Quadrotor: Classical Roll and Pitch Axis Control

Mike Timmons, www.mtwallets.com

October, 2018

Ignoring Gyroscopic effects of rigid body and propellers, and with the modified, linearized propeller speed equation we arrive at a roll-axis transfer function and design a lead compensator.

Loading Units:-Standard

$with(DynamicSystems):$

$with(inttrans):$

Modelling Parameters

Motor Terms

Assume a motor advertized with Kv~2000 RPM/V.

$$\textbf{Motor Torque Constant} \quad K_t := 10^{-2}\,\frac{\mathrm{N\,m}}{\mathrm{A}}:$$

Here is an interesting equality better explained in sources you can look up...

$$\textbf{Motor Back EMF Constant} \quad K_b := 10^{-2}\,\frac{\mathrm{V}}{\frac{1}{\mathrm{s}}}:$$

Internal resistance of motor (safe assumption):

$$R := 1\,\frac{\mathrm{V}}{\mathrm{A}}:$$

Propeller Terms

Thrust Factor

$$b := \frac{C_t\,\rho\,\mathrm{D}_p^{\,4}}{4\,\pi^2}:$$

Drag Factor

$$d := \frac{C_p\,\rho\,\mathrm{D}_p^{\,5}}{8\,\pi^3}:$$

Where these power and torque coefficients are unitless (if you cancel terms), but resulting calculated units work out if we leave in this form

$$C_p := 0.7\,\frac{\mathrm{N\,s}^2}{\mathrm{kg\,m}}:$$

$$C_t := 0.1\,\frac{\mathrm{N\,s^2}}{\mathrm{kg\,m}}:$$

$$\rho := 1.225\,\frac{\mathrm{kg}}{\mathrm{m^3}}:$$

$$D_p := 0.1\,\mathrm{m}:$$

This gets us to thrust factor, 'b':
evalf(*b*)

$$3.102961248\;10^{-7}\ \mathrm{N\ s^2} \tag{1}$$

And drag factor, 'd':
evalf(*d*)

$$3.456961346\;10^{-8}\ \mathrm{N\ s^2\ m} \tag{2}$$

Gearbox

No gearbox, so:

$$r := 1:$$
$$\eta := 1:$$

Body Moment of Inertia

For now model as a dumbell with 100-gram motor-propeller assemblies

$$m_m := 0.1\ \mathrm{kg}:$$
$$l_a := 0.1\,\mathrm{m}:$$
$$I_x := 2\,m_m l_a^{\,2}$$

$$I_x := 0.002\ \mathrm{kg\ m^2} \tag{3}$$

Motor-Propeller Moment of Inertia

Model the Prop as a 2 gram spinning rod. This will over-estimate J_P because in fact is has most mass concentrated nearer to hub.

$$m_p := 0.002\,\mathrm{kg}:$$

$$J_p := \frac{1}{12}\,m_p D_p^{\,2}:$$

Model spinning motor shaft and rotor as spinning cylinder. Assign half motor weight to this spinning core. That's likely too much but a safe estimate.

$m_{motor} := 0.03\,\mathrm{kg}$: $m_{rotor} := \frac{m_{motor}}{2}$: $r_{rotor} := 0.01\,\mathrm{m}$: $J_m := \frac{1}{2} m_{rotor} r_{rotor}^{\;2}$:

Let's see if motor or its propeller dominate the assembly's moment of intertia...

$$\frac{J_p}{J_m}$$

$$2.222222223 \tag{4}$$

Looks like the prop dominates, but the rotor+shaft not negligible, so lump the total as Jt...

$$J_t := J_p + J_m :$$

Initial conditions

Nominal propeller speed at hover

$$\omega_0 := 100\,\mathrm{s}^{-1} :$$

Motor Speed Equation Parameters

$$A := convert\left(\left(\frac{K_t \cdot K_b}{R \cdot J_t} + \frac{d\,\omega_0}{\eta \cdot r^3 \cdot J_t}\right), unit\_free\right)$$

$$A := 42.80977710 \tag{5}$$

$$B := convert\left(\frac{K_t}{R\,J_t}, unit\_free\right)$$

$$B := 4137.931034 \tag{6}$$

'A' term relative assessment: motor parameters or drag term?

$$\frac{\frac{K_t \cdot K_b}{R \cdot J_t}}{\frac{d\,\omega_0}{\eta \cdot r^3 \cdot J_t}}$$

$$28.92713859 \tag{7}$$

Although confusing, if you expand the units and recall that 'A' multiplies an ω and 'B' multiplies command voltage input, then each term contributes a $\frac{1}{s^2}$: as we expect for the differential equation.

