ECGR 6114
Computer Project: filter design
Student Name:___________

Part 1
Continuous-time Butterworth filter

The Butterworth filter is defined by

$$(|H(s)|)^2 := \frac{1}{1 + \left(\frac{s}{i \cdot \Omega c}\right)^{2 \cdot N}}$$

The poles of |H(s)|^2 are at the roorts of the denominator, Dbutter() below

$$\text{Dbutter}(s, \Omega c) := 1 + \left(\frac{s}{i \cdot \Omega c}\right)^{2 \cdot N}$$

To find the roots, first define the filter order, N, and cutoff frequency Ωc in radians/second

$N := 2$ $\quad\quad$ $\Omega c := 1$ $\quad\quad$ $Ts := \frac{1}{\Omega c}$

To use the Mathcad polyroots function,
ceate a vector that contains the coefficients of the polynomial beginning with the constant term and proceeding with increasing powers of s, i.e., a0 + a1 s + a2 s^2.

$k := 0..2 \cdot N$

$\text{DenomCoeffs}_k := 0$ $\quad\quad$ $\text{DenomCoeffs}_0 := 1$ $\quad\quad$ $\text{DenomCoeffs}_{2 \cdot N} := \left(\frac{1}{i \cdot \Omega c}\right)^{2 \cdot N}$

$\text{FilterPoles} := \text{polyroots}(\text{DenomCoeffs})$

These are then the poles of |H(s)|^2.

$$\text{FilterPoles} = \begin{pmatrix} -0.7071 + 0.7071i \\ -0.7071 - 0.7071i \\ 0.7071 + 0.7071i \\ 0.7071 - 0.7071i \end{pmatrix}$$

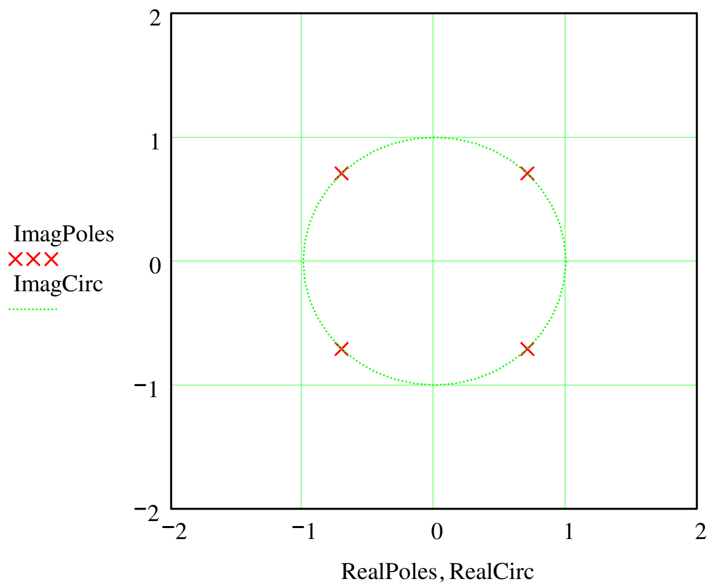
Part 2
Plotting the poles and zeroes of continuous-time filter in the s-plane

and the pole-zero plot

$ImagPoles := Im(FilterPoles)$ $RealPoles := Re(FilterPoles)$

Lets plot a unit circle too

$un := 0..40$ $ImagCirc_{un} := \Omega c \cdot \sin\left(2 \cdot \pi \cdot \frac{un}{40}\right)$ $RealCirc_{un} := \Omega c \cdot \cos\left(2 \cdot \pi \cdot \frac{un}{40}\right)$



Q1. Replot for a Chebyshev filter with N=5, ε=0.5, Ωc=1, Ts=2/Ωc.

Part 3
Finding the stable poles in the s-plane

The stable poles are in the left half plane and gibve rise to H(s) from |H(s)|^2

$$\text{StablePoles}(vv) := \left| \begin{array}{l} n \leftarrow \text{rows}(vv) \\ cc \leftarrow 0 \\ \text{for } nc \in 0..n-1 \\ \quad \left| \begin{array}{l} xx_{cc} \leftarrow vv_{nc} \ \text{ if } \operatorname{Re}(vv_{nc}) < 0 \\ cc \leftarrow cc + 1 \ \text{ if } \operatorname{Re}(vv_{nc}) < 0 \end{array} \right. \\ xx \end{array} \right.$$

$$\text{StableFilterPoles} := \text{StablePoles}(\text{FilterPoles})$$

$$\text{StableFilterPoles} = \begin{pmatrix} -0.7071 + 0.7071i \\ -0.7071 - 0.7071i \end{pmatrix}$$

