236595225258	-0.06232992055387	-0.06235323232102
4	0.07568267286407	0.07480846476052	0.07561109368412
5	0.00000000000000	0.00038455366668	0.00002979805319

which are the components of the Fourier Series. Based on these coefficients, the following discussion focuses on a comparison between the filter design methods.

First, the fir2() routine, which involves applying equation (5) in MATLAB, is used to design the FIR filter coefficients with order $N = 10$, and f and m as given above. The design results without applying windowing are shown in the column under $b_{fir2}(n)$ in Table I.

Then the OLS based approach is applied for the design with the following considerations:

1) $H_d(e^{j\omega})$ are equally sampled over $[-\pi, \pi]$ with a sampling interval of $\frac{\pi}{1000}$ to produce the sampled frequency vector $\omega = [\omega_{-m}, \ldots, \omega_m]$ and the desired frequency response over the sampled frequencies $H = [H(e^{-j\omega_{-m}}), \ldots, H(e^{-j\omega_m})]^\tau$.

2) Assume that the filter coefficients are symmetric, i.e. $b_{-i} = b_i$, $0 \le i \le N$. Therefore, only half the coefficients β_1, the $(N+1) \times 1$ column vector

$$\beta_1 = \begin{pmatrix} b_{-N} \\ \vdots \\ 0 \end{pmatrix} \quad (30)$$

need to be determined and matrix U in equation (25) can consequently be reduced to

$$U_1 = \begin{pmatrix} (e^{-j(-N)\omega_{-m}} + e^{-j(N)\omega_{-m}}) & \cdots & e^{-j(0)\omega_{-m}} \\ \vdots & \ddots & \vdots \\ (e^{-j(-N)\omega_m} + e^{-j(N)\omega_m}) & \cdots & e^{-j(0)\omega_m} \end{pmatrix}$$
$$(31)$$

The design results (again without applying windowing) are shown in the column under $b_{OLS}(n)$ in Table I.

3) Using the forward OLS algorithm based on equation (26), the term selection stopped when $\sum ERR_{10} = 0.96631922625219$ and altogether 10 terms were included in FIR filter.

B. Algorithm Analysis

From Table I, it can be observed that the difference between the two design methods is not significant at all, but the OLS based design is slightly better than the design using the fir2() routine. Two aspects of analysis are conducted in the following for the two different design methods.

1) Analysis of the Filter Coefficients: According to equation (3) and (4),

$$h[n] = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=-\infty}^{\infty} h_d[l] W_N^{kl} W_N^{-kn} \quad (32)$$

and the relation between the ideal filter Fourier series and the frequency sampling impulse response coefficients can be simplified by swapping the order of summation [6],

$$h[n] = \sum_{m=-\infty}^{\infty} h_d(n + mN) \quad 0 \le n \le N-1 \quad (33)$$

Equation (33) implies that $h[n]$ from the fir2() routine is obtained from $h_d[n]$ by adding an infinite number of shifted replicas of $h_d[n]$, with each replica shifted by an integer multiple of N sampling instants. Since $h_d[n]$ is an infinite length sequence according to equation (2), which shows that the coefficients $h[n]$ computed from the inverse-DFT can not be the same as the ideal result. This is the intrinsic shortcoming of the fir2() routine.

However, the OLS based method doesn't possess this problem. This explains the slightly better performance of the OLS algorithm compared to the fir2() routine.

Another point is that fir2() may miscalculate coefficients when a higher frequency resolution is used in the computation. Consider the case $f = [0, 0.6, 0.6, 1]$ and frequency interval as $\frac{\pi}{1000}$ cases for the fir2() routine, the coefficients estimation through this approach is listed in Table II.

Considering next the frequency interval as $\frac{\pi}{1000}$ and $\frac{\pi}{10000}$, the results for the OLS algorithm are given in Table III.

It is obvious that for the OLS method the higher the frequency resolution, the closer the results converge to the ideal result, and the better the effect of the design. While for the fir2() routine, the limitation is that for higher frequency resolution situations, it may perform worse.

