DESIGN OF N-DIMENSIONAL MCCLELLAN TRANSFORM BY SIMULTANEOUS LEAST-SQUARES OPTIMIZATION OF CONTOUR MAPPING AND CUTOFF FREQUENCY OF 1-D PROTOTYPE

William C. Ogle

Science Applications International Corp. (SAIC)
4001 N. Fairfax Dr.
Ste. 675
Arlington, VA 22203
USA
`oglew@saic.com`

Mark J.T. Smith

Center for Signal and Image Processing (CSIP)
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332
USA
`mark.smith@ece.gatech.edu`

ABSTRACT

A new technique is proposed for determining simultaneously the coefficients of the N-dimensional (N-D) McClellan Transformation and the cutoff frequency of the one-dimensional (1-D) prototype. This method is applicable to arbitrary centro-symmetric N-D zero-phase frequency responses. It employs a mapping of the 1-D cutoff frequency to a series of points along the corresponding contour of the ideal N-D response. The formulation results in an analytical, closed-form, least-squares solution for both the coefficients and the cutoff frequency. A pair of general constraints are imposed that avoid the transformation scaling problem. A detailed example is given for a 3D ellipsoidal frequency response, including a performance measurement.

1. INTRODUCTION

For N-D adaptive filtering, it is desirable to have a flexible technique for quickly designing N-D digital finite impulse response (FIR) filters. The technique should easily allow a high degree of control over the features or shape of the filter's frequency response. Furthermore, an efficient implementation of the resulting filter is extremely valuable when one considers the computational burden of the N-D data, and the real-time constraints normally imposed on the application. The McClellan transformation is a powerful method for constructing N-D digital FIR filters [1, 2]. These reasons provide ample motivation to pursue the development of N-D McClellan transformation design methods, since it is known that the corresponding filters can be implemented in very computationally efficient architectures [3]. A method for 2-D least-squares contour mapping is proposed in [4], of which this method is not only a generalization to N-D filters, but also an improvement by simultaneous determination of the optimal 1-D cutoff frequency. Least-squares 2-D transformation design is also discussed in [5], in which a recursive system identification approach is taken. Another 2-D design algorithm [6] finds the coefficients, but not the 1-D cutoff contour simultaneously. In [7], the 2-D eigen-system solution requires numerical integration , which is computationally intensive. In [8], a fast least-squares 2-D design method is introduced, but the McClellan transformation is not used, so the resulting filter may not have an efficient implementation. The work done in [9] assumes the cutoff frquency is known before hand, whereas the method herein computes it automatically. In [10], the 2-D solution requires numerical integration over a fixed contour, which makes it unwieldy for rapid adaptive N-D filters. Some recent 3-D work [11, 12] does not treat the cutoff frequency as a separate unknown parameter, and is focused on a particular symmetry class (spherical).

The approach discussed here results in a closed form solution for optimizing both the transform coefficients and the 1-D cutoff frequency. It is stated in a form general enough to be used for filters of any dimensionality and any type of centro-symmetry, and the transformation scaling problem can be avoided through appropriately chosen constraints. The design method is derived below, followed by a 3-D numerical example, including a performance measurement to demonstrate its effectiveness.

2. TRANSFORMATION DESIGN

The two primary parts of the McClellan transformation are the 1-D filter design, including the cutoff frequency ω_c, and the design of the transformation function. Since the performance of a given transformation function varies with respect to both ω_c and the coefficients themselves [4], it is desirable to simultaneously optimize the coefficients and ω_c. The approach described here treats $\cos(\omega_c)$ as another free parameter, and optimizes over both the transform coefficients and $\cos(\omega_c)$ simultaneously.

The essence of the McClellan transformation is captured by the 1-D to N-D mapping

$$\begin{aligned}\cos(\omega) &= F(\omega_1, \omega_2, \ldots, \omega_N) \\ &= F(\boldsymbol{\omega}), \end{aligned} \qquad (1)$$

where $F(\boldsymbol{\omega})$ is the transformation function, and $\boldsymbol{\omega}$ denotes the vector $(\omega_1, \omega_2, \ldots, \omega_N)^T$. If $F(\boldsymbol{\omega})$ is the frequency response of a zero-phase filter, then so will be the resulting

---

W.C. Ogle is also with the School of Physics, Georgia Institute of Technology, Atlanta, GA 30332, USA filter. The problem, then, is to find the transform coefficients whose frequency response equals or optimally approximates the ideal $F(\boldsymbol{\omega})$. The optimality criterion used here is least-squares minimization.

