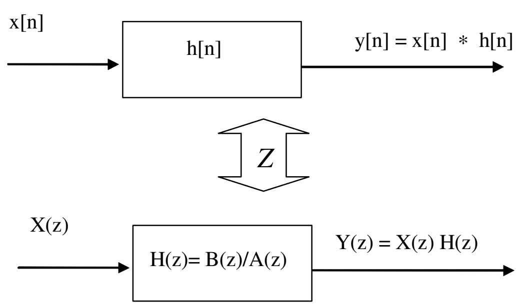
ECGR 6114
Computer Project: Pade approximation method
Student Name:___________

Part 1
Approximating a system response



In the above figure, a system is represented by its impulse response h[n] in the time domain, or by its Z-transform H(z) = B(z)/A(z) in the z-domain. As indicated above, the z-transform may be taken across the entire block diagram.

Suppose that we want to approximate a signal g[n] as the output of the above system for input signal x[n] = δ[n]. For example, we may wish to do this to find some more compact representation of g[n] that may require less memory for storage or less buts for some data transmission.

In most cases, the input signal is presumed to be the unit sample, x[n] = δ[n]. In the Pade approximation method, the focus is on finding some compact polynomial H(z) = B(z)/A(z) such that the output y[n] = h[n] approximately equals the desired signal g[n] when the input to the syatem is x[n] = δ[n].

The trick is then to solve for polnomials A(z) and B(z), given some signal g[n] to be approximated, and given that x[n] = δ[n].

Finally, note that:
1. Pade method does not guarantee a stable solution for H(z)
2. Pade exactly matches the first na + nb + 1 points of g[n], where Na and Nb are the order of the polynomials A(z) and B(z)
3. High order models are needed to match a large number of points of y[n] to g[n], unless by chance g[n] is exactly modeled by a low order H(z).
4. Items 2 and 3 might seem to imply that the pade result is equivalent to truncating g[n]. However, the Pade approximation will not generally be finite length, since the general form of the Pade approximation is IIR.

To solve for H(z), first rearrange the equatons:

H(z) = B(z)/A(z)

To

H(z)A(z) = B(z)

But H(z)=Y(z)/X(z)
and .

So,

H(z))A(z) = Y(z))A(z)/X(z) = B(z)

and since x[n] = δ[n], and therefore X(z) = 1, then

A(z))Y(z) = B(z)

and we wany y[n] to eqaul g[n], so set Y(z) = G(z)

A(z))G(z) = B(z)

Finally, taking the inverse z-transform

a[n] * g[n] = b[n]

And using the definition of convolution

$$\sum_{p=0}^{amax} a[p]\, g[n-p] = b[n]$$

Which can be written in matrix form as follows

$$\begin{pmatrix} g_0 & 0 & 0 \\ g_1 & g_0 & 0 \\ g_2 & g_1 & g_0 \\ g_3 & g_2 & g_1 \\ g_4 & g_3 & g_2 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ 0 \\ 0 \end{pmatrix}$$

Q1. Is the Pade approximation IIR?

Part 2
Using Mathcad Solve Block to solve for Pade H(z)

Mathcad can directly solve for the coefficients a1, a2, b0, b1, b2
as follows using a solve block. Set the order of numerator as Nb, order of denominator as Na where a0, a1, ... aNa, bo, b1, b2, ... bNb

$Na := 2 \qquad Nb := 2$

First define the vectors representing the numberator coefficients as b,
and denominator coefficients as a. Also, load these vectors with initial values (i.e., 0)
for the coefficients and define the desired output response g[n]

$na := 0..Na \qquad nb := 0..Nb$

$a_{na} := 0 \qquad b_{nb} := 0$

Then, define the desired output response g[n]

$$g := \begin{pmatrix} 1 \\ 0.5 \\ 0.25 \\ 00.125 \\ 0.0625 \end{pmatrix}$$

Then set up the matrix G that corresponds to the convolution g[n] * a[n]
when G is multioplied by column vector a.