Transfer Function for Roll or Pitch

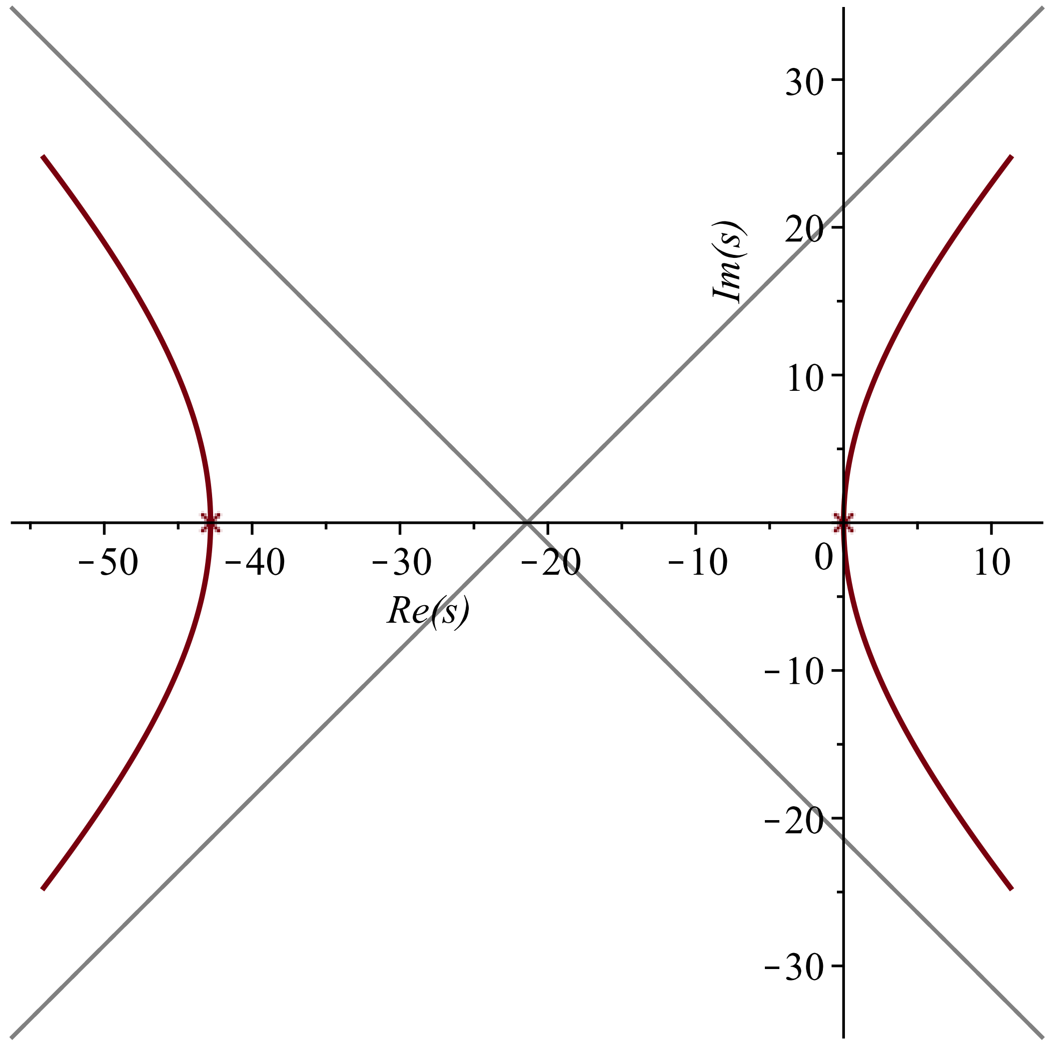
Plant gain

$$K_g := convert\left(\frac{l_a\, b\, B^2}{I_x}, unit\_free\right)$$

$$K_g := 265.6518546 \tag{8}$$

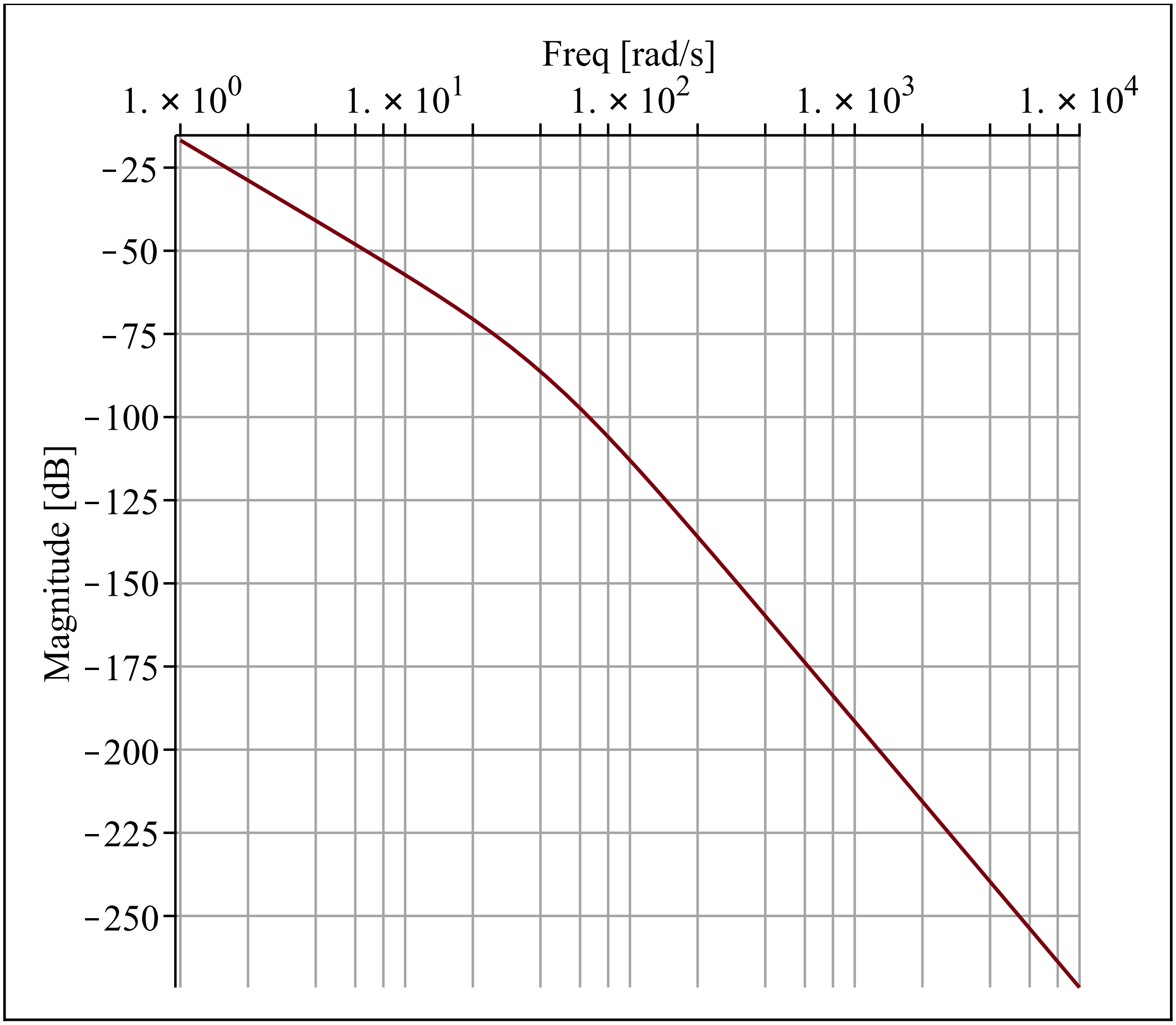
$$G := TransferFunction\left(\frac{K_g}{s^2 \cdot (s+A)^2}\right) :$$