2.1. N-Dimensional Transformation Function

Let the support of the transform coefficients be an N-D region defined by $-M_d \leq n_d \leq M_d$, $d \leq N$ where $M_d > 0$ and n_d is the value of the index in dimension d. For example, a 1-D zero-phase filter of length 5 would have $N = 1$, $M_1 = 2$, and $n_1 = (-2, -1, 0, 1, 2)$. Similarly, a 2-D zero-phase 5×3 filter would have $N = 2$, and

$$\begin{array}{rclcrcl} M_1 & = & 2 & \quad & n_1 & = & (-2,-1,0,1,2) \\ M_2 & = & 1 & & n_2 & = & (-1,0,1). \end{array} \tag{2}$$

Continuing with the 2-D example, a specific coefficient s is referenced by a vector of indices, so that $s = s[\mathbf{n}]$, where $\mathbf{n} = [n_1, n_2]$. In addition, every $\mathbf{n}$ and $-\mathbf{n}$ form a unique centro-symmetric *pair* (we don't include the degenerate pair of the origin and itself). For the 2-D example given above, the number of such pairs is 7.

The extension of this 2-D example to the N-D case is straightforward and forms the initial basis for the formulation in this paper. Thus, the N-D transformation function $F(\boldsymbol{\omega})$ is expressed as the following Discrete Fourier Transform (DFT):

$$F(\boldsymbol{\omega}) = \sum_{i=-N_p}^{N_p} s[\mathbf{n}_i] \exp(-j\boldsymbol{\omega}^T \mathbf{n}_i), \tag{3}$$

where the N-D index vector $\mathbf{n}_i$ is ordered such that it is odd under reflection about the origin:

$$\begin{array}{rcl} \mathbf{n}_i & = & [n_1[i], n_2[i], \ldots, n_N[i]]^T \\ & = & -\mathbf{n}_{-i}, \end{array} \tag{4}$$

and

$$N_p = \frac{\left(\prod_{i=1}^{N}(2M_i+1)\right) - 1}{2} \tag{5}$$

is the number of centro-symmetric pairs of $\mathbf{n}_i$, and $s[\mathbf{n}_i]$ are the sought-after transform coefficients.

Notice that (4) implies that $\mathbf{n}_0 = \mathbf{0}$. The analysis herein exploits the symmetry property of the indices, as illustrated in (2) and captured in (4). Furthermore, if one incorporates the assumption that $s[\mathbf{n}_i]$ is real, and the zero-phase condition that $s[\mathbf{n}_i] = s[\mathbf{n}_{-i}]$, (3) reduces to

$$F(\boldsymbol{\omega}) = s[\mathbf{0}] + \sum_{i=1}^{N_p} 2s[\mathbf{n}_i] \cos(\boldsymbol{\omega}^T \mathbf{n}_i). \tag{6}$$

Referring once again to the 2-D example discussed above, there are $N_p = 7$ pairs of coefficients whose indices satisfy the reflection symmetry in (4), and so there are a total of 8 coefficients to be found, including the one at the origin.

As a note, we mention that since the arguments of the cosine terms in (6) are of the form $\omega_1 n_1 + \omega_2 n_2 + \cdots + \omega_N n_N$, then application of an elementary trigonometric identity produces the

$$\begin{array}{c} \cos(\omega_1 n_1)\cos(\omega_2 n_2)\cdots\cos(\omega_N n_N) \\ +\sin(\omega_1 n_1)\sin(\omega_2 n_2)\cdots\sin(\omega_N n_N) \end{array} \tag{7}$$

formulation more often seen in the literature to denote the generalized McClellan transformation for both symmetric and anti-symmetric frequency responses. However, such a conversion only makes the analysis in this paper more cumbersome, and does not add value. Therefore, the cosine form in (6) will be kept, since it still provides the same flexibility for symmetric and anti-symmetric frequency responses.

2.2. Constraints

In order to avoid the transformation scaling problem [13], we impose a pair of constraints, which are specific mappings from 1-D to N-D, generally expressed as

$$\omega = 0 \quad \mapsto \quad \boldsymbol{\omega} = \boldsymbol{\theta_1} \tag{8}$$

$$\omega = \pi \quad \mapsto \quad \boldsymbol{\omega} = \boldsymbol{\theta_2}. \tag{9}$$