$grows := rows(a) + rows(b) - 1 \qquad gcols := rows(a)$

$gr := 0..grows - 1 \qquad gc := 0..gcols - 1$

$$G_{gr,gc} := if\left[(gr - gc \geq 0) \wedge (gr - gc \leq rows(g) - 1), g_{gr-gc}, 0\right]$$

$$G = \begin{pmatrix} 1 & 0 & 0 \\ 0.5 & 1 & 0 \\ 0.25 & 0.5 & 1 \\ 0.125 & 0.25 & 0.5 \\ 0.063 & 0.125 & 0.25 \end{pmatrix}$$

Next set up the equations
and use the Mathcad solve block to find the solutionn
(Given is a keyword to start a solve block)

Given

$$G \cdot \begin{pmatrix} 1 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ 0 \\ 0 \end{pmatrix}$$

where

$$G = \begin{pmatrix} 1 & 0 & 0 \\ 0.5 & 1 & 0 \\ 0.25 & 0.5 & 1 \\ 0.125 & 0.25 & 0.5 \\ 0.063 & 0.125 & 0.25 \end{pmatrix}$$

Note: if no soultion exists using Find() below, see the Minerr() help page in the manual.

$$\text{pade\_solution} := \text{Find}(a_1, a_2, b_0, b_1, b_2)$$

$$\text{pade\_solution} = \begin{pmatrix} -0.5 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

And place the solutions back in vectors a and b

$$a_{na} := \text{if}(na = 0, 1, \text{pade\_solution}_{na-1})$$

$$a = \begin{pmatrix} 1 \\ -0.5 \\ 0 \end{pmatrix}$$

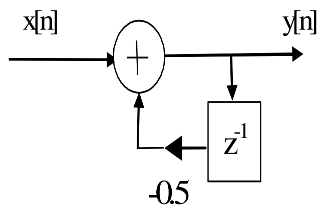
$$b_{nb} := \text{pade\_solution}_{nb+\text{rows}(a)-1}$$

$$b = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Q2. If the coefficient a2 in vecor "a" is zero, does this mean that the solution for the denominator A(z) is a first order polynomial instead of quadratic?

Q3. If the coefficients b1 and b2 in vecor "b" are zero, does this mean that the solution for the numerator B(z) is a first order polynomial?

This solution corresponds to the block diagram below::



Note that these are the coefficients for the IIR system with impulse response $(1/2)^n$ u[n] shown, with difference equation: y[n] - 0.5 y[n-1] = x[n].

The result in the example is of reduced order, i.e., the order of the solution is lower than the initial assumption, since a2 = 0.

And the system response is H(z)=N(z)/D(z)=Num(z)/Den(z) where

$$\mathrm{Num}(b,z) := \sum_{nn=0}^{\mathrm{rows}(b)-1} b_{nn}\cdot z^{-nn} \qquad \mathrm{Den}(a,z) := \sum_{nn=0}^{\mathrm{rows}(a)-1} a_{nn}\cdot z^{-nn}$$

or

$$H(a,b,z) := \frac{\displaystyle\sum_{nn=0}^{\mathrm{rows}(b)-1} b_{nn}\cdot z^{-nn}}{\displaystyle\sum_{nn=0}^{\mathrm{rows}(a)-1} a_{nn}\cdot z^{-nn}}$$

Note also, that the Pade method would give the same solution if the desired system response was either h[n] = $(1/2)^n$ u[n] or h[n] = $(1/2)^n$ { u[n] - u[n-3] }.

This is because the Pade method exactly matches the first few points of the desired response h[n]. In the case where the desired response was $(1/2)^n$ { u[n] - u[n-3] }, the pade method may result in considerable error, since the pade approximation would be $(1/2)^n$ u[n] instead of the desired $(1/2)^n$ { u[n] - u[n-3] }

Part 3
Checking stability of Pade H(z)

Next, let us see the full form of H(z)

$a\_order := rows(a) - 1$ $\quad$ $b\_order := rows(b) - 1$

$a\_order = 2$ $\quad$ $b\_order = 2$

Then the polynomials are

Use symbolic operator "->" to expand

$$A(z) := \sum_{p=0}^{rows(a)-1} a_p \cdot \left(z^{-p}\right) \qquad B(z) := \sum_{p=0}^{rows(b)-1} b_p \cdot z^{-p}$$

In some editions of Mathcad, the symbolic expansion of H(z) will not evaluate properly below if the elements of a and b are decimal (i.e., 0.5) instead of integer rational fractions (i.e., 1/2).