$$RootLocusPlot(G, 0\, ..10000)$$

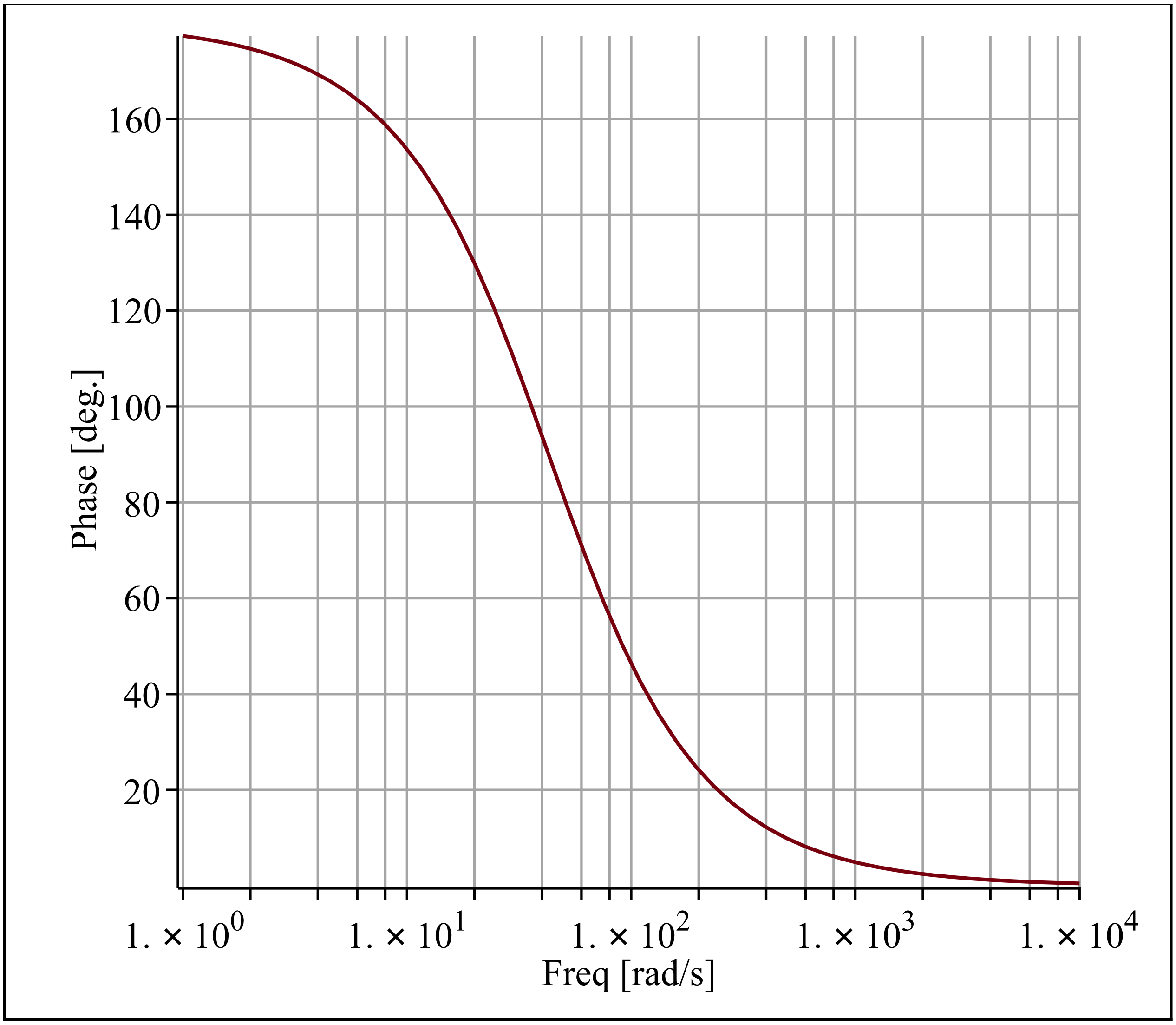


We can see above that our double-pole at zero is immediately unstable, and worse-so with any gain as the loci head-off into the righ-half plane of the proportional gain (Plant) root locus. We need to stablaize the platform.

$$BodePlot(G)$$



Freq [rad/s]
1. × 10^0
1. × 10^1
1. × 10^2
1. × 10^3
1. × 10^4
−25
−50
−75
−100
−125
−150
−175
−200
−225
−250
Magnitude [dB]



Plant is unstable: Need a Lead Compensator

The plant transfer function exhibits a classic, "double integrator" for the platform subject to thrust from the opposing propellers. To control such a plant, a lead compensator is employed to provide a phase-margin, "hump" at a desired, practical unity gain crossover frequency.

"Practical" can be a bit vaque, for example a textbook problem for satellite attitude control typically presents a space capsule with thrusters. The thrusters are not typically modelled, so we solve the textbook problem assuming we have immediate thrust available to actuate the capsule. I remember this problem from school, and I also remember wondering where to pick the unity-gain crossover frequency and then my lead zero location on the frequency axis.

In our case, the, "practical" limit for our lead compensating zero is going to be below the double-pole at our, 'A' term. The 'A' term is the, "practical" limit in our case. It represents the frequency response limits of our actuating propeller pair. Theoretically, the most pahse we can gain with a lead if 90-degrees, before we cancel it with a higher-frequency pole. However, you can see above that the double-pole at A is lowering our already zero phase-margin plant beyond any phase angle we can recover with a lead compensator.