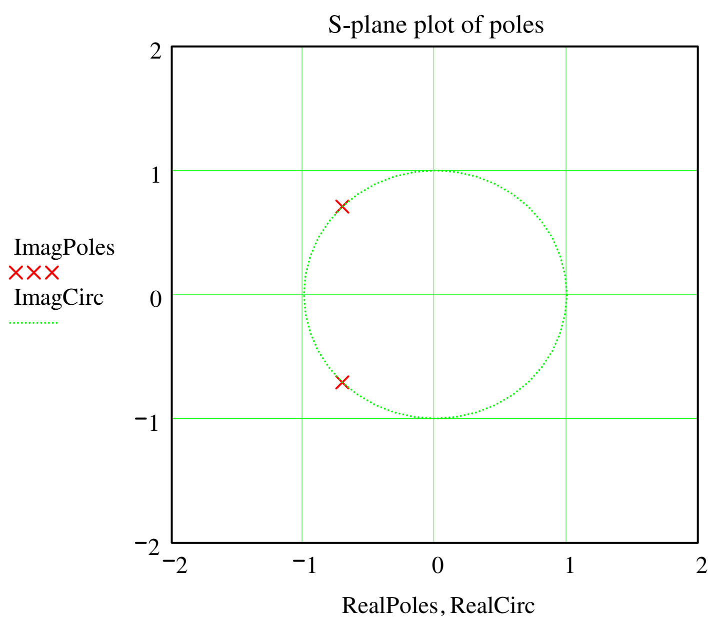
Q2. Find the stable poles for a Chebyshev filter with N=5, ε=0.5, Ωc=1, Ts=2/Ωc.

and the pole-zero plot of the stable poles

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

Lets plot a unit circle too

$un := 0..40$ $\text{ImagCirc}_{un} := \Omega c \cdot \sin\left(2 \cdot \pi \cdot \frac{un}{40}\right)$ $\text{RealCirc}_{un} := \Omega c \cdot \cos\left(2 \cdot \pi \cdot \frac{un}{40}\right)$



Q3. Replot for a Chebyshev filter with N=5, ε=0.5, Ωc=1, Ts=2/Ωc.

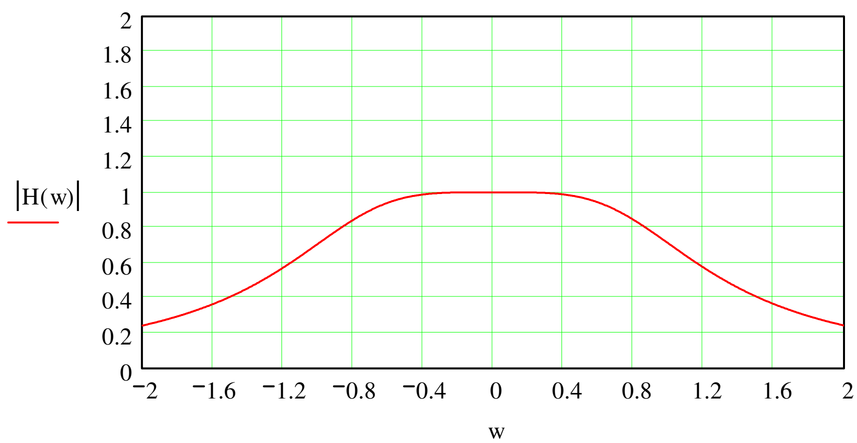
Part 4
The continuous-time filter polynomial in the s-domain

$$\text{ContinuousTimeFilter}(v,s) := \frac{\prod_{n=0}^{\text{rows}(v)-1} \left[(-v)_n\right]}{\prod_{n=0}^{\text{rows}(v)-1} \left(s - v_n\right)}$$

For some reason, the Symbolic expansion below does not work in some versions of Mathcad.

ContinuousTimeFilter(StableFilterPoles, s) complex →

$H(\Omega) := \text{ContinuousTimeFilter}(\text{StableFilterPoles}, i\cdot\Omega)$ $\quad H(\Omega c) = 1.9781 \times 10^{-9} - 0.7071i$



Q4. Replot for a Chebyshev filter with N=5, ε=0.5, Ωc=1, Ts=2/Ωc.

Part 5
Impulse response of continuous-time filter

$H(s) := \text{ContinuousTimeFilter}(\text{StableFilterPoles}, s)$

Take inverse Laplace transform along the imaginary axis

$nw := 5$

$$h(t) := \frac{1}{2\cdot\pi}\cdot\int_{-nw\cdot\Omega c}^{nw\cdot\Omega c} H(i\Omega)\, e^{i\Omega\cdot t}\, d\Omega$$

$\Omega c = 1$

Check against piecewise approximation to integral below

$npts := 200$

$$h2(t) := \sum_{k=-npts}^{npts} \frac{H\left(ik\cdot\frac{nw\cdot\Omega c}{npts}\right) e^{i\left(k\cdot\frac{nw\cdot\Omega c}{npts}\right)\cdot t}}{2\pi}\cdot\left(\frac{nw\cdot\Omega c}{npts}\right)$$

$h(0) = 0.0628 + 3.4636i \times 10^{-10}$

$h2(0) = 0.0626 + 3.4555i \times 10^{-10}$

$h(1) = 0.4646 + 1.484i \times 10^{-9}$

$h2(1) = 0.4645 + 1.4834i \times 10^{-9}$

$\frac{nw\cdot\Omega c}{npts} = 0.025$

Q5. Replot for a Chebyshev filter with N=5, ε=0.5, Ωc=1, Ts=2/Ωc.