$$H(z) := \frac{B(z)}{A(z)} \qquad H(z) \rightarrow$$

Check the DC gain of the system

$H(1) = 2$

Find poles and zeroes

$a\_roots := polyroots(a)$ $\quad$ $a\_roots = 2$

$Hpoles := \frac{1}{a\_roots}$ $\quad$ $Hpoles = 0.5$

$b\_roots := polyroots(b)$ $\quad$ $b\_roots = ▮$

in this case we have no roots

$Hzeroes := \frac{1}{b\_roots}$ $\quad$ $Hzeroes = ▮$

Part 4
Pole-zero plot in z-plane for Pade H(z)

Lets plot poles and Zeroes in z-plane

$\text{ImagZeroes} := \text{Im}(\text{Hzeroes})$ $\text{RealZeroes} := \text{Re}(\text{Hzeroes})$

Lets just force a zero way off plot to let us plot below

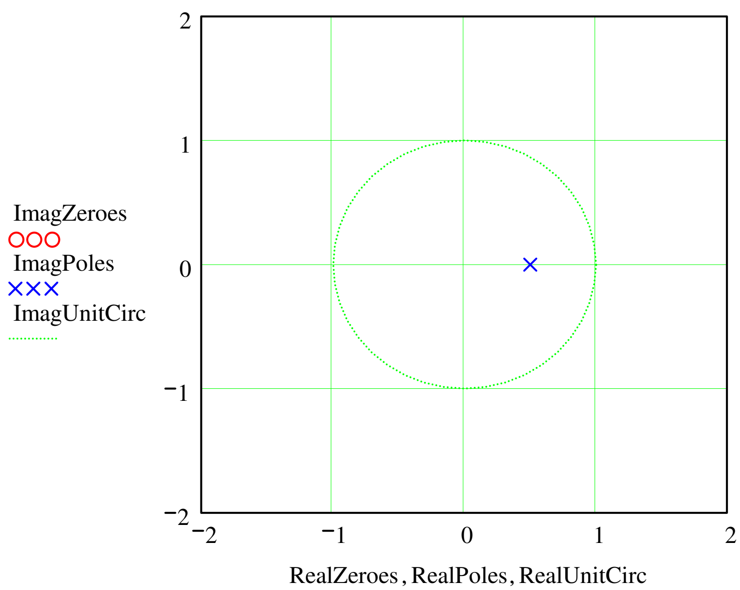
$\text{ImagZeroes} := 1000$ $\text{RealZeroes} := 1000$

$\text{ImagPoles} := \text{Im}(\text{Hpoles})$ $\text{RealPoles} := \text{Re}(\text{Hpoles})$

Lets plot a unit circle too

$\text{un} := 0..40$

$\text{ImagUnitCirc}_{\text{un}} := \sin\left(2\cdot\pi\cdot\frac{\text{un}}{40}\right)$ $\text{RealUnitCirc}_{\text{un}} := \cos\left(2\cdot\pi\cdot\frac{\text{un}}{40}\right)$



Q4. Is the above Pade approximation stable?

Part 5
Using Split G-matrix and Matrix Inverse to solve for Pade H(z)

Returning to the original problem:

$$G\cdot\begin{pmatrix}1\\a_1\\a_2\end{pmatrix} = \begin{pmatrix}b_0\\b_1\\b_2\\0\\0\end{pmatrix} \quad \text{where } g = \begin{pmatrix}1\\0.5\\0.25\\0.125\\0.063\end{pmatrix} \quad \text{and} \quad G = \begin{pmatrix}1 & 0 & 0\\0.5 & 1 & 0\\0.25 & 0.5 & 1\\0.125 & 0.25 & 0.5\\0.063 & 0.125 & 0.25\end{pmatrix}$$

The Pade method using split G-matrix and matrix inverse is:

Step 1. Solve for a1, a2 using lower portion of matrix where there are zeroes in the vector on the right hand side of the equation (below the b2 term).

Step 2. Solve for b0, b1, b2, using the top of the matrix and the solutions for a1, a2.