We must place our lead zero far enough below, 'A', and a unity-gain crossover to create a phase-margin hump before the double-pole at 'A' kills the phase beyond our ability to gain any with the lead zero.

Scale the lead zero frequency down-frequency from the 'A' double-pole

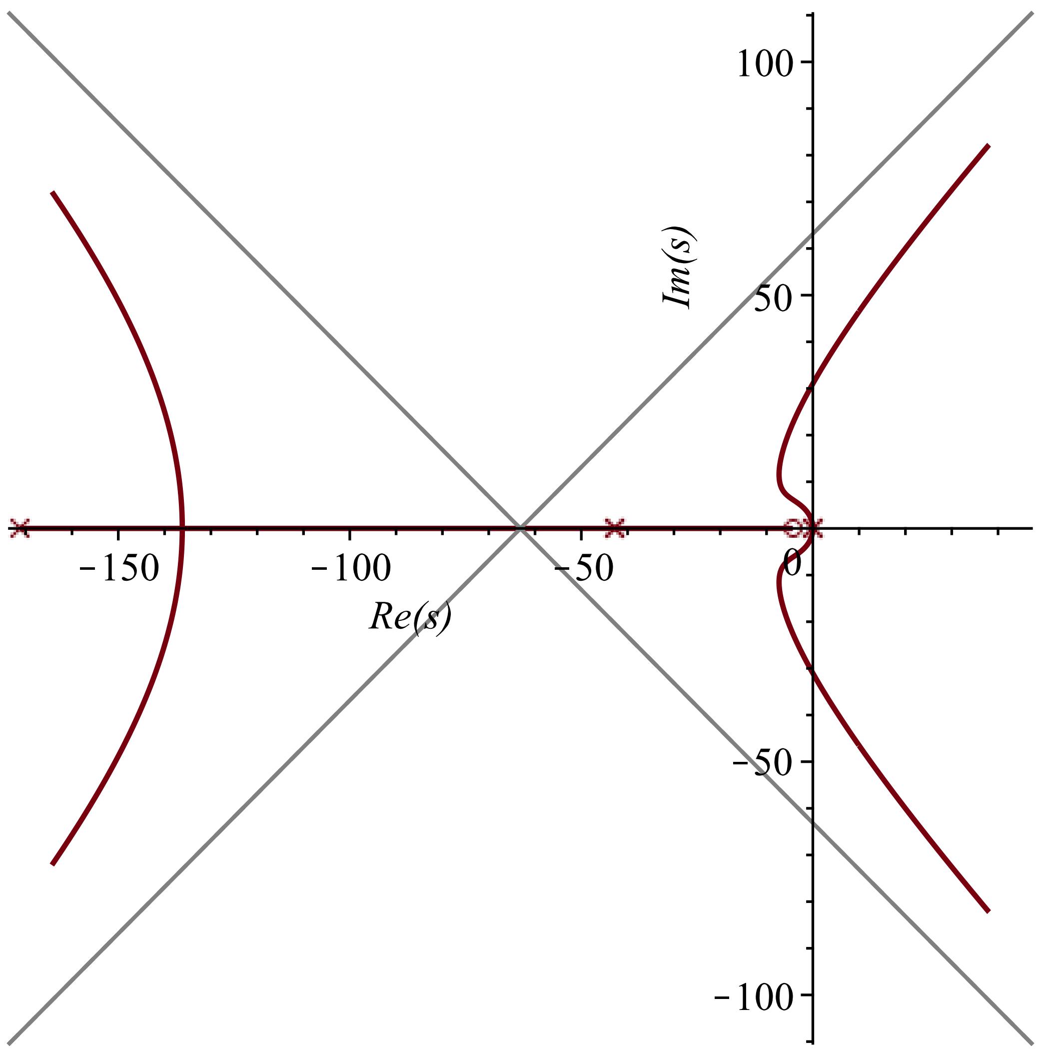
$lz := \frac{A}{10}$:

$lp := 4 \cdot A$:

Compare the resulting root-locus for the compensated plant with the plant root-locus above. You can see the effect of the zero: It's, "pulling" the loci to the left-half plane for a range of relatively low gain. Sketching rules for root-locus diagrams tell us we could pick a spot on the bulge of the root locus near the origin and estimate frequency response, phase margin and/or time-domain characteristics to expect.

$$G_c := TransferFunction\left(\frac{s + lz}{s + lp} \cdot \frac{K_g}{s^2 \cdot (s + A)^2}\right) :$$

$$RootLocusPlot(G_c, 0 ..1000000)$$



Compute Proportional Term for Desired unity-gain crossover

The Bode plot will help determine what our compensator gain needs should be. We want unity-gain crossover where our phase-margin hump peaks near 10 rad/sec. Get the exact frequency as an index from the phase data to the gain data. Then we'll know what proportional gain is required for unity-gain crossover at this frequency.

$$phase := BodePlot\left(G_c, output = [phasedata, magnitudedata]\right) :$$

Get the index of the maximum phase from the Bode plot

$$mpi := \max[index](phase[1][[1..-1], [2]])$$

$$mpi := 70, 1 \tag{9}$$

Phase margin at this frequency is...

$$pm := phase[1][[mpi, 1]][1, 2] + 180$$

$$pm := 37.9206762881709 \tag{10}$$

Plant gain at this frequency

$$ugcog := phase[2][mpi[1]][2]$$

$$ugcog := -79.63995814 \tag{11}$$

"uccog" is the gain of the lead-compensated system at our frequency of maximum phase margin for a proportional gain of one. The above number is in dB. We calculate what proportional gain we'll need for unity gain as..

