# Designing FIR Filters in a New Vector Space

James K Beard, *Life Senior Member, IEEE*

*Abstract*— The time and frequency response of a FIR filter are vectors mapped by the DFT matrix. A new vector space can be defined by expressing the DFT matrix in terms of its spectral decomposition, and expressing the filter as a vector adjacent to the characteristic value matrix. For linear phase filters, all operations and characteristic values are real, and design constraints in the time and frequency domains have identical formats. Simple methods are given here for computing the orthogonalized characteristic vector matrix, which is used in transforming the design constraints and in computing the filter weights. The procedure is efficient enough to be useful on low cost personal computers.

*Index Terms*—FIR digital filters, Fourier transforms, linear phase filters, linear programming.

## I. INTRODUCTION

FINITE impulse response (FIR) or convolution filters are used as windows for FFT's and other purposes [1]. The relationship between the time domain response, or weights, and the frequency response is

$$\underline{a} = D \cdot \underline{h} \qquad (1.1)$$

where the underlined small letters are vectors and $D$ is the DFT matrix, properly normalized (see below). If there are $N$ weights in the filter, $D$ is order $N$ or larger.

In the Remez exchange FIR filter design technique [2], [3], the filter design is characterized and designed as a polynomial in the frequency domain and the DFT is used to find the filter weights from the frequency response. This technique is known to provide good accuracy and efficiency and is useful for filters having a few hundreds weights.

A linear programming problem is one in which a solution vector is defined in terms of optimization of a cost function subject to a set of constraints or inequalities. The cost function and the inequalities are defined in terms of linear combinations of the parameters in the solution vector. FIR filters can be designed in this way [4, 5], using rows from the inverse of equation (1) as constraints on the impulse response. Frequency domain constraints are applied directly, and a DFT is used to find the weights. The linear programming approach is noted for less efficiency than Remez exchange, and large FIR filters are rarely designed this way.

Simultaneous time and frequency domain constraints in digital filter design have been investigated [5], and the linear programming approach lends itself easily to this. A very real advantage is realized when filters are designed in this way, because designs are realized which achieve lower overall processing losses for a given filter complexity.

A third avenue is presented here for the first time. The DFT matrix is factored,

$$D = V \cdot L \cdot V^T \qquad (1.2)$$

where $V^T$ is the transpose of $V$ (and also its inverse) and $L$ is a diagonal matrix. Given this factorization of the DFT matrix, the filter can be characterized as a vector $\underline{s}$ given by

$$\underline{s} = V^T \cdot \underline{h} = L^* \cdot V^T \cdot \underline{a} \qquad (1.3)$$

where $L^*$ is the complex conjugate of the diagonal matrix $L$ (and its inverse). In this vector space, constraints on $\underline{a}$ and $\underline{h}$ are posed in the same format.

## II. LINEAR PROGRAMMING

Posing FIR filter design as a linear programming problem consists of defining a cost function vector $\underline{c}$ and a constraint matrix $A$ and vector $\underline{b}$. The constraints and cost function $J$ are defined by the linear relationships

$$\left. \begin{array}{c} A \cdot \underline{s} \leq \underline{b} \\ J = \underline{c}^T \cdot \underline{s} . \end{array} \right\} \qquad (2.1)$$

The rows of (2.1) can be found by taking rows from the equations

$$\underline{a} = V \cdot L \cdot \underline{s} \qquad (2.2)$$

and

$$\underline{h} = V \cdot \underline{s} . \qquad (2.3)$$

An efficient, accurate linear programming method is to use a variation of the Simplex method [7], [8] roughly following [7], but using Householder transformations to triangularize the matrices involved instead of Givens rotations. The algorithm is customized for FIR filter design by other modifications. For example, the cost vector $\underline{c}$ can be made adaptive to minimize the number of nonzero weights in $\underline{h}$ by using

$$J = \underline{c}^T \cdot V^T \cdot \underline{h} \qquad (2.4)$$

and defining the equivalent cost vector

$$\underline{c}' = V \cdot \underline{c} \qquad (2.5)$$

to force the solution to be favored which uses a smaller number of weights. This is done by defining the elements of $\underline{c}'$ as having the sign of the corresponding element of $\underline{h}$, and having a magnitude which increases as the distance from the center of the filter increases. Another important custom feature is used to define the bracketed limits expressing the filter specifications. Specifications for stopband attenuation, passband ripple or scalloping loss, and time domain scalloping loss all have the same form:

$$\left| \underline{a}^T \cdot \underline{s} - \underline{b} \right| \leq \underline{d} \tag{2.6}$$

where $\underline{d}$ is a vector of small values expressing stopband leakage (b=0 for stopband leakage), passband peak to peak ripple, or time domain scalloping loss. An efficient, trouble-free way to place these conditions into the linear programming generic format is to define two sets of slack variables $\underline{xs1}$ and $\underline{xs2}$ and use the equations

$$\underline{a}^T \cdot \underline{s} + \underline{xs1} = \underline{b} - \underline{d} \tag{2.7}$$

$$\underline{xs1} - \underline{xs2} = 2 \cdot \underline{d} \tag{2.8}$$

$$\left. \begin{array}{c} \underline{xs1} \geq 0 \\ \underline{xs2} \geq 0 . \end{array} \right\} \tag{2.9}$$