Pade Step 1 is to first solve for a1, a2 values.
First get the lower right G submatrix, G_br, and lower left column, g_left_col

$$G\_br := \text{submatrix}(G, b\_order + 1, \text{rows}(G) - 1, 1, \text{cols}(G) - 1)$$

$$g\_left\_col := \text{submatrix}(G, b\_order + 1, \text{rows}(G) - 1, 0, 0)$$

$$G\_br = \begin{pmatrix}0.25 & 0.5\\0.125 & 0.25\end{pmatrix} \qquad g\_left\_col = \begin{pmatrix}0.125\\0.063\end{pmatrix}$$

then we have

$$G\_br\cdot\begin{pmatrix}a_1\\a_2\end{pmatrix} = -g\_left\_col \qquad \text{or} \qquad \begin{pmatrix}0.25 & 0.5\\0.125 & 0.25\end{pmatrix}\cdot\begin{pmatrix}a_1\\a_2\end{pmatrix} = -\begin{pmatrix}0.125\\0.063\end{pmatrix}$$

The soultion (if G_br has an inverse) is

$$\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = [-(G\_br)]^{-1} \cdot g\_left\_col \qquad \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} := [-(G\_br)]^{-1} \cdot g\_left\_col$$

Unfortunately the above matrix is singular, with an infinite number of soultions since g_left_col lies in the column space of G_br.
This implies that a lower order solution could be used (i.e., a2=0)

Letting a2 become zero, the equations become

$$G\_br \cdot \begin{pmatrix} a_1 \\ 0 \end{pmatrix} = -a\_left\_col$$ or $$\begin{pmatrix} 0.25 & 0.5 \\ 0.125 & 0.25 \end{pmatrix} \cdot \begin{pmatrix} a_1 \\ 0 \end{pmatrix} = -\begin{pmatrix} 0.125 \\ 0.063 \end{pmatrix}$$

or $$G\_br^{\langle 0 \rangle} \cdot a_1 = -g\_left\_col$$ or $$\begin{pmatrix} 0.25 \\ 0.125 \end{pmatrix} \cdot a_1 = -\begin{pmatrix} 0.125 \\ 0.063 \end{pmatrix}$$

so $$a_1 := \frac{-G\_br^{\langle 0 \rangle} \cdot g\_left\_col}{G\_br^{\langle 0 \rangle} \cdot G\_br^{\langle 0 \rangle}}$$

Note: the ctrl-6 key gives the <0> superscript operator which selects the oth column of the matrix.

$a_1 = -0.5$ $\qquad a_2 := 0$

Summarizing:

$$\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} -0.5 \\ 0 \end{pmatrix}$$

Pade Step 2 is to solve for b0, b1, b2 values.
First get the top G submatrix, G_t, and solve for the upper right side of the equations

$$G\_t := \text{submatrix}(G, 0, b\_order, 0, \text{cols}(G) - 1)$$

$$G\_t = \begin{pmatrix} 1 & 0 & 0 \\ 0.5 & 1 & 0 \\ 0.25 & 0.5 & 1 \end{pmatrix}$$

or

$$\begin{pmatrix} 1 & 0 & 0 \\ 0.5 & 1 & 0 \\ 0.25 & 0.5 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix}$$

$$\begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} := G\_t \cdot \begin{pmatrix} 1 \\ a_1 \\ a_2 \end{pmatrix}$$

so

$$\begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Summarizing steps 1 and 2:

$$\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} -0.5 \\ 0 \end{pmatrix} \qquad \begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Q5. Is this the same answer as when the Mathcad Solve block was first used to solve the entire system at once.?

Part 6
Another example of Pade

Mathcad can directly solve for the coefficients a1, a2, b0, b1, b2
as follows using a solve block. Set the order of numerator as Nb, order of denominator as Na where a0, a1, ... aNa, bo, b1, b2, ... bNb

$Na := 2 \qquad Nb := 2$

First define the vectors representing the numberator coefficients as b,
and denominator coefficients as a. Also, load these vectors with initial values (i.e., 0)
for the coefficients and define the desired output response g[n]

$na := 0 .. Na \qquad nb := 0 .. Nb$

$a_{na} := 0 \qquad b_{nb} := 0$

Then, define the desired output response y[n]

$$g := \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{pmatrix}$$

Then set up the matrix G that corresponds to the convolution g[n] * a[n]
when G is multiplied by column vector a.