$$K_p := 10^{\frac{-ugcog}{20}}$$

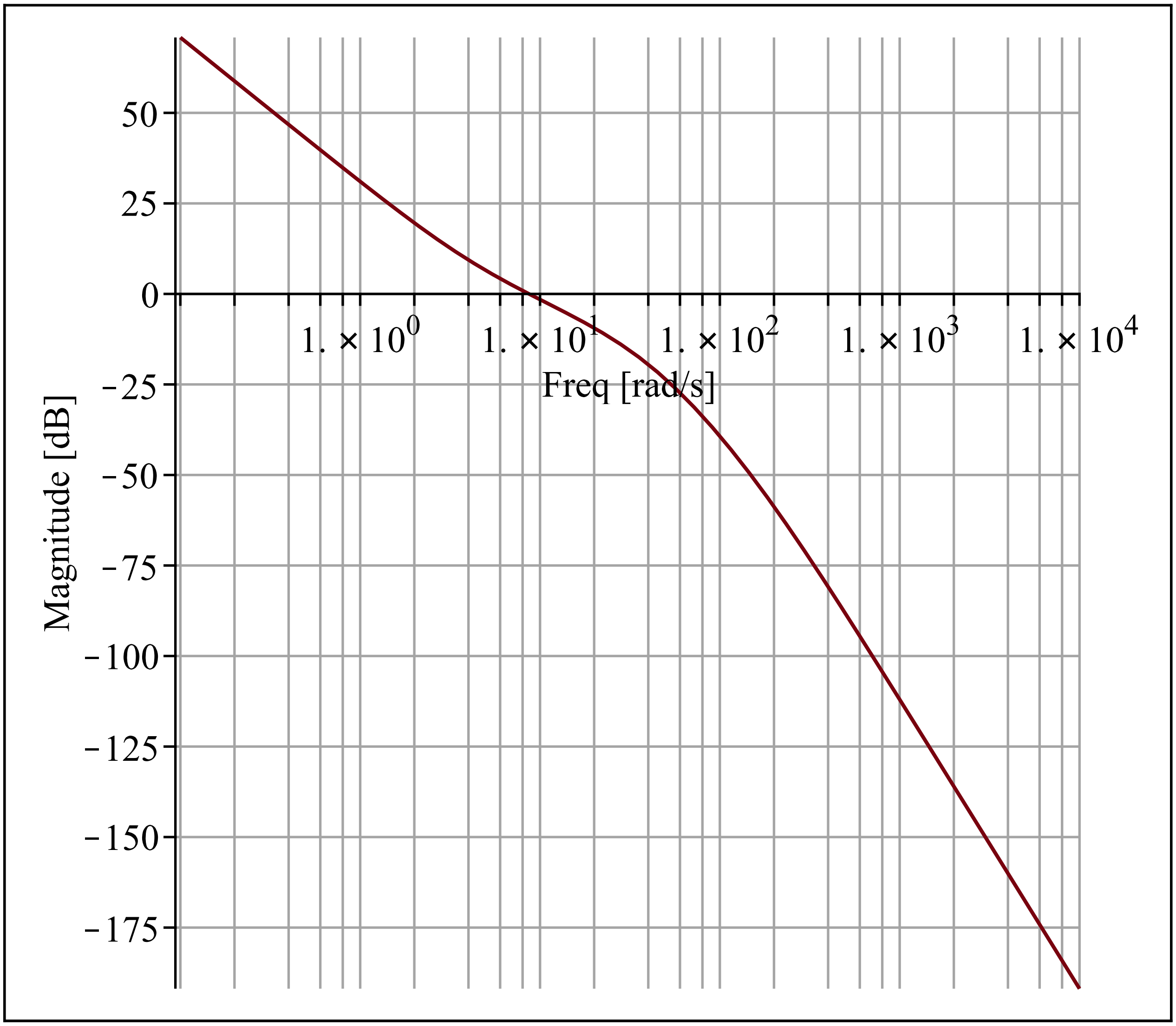
$$K_p := 9593.960079 \tag{12}$$

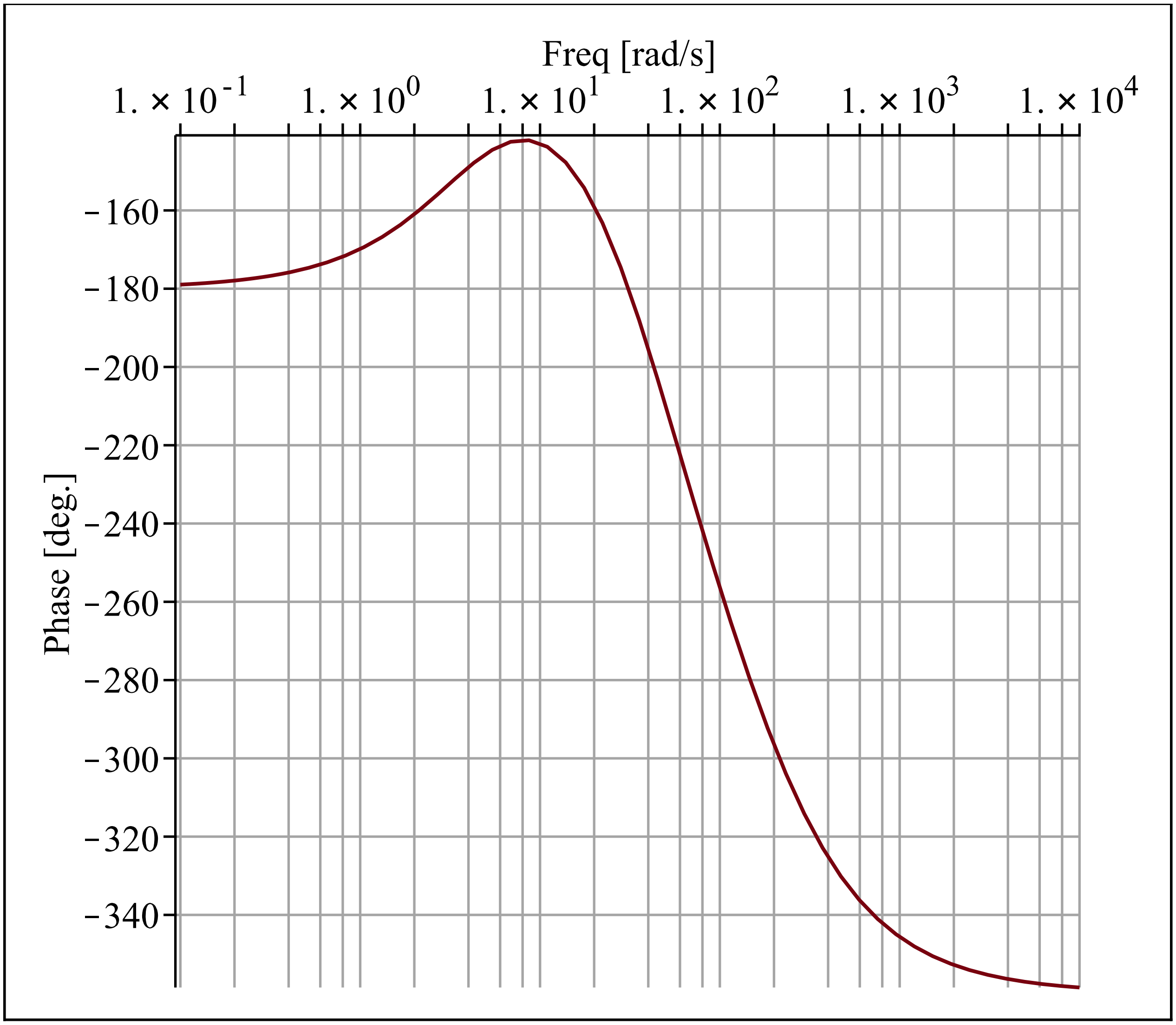
$$G_{cd} := TransferFunction\left(\frac{s + lz}{s + lp} \cdot \frac{K_p \cdot K_g}{s^2 \cdot (s + A)^2}\right) :$$