2) Term Selection Study: Normally, the OLS based method needs to select all the relevant terms to achieve an ideal filtering effect. However, by control of the summation of the

TABLE II

COEFFICIENTS COMPARISON OF FIR2() AND IDEAL ONES.(ORDER-10)

n	$h_d(n)$	$f = [0, 0.6, 0.6, 1]$	interval=$\frac{\pi}{1000}$
-5	0.00000000000000	0.00038455366668	0.00474422289254
-4	0.07568267286407	0.07480846476052	0.07381197433722
-3	-0.06236595225258	-0.06232992055387	-0.06603956659887
-2	-0.09354892837886	-0.09299995683258	-0.08955904050756
-1	0.30273069145626	0.30266212174209	0.30410856041926
0	0.60000000000000	0.59960937500000	0.59521484375000
1	0.30273069145626	0.30266212174209	0.30410856041926
2	-0.09354892837886	-0.09299995683258	-0.08955904050756
3	-0.06236595225258	-0.06232992055387	-0.06603956659887
4	0.07568267286407	0.07480846476052	0.07381197433722
5	0.00000000000000	0.00038455366668	0.00474422289254

TABLE III

COEFFICIENTS COMPARISON OF OLS AND IDEAL ONES.(ORDER-10)

n	$h_d(n)$	interval=$\frac{\pi}{1000}$	interval=$\frac{\pi}{10000}$
-5	0.00000000000000	0.00002979805319	0.00000353824178
-4	0.07568267286407	0.07561109368412	0.07567239400313
-3	-0.06236595225258	-0.06235323232102	-0.06236359287874
-2	-0.09354892837886	-0.09359782644052	-0.09354792546414
-1	0.30273069145626	0.30275145161048	0.30273168582088
0	0.60000000000000	0.59991711964520	0.59998368155733
1	0.30273069145626	0.30275145161048	0.30273168582088
2	-0.09354892837886	-0.09359782644052	-0.09354792546414
3	-0.06236595225258	-0.06235323232102	-0.06236359287874
4	0.07568267286407	0.07561109368412	0.07567239400313
5	0.00000000000000	0.00002979805319	0.00000353824178

TABLE IV

ERR VALUE ANALYSIS FOR EACH STEP.(ORDER-10, FREQUENCY INTERVAL=$\frac{\pi}{1000}$)

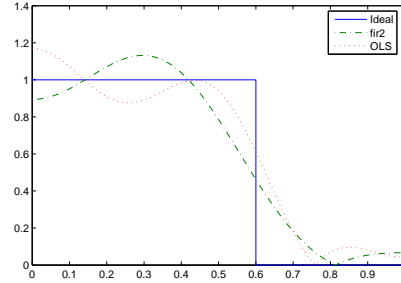| Step	$ERR_{(i)}$	$\sum_{(i)} ERR$	Selected terms
1	0.59978262523488	0.59978262523488	$e^{-j(0)\omega}$
2	0.30537642402055	0.90515904925543	$e^{-j(1)\omega} + e^{-j(-1)\omega}$
3	0.02912828996622	0.93428733922166	$e^{-j(2)\omega} + e^{-j(-2)\omega}$
4	0.01905095914553	0.95333829836719	$e^{-j(4)\omega} + e^{-j(-4)\omega}$
5	0.01298092491840	0.96631922328558	$e^{-j(3)\omega} + e^{-j(-3)\omega}$
6	0.00000000296661	0.96631922625219	$e^{-j(5)\omega} + e^{-j(-5)\omega}$

Fig. 2. Frequency Response Magnitudes with Term Selection ($\sum ERR = 0.95333829836719$)

the value of ERR for term $\left(e^{-j(3)\omega} + e^{-j(-3)\omega}\right)$ on the fourth step is 0.01296523967681, which makes $\sum ERR' = 0.94725257889847$, unqualified for the design requirement.

The comparison of design results and corresponding coefficients are depicted respectively in Figure 2 and 3. The significance of frequency component $n = 4$ is greater than that of $n = 3$, therefore, the OLS algorithm takes advantage and picks up the most significant term to improve the overall system design.

