

# The **bodeplot** package\*

Rushikesh Kamalapurkar  
rlkamalapurkar@gmail.com

October 29, 2021

## 1 Introduction

Generate Bode, Nyquist, and Nichols plots for transfer functions in the canonical (TF) form

$$G(s) = e^{-Ts} \frac{b_m s^m + \cdots + b_1 s + b_0}{a_n s^n + \cdots + a_1 s + a_0} \quad (1)$$

and the zero-pole-gain (ZPK) form

$$G(s) = K e^{-Ts} \frac{(s - z_1)(s - z_2) \cdots (s - z_m)}{(s - p_1)(s - p_2) \cdots (s - p_n)}. \quad (2)$$

In the equations above,  $b_m, \dots, b_0$  and  $a_n, \dots, a_0$  are real coefficients,  $T \geq 0$  is the loop delay,  $z_1, \dots, z_m$  and  $p_1, \dots, p_n$  are complex zeros and poles of the transfer function, respectively, and  $K \in \mathbb{R}$  is the loop gain. For transfer functions in the ZPK format in (2) with zero delay, this package also supports linear and asymptotic approximation of Bode plots.

## 2 Usage

### 2.1 Bode plots

```
\BodeZPK \BodeZPK [ $\langle obj1/typ1/\{opt1\} \rangle, obj2/typ2/\{opt2\}, \dots \rangle$ ]  
           $\{ \langle z/\{zeros\} \rangle, p/\{poles\} \rangle, k/\{gain\} \rangle, d/\{delay\} \rangle \}$   
           $\{ \langle min-freq \rangle \} \{ \langle max-freq \rangle \}$ 
```

Plots the Bode plot of a transfer function given in ZPK format using the **groupplot** environment. The three mandatory arguments include: (1) a list of tuples, comprised of the zeros, the poles, the gain, and the transport delay of the transfer function, (2) the lower end of the frequency range for the  $x$ -axis, and (3) the higher end of the frequency range for the  $x$ -axis. The zeros and the poles are complex numbers, entered as a comma-separated

---

\*This document corresponds to **bodeplot** ?, dated ?.

list of comma-separated lists, of the form `{{real part 1,imaginary part 1},{real part 2,imaginary part 2},...}`. If the imaginary part is not provided, it is assumed to be zero.

The optional argument is comprised of a comma separated list of tuples, either `obj/typ/{opt}`, or `obj/{opt}`, or just `{opt}`. Each tuple passes options to different `pgfplots` macros that generate the group, the axes, and the plots according to:

- Tuples of the form `obj/typ/{opt}`:
  - `plot/typ/{opt}`: modify plot properties by adding options `{opt}` to the `\addplot` macro for the magnitude plot if `typ` is `mag` and the phase plot if `typ` is `ph`.
  - `axes/typ/{opt}`: modify axis properties by adding options `{opt}` to the `\nextgroupplot` macro for the magnitude plot if `typ` is `mag` and the phase plot if `typ` is `ph`.
- Tuples of the form `obj/{opt}`:
  - `plot/{opt}`: adds options `{opt}` to `\addplot` macros for both the magnitude and the phase plots.
  - `axes/{opt}`: adds options `{opt}` to `\nextgroupplot` macros for both the magnitude and the phase plots.
  - `group/{opt}`: adds options `{opt}` to the `groupplot` environment.
  - `approx/linear`: plots linear approximation.
  - `approx/asymptotic`: plots asymptotic approximation.
- Tuples of the form `{opts}` add all of the supplied options to `\addplot` macros for both the magnitude and the phase plots.

The options `{opt}` can be any `key=value` options that are supported by the `pgfplots` macros they are added to. *Linear or asymptotic approximation of transfer functions that include a transport delay is not supported.*

For example, given a transfer function

$$G(s) = 10 \frac{s(s + 0.1 + 0.5i)(s + 0.1 - 0.5i)}{(s + 0.5 + 10i)(s + 0.5 - 10i)}, \quad (3)$$

its Bode plot over the frequency range `[0.01,100]` can be generated using `\BodeZPK`

```
{z/{0,{-0.1,-0.5},{-0.1,0.5}},p/{{-0.5,-10},{-0.5,10}},k/10}
{0.01}{100}
```

which generates the plot in Figure 1. If a delay is not specified, it is assumed to be zero. If a gain is not specified, it is assumed to be 1. By default, each of the axes, excluding ticks and labels, are 5cm wide and 2.5cm high. The width and the height, along with other properties of the plots, the axes, and the group can be customized using native `pgf` keys as shown in the example below.