Resultant Bode Plot for the Compensated System

Here we see unity gain crossover for the compensated plant with a design phase margin of $pm = 38.$ degrees.

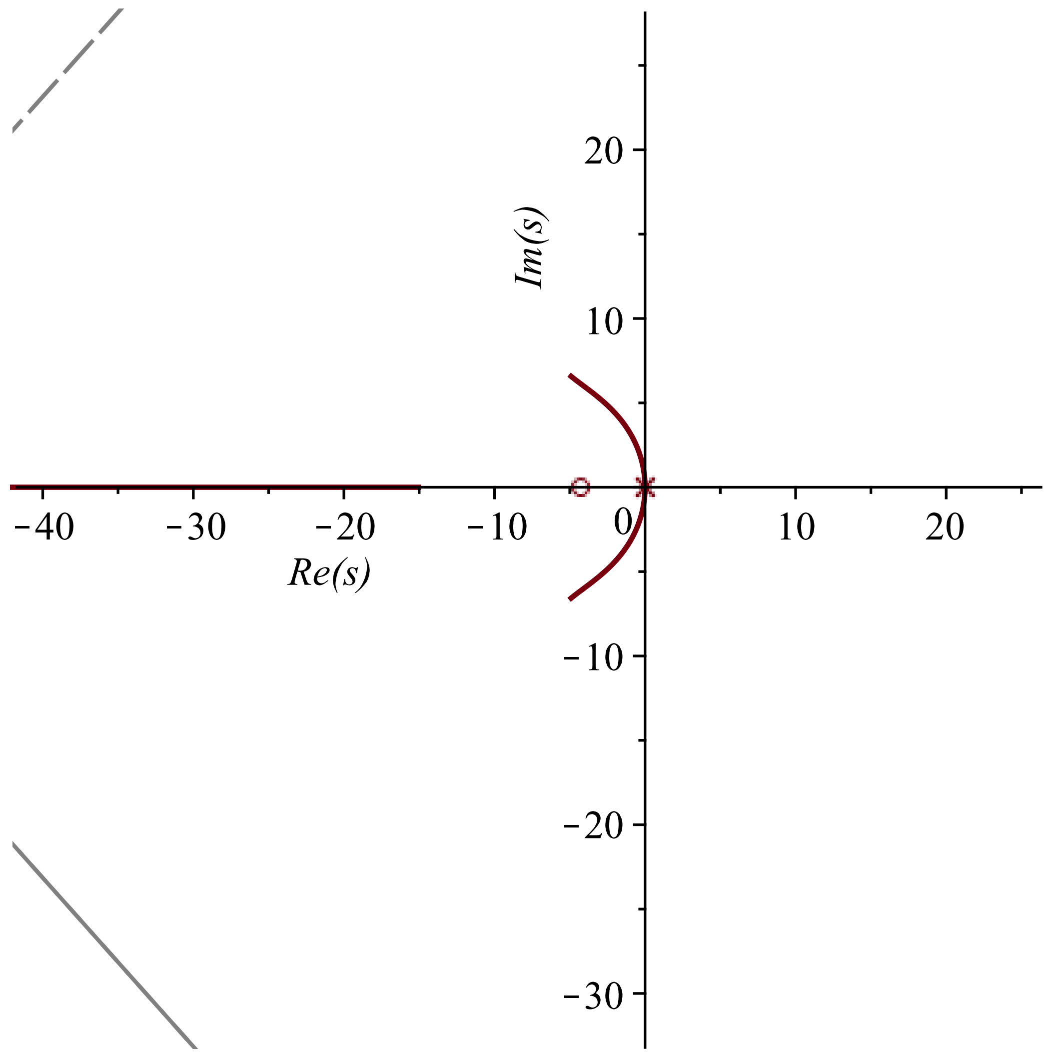
$BodePlot(G_{cd})$





Plotting the root-locus up to this proprtional gain illustrates what we would expect from a pole-placement perspective: our conditional stability based on this compensator and gain gives us a pair of complex conjugate poles in the left-half-plane (LHP) where we desire them. Selecting unity gain crossover at the maximim phase margin location above is equivalent to selecting a gain from the root-locus plot at the point of inflection representing the maximum angle between the complex axis and a line from the origin to one of these poles.