The particular choice of $\boldsymbol{\theta_1}$ and $\boldsymbol{\theta_2}$ depends on the desired frequency response. For the centro-symmetric case, we want the 1-D frequency origin to correspond to the N-D frequency origin, so that $\boldsymbol{\theta_1} = \mathbf{0}$ is an easy choice. To avoid the scaling problem, $\boldsymbol{\theta_2}$ should be chosen to be on the boundary of the frequency space. Therefore,

$$\boldsymbol{\theta_2} = \pi\boldsymbol{\nu}, \tag{10}$$

where $\boldsymbol{\nu}$ is an length-N column vector, each of whose elements is either 1, -1, or 0, with the constraint that *at least one* is non-zero. This guarantees that $\boldsymbol{\theta_2}$ resides at some point on the boundary of $[-\pi, \pi]^N$. Note that our choice of $\boldsymbol{\theta_2}$ restricts it to be either on a frequency axis (e.g. $\boldsymbol{\theta_2} = [\pi, 0, \pi]$) or on an extremal corner (e.g. $\boldsymbol{\theta_2} = [\pi, \pi, \pi]$). This restrictiveness provides convenient structure. This paper does not address the more general case, in which the only requirement on $\boldsymbol{\theta_2}$ is that at least one of its elements be $\pm\pi$, so that it is on the extremal boundary, but not necessarily on an axis or corner.

The constraints are then summarized as

$$\cos(0) \quad = \quad F(\mathbf{0}) \tag{11}$$

$$\cos(\pi) \quad = \quad F(\pi\boldsymbol{\nu}). \tag{12}$$

Evaluation of (11) and (12) in (6) yields

$$1 \quad = \quad s[\mathbf{0}] + \sum_{i=1}^{N_p} 2s[\mathbf{n}_i] \tag{13}$$

$$-1 \quad = \quad s[\mathbf{0}] + \sum_{i=1}^{N_p} 2s[\mathbf{n}_i] \cos(\pi\boldsymbol{\nu}^T \mathbf{n}_i). \tag{14}$$

We will use these constraints to solve for two of the unknown coefficients in terms of the others, and insert the resulting expressions into (6), the transformation equation. This reduces by 2 the number of unknown coefficients we need to find.

Before proceeding, however, a notation is introduced solely for compactness. Let $\eta_i = \boldsymbol{\nu}^T \mathbf{n}_i$ so that

$$\cos(\pi\boldsymbol{\nu}^T \mathbf{n}_i) = \cos(\pi\eta_i). \tag{15}$$

Furthermore, since both $\boldsymbol{\nu}$ and $\mathbf{n}_i$ have integer elements, then η_i is an integer, and (15) becomes

$$\begin{aligned} \cos(\pi\boldsymbol{\nu}^T\mathbf{n}_i) &= \cos(\pi\eta_i) \\ &= (-1)^{\eta_i}. \end{aligned} \qquad (16)$$

In addition, let

$$i_o = \min\arg((\eta_i \pmod 2) = 1) \qquad (17)$$

be the smallest index i such that η_i is odd (and non-zero).

We now continue to derive expressions for two of the transform coefficients in terms of the others. This is simply done by combining equations (13) and (14) to get

$$s[\mathbf{0}] = -\sum_{i=1}^{N_p} s[\mathbf{n}_i][1+(-1)^{\eta_i}] \qquad (18)$$

$$s[\mathbf{n}_{i_o}] = \frac{1}{2}\left(1 - \sum_{\substack{i=1 \\ i\neq i_o}}^{N_p} s[\mathbf{n}_i][1-(-1)^{\eta_i}]\right) \qquad (19)$$

We point out that, although not explicitly shown, the right hand side of (18) only has nonzero terms for even η_i, so $s[\mathbf{n}_{i_o}]$ is not represented in the summation. This means that both $s[\mathbf{0}]$ and $s[\mathbf{n}_{i_o}]$ are expressed in terms of the other unknown coefficients. Similarly, only the odd η_i contribute non-vanishing terms for (19). It is easy to show that substitution of (18) and (19) back into (6) produces

$$\begin{aligned} F(\boldsymbol{\omega}) &= \sum_{\substack{i=1 \\ i\neq i_o}}^{N_p} s[\mathbf{n}_i]\left[2\cos(\boldsymbol{\omega}^T\mathbf{n}_i) - \alpha_i - \beta_i\cos(\boldsymbol{\omega}^T\mathbf{n}_{i_o})\right] \\ &\quad + \cos(\boldsymbol{\omega}^T\mathbf{n}_{i_o}) \qquad (20) \\ &\triangleq \mathbf{g}(\boldsymbol{\omega})^T\mathbf{s} + \cos(\boldsymbol{\omega}^T\mathbf{n}_{i_o}), \qquad (21) \end{aligned}$$

where the column vector $\mathbf{g}(\boldsymbol{\omega})$ has elements $\mathbf{g}(\boldsymbol{\omega}, i), 1 \leq i \leq N_p, i \neq i_o$, defined by

$$\mathbf{g}(\boldsymbol{\omega}, i) = 2\cos(\boldsymbol{\omega}^T\mathbf{n}_i) - \alpha_i - \beta_i\cos(\boldsymbol{\omega}^T\mathbf{n}_{i_o}). \qquad (22)$$