Part 6
Partial fraction expansion of continuous-time filter design

We already have the poles of the filte, all we need is residues

```
FilterResidues(pol) := | n ← rows(pol)
                       | num ← 1
                       | for na ∈ 0..n − 1
                       |   num ← num·pol_na
                       | num ← num
                       | for nr ∈ 0..n − 1
                       |   | dd ← 1
                       |   | for nc ∈ 0..n − 1
                       |   |   dd ← dd·(pol_nr − pol_nc) if nc ≠ nr
                       |   | res_nr ← num / dd
                       | res
```

FilterRes := FilterResidues(StableFilterPoles)

$$\text{PartialFracExpan(res, poles, s)} := \sum_{k=0}^{\text{rows(poles)}-1} \frac{\text{res}_k}{\left(s - \text{poles}_k\right)}$$

PartialFracExpan(FilterRes, StableFilterPoles, s) expand, s →

$$\text{StableFilterPoles} = \begin{pmatrix} -0.7071 + 0.7071i \\ -0.7071 - 0.7071i \end{pmatrix} \qquad \text{FilterRes} = \begin{pmatrix} 1.9781 \times 10^{-9} - 0.7071i \\ -1.9781 \times 10^{-9} + 0.7071i \end{pmatrix}$$

Q6. Find the FilterRes (Filter residues) for a Chebyshev filter with N=5, ε=0.5, Ωc=1, Ts=2/Ωc.

$$\text{PartialFracExpan}(\text{FilterRes}, \text{StableFilterPoles}, i\cdot\Omega c) = 1.9781 \times 10^{-9} - 0.7071i$$

$$\text{PartialFracExpan(FilterRes, StableFilterPoles, 0)} = 1$$

Part 7
Impulse-invariance filter design

Using the partial fraction expansion form, it is simple to created the impulse-invariance filter.
Given the partial fraction expansion of H(s) as follows:

$$H(s) := \sum_{k=0}^{NumberOfPoles-1} \frac{res_k}{\left(s - poles_k\right)}$$

The impulse-invariance filter design in the z-domain is:

$$H(s) := \sum_{k=0}^{NumberOfPoles-1} \frac{Ts \cdot res_k}{\left(1 - z^{-1} e^{Ts \cdot poles_k}\right)}$$

So define a function to create the impulse invariance design:

$$ImpInvarDesign(res, poles, z, Ts) := \sum_{k=0}^{rows(poles)-1} \frac{Ts \cdot res_k}{\left(1 - z^{-1} e^{Ts \cdot poles_k}\right)}$$

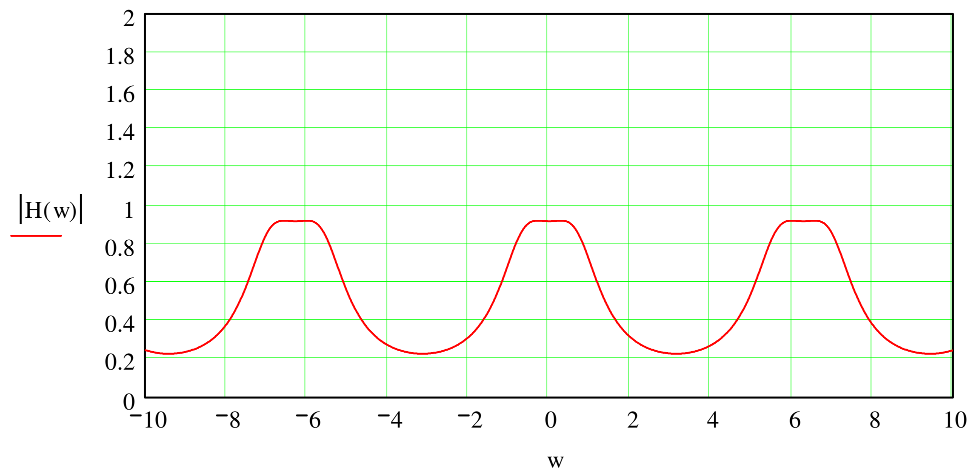
$$Himpinv(z) := ImpInvarDesign(FilterRes, StableFilterPoles, z, Ts)$$

$$Himpinv(1) = 0.9181 - 4.5017i \times 10^{-10}$$

$$H(\omega) := Himpinv\left(e^{i \cdot \omega}\right)$$

Frequency response of impulse-invariance design

$\Omega c = 1 \qquad fc := \frac{\Omega c}{2\pi} \qquad fc = 0.1592 \qquad Ts = 1 \qquad fs := \frac{1}{Ts} \qquad fs = 1 \qquad \frac{fs}{fc} = 6.2832$



Q7. Replot for a Chebyshev filter with N=5, ε=0.5, Ωc=1, Ts=2/Ωc.

Part 8
Bilinear transform filter design

The bilinear design is taken by substituting s= (2/Ts)(z-1)/(z+1)

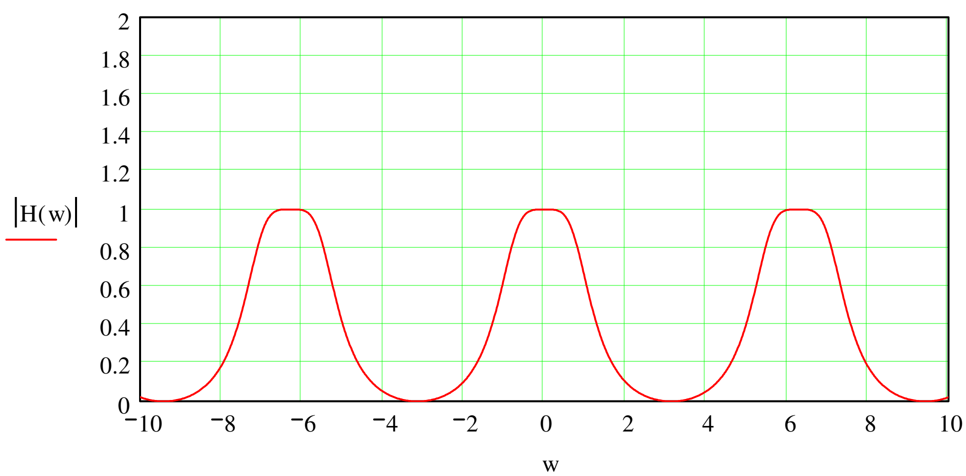
$$\text{BilinearDesign}(res, poles, z, Ts) := \sum_{k=0}^{rows(poles)-1} \frac{res_k}{\left[\frac{2\cdot(z-1)}{Ts\cdot(z+1)} - poles_k\right]}$$