$RootLocusPlot\left(G_c, 0 ..K_p\right)$



Im(s)
Re(s)
20
10
0
−10
−20
−30
−40
−30
−20
−10
10
20

Yaw Axis Model

When we command for pitch and roll using the designed controller our input is the difference of the squared cross-body motor drive command inputs. Yaw is controlled by the difference between the squared clockwise and counterclockwise motor drive inputs.

The first per-iteration step will be to employ the above control scheme to determine the next commanded value required as the difference-squared between cross-body motor drive for roll and pitch.

These are then differenced relative to the last iteration to produce a positive or negative change relative to the cross-body difference of squared commanded inputs for roll and pitch from the last iteration. The difference is then queued for each motor command, but not yet issued.

The per-iteration control strategy will then assess yaw error to establish a ratio of the clockwise and counterclockwise commands required to drive to a particular yaw angle or regulate it to zero.

A positive or negative difference of the squares computed to drive yaw is then applied to the queued roll and pitch commands. The clockwise motor command is increased the same ammount as the counter-clockwise command is decreased, or vice versa. This maintains constant along-body-z axis thrust. for the near-horizontal hover model assumptions thus far this implies constant fixed-frame Z-axis, or vertical thrust with respect to gravity vector.

Body Z-axis moment of inertia

The Body Z-axis moment of inertia is modelled as 4 motors on massless rods, just as the roll and pitch are modeled as 2 on massless rods of the same length. This neglects near-center-of-mass control unit, batteries, and the frame mass. We can account for this later. For now, this simple model will get our controller design going.

$I_z := 4\, m_m l_a^2$

$$I_z := 0.004\ \mathrm{kg}\ \mathrm{m}^2 \tag{13}$$

Yaw-axis transfer function and lead compensator

The plant gain for the yaw axis involves the, "drag factor" not the "thrust factor" as in the roll and pitch case. This is the, 'd' term here. Otherwise the caracteristics of the plant are the same: a double-integrator with bandwidth-limiting double-pole at A that represents the bandwidth of our motor-propeller system in providing thrust relative to input drive voltage to the motors.

$$K_{gy} := convert\left(\frac{l_a\, d\, B^2}{I_z}, unit\_free\right)$$

$$K_{gy} := 14.79793204 \tag{14}$$

The plant poles are the same as for the roll and pitch axes so the lead compensator structure is the same: same lead zero and lead-cancelling pole.

$$G_{cy} := TransferFunction\left(\frac{s+lz}{s+lp}\cdot\frac{K_{gy}}{s^2\cdot(s+A)^2}\right):$$

Yaw-axis gain proportional gain determination

As above, we desire unity-gain crossover at our frequency of maximum design phase margin. Here we use the Maple tools to find this peak and compute a needed gain for the yaw-axis unity-gain crossover as desired.

$$phasey := BodePlot\left(G_{cy}, output = [phasedata, magnitudedata]\right):$$

Get the index of the maximum phase from the Bode plot

$$mpiy := \max[index](phasey[1][[1..-1],[2]])$$

$$mpiy := 70, 1 \tag{15}$$

Phase margin at this frequency is...

$$pmy := phasey[1][[mpiy, 1]][1, 2] + 180$$

$$pmy := 37.9206762839632 \tag{16}$$

Plant gain at this frequency

$$ugcogy := phasey[2][mpiy[1]][2]$$

$$ugcogy := -104.7221946 \tag{17}$$

"uccogy" is the gain of the lead-compensated system at our frequency of maximum phase margin for a proportional gain of one. The above number is in dB. We calculate what proportional gain we'll need for unity gain as..