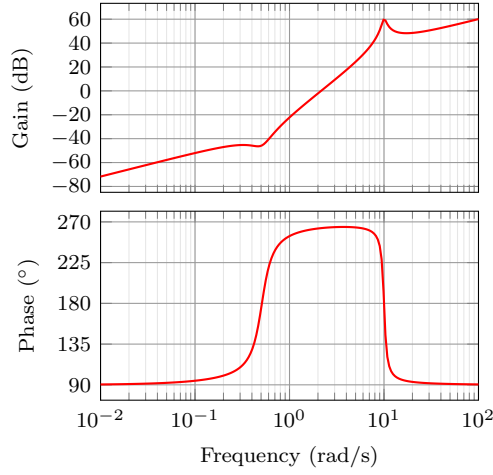


Figure 1: Output of the default `\BodeZPK` macro.

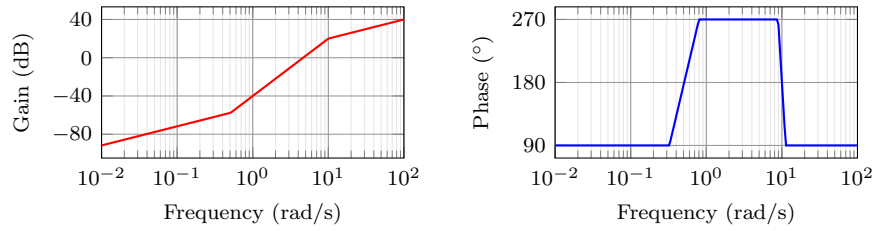


Figure 2: Customization of the default `\BodeZPK` macro.

A linear approximation of the Bode plot with customization of the plots, the axes, and the group can be generated using

```
\BodeZPK[plot/mag/{red,thick},plot/ph/{blue,thick},
  axes/mag/{ytick distance=40,xmajorticks=true,
  xlabel={Frequency (rad/s)}},axes/ph/{ytick distance=90},
  group/{group style={group size=2 by 1,horizontal sep=2cm,
  width=4cm,height=2cm}},approx/linear]
{z/{0,{-0.1,-0.5},{-0.1,0.5}},p/{{-0.5,-10},{-0.5,10}},k/10}
{0.01}{100}
```

which generates the plot in Figure 2.

```
\BodeTF [obj1/typ1/{opt1}obj2/typ2/{opt2}...]
  {num/{coeffs}den/{coeffs}d/{delay}}
  {min-freq}{max-freq}}
```

Plots the Bode plot of a transfer function given in TF format. The three mandatory arguments include: (1) a list of tuples comprised of the coefficients in the

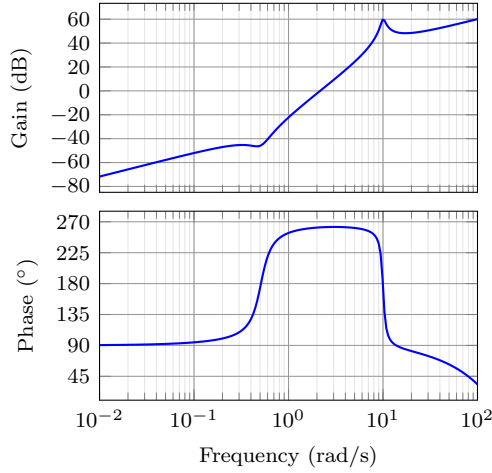


Figure 3: Output of the `\BodeTF` macro.

numerator and the denominator of the transfer function and the transport delay, (2) the lower end of the frequency range for the  $x$ -axis, and (3) the higher end of the frequency range for the  $x$ -axis. The coefficients are entered as a comma-separated list, in order from the highest degree of  $s$  to the lowest, with zeros for missing degrees. The optional arguments are the same as `\BodeZPK`, except that linear/asymptotic approximation is not supported, so `approx/...` is ignored.

For example, given the same transfer function as (3) in TF form and with a small transport delay,

$$G(s) = e^{-0.01s} \frac{s(10s^2 + 2s + 2.6)}{(s^2 + s + 100.25)}, \quad (4)$$

its Bode plot over the frequency range  $[0.01, 100]$  can be generated using

```
\BodeTF[blue,thick]
  {num/{10,2,2.6,0},den/{1,0.2,100},d/0.01}
  {0.01}{100}
```

which generates the plot in Figure 3. Note the 0 added to the numerator coefficients to account for the fact that the numerator does not have a constant term in it. As demonstrated in this example, if a single comma-separated list of options is passed, it applies to both the magnitude and the phase plots.

```
BodePlot \begin{BodePlot}[\langle axis-options \rangle]{\langle min-frequency \rangle}{\langle max-frequency \rangle}
\addBode...
\end{BodePlot}
```

The `BodePlot` environment works in conjunction with the parametric function generator macros `\addBodeZPKPlots`, `\addBodeTFPlot`, and `\addBodeComponentPlots`. If supplied, `axis-options` are passed directly to the `semilogaxis` environment and the frequency limits are translated to the  $x$ -axis limits and the do-

main of the `semilogaxis` environment. Example usage in the description of `\addBodeZPKPlots`, `\addBodeTFPlot`, and `\addBodeComponentPlots`.