$$\text{Hbilin}(z) := \text{BilinearDesign}(\text{FilterRes}, \text{StableFilterPoles}, z, Ts)$$

$$\text{Hbilin}(1) = 1$$

$$\text{Hbilin}\left(e^{i\cdot 2\pi\cdot\frac{fc}{fs}}\right) = -0.0799 - 0.6372i$$

$$H(\omega) := \text{Hbilin}\left(e^{i\cdot\omega}\right)$$



Q8. Replot for a Chebyshev filter with N=5, ε=0.5, Ωc=1, Ts=2/Ωc.

The partial fraction expanson can be rewritten as

$$\mathrm{BilinearDesign}(res, poles, z, Ts) := \sum_{k=0}^{rows(poles)-1} \frac{1}{2 - Ts \cdot poles_k} \cdot \frac{res_k \cdot Ts \cdot (z+1)}{z - \frac{2 + Ts \cdot poles_k}{2 - Ts \cdot poles_k}}$$

or can be rewriten as

$$\mathrm{BilinearDesign}(res, poles, z, Ts) := (z+1) \cdot \sum_{k=0}^{rows(poles)-1} \frac{res_k \cdot Ts}{2 - Ts \cdot poles_k} \cdot \frac{1}{z - \frac{2 + Ts \cdot poles_k}{2 - Ts \cdot poles_k}}$$

$$\mathrm{Hbilin}(z) := \mathrm{BilinearDesign}(FilterRes, StableFilterPoles, z, Ts)$$

$$\mathrm{Hbilin}(1) = 1$$

$$\mathrm{Hbilin}\left(e^{i \cdot 2\pi \cdot \frac{fc}{fs}}\right) = -0.0799 - 0.6372i$$

hence, the Bilinear poles are at

$$n := 0 .. N - 1$$

$$BilinPoles_n := \frac{2 + Ts \cdot StableFilterPoles_n}{2 - Ts \cdot StableFilterPoles_n} \qquad BilinPoles = \begin{pmatrix} 0.3832 + 0.3613i \\ 0.3832 - 0.3613i \end{pmatrix}$$

Part 9
Expressing the bilinear transform filter as H(z) = Num(z) / Den(z)

As the first step,
the BilinNum() function below returns the numerator polynomial coefficients Num(z), in increasing powers of z for the bilinear filter response Hbilin(z)=Num(z)/Den(z).
The arguments are: rez=residues of partial fraction expansion of the continuous time filter H(s)
poles= poles of partial fraction expansion of the continuous time filter H(s)
Ts= sampling period of the discrete-time system

```
BilinNum(rez, poles, Ts) :=  | n ← rows(rez)
                             | num ← 0
                             | for na ∈ 0..n − 1
                             |    | res_na ← (rez_na·Ts) / (2 − Ts·poles_na)
                             |    | pol_na ← (2 + Ts·poles_na) / (2 − Ts·poles_na)
                             | for na ∈ 0..n
                             |    nx_na ← 0
                             | dummya ← 1
                             | for nr ∈ 0..n − 1
                             |    | nx ← nx·0
                             |    | ny ← nx·0
                             |    | flg ← 0
                             |    | cnt ← 0
                             |    | dummya ← 1
                             |    | for np ∈ 0..n − 1
                             |    |    | ny ← ny·0
                             |    |    | ny ← ny·0
                             |    |    | nx_0 ← −(pol_np)  if (np + nr ≠ 0) ∧ (flg = 0)
                             |    |    | nx_1 ← 1  if (np + nr ≠ 0) ∧ (flg = 0)
                             |    |    | flg ← 1  if (np + nr ≠ 0) ∧ (flg = 0)
                             |    |    | cnt ← cnt + 1  if (flg > 0) ∧ (np ≠ nr)
                             |    |    | cnta_(nr, np) ← cnt
                             |    |    | for nc ∈ 1..cnt  if (np ≠ nr) ∧ (flg = 2)
                             |    |    |    ny_nc ← nx_(nc−1)
```

```
            for  nc ∈ 1 .. cnt                              if  (np ≠ nr) ∧ (flg = 2)
              nx_{nc-1} ← ny_{nc-1} + -(pol_np)·ny_nc
            nx_cnt ← ny_cnt   if  (np ≠ nr) ∧ (flg = 2)
            flg ← 2   if  (flg = 1)
        nz_nr ← nx·res_nr
        num ← num + nz_nr
    ny ← ny·0
    numa ← num
    nx ← num
    for  nc ∈ 1 .. n
      ny_nc ← nx_{nc-1}
    for  nc ∈ 0 .. n
      nx_nc ← nx_nc + ny_nc
    num ← nx
    num
```

$$\mathrm{BilinNum}(\mathrm{FilterRes}, \mathrm{StableFilterPoles}, \mathrm{Ts}) = \begin{pmatrix} 0.1277 + 4.943i \times 10^{-10} \\ 0.2555 + 9.8859i \times 10^{-10} \\ 0.1277 + 4.943i \times 10^{-10} \end{pmatrix}$$