Similarly, the column vector $\mathbf{s}$ has elements $\mathbf{s}[i], 1 \leq i \leq N_p, i \neq i_o$ defined by

$$\mathbf{s}[i] = s[\mathbf{n}_i]. \qquad (23)$$

Finally, $\alpha_i = [1+(-1)^{\eta_i}]$ and $\beta_i = [1-(-1)^{\eta_i}]$. It is interesting to note that for any given i, then the pair (α_i, β_i) has values (0,2) if η_i is odd, or (2,0) if η_i is even. It may be possible to exploit this structure to gain analytical or computational efficiency.

Examples of $\mathbf{g}(\boldsymbol{\omega})$ and $\mathbf{s}$ are shown below. If $i_o = 1$ (which is often the case), then the $i = 1$ term is not included, so that

$$\mathbf{g}(\boldsymbol{\omega}) = \begin{bmatrix} 2\cos(\boldsymbol{\omega}^T\mathbf{n}_2) - \alpha_2 - \beta_2\cos(\boldsymbol{\omega}^T\mathbf{n}_2) \\ 2\cos(\boldsymbol{\omega}^T\mathbf{n}_3) - \alpha_3 - \beta_3\cos(\boldsymbol{\omega}^T\mathbf{n}_3) \\ \vdots \\ 2\cos(\boldsymbol{\omega}^T\mathbf{n}_{N_p}) - \alpha_N - \beta_N\cos(\boldsymbol{\omega}^T\mathbf{n}_{N_p}) \end{bmatrix}$$

and

$$\mathbf{s} = \begin{bmatrix} s[\mathbf{n}_2] \\ s[\mathbf{n}_3] \\ \vdots \\ s[\mathbf{n}_{N_p}] \end{bmatrix},$$

which are both length $N_p - 1$, regardless of the value of i_o.

2.3. Least-Squares Contour Mapping

The objective is to map the 1-D cutoff frequency to the N-D cutoff contour of the transformation function. The contour is specified by a series of N_c points $\boldsymbol{\omega}_k$, $1 \leq k \leq N_c$. For each contour point $\boldsymbol{\omega}_k$, we want

$$\begin{aligned} \cos(\omega_c) &= F(\boldsymbol{\omega}_k) \\ &= \mathbf{g}(\boldsymbol{\omega}_k)^T\mathbf{s} + \cos(\boldsymbol{\omega}_k{}^T\mathbf{n}_{i_o}). \end{aligned} \qquad (24)$$

This results in a non-homogeneous system of N_c equations, with the $N_p - 1$ unknown coefficients $s[\mathbf{n}_i]$ contained in $\mathbf{s}$. The system can be re-written as

$$\begin{array}{ccccc} \mathbf{g}(\boldsymbol{\omega}_1)^T\mathbf{s} & - & \cos(\omega_c) & = & -\cos(\boldsymbol{\omega}_1{}^T\mathbf{n}_{i_o}) \\ \mathbf{g}(\boldsymbol{\omega}_2)^T\mathbf{s} & - & \cos(\omega_c) & = & -\cos(\boldsymbol{\omega}_2{}^T\mathbf{n}_{i_o}) \\ \vdots & & \vdots & & \vdots \\ \mathbf{g}(\boldsymbol{\omega}_{N_c})^T\mathbf{s} & - & \cos(\omega_c) & = & -\cos(\boldsymbol{\omega}_{N_c}{}^T\mathbf{n}_{i_o}) \end{array}. \qquad (25)$$

We consider $\cos(\omega_c)$ to also be an unknown, so that (25) is now a system of equations *linear* in all the N_p unknowns. Furthermore, if $N_c > N_p$, then the system is overdetermined, and can easily be solved by the least-squares approximation. It is easy to satisfy this criterion: one typically chooses a transformation function with ($M_d = 1$, $\forall d \in [1, N]$), so that $N_p = 13$ in the 3-D case, which is at least an order of magnitude less than the number of points N_c needed to reasonably specify the 3-D cutoff contour. Therefore, in this paper, we impose the easily-satisfied restriction that $N_c > N_p$.