`\addBodeZPKPlots`      `\addBodeZPKPlots` [*approx1*/*{opt1}*],*approx2*/*{opt2}*],...]  
                           *{plot-type}*  
                           *{z*/*{zeros}*},*p*/*{poles}*},*k*/*{gain}*},*d*/*{delay}*}]

Generates the appropriate parametric functions and supplies them to multiple `\addplot` macros, one for each `approx`/*{opt}* pair in the optional argument. If no optional argument is supplied, then a single `\addplot` command corresponding to the true Bode plot is generated. This macro can be used inside any `semilogaxis` environment as long as a domain for the x-axis is supplied through either the `approx`/*{opt}* interface or directly in the optional argument of the `semilogaxis` environment. Use with the `BodePlot` environment supplied with this package is recommended. The second mandatory argument, `plot-type` is either `magnitude` or `phase`. If it is not equal to `phase`, it is assumed to be `magnitude`. The last mandatory argument is the same as `\BodeZPK`.

For example, given the transfer function in (3), its linear, asymptotic, and true Bode plots can be superimposed using

```
\begin{BodePlot}[ ylabel={Gain (dB)}, ytick distance=40,
  height=2cm, width=4cm] {0.01} {100}
  \addBodeZPKPlots[
    true/{black,thick},
    linear/{red,dashed,thick},
    asymptotic/{blue,dotted,thick}]
    {magnitude}
    {z/{0,-0.1,-0.5},{-0.1,0.5}},p/{-0.1,-10},{-0.1,10}},k/10}
\end{BodePlot}
```

```
\begin{BodePlot}[ylabel={Phase ( $\angle$ )},
  height=2cm, width=4cm, ytick distance=90,] {0.01} {100}
  \addBodeZPKPlots[
    true/{black,thick},
    linear/{red,dashed,thick},
    asymptotic/{blue,dotted,thick}]
    {phase}
    {z/{0,-0.1,-0.5},{-0.1,0.5}},p/{-0.1,-10},{-0.1,10}},k/10}
\end{BodePlot}
```

which generates the plot in Figure 4.

`\addBodeTFPlot`      `\addBodeTFPlot` [*plot-options*]  
                           *{plot-type}*  
                           *{num*/*{coeffs}*},*den*/*{coeffs}*},*d*/*{delay}*}]

Generates a single parametric function for either Bode magnitude or phase plot of a transfer function in TF form. The generated parametric function is passed to the `\addplot` macro. This macro can be used inside any `semilogaxis` environment as long as a domain for the x-axis is supplied through either the `plot-options` interface or directly in the optional argument of the container `semilogaxis` environment. Use with the `BodePlot` environment supplied with this package is

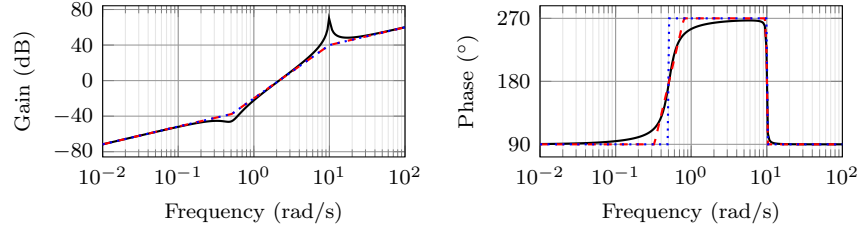


Figure 4: Superimposed approximate and true Bode plots using the `BodePlot` environment and the `\addBodeZPKPlots` macro.

recommended. The second mandatory argument, `plot-type` is either `magnitude` or `phase`. If it is not equal to `phase`, it is assumed to be `magnitude`. The last mandatory argument is the same as `\BodeTF`.

`\addBodeComponentPlot`

`\addBodeComponentPlot[<plot-options>]{<plot-command>}`

Generates a single parametric function corresponding to the mandatory argument `plot-command` and passes it to the `\addplot` macro. The plot command can be any parametric function that uses `t` as the independent variable. The parametric function must be `gnuplot` compatible (or `pgfplots` compatible if the package is loaded using the `pgf` option). The intended use of this macro is to plot the parametric functions generated using the basic component macros described in Section 2.1.1 below.