## III. FACTORING THE DFT MATRIX

The DFT matrix is here defined to be the symmetrical complex matrix

$$D = \frac{1}{r} \cdot \left[ \exp\left( j \cdot \frac{2\pi \cdot p \cdot q}{N} \right) \right] \tag{3.1}$$

where $j$ is the unit imaginary, $p$ and $q$ are the row and column indices, N is the order of the matrix, and r is the square root of N. With this normalization, the DFT matrix is complex, symmetrical, unitary, and, without the division by r, it is a Vandermonde matrix [9]. A Vandermonde matrix is of the form

$$\left[ (t_i)^{(j-1)} \right]. \tag{3.2}$$

Vandermonde matrices have been extensively studied as they result when polynomial fits are computed. This is no coincidence, as the inverse DFT can be interpreted as the result of solving for the frequency response function as coefficients of a polynomial in $z$, where the polynomial is fitted for $N$ values of $a(z)$ spaced evenly around the unit circle. The determinant of the Vandermonde matrix is well known [9]. The determinant of D is

$$|D| = \frac{1}{N^{N/2}} \cdot \prod_{k=1}^{N-1} \left( \exp\left( -j \cdot \frac{2\pi \cdot k}{N} \right) - \exp\left( -j \cdot \frac{2\pi \cdot i}{N} \right) \right). \tag{3.3}$$

This equation and the property that the determinant of a unitary matrix has a magnitude of one lead to the expression

$$|D| = \exp\left( -j \cdot \frac{2\pi}{8} \cdot (N-1) \cdot (3 \cdot N - 2) \right). \tag{3.4}$$

A very simple equation can be used to draw a very general conclusion about the characteristic values of the DFT matrix. Specifically,

$$D^2 = E \tag{3.5}$$

where the real, symmetric matrix $E$ contains a 1 in the first row and column position, ones in positions whre the sum of the row and column indices is $N + 2$, and zeros elsewhere. In addition,

$$D^4 = E^2 = I , \tag{3.6}$$

so that $E$ is its own inverse. Equation (3.6) also proves that all the characteristic values of $D$ are fourth root of $+1$. The characteristic vector matrix of $D$ must follow the format

$$D \cdot V = V \cdot L \tag{3.7}$$

where $V$ is a matrix whose columns are characteristic vectors and $L$ is a matrix containing characteristic values of $D$ on the main diagonal and zeros elsewhere. Equation (3.6) can be written as

$$D^5 - L^5 = D - L \tag{3.8}$$

which, with the polynomial factorization

$$\left. \begin{array}{l} x^5 - y^5 \\ = (x-y) \cdot (x^4 + x^3 \cdot y + x^2 \cdot y^2 + x \cdot y^3 + y^4) \end{array} \right\} \tag{3.9}$$

suggests the form

$$\left. \begin{array}{l} D^5 - L^5 = D - L \\ = D \cdot (Q + I) - (Q + I) \cdot L \end{array} \right\} \tag{3.10}$$

where $Q$ is given by

$$Q = I + D \cdot L^3 + D^2 \cdot L^2 + D^3 \cdot L . \tag{3.11}$$

Equation (3.7) suggests the equivalence

$$\left. \begin{array}{l} V = \frac{1}{4} \cdot Q \\ = \frac{1}{4} \cdot (I + D \cdot L^3 + D^2 \cdot L^2 + D^3 \cdot L) \\ = \frac{1}{4} \cdot (I + E + D \cdot (L^2 + E) \cdot L) \end{array} \right\} \tag{3.12}$$

as a candidate for a matrix containing some characteristic vectors. The factor of $1/4$ is an arbitrary normalization chosen for later convenience.

Equation (3.12) will yield a $V$ matrix which satisfies (3.7) so long as the diagonal elements of $L$ are fourth roots of $+1$. If they are all the same root, $V$ cannot be full rank for $N > 1$. However, the $V$ matrices thus found do span the characteristic vector space for any one characteristic value. Any characteristic vector $\underline{x}$ of $D$ satisfies

$$D \cdot \underline{x} = e \cdot \underline{x} \tag{3.13}$$

where $e$ is a fourth root of $+1$. Then, if $V$ is computed from (3.12) using

$$L = e \cdot I , \tag{3.14}$$

then

$$V^T \cdot \underline{x} = \underline{x} . \qquad (3.15)$$

A process very similar to that shown above establishes that characteristic vectors corresponding to different fourth roots of +1 are orthogonal.