$$K_{py} := 10^{\frac{-ugcogy}{20}}$$

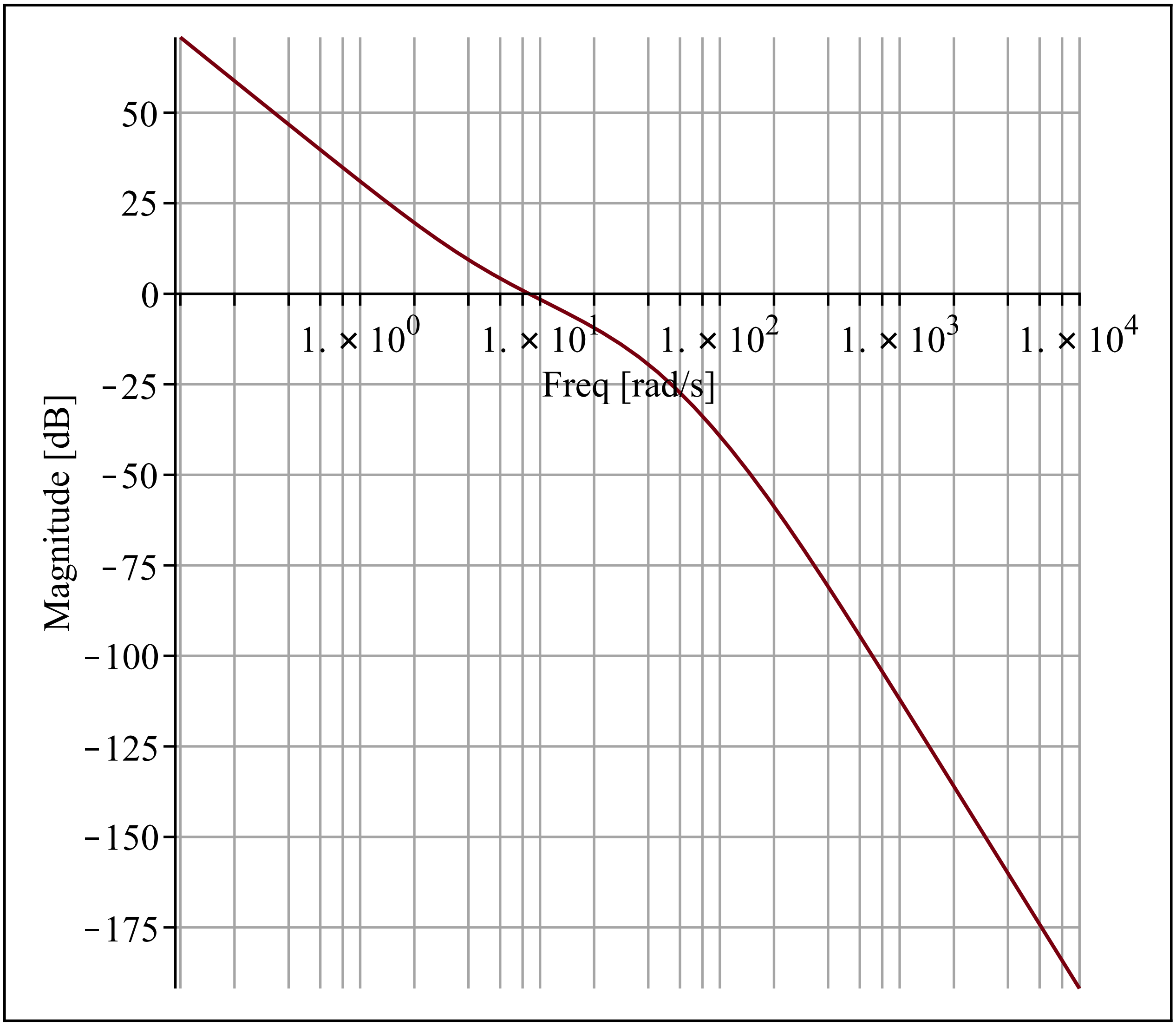
$$K_{py} := 172230.3682 \tag{18}$$

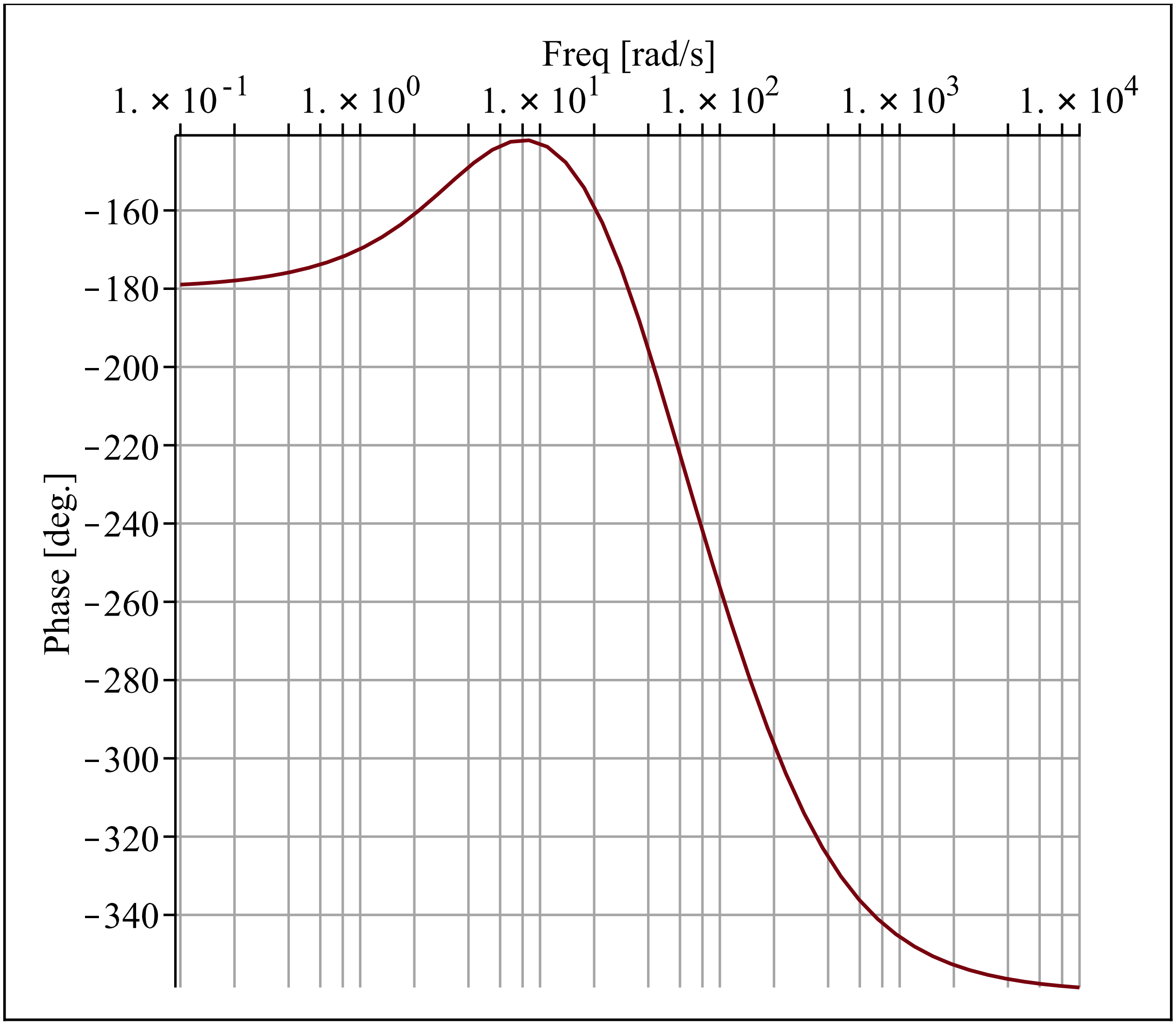
Compensated Plant Transfer Function

Here again we see this gain will give us unity-gain crossover at our maximum design phase-margin.

$$G_{cdy} := TransferFunction\left(\frac{s+lz}{s+lp}\cdot\frac{K_{py}\cdot K_{gy}}{s^2\cdot(s+A)^2}\right):$$

$BodePlot(G_{cdy})$





Freq [rad/s]
1. × 10^-1
1. × 10^0
1. × 10^1
1. × 10^2
1. × 10^3
1. × 10^4
Phase [deg.]
−160
−180
−200
−220
−240
−260
−280
−300
−320
−340