To express the problem in matrix notation, define the following arrays. Let

$$\mathsf{G} \triangleq \begin{bmatrix} \mathbf{g}(\boldsymbol{\omega}_1)^T & -1 \\ \mathbf{g}(\boldsymbol{\omega}_2)^T & -1 \\ \vdots & \vdots \\ \mathbf{g}(\boldsymbol{\omega}_{N_c})^T & -1 \end{bmatrix}, \qquad (26)$$

collect all the unknowns in the vector

$$\hat{\mathbf{s}} \triangleq [\mathbf{s}^T \ \cos(\omega_c)]^T, \qquad (27)$$

and denote

$$\mathbf{b} \triangleq \begin{bmatrix} -\cos(\boldsymbol{\omega}_1{}^T\mathbf{n}_{i_o}) \\ -\cos(\boldsymbol{\omega}_2{}^T\mathbf{n}_{i_o}) \\ \vdots \\ -\cos(\boldsymbol{\omega}_{N_c}{}^T\mathbf{n}_{i_o}) \end{bmatrix} \qquad (28)$$

as the effective constant for a given desired contour. With these definitions, (25) becomes

$$\mathsf{G}\hat{\mathbf{s}} = \mathbf{b}. \qquad (29)$$

The least-squares solution is given by

$$\hat{\mathbf{s}} = (\mathsf{G}^T\mathsf{G})^{-1}\mathsf{G}^T\mathbf{b} \qquad (30)$$

3. 3-D EXAMPLE: ELLIPSOIDAL FREQUENCY RESPONSE

We now illustrate the usefulness of this technique by applying it to a 3-D ellipsoidal transformation function whose coefficients have a $3 \times 3 \times 3$ region of support, i.e. $M_1 = M_2 = M_3 = 1$. In Section 3.1, the explicit enumeration of the $\mathbf{n}_i$ is carried out, in order to allow calculation of (22). Section 3.2 defines the particular 3-D constraints that were chosen, and Section 3.3 derives the particular N-D cutoff contour points that are used in the least-squares solution presented in Section 3.4. A standard measure of performance is defined and evaluated in Section 3.5, demonstrating the effectiveness of the method.

3.1. Enumeration of Anti-Symmetric 3-D Indices

In order to apply the solution given by (30), it is necessary to numerically evaluate (22), which requires the 3-D contour points and the values of the anti-symmetric indices $\mathbf{n}_i = [n_1[i], n_2[i], n_3[i]]$, of which there are N_p pairs that satisfy (4), not including the origin. One *and only one* member of each $\mathbf{n}_i$ pair must be found. This process is essentially the bisection of the 3-D support of the coefficients. For this example, the task now is to enumerate all of these triplet members. It can be shown that there are 9 different ways of bisecting the $3 \times 3 \times 3$ cube of coefficients. The choice of a particular one has no effect on the solution, so it may be chosen arbitrarily. Again, the only constraint is that the reflection symmetry in (4) must be satisfied, and the origin $[0, 0, 0]$ is not included ($s[0, 0, 0]$ will be determined by (18) after (30) is calculated). The following formula is what we have chosen for this example, and is valid for any length in each dimension:

$$\begin{aligned}\sum_{i=1}^{N_p}(\cdot) \quad = \quad & \sum_{n_1=1}^{M_1}\ \sum_{n_2=-M_2}^{M_2}\ \sum_{n_3=-M_3}^{M_3}(\cdot)\\ & + \sum_{n_1=0}^{0}\ \sum_{n_2=0}^{M_2}\ \sum_{n_3=1}^{M_3}(\cdot)\ . \\ & + \sum_{n_1=0}^{0}\ \sum_{n_2=1}^{M_2}\ \sum_{n_3=-M_3}^{0}(\cdot)\end{aligned} \tag{31}$$

For our transformation function, $M_1 = M_2 = M_3 = 1$, so we can use (31) to explicitly list the $N_p = 13$ $\mathbf{n}_i$:

$$\begin{array}{lcl} \mathbf{n}_1 & = & [\ 1\ \ -1\ \ -1\ ]^T \\ \mathbf{n}_2 & = & [\ 1\ \ -1\ \ 0\ ]^T \\ \mathbf{n}_3 & = & [\ 1\ \ -1\ \ 1\ ]^T \\ \mathbf{n}_4 & = & [\ 1\ \ 0\ \ -1\ ]^T \\ \mathbf{n}_5 & = & [\ 1\ \ 0\ \ 0\ ]^T \\ \mathbf{n}_6 & = & [\ 1\ \ 0\ \ 1\ ]^T \\ \mathbf{n}_7 & = & [\ 1\ \ 1\ \ -1\ ]^T \\ \mathbf{n}_8 & = & [\ 1\ \ 1\ \ 0\ ]^T \\ \mathbf{n}_9 & = & [\ 1\ \ 1\ \ 1\ ]^T \\ \mathbf{n}_{10} & = & [\ 0\ \ 0\ \ 1\ ]^T \\ \mathbf{n}_{11} & = & [\ 0\ \ 1\ \ 1\ ]^T \\ \mathbf{n}_{12} & = & [\ 0\ \ 1\ \ -1\ ]^T \\ \mathbf{n}_{13} & = & [\ 0\ \ 1\ \ 0\ ]^T \end{array} \ . \tag{32}$$

Notice that if one combines the 13 indices listed in (32) with their respective reflections about the origin, and then includes the index at the origin $\mathbf{n}_0 = [0, 0, 0]$, then all 27 indices of the $3 \times 3 \times 3$ transformation filter are accounted for.