VI. CONCLUSIONS

In this paper, a new OLS based FIR filter design procedure has been proposed and compared with the traditional FIR filter design routine fir2() in MATLAB. Several advantages of the new method over traditional method have been demonstrated

ERR value, the number of terms in the desired FIR filter can be actively controlled to achieve a compromise between the filter complexity and the filter performance. This is another advantage of the OLS based method over the fir2() routine, which can do nothing on this point.

To demonstrate the merits, the term selection process of the OLS based design is illustrated in Table IV. Clearly, if the target value of $\sum ERR$ is only 95%, then the FIR filter of seven terms

$$H_{OLS}(e^{j\omega}) = b_{-4}e^{-j(-4)\omega} + b_{-2}e^{-j(-2)\omega} + b_{-1}e^{-j(-1)\omega}$$
$$+ b_0 e^{-j(0)\omega} + b_1 e^{-j(1)\omega} + b_2 e^{-j(-2)\omega} + b_4 e^{-j(4)\omega} \quad (34)$$

is sufficient to satisfy the design requirements. After the fourth step, the summation of ERR reaches $\sum ERR = 0.95333829836719$.

However, the seven terms FIR filter design using fir2() is

$$H_{fir2}(e^{j\omega}) = b_{-3}e^{-j(-3)\omega} + b_{-2}e^{-j(-2)\omega} + b_{-1}e^{-j(-1)\omega}$$
$$+ b_0 e^{-j(0)\omega} + b_1 e^{-j(1)\omega} + b_2 e^{-j(-2)\omega} + b_3 e^{-j(3)\omega} \quad (35)$$

The difference is that the fir2() routine selects frequency component $n = 3$ but not $n = 4$. Through further calculation,
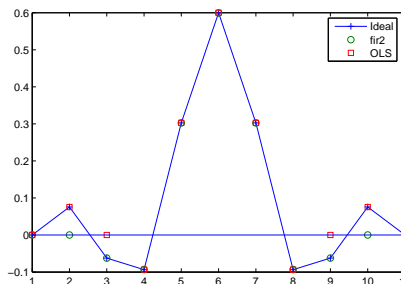

Fig. 3. Filter Coefficients Comparison with Term Selection. (Order-10)

using simple design examples. The simulation studies verify the effectiveness and value of this new approach. We are currently extending the OLS based method to deal with the issue of nonlinear filter designs. These results will be presented in a future publication.

ACKNOWLEDGMENT

REFERENCES

[1] Lertniphonphun, W. & McClellan, J.H. [1998] Complex frequency response FIR filter design, *Proceedings of the 1998 IEEE International Conference* Vol. 3, pp. 1301-1304

[2] Wu, S.P. & Boyd, S. & Vandenberghe, L. [1997] FIR Filter Design via Spectral Factorization and Convex Optimization, *Applied Computational Control, Signal and Communications* pp. 1-33

[3] Yong C.L. & Sun, Y. & Ya J.Y. [2002] Design of discrete-coefficient FIR filters on loosely connected parallel machines, *Signal Processing, IEEE Transactions* Vol. 50, pp. 1409-1416

[4] Chen, S. & Billings, S.A. & Luo, W. [1989] Orthogonal Least Squares Methods and Their Application to Nonlinear System Identification, *International Journal of Control* 50(5):1873–1896, 1989.

[5] Lay, D.C. [1998] Linear Algebra and Its Applications, *Second Edition, Addison-Wesley*, pp. 378-402.

[6] Mitra, S.K. [1998] Digital Signal Processing A Computer Based Approach, *First Edition, McGraw-Hill, New York*, pp. 462-468.

[7] Jackson, L.B. [1996] Digital Filters and Signal Processing, *Third Edition, Kluwer Academic Publishers, Boston*, pp. 301-307.

[8] Mulgrew, B. & Grant, P. & Thompson, J. [1999] Digital Signal Processing Concepts & Applications, *Palgrave, New York*, pp. 150-154.

[9] Swain, A.K. & Billings S.A. [1998] Weighted Complex Orthogonal Estimator for Identifying Linear and Nonlinear Continuous Time Models from Generalised Frequency Response Functions, *Mechanical Systems and Signal Processing*, pp. 269-292.