$grows := rows(a) + rows(b) - 1 \qquad gcols := rows(a)$

$gr := 0 .. grows - 1 \qquad gc := 0 .. gcols - 1$

$$G_{gr,gc} := \text{if}\left[(gr - gc \geq 0) \wedge (gr - gc \leq rows(g) - 1), g_{gr-gc}, 0\right]$$

$$G = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 3 & 2 & 1 \\ 4 & 3 & 2 \\ 5 & 4 & 3 \end{pmatrix}$$

Next set up the equations (Given is a keyword to start a solve block)

Given

$$G \cdot \begin{pmatrix} 1 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ 0 \\ 0 \end{pmatrix}$$

where

$$G = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 3 & 2 & 1 \\ 4 & 3 & 2 \\ 5 & 4 & 3 \end{pmatrix}$$

$$\text{Find}(a_1, a_2, b_0, b_1, b_2) = \begin{pmatrix} -2 \\ 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

Note: if no soultion exists using Find(),
see the Minerr() help page in the manual.

So here is the solution:

$$a := \begin{pmatrix} 1 \\ -2 \\ 1 \end{pmatrix} \qquad b := \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Find poles and zeroes

$$a\_roots := \text{polyroots}(a) \qquad a\_roots = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

$$Hpoles := \frac{1}{a\_roots} \qquad Hpoles = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

$$b\_roots := \text{polyroots}(b) \qquad b\_roots = \blacksquare$$

in this case we have no roots

$$Hzeroes := \frac{1}{b\_roots} \qquad Hzeroes = \blacksquare$$

Lets plot poles and Zeroes in z-plane

$ImagZeroes := Im(Hzeroes)$ $\qquad$ $RealZeroes := Re(Hzeroes)$

Lets just force a zero way off plot to let us plot below

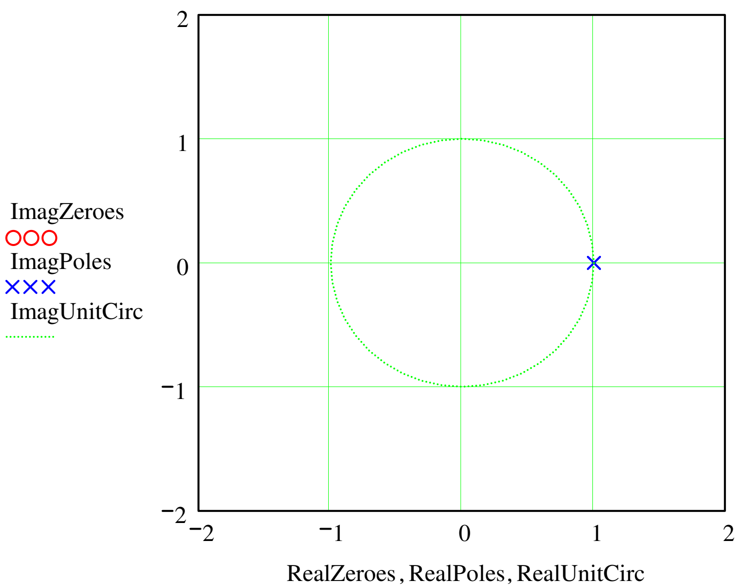
$ImagZeroes := 1000$ $\qquad$ $RealZeroes := 1000$

$ImagPoles := Im(Hpoles)$ $\qquad$ $RealPoles := Re(Hpoles)$

Lets plot a unit circle too

$un := 0..40$

$ImagUnitCirc_{un} := \sin\left(2\cdot\pi\cdot\frac{un}{40}\right)$ $\qquad$ $RealUnitCirc_{un} := \cos\left(2\cdot\pi\cdot\frac{un}{40}\right)$



Note: the pole is actually a double pole.

Q6. Is the above Pade approximation stable (g[n] = {1,2,3,4,5})?

Q7. Is the above system (g[n] = {1,2,3,4,5}) a double-accumulator (discrete-time equivalent to a double integrator)?

Q8-Q10. Change to g[n] = { 1, 2.25, 3.8125, 5.7656, 2.957, 3.134, 2.964 } in Part 6 above and find the new Pade approximation vectors "a" and "b" for Na=3 and Nb=3, and replot the poles and zeroes above (remove the forced zeroes above, RealZeroes=ImagZeroes=1000).