3.2. 3-D Constraints

For an ellipsoidally symmetric frequency response, we impose the constraint that $\boldsymbol{\theta_1} = \mathbf{0}$ and $\boldsymbol{\theta_2} = [0, 0, \pi]$, so that $\boldsymbol{\nu} = [0\ 0\ 1]^T$ and $\eta_i = n_3[i]$ is just the third element of $\mathbf{n}_i$. In addition, since $\eta_{(i=1)} = -1$ is odd, then $i_o = 1$.

3.3. Determination of 3-D Contour Points

The next step is to determine the 3-D contour points to which the transform will be fitted. Recall that an ellipsoid is the set of points $[\omega_1, \omega_2, \omega_3]$ that satisfy

$$\frac{\omega_1^2}{a_1^2} + \frac{\omega_2^2}{a_2^2} + \frac{\omega_3^2}{a_3^2} = 1, \tag{33}$$

where $[a_1, a_2, a_3]$ are the semimajor axes. For this example, the desired transformation function is an oblate ellipsoid for which two of the semimajor axes are equal, and are greater than the remaining axis. We have chosen $a_1 = a_2 > a_3$, specifically $[a_1, a_2, a_3] = [\pi/2, \pi/2, \pi/10]$. The approach is to find groups of constant-ω_3 contour points, with a constant interval between groups, effectively sampling the circumference of the ellipsoid many times at each increment in ω_3. Using spherical coordinates, this will be done by first finding the angular coordinates of all the points, then using (33) to find the radial distances.

Let N_{ω_3} be the number of constant-ω_3 planes. In order to ensure that the contour point algorithm includes the poles at $(0, 0, \pm a_3)$ and points in the (ω_1, ω_2)-plane, it must satisfy $N_{\omega_3} = 3 + 2m$, where m is any non-negative integer. Then

$$\Delta\omega_3 = \frac{2a_3}{N_{\omega_3} - 1} \tag{34}$$

is the step-size in the ω_3 direction. By symmetry, only the planes with $\omega_3 \geq 0$ need to be computed. The ω_3 coordinate of each of these planes is

$$\omega_{3l} = l\Delta\omega_3, \tag{35}$$

where $0 \leq l \leq (N_{\omega_3} - 1)/2$.

Spherical coordinates (r, θ, ϕ) will be used, where r is the distance from the origin, θ is the elevation angle with respect to the ω_3-axis and ϕ is the azimuthal angle in the (ω_1, ω_2)-plane. To preserve a four-fold sampling symmetry in each slice, an additional constraint is imposed that there be points on at least the ω_1- and ω_2-axes for each slice, except the single-sample slices at the poles. The algorithm to find these described contour points is summarized below, where L denotes the set of integers $[1, (N_{\omega_3} - 1)/2]$, and $\hat{L}$ denotes the set of integers $[1, ((N_{\omega_3} - 1)/2) - 1]$:

1.

$$\theta_l = \begin{cases} \pi/2 & \text{if } l = 0 \\ \arctan\left(a_2\sqrt{{\omega_{3l}}^{-2} - {a_3}^{-2}}\right) & \text{if } l \in L \end{cases} \tag{36}$$

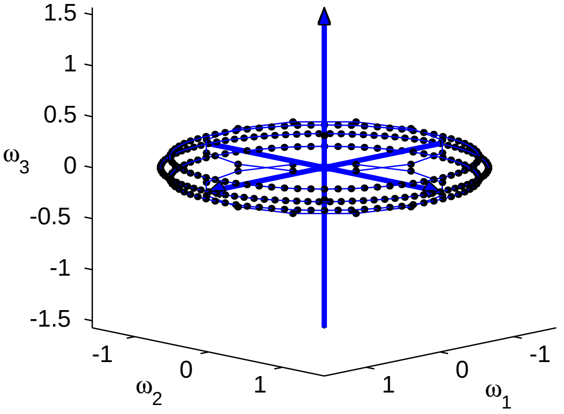

Figure 1: Contour points on surface of 3-D ellipsoid used for this example. For illustrative purposes, note the shortened axes do not extend to $\pm\pi$.