Next, clean up the numerical errors that give rise to the negligibly small imaginary parts in the coefficients:

$$\mathrm{num} := \mathrm{Re}(\mathrm{BilinNum}(\mathrm{FilterRes}, \mathrm{StableFilterPoles}, \mathrm{Ts})) \qquad \mathrm{num} = \begin{pmatrix} 0.1277 \\ 0.2555 \\ 0.1277 \end{pmatrix}$$

Q9. Find num, the numerator coefficients of H(z) for a Chebyshev filter with N=5, ε=0.5, Ωc=1, Ts=2/Ωc.

Finally, the numerator Num(z) of Hbilin(z) = Num(z) / Den(z) is

$$\mathrm{numerator}(z) := \sum_{k=0}^{\mathrm{rows(num)}-1} \mathrm{num}_k \cdot z^k \qquad \mathrm{numerator}(x) \text{ expand}, x \rightarrow$$

Next,
the BilinDenom() function below returns the denominator polynomial coefficients Den(z), in increasing powers of z for the bilinear filter response Hbilin(z)=Num(z)/Den(z).
The arguments are: rez=residues of partial fraction expansion of the continuous time filter H(s)
poles= poles of partial fraction expansion of the continuous time filter H(s)
Ts= sampling period of the discrete-time system

```
BilinDenom(rez, poles, Ts) := | n ← rows(rez)
                              | den ← 0
                              | cnt ← 0
                              | for na ∈ 0..n − 1
                              |   | res_na ← (rez_na·Ts) / (2 − Ts·poles_na)
                              |   | pol_na ← (2 + Ts·poles_na) / (2 − Ts·poles_na)
                              | for na ∈ 0..n
                              |   nx_na ← 0
                              | ny ← nx
                              | for np ∈ 0..n − 1
                              |   | ny ← ny·0
                              |   | ny ← ny·0
                              |   | nx_0 ← −(pol_np)  if np = 0
                              |   | nx_1 ← 1  if np = 0
                              |   | cnt ← cnt + 1
                              |   | cnta_np ← cnt
                              |   | for nc ∈ 1..cnt    if cnt > 1
                              |   |   ny_nc ← nx_nc−1
                              |   | for nc ∈ 1..cnt                         if cnt > 1
                              |   |   nx_nc−1 ← ny_nc−1 + −(pol_np)·ny_nc
                              |   | nx_cnt ← ny_cnt  if cnt > 1
                              | den ← nx
                              | den
```

$$den := BilinDenom(FilterRes, StableFilterPoles, Ts)$$

$$den = \begin{pmatrix} 0.2774 - 7.7467i \times 10^{-10} \\ -0.7664 + 2.7519i \times 10^{-9} \\ 1 \end{pmatrix}$$

Again, clean up the numerical errors that give rise to the negligibly small imaginary parts in the coefficients:

$$den := Re(BilinDenom(FilterRes, StableFilterPoles, Ts))$$

$$den = \begin{pmatrix} 0.2774 \\ -0.7664 \\ 1 \end{pmatrix}$$

Q10. Find den, the denomonator coefficients of H(z) for a Chebyshev filter with N=5, ε=0.5, Ωc=1, Ts=2/Ωc.

Finally, the denominator Den(z) of Hbilin(z) = Num(z) / Den(z) is

$$denominator(z) := \sum_{k=0}^{rows(den)-1} den_k z^k$$

$den(x)$ expand, $z \rightarrow$

After all the calculations,
there is potential for numerical error that could lead to instability,
so check that the poles of the bilinar transform filter are stable

$$polecheck := polyroots(den) \qquad polecheck = \begin{pmatrix} 0.3832 - 0.3613i \\ 0.3832 + 0.3613i \end{pmatrix}$$

Check the magnitude of the complex poles.

$$\overrightarrow{|polecheck|} = \begin{pmatrix} 0.5267 \\ 0.5267 \end{pmatrix}$$

Q11. Is the filter stable? Why?

Finally, we have Hbilin(z)=Num(z)/Den(z)

$$Hbilin(num, den, z) := \frac{\displaystyle\sum_{k=0}^{rows(num)-1} num_k \cdot z^k}{\displaystyle\sum_{k=0}^{rows(den)-1} den_k \cdot z^k}$$

So check a few important points:

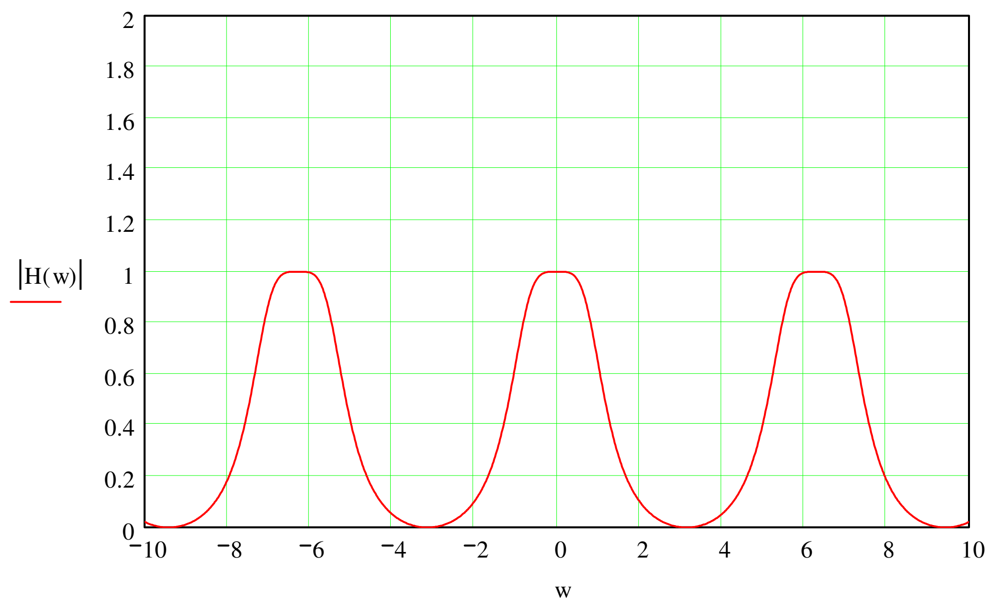
$$Hbilin(num, den, 1) = 1 \qquad fc = 0.1592$$

$$Hbilin(num, den, -1) = 0 \qquad fs = 1$$

$$Hbilin\left[num, den, e^{i\cdot\left(2\cdot\pi\cdot\frac{fc}{fs}\right)}\right] = -0.0799 - 0.6372i$$

Plot the frequency reponse of the bilinear transform filter

$$H(\omega) := Hbilin\left(num, den, e^{i\cdot\omega}\right)$$



Q12. Replot for a Chebyshev filter with N=5, ε=0.5, Ωc=1, Ts=2/Ωc.

Part 10
Plotting the poles and zeroes of bilinear transform filter in the z-plane

and the pole-zero plot

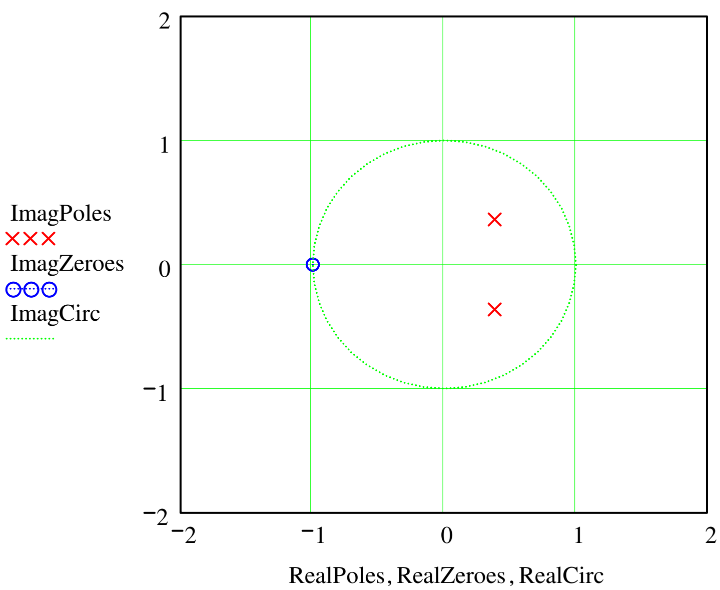
$$\text{polyroots(num)} = \begin{pmatrix} -1 \\ -1 \end{pmatrix} \qquad \text{polyroots(den)} = \begin{pmatrix} 0.3832 - 0.3613i \\ 0.3832 + 0.3613i \end{pmatrix}$$

$$\text{ImagPoles} := \text{Im(polyroots(den))} \qquad \text{RealPoles} := \text{Re(polyroots(den))}$$

$$\text{ImagZeroes} := \text{Im(polyroots(num))} \qquad \text{RealZeroes} := \text{Re(polyroots(num))}$$

Lets plot a unit circle too

$$un := 0..40 \qquad \text{ImagCirc}_{un} := \sin\left(2\cdot\pi\cdot\frac{un}{40}\right) \qquad \text{RealCirc}_{un} := \cos\left(2\cdot\pi\cdot\frac{un}{40}\right)$$



Q13. Replot for a Chebyshev filter with N=5, ε=0.5, Ωc=1, Ts=2/Ωc.

Part 11
Experiment with effects of coefficient precision on the performance of the bilinear transform filter