If the subscripts 1, 2, 3 and 4 are used for +1, -j, -1 and +j respectively, the characteristic vector matrices can be written as

$$
\left.
\begin{aligned}
V_1 &= \frac{1}{4} \cdot (I + E) + \frac{1}{2} \cdot C, \text{ rank is } \left[\frac{N+4}{4}\right] \\[2mm]
V_2 &= \frac{1}{4} \cdot (I - E) - \frac{1}{2} \cdot S, \text{ rank is } \left[\frac{N+2}{4}\right] \\[2mm]
V_3 &= \frac{1}{4} \cdot (I + E) - \frac{1}{2} \cdot C, \text{ rank is } \left[\frac{N-1}{4}\right] \\[2mm]
V_4 &= \frac{1}{4} \cdot (I - E) + \frac{1}{2} \cdot S, \text{ rank is } \left[\frac{N+1}{4}\right]
\end{aligned}
\right\} \quad (3.16)
$$

where $C$ and $S$ are the real and imaginary parts of the DFT matrix,

$$D = C - j \cdot S . \qquad (3.17)$$

Note that (3.16) shows that all of the V matrices are real and symmetrical. Pure real numbers are important in practical applications because it means that involvement with the $V$ matrices does not automatically make complex arithmetic necessary. It is also important to note that $V_1$ and $V_3$ are characteristic vectors of $C$ as well as $D$, and that $V_2$ and $V_4$ are characteristic vectors of $S$ as well as $D$. This is used to establish the ranks of the $V_i$ as the number of distinct columns of $C$ and $S$. The ranks can also be established from the fact that the ranks are nondecreasing with $N$ and the determinant of $D$ as a function of $N$ given by (3.4). Furthermore, it can be seen from (3.13) and (3.15) and the fact that each row and column of V satisfies these equations that

$$(V_i)^2 = V_i, \, i = 1, 2, 3, 4 \qquad (3.18)$$

so that each of the V matrices is its own characteristic vector matrix. This is important in the process of orthogonalization and normalization. Orthogonalization and normalization are necessary because, while the columns of each of the four $V$ matrices are orthogonal to the columns of all the other $V$ matrices, they are not necessarily orthogonal to each other; the fact (3.18) shows that are definitely NOT orthogonal to each other, and in addition that their lengths are not 1.

The ranks of the $V$ matrices can be determined by a count of the corresponding characteristic values. In addition, it can be seen from (3.12) that each of the columns of that equation can be considered separate equations. Which columns to select when synthesizing the final $V$ matrix from the $V_i$ can be determined from the following properties:

(a) The columns of $V_1$ and $V_3$ exhibit even symmetry about column $N/2$ for even $N$, or about the equal columns $(N-1)/2$ and $(N+1)/2$ for odd $N$.

(b) Columns 1 and (for even $N$) $N/2$ of $V_2$ and $V_4$ are zero, and the columns of these matrices exhibit odd symmetry about the center column or columns.

(c) Of the remaining approximately $N/2$ columns, only approximately $N/4$ are distinct, but any of them can be used with Gram-Schmidt orthogonalization to form a basis set which spans the characteristic vector space.

So, when the $V$ matrix is formed from the $V_i$, only vectors from the first half of the matrices are used, and the first and center column (if any) of $V_2$ and $V_4$ are avoided.

One last very important fact about orthogonalization and normalization of $V$ follows from (3.18). Gram-Schmidt Orthogonalization uses dot products of the columns of the matrix to determine weights which are then used to add a column to other columns to produce orthogonal column vectors. Equation (3.18) shows that these dot products can be taken as elements of the matrix and need not be computed. Furthermore, careful investigation of the Gram-Schmidt orthogonalization process shows that this property is not destroyed by the orthogonalization process. This also implies that the dot products required for normalization of the vectors to length 1 can also be found from the diagonal of the orthogonalized matrix and also need not be computed. This makes the orthogonalization and normalization of $V$ very simple and efficient

## IV. REMEZ EXCHANGE

The Remez exchange approach [2], [3] can be used if the filter can be characterized as a polynomial in the $\underline{s}$ vector space, and the design constraints from both the time and frequency domains can be mapped to the $\underline{s}$ vector space. The key point is that the filter must be characterized as a polynomial. This is natural for the relationship between the filter weights $\underline{h}$ and the frequency response $\underline{a}$ given in (1.1) because, as stated above, computing a DFT can be interpreted as a polynomial fit computation. At this time, a polynomial formulation in the $\underline{s}$ domain has not been formulated. However, if the filter is characterized as a polynomial in the frequency domain, time domain constraints can be mapped to the frequency domain using the inverse of (1.1) and used as part of the Remez cost function. A real cosine transform or the factorization of the DFT matrix given above can be used to avoid complex arithmetic.

