Digital Biquad Design by Magnitude Requirements

Robin Schmidt (www.rs-met.com)

January 5, 2012

We consider the problem of finding the coefficients $b_0, b_1, b_2, a_0, a_1, a_2$ of a discrete time biquad transfer-function of the general form:

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{a_0 + a_1 z^{-1} + a_2 z^{-2}} \tag{1}$$

given 5 requirements on the magnitude response. These requirements are given as pairs of a normalized radian frequency ω_n and a desired magnitude-response value g_n which the filter should have at ω_n such that $g_n = |H(e^{j\omega_n})|$, where $n = 1, \ldots, 5$. As we shall see later, one of the biquad coefficients is actually redundant and may be set to unity. For implementation purposes, it makes most sense to normalize the transfer-function such that $a_0 = 1$. We will take care of this (re)normalization later - for the design algorithm, we have to take a_0 as variable.

From Requirements to a System of Equations

The magnitude-squared response of a general biquad transfer-function (1) is given by:

$$g^2(\omega) \hat{=} |H(e^{j\omega})|^2 = \frac{b_0^2 + b_1^2 + b_2^2 + 2b_1(b_0 + b_2)\cos(\omega) + 2b_0 b_2 \cos(2\omega)}{a_0^2 + a_1^2 + a_2^2 + 2a_1(a_0 + a_2)\cos(\omega) + 2a_0 a_2 \cos(2\omega)} \tag{2}$$

We define:

$$\begin{aligned} B_0 &\hat{=} b_0^2 + b_1^2 + b_2^2, \quad B_1 \hat{=} b_1(b_0 + b_2), \quad B_2 \hat{=} b_0 b_2 \\ A_0 &\hat{=} a_0^2 + a_1^2 + a_2^2, \quad A_1 \hat{=} a_1(a_0 + a_2), \quad A_2 \hat{=} a_0 a_2 \end{aligned} \tag{3}$$

We have 5 magnitude-requirements at 5 normalized radian frequencies given as pairs $(\omega_n, g_n),\ n = 1, \ldots, 5$. Because we will mostly deal with the squared magnitude, we'll assign a name to that squared magnitude by defining $p_n \hat{=} g_n^2$ such that may now deal with the 5 pairs (ω_n, p_n). Using the definitions (3) in (2), these 5 requirements lead to the system of 5 equations:

$$p_n = \frac{B_0 + 2B_1 \cos(\omega_n) + 2B_2 \cos(2\omega_n)}{A_0 + 2A_1 \cos(\omega_n) + 2A_2 \cos(2\omega_n)} \qquad n = 1, \ldots, 5 \tag{4}$$

Defining furthermore:

$$u_n = 2\cos(\omega_n), \qquad v_n = 2\cos(2\omega_n) \tag{5}$$

we may simplify this system to:

$$p_n = \frac{B_0 + B_1 u_n + B_2 v_n}{A_0 + A_1 u_n + A_2 v_n} \qquad n = 1, \ldots, 5 \tag{6}$$

We now fix $A_0 = 1$ and multiply both sides by the denominator - we get:

$$p_n(1 + A_1 u_n + A_2 v_n) = p_n + p_n A_1 u_n + p_n A_2 v_n = B_0 + B_1 u_n + B_2 v_n \qquad n = 1, \ldots, 5 \tag{7}$$

Bringing the terms containing A_1, A_2 over to the right hand side, this beocmes:

$$p_n = B_0 + B_1 u_n + B_2 v_n - p_n A_1 u_n - p_n A_2 v_n \qquad n = 1, \ldots, 5 \tag{8}$$

This is a linear system of 5 equations for 5 unknowns B_0, B_1, B_2, A_1, A_2. We can also write this in matrix form:

$$\begin{pmatrix} 1 & u_1 & v_1 & -p_1 u_1 & -p_1 v_1 \\ 1 & u_2 & v_2 & -p_2 u_2 & -p_2 v_2 \\ 1 & u_3 & v_3 & -p_3 u_3 & -p_3 v_3 \\ 1 & u_4 & v_4 & -p_4 u_4 & -p_4 v_4 \\ 1 & u_5 & v_5 & -p_5 u_5 & -p_5 v_5 \end{pmatrix} \begin{pmatrix} B_0 \\ B_1 \\ B_2 \\ A_1 \\ A_2 \end{pmatrix} = \begin{pmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \end{pmatrix} \tag{9}$$

This system may now be solved for our intermediate variables B_0, B_1, B_2, A_1, A_2 by standard techniques for linear systems (such as Gaussian elimination).

Computing the Biquad Coefficients

Having obtained our intermediate variables B_0, B_1, B_2, A_1, A_2, the next task is to go back to the actual biquad coefficients $b_0, b_1, b_2, a_0, a_1, a_2$. From the first line of (3), we see:

$$B_2 \hat{=} b_0 b_2 \Leftrightarrow b_2 = \frac{B_2}{b_0}, \qquad B_1 \hat{=} b_1(b_0 + b_2) \Leftrightarrow b_1 = \frac{B_1}{b_0 + b_2} = \frac{B_1}{b_0 + B_2/b_0} \tag{10}$$

and so:

$$B_0 \hat{=} b_0^2 + b_1^2 + b_2^2 = b_0^2 + \left(\frac{B_1}{b_0 + B_2/b_0} \right)^2 + \left(\frac{B_2}{b_0} \right)^2 \tag{11}$$

Bringing B_0 over to the right hand side and evaluating the squares gives:

$$0 = b_0^2 + \frac{B_1^2}{b_0^2 + 2B_2 + B_2^2/b_0^2} + \frac{B_2^2}{b_0^2} - B_0 \tag{12}$$

which we may consider as a root-finding problem for b_0^2. We must now solve this root-finding problem and take the square-root to obtain b_0 from b_0^2. Once b_0 is known, b_1 and b_2 may be computed straightforwardly by plugging b_0 back into (10). Empirically, it turns out that the root-finding problem may be solved by Newton iteration using $b_0^2 = 1$ as start value. The very same procedure can be used to compute a_0, a_1, a_2 from $A_0 = 1, A_1, A_2$. [TODO: investigate convergence properties of the Newton iteration more thoroughly]

A Normalized, Stable, Minimum-Phase Biquad

Our biquad coefficients so obtained do not necessarily lead to a stable, minimum-phase filter nor do they satisfy the $a_0 = 1$ normalization, which is usually desired for implementation purposes. Transforming the coefficient-set to a stable, minimum-phase set is done by reflecting poles and zeros outside the unit circle into the unit circle. A re-normalization to $a_0 = 1$ is straightforward by dividing all coefficients by a_0.