Experiment using finite precision coefficients in the filter, using the variable "precision"

$$\text{precision} := 1000$$

Use precision to control truncation of the coefficients

$$\text{min\_num} := \overrightarrow{\min(|\text{num}|)} \qquad \text{min\_num} = 0.1277$$

$$\text{numscale} := \text{round}\left(\frac{\text{precision}}{\text{min\_num}}, 0\right) \qquad \text{numscale} = 7.828 \times 10^{3}$$

$$k := 0 .. \text{rows(num)} - 1$$

$$\text{scaled\_num}_k := \text{round}\left(\text{num}_k \cdot \text{numscale}, 0\right)$$

$$\text{min\_den} := \overrightarrow{\min(|\text{den}|)} \qquad \text{min\_den} = 0.2774$$

$$\text{denscale} := \text{round}\left(\frac{\text{precision}}{\text{min\_den}}, 0\right) \qquad \text{denscale} = 3.605 \times 10^{3}$$

$$k := 0 .. \text{rows(den)} - 1$$

$$\text{scaled\_den}_k := \text{round}\left(\text{den}_k \cdot \text{denscale}, 0\right)$$

$$\text{scaled\_num} = \begin{pmatrix} 1 \times 10^{3} \\ 2 \times 10^{3} \\ 1 \times 10^{3} \end{pmatrix} \quad \text{nscale} = 7.828 \times 10^{3} \qquad \text{scaled\_den} = \begin{pmatrix} 1 \times 10^{3} \\ -2.763 \times 10^{3} \\ 3.605 \times 10^{3} \end{pmatrix} \quad \text{le} = 3.605 \times 10^{3}$$

$$\text{HbilinPrecision}(\text{s\_num}, \text{s\_den}, \text{nscale}, \text{dscale}, z) := \frac{\text{dscale}}{\text{nscale}} \cdot \frac{\displaystyle\sum_{k=0}^{\text{rows(s\_num)}-1} \text{s\_num}_k \cdot z^{k}}{\displaystyle\sum_{k=0}^{\text{rows(s\_den)}-1} \text{s\_den}_k \cdot z^{k}}$$

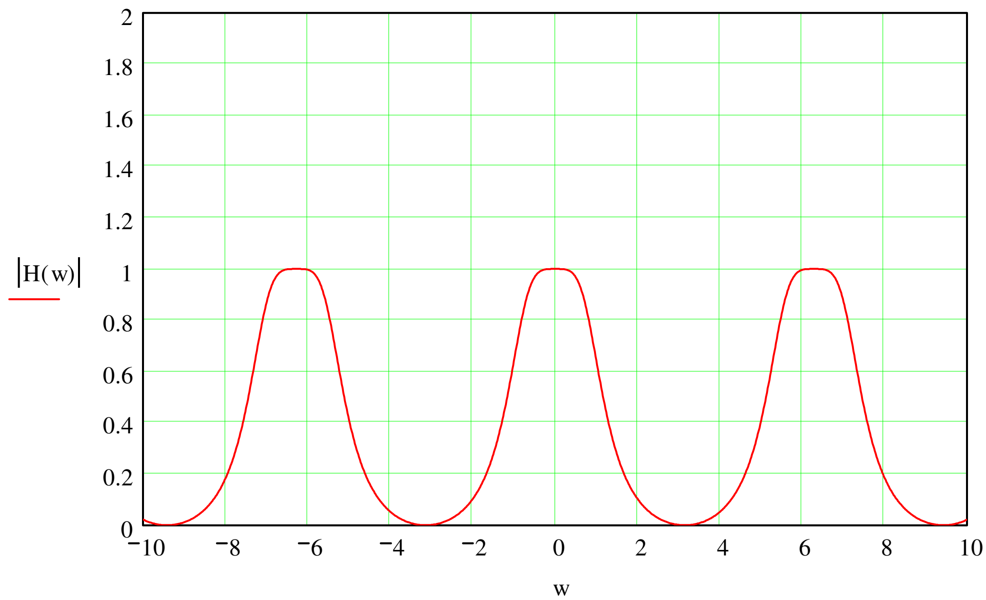
HbilinPrecision(scaled_num, scaled_den, numscale, denscale, z) →

HbilinPrecision(scaled_num, scaled_den, numscale, denscale, z) →

HbilinPrecision(scaled_num, scaled_den, numscale, denscale, 1) = 1.0001

$$H(\omega) := \mathrm{Hbilin}\left(\mathrm{num}, \mathrm{den}, e^{i\cdot\omega}\right)$$

$$H(\omega) := \mathrm{HbilinPrecision}\left(\mathrm{scaled\_num}, \mathrm{scaled\_den}, \mathrm{numscale}, \mathrm{denscale}, e^{i\cdot\omega}\right)$$



Q14. Replot for a Chebyshev filter with N=5, ε=0.5, Ωc=1, Ts=2/Ωc.

$$\mathrm{polyroots}(\mathrm{den}) = \begin{pmatrix} 0.3832 - 0.3613i \\ 0.3832 + 0.3613i \end{pmatrix} \qquad \mathrm{polyroots}(\mathrm{scaled\_den}) = \begin{pmatrix} 0.3832 + 0.3613i \\ 0.3832 - 0.3613i \end{pmatrix}$$

Q15. Reset the precision parameter to 10. With this setting, by what percenage did the poles of the filter change from their original value.