## V. CONCLUSIONS

Results at this writing are conclusive in that the $V$ matrix can be efficiently and accurately computed as given above,

and that the linear programming method outlined can be used to successfully design FIR filters. Implementation of the method given here benefits from ongoing work by others [9], [10].

Plots showing the relative speed and accuracy of direct use of linear programming versus the application of the same linear programming methods in $s$ space will be shown at the presentation. The method will also be compared with the Remez exchange method when only frequency domain constraints are applied. Listings of the key implementations will be made available as the work progresses.

## REFERENCES

[1] Schafer, R. W. and Rabiner, R. L., "A Digital Signal Processing Approach to Interpolation", *Proc. IEEE*, V. 61 No. 6, June 1973, pp 692-702

[2] J. H. McClellan, T. W. Parks and L. R. Rabiner, "A Program for the Design of Linear Phase Finite Impulse Response Digital Filters*", IEEE Trans. on Audio and Electroacoustics*, AU-21, No. 6, 506-526, Dec., 1973.

[3] L. R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing*, Prentice Hall, 1975.

[4] L. R. Rabiner, "Linear Program Design of Finite Impulse Response (FIR) Digital Filters", *IEEE Trans. on Audio and Electroacoustics*, AU-20, No. 4, 280-288, Oct., 1972.

[5] Chen, X. and Parks, T. W., "Design of FIR Filters in the Complex Domain", *IEEE Trans. ASSP*, ASSP-35, No. 2, 144-153, February 1987.

[6] Dembro, A. and Malah, D., "The Design of Optimal Uniform Filter Banks with Specified Composite Response", *IEEE Trans. ASSP*, ASSP-35, No. 6, 807-817, June 1987.

[7] Bartels, R. H., Stoer, J., and Zenger, Ch., "A Realization of the Simplex Method Based on Triangular Decompositions", contribution I/11 in *Handbook for Automatic Computation, Volume II, Linear Algebra*, J. H. Wilkinson and C. Reinsch, Eds., Springer-Verlag, 1971.

[8] William L. Broga*, Modern Control Theory*, Quantum Books, NY (1974). See especially pp 87-90 and 95-102 for an excellent overview of linear programming and the Simplex method.

[9] Richard Bellman, *Introduction to Matrix Analysis (Second Edition)*, McGraw-Hill, 1970, p. 193.

[10] Sorenson, H. V., Jones, D. L., Heideman, M. T., and Burrus, C. S., "Real- Valued Fast Fourier Transform Algorithms*", IEEE Trans. ASSP*, ASSP-35, No. 6, 849-863, June, 1987.

[11] Hou, H. S., "A Fast Recursive Algorithm for Computing the Discrete Cosine Transform*", IEEE Trans. ASSP*, ASSP-35, No. 10, 1455-1461, October 1987.

**James K Beard** (M'64–LM'04–LSM'05) became a Member (M) of IEEE in 1964, Life Member (LM) in 2004, and a Life Senior Member in 2005. He was born in Austin, TX in 1939. He received a BS degree from the University of Texas at Austin in 1962, an MS from the University of Pittsburgh in 1963, and the Ph. D. from the University of Texas at Austin in 1968, all in electrical engineering.

Between 1959 and 2004, he worked in Government laboratories, industry, and as an individual consultant. Employers include precursors or current divisions of Northrop Grumman, Raytheon, and Lockheed Martin, most recently Lockheed Martin MS2 in Moorestown, NJ. He is currently an individual consultant based in Medford, NJ near Philadelphia. He is the author of a number of symposia papers and a book, "The FFT in the 21$^{st}$ Century" (Kluwer, 2003). Current research interests include system engineering solutions to homeland defense issues, estimation and decision theory, radar and communications concept and waveform design, and digital radar concepts. He is an adjunct professor at Temple University in Philadelphia and teaches graduate and undergraduate analog and digital communications.

Dr. Beard is a member of AIAA, AOC, SPIE and ACM. He was Publications Chairman for FUSION2005. He is a member of Phi Eta Sigma, Eta Kappa Nu, Tau Beta Pi, and Sigma Xi. He studied for his Ph. D. under a GSRF Fellowship (matched U. Texas Austin and Ford Foundation funding, administered by U. Texas Austin) and a NSF Fellowship.