2.

$$\Delta\theta_l = \begin{cases} \theta[1] - \theta[0] & \text{if } l = 0 \\ (\theta[l+1] - \theta[l-1])/2 & \text{if } l \in \hat{L} \end{cases} \tag{37}$$

3.

$$n_{\phi_l} = \begin{cases} 4\lceil(2\pi/(4\Delta\theta_l))\rceil & \text{if } l = 0 \\ \lceil(2\pi\sin(\theta_l)/\Delta\theta_l)\rceil & \text{if } l \in \hat{L} \\ 1 & \text{if } l = \frac{N_{\omega_3}-1}{2} \end{cases} \tag{38}$$

4.

$$\Delta\phi_l = \; 2\pi/n_{\phi_l} \quad \forall l \in (\{0\} \bigcup L) \tag{39}$$

5.

$$\phi_{k,l} = \; k\Delta\phi_l \quad \begin{array}{l} \forall l \in (\{0\} \bigcup L), \\ \forall k \in [0, (n_{\phi_l} - 1)] \end{array} \tag{40}$$

6.

$$\begin{aligned} r_{k,l} \; = \; & \left[\frac{\sin^2(\theta_l)\cos^2(\phi_{k,l})}{a_1^2} \; + \; \frac{\sin^2(\theta_l)\sin^2(\phi_{k,l})}{a_2^2} \right. \\ & \left. + \; \frac{\cos^2(\theta_l)}{a_3^2}\right]^{-1/2} \end{aligned} \tag{41}$$

The operator $\lceil\cdot\rceil$ returns the smallest integer greater than or equal to its argument. For this example, we chose $N_{\omega_3} = 7$, resulting in $N_c = 262$ contour points. These are shown in Figure (1).

3.4. 3-D Least-Squares Solution

Now that we have the spherical coordinates for each contour point, they are transformed back into Cartesian coordinates, and used to calculate (26) and (28). Those results are used to compute the unknown coeffiencts via (30), (18), and (19), giving the solutions below for $s[\mathbf{n}_i]$ and ω_c:

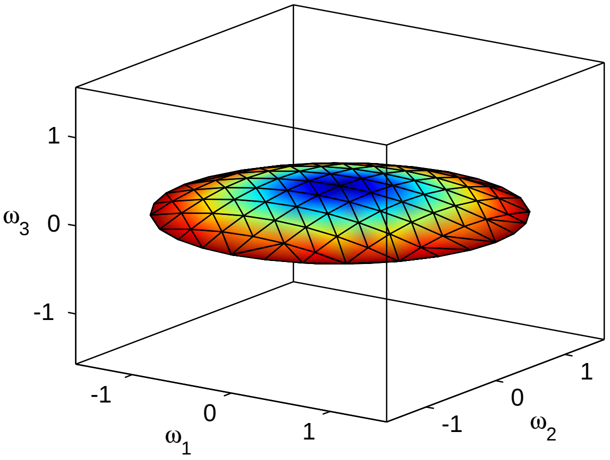

Figure 2: 3-D contour for which $\cos(\omega_c) = F(\boldsymbol{\omega})$, using the coefficients. For illustrative purposes, note the shortened axes do not extend to $\pm\pi$.

$$\begin{array}{rcr} s[\mathbf{0}] & = & 0.2545 \\ s[\mathbf{n}_1] & = & 0.0109 \\ s[\mathbf{n}_2] & = & -0.0174 \\ s[\mathbf{n}_3] & = & 0.0109 \\ s[\mathbf{n}_4] & = & 0.0309 \\ s[\mathbf{n}_5] & = & -0.0463 \\ s[\mathbf{n}_6] & = & 0.0309 \\ s[\mathbf{n}_7] & = & 0.0109 \\ s[\mathbf{n}_8] & = & -0.0174 \\ s[\mathbf{n}_9] & = & 0.0109 \\ s[\mathbf{n}_{10}] & = & 0.3327 \\ s[\mathbf{n}_{11}] & = & 0.0309 \\ s[\mathbf{n}_{12}] & = & 0.0309 \\ s[\mathbf{n}_{13}] & = & -0.0463 \end{array} \tag{42}$$

and

$$\frac{\omega_c}{2\pi} = 0.0500. \tag{43}$$

The cutoff contour surface generated by these coefficients is shown in Fig (2). The symmetries evident in the coefficients above are most likely an artifact of this particular example, in which the ellipsoid itself has a four-fold symmetry in the (ω_1, ω_2)-plane, and two-fold symmetry about both the ω_1- and ω_2-axes; the choice of $\boldsymbol{\theta_2}$ attempts to exploit that symmetry. In general however, only centro-symmetry can be expected.