Appendix
Continuous-time Chebyshev filter

The Chebyshev filter is defined by

$$(|H(s)|)^2 := \frac{1}{1 + \left(\varepsilon \cdot Cn\left(\frac{s}{i \cdot \Omega c}\right)\right)^2}$$

where Cn(x) is the Chebyshev polynonial of the first kind of order n.
The poles of |H(s)|^2 are at the roorts of the denominator, Dbutter() below

$$\mathrm{DenCheby}(s, \Omega c) := 1 + \left(\varepsilon \cdot Cn\left(\frac{s}{i \cdot \Omega c}\right)\right)^2$$

To find the roots, first define the filter order, N, and cutoff frequency Ωc in radians/second

$N := 4$ $\quad \Omega c := 1$ $\quad Ts := \frac{1}{\Omega c}$ $\quad \varepsilon := 0.5$

The parameter ε determines minima of the passband ripple of $|H(\Omega)|^2$

$$\mathrm{minPassband} := \frac{1}{\left(1 + \varepsilon^2\right)} \qquad \mathrm{minPassband} = 0.8$$

The minima of the passband ripple of |H(Ω)| is

$$\sqrt{\mathrm{minPassband}} = 0.8944$$

The passband ripple in dB is

$$-10 \log(\mathrm{minPassband}) = 0.9691$$

the Chebyshev polynomials are recursively defined.
$C_0(x)=1$
$C_1(x)=x$
$C_n(x)=2xC_{n-1}(x) - C_{n-2}(x)$

The following code snippet works for n=2,
fix it to work for any value of n.
DenCheby(n,ε) computes the entire Chebyshev filter denominator:
DenCheby(n,ε) = 1 + [ε Cn(s/iΩc)]^2

```
DenCheby(n, eps) := | for  na ∈ 0..n
                    |   x_na ← 0
                    | pa ← x
                    | pb ← x
                    | p ← x
                    | pa_0 ← 1
                    | pb_1 ← 1
                    | ny ← 0·pa
                    | nx ← pa
                    | nz ← pb
                    | for  nc ∈ 0..n − 1
                    |   ny_(nc+1) ← 2·nz_nc
                    | ny ← ny − nx
                    | p ← ny
                    | pa ← pb
                    | pb ← ny
                    | for  na ∈ 0..n·2
                    |   x_na ← 0
                    | for  na ∈ 0..n
                    |   | ny ← 0·x
                    |   | for  nc ∈ 0..n
                    |   |   | ny_(nc+na) ← p_na·p_nc
                    |   |   | x_(nc+na) ← x_(nc+na) + ny_(nc+na)
                    | x ← x·eps·eps
                    | x_0 ← x_0 + 1
                    | for  na ∈ 0..n·2
                    |   x_na ← i^na·x_na
                    | x
```

The coefficients of the denominator polynomial of the chebyshev filter is then

$$\text{den} := \text{DenCheby}(2,\varepsilon)$$

$$\text{den} = \begin{pmatrix} 1.25 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix}$$

where

$$\text{denominator}(s) := \sum_{k=0}^{\text{rows(den)}-1} \text{den}_k\, s^k$$

$$\text{denominator}(s) \rightarrow$$

To use the Mathcad polyroots function,
ceate a vector that contains the coefficients of the polynomial beginning with the constant term and proceeding with increasing powers of s, i.e., a0 + a1 s + a2 s^2.

$$\text{FilterPoles} := \text{polyroots}(\text{DenCheby}(N,\varepsilon))$$

$$\text{FilterPoles} = \begin{pmatrix} -0.5559 + 0.8995i \\ -0.5559 - 0.8995i \\ 0.5559 + 0.8995i \\ 0.5559 - 0.8995i \end{pmatrix}$$

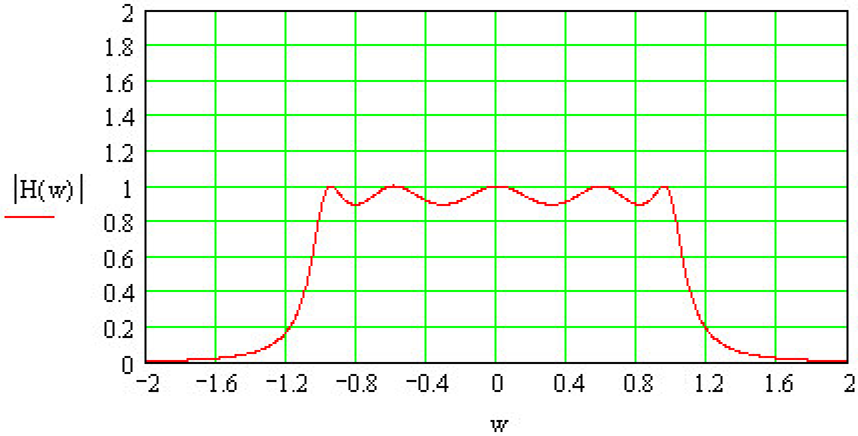
To test your code, for N=4, Ωc =1, and ε=0.5 the poles of |H(s)|^2 should be.

$$\begin{pmatrix} -0.3407 + 0.4079i \\ -0.3407 - 0.4079i \\ -0.1411 + 0.9847i \\ -0.1411 - 0.9847i \\ 0.1411 - 0.9847i \\ 0.1411 + 0.9847i \\ 0.3407 - 0.4079i \\ 0.3407 + 0.4079i \end{pmatrix}$$

To test your code, an example Chebyshev continuous time filter would be

$H(\Omega) := \text{ContinuousTimeFilter}(\text{StableFilterPoles}, i \cdot \Omega)$

$\Omega c = 1$ $\quad N = 5$ $\quad \varepsilon = 0.5$



To test your code, an example Chebyshev impulseinvariance filter is:

Frequency response of impulse-invariance design

$H(\omega) := \text{Himpinv}\left(e^{i \cdot \omega}\right)$ $\quad \Omega c = 1$ $\quad N = 5$ $\quad \varepsilon = 0.5$ $\quad Ts = 1$