3.5. Performance Metric: Deviation from Ideal Response

In order to measure the performance of this algorithm, we simply measure the root-mean-square (RMS) error ε between the desired and actual values of the transformation function. This is done by inserting the values determined in (42) and (43) into (6), which is evaluated at each contour point and subtracted from the ideal value $\cos(\omega_c)$:

$$\varepsilon \triangleq \left\{ \frac{1}{N_c} \sum_{k=1}^{N_c} [(F(\boldsymbol{\omega}_k) - \cos(\omega_c))]^2 \right\}^{1/2} \tag{44}$$

For this example, $\varepsilon = 1.7508 \times 10^{-5}$.

4. SUMMARY

We have devised and formulated a method for designing the optimal coefficients of an N-D McClellan Transformation by mapping the 1-D cutoff frequency to the N-D cutoff contour, while simultaneously determining what the optimal 1-D cutoff should be. The optimality criterion is least-squares minimization of the difference between the desired and actual transformation values along the contour points. A pair of general constraints are imposed, which, if chosen properly, mitigate the transformation scaling problem. The proposed solution is closed form, so it does not require numerical optimization techniques, such as linear or evolutionary programming, which are computationally expensive and have non-deterministic runtimes. The algorithm as described is general enough for N-D transformations for which the desired frequency response is centro-symmetric. This technique is attractive for general adaptive multi-dimensional filtering. In the next phase of our research, we will apply this method to problems in object tracking and motion parameter estimation.

REFERENCES

[1] J.M. McClellan, “The design of 2-D digital filters by transformations,” in *Proc. 7th. Annual Princeton Conference on Information Sciences and Systems*, 1973, pp. 247–251.

[2] R.M. Mersereau, W.F.G. Mecklenbrauker, and T.F. Quatieri, “McClellan transformations for two-dimensional digital filtering: Part I–design,” *IEEE Transactions on Circuits and Systems*, vol. CAS–23, no. 7, pp. 405–414, July 1976.

[3] J.M. McClellan and D.S.K. Chan, “A 2-D FIR filter structure derived from the Chebyshev recursion,” *IEEE Transactions on Circuits and Systems*, vol. CAS–24, no. 7, pp. 372–378, July 1977.

[4] S.-C. Pei and J.-J. Shyu, “Design of two-dimensional FIR digital filters by McClellan transformation and least-squares contour mapping,” *Signal Processing*, vol. 44, pp. 19–26, 1995.

[5] H.-C. Lu and K.-H. Yeh, “Optimal design of 2-D digital filters by scaling-free McClellan transformation using least-squares estimation,” *Signal Processing*, vol. 58, pp. 303–308, 1997.

[6] T. Sekiguchi and S. Takahashi, “McClellan transformations for multidimensional FIR digital filters with complex coefficients,” *Electronics and Communications in Japan*, vol. 72, no. 4, pp. 79–93, 1989.

[7] Hong Chen and G.E. Ford, “A unified eigenfilter approach to the design of two-dimensional zero-phase FIR filters with the McClellan transform,” *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 43, no. 8, pp. 622–626, August 1996.

[8] S.-C. Pei and J.-J. Shyu, “Efficient fast design of three- and higher-dimensional FIR digital filters by analytical least squares method,” *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 43, no. 4, pp. 339–344, April 1996.

[9] Lina Karam, “Two-dimensional FIR filter design by transformation,” *IEEE Transactions on Signal Processing*, vol. 27, no. 5, pp. 1474–1478, May 1999.

[10] E.Z. Psarakis and G.V. Moustakides, “Design of two-dimensional FIR filters via the generalized McClellan transform,” *IEEE Transactions on Circuits and Systems*, vol. 38, no. 11, pp. 1355–1363, November 1991.

[11] J. Xiaoying, L. Jixing, and C. Shuzhen, “The parameter determination in the design of 3-D spherically symmetric FIR filters via the McClellan transformation,” *Wuhan University Journal of Natural Sciences*, vol. 3, no. 1, pp. 65–70, 1998.

[12] T. Yoshida, A. Nishihara, and N. Fujii, “A design method for 3-dimensional band-limiting FIR filters using McClellan transformation,” *IEICE Trans. Fundamentals of Electronics Communications*, vol. E76-A, no. 8, pp. 1283–1292, August 1993.

[13] N. Nagamuthu and M.N.S. Swamy, “Scaled and scaling-free McClellan transformations for the design of 2-D digital filters,” *Journal of Circuits, Systems, and Computers*, vol. 1, no. 1, pp. 105–124, 1991.