### 2.1.1 Basic components up to first order

`\TypeFeatureApprox`

`\TypeFeatureApprox{<real-part>}{<imaginary-part>}`

This entry describes 20 different macros of the form `\TypeFeatureApprox` that take the real part and the imaginary part of a complex number as arguments. The **Type** in the macro name should be replaced by either **Mag** or **Ph** to generate a parametric function corresponding to the magnitude or the phase plot, respectively. The **Feature** in the macro name should be replaced by one of **K**, **Pole**, **Zero**, or **Del**, to generate the Bode plot of a gain, a complex pole, a complex zero, or a transport delay, respectively. If the **Feature** is set to either **K** or **Del**, the *imaginary-part* mandatory argument is ignored. The **Approx** in the macro name should either be removed, or it should be replaced by **Lin** or **Asymp** to generate the true Bode plot, the linear approximation, or the asymptotic approximation, respectively. If the **Feature** is set to **Del**, then **Approx** has to be removed. For example,

- `\MagK{k}{0}` or `\MagK{k}{400}` generates a parametric function for the true Bode magnitude of  $G(s) = k$
- `\PhPoleLin{a}{b}` generates a parametric function for the linear approximation of the Bode phase of  $G(s) = \frac{1}{s-a-ib}$ .

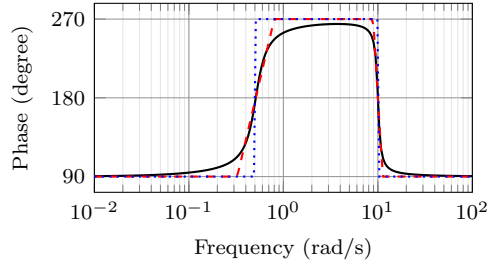


Figure 5: Superimposed approximate and true Bode Phase plot using the `BodePlot` environment, the `\addBodeComponentPlot` macro, and several macros of the `\TypeFeatureApprox` form.

- `\PhDel{T}{200}` or `\PhDel{T}{0}` generates a parametric function for the Bode phase of  $G(s) = e^{-Ts}$ .

All 20 of the macros defined by combinations of `Type`, `Feature`, and `Approx`, and any `gnuplot` (or `pgfplot` if the `pgf` class option is loaded) compatible function of the 20 macros can be used as `plot-command` in the `addBodeComponentPlot` macro. This is sufficient to generate the Bode plot of any rational transfer function with delay. For example, the Bode phase plot in Figure 4 can also be generated using:

```
\begin{BodePlot}[ylabel={Phase (degree)},ytick distance=90]{0.01}{100}
  \addBodeComponentPlot[black,thick]{\PhZero{0}{0} + \PhZero{-0.1}{-0.5} +
    \PhZero{-0.1}{0.5} + \PhPole{-0.5}{-10} + \PhPole{-0.5}{10} +
    \PhK{10}{0}}
  \addBodeComponentPlot[red,dashed,thick]{\PhZeroLin{0}{0} +
    \PhZeroLin{-0.1}{-0.5} + \PhZeroLin{-0.1}{0.5} +
    \PhPoleLin{-0.5}{-10} + \PhPoleLin{-0.5}{10} + \PhKLin{10}{20}}
  \addBodeComponentPlot[blue,dotted,thick]{\PhZeroAsymp{0}{0} +
    \PhZeroAsymp{-0.1}{-0.5} + \PhZeroAsymp{-0.1}{0.5} +
    \PhPoleAsymp{-0.5}{-10} + \PhPoleAsymp{-0.5}{10} + \PhKAsymp{10}{40}}
\end{BodePlot}
```

which gives us the plot in Figure 5.

### 2.1.2 Basic components of the second order

`\TypeS0FeatureApprox` `\TypeS0FeatureApprox{<a1>}{<a0>}`

This entry describes 12 different macros of the form `\TypeS0FeatureApprox` that take the coefficients  $a_1$  and  $a_0$  of a general second order system as inputs. The **Feature** in the macro name should be replaced by either **Poles** or **Zeros** to generate the Bode plot of  $G(s) = \frac{1}{s^2 + a_1s + a_0}$  or  $G(s) = s^2 + a_1s + a_0$ , respectively. The **Type** in the macro name should be replaced by either **Mag** or **Ph** to generate a parametric function corresponding to the magnitude or the phase plot, respectively. The **Approx** in the macro name should either be removed, or it should be replaced

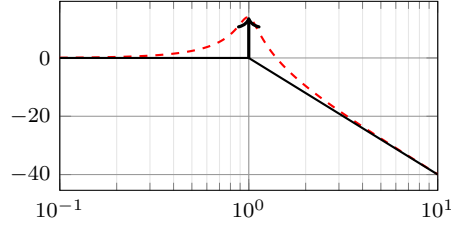


Figure 6: Resonant peak in asymptotic Bode plot using `\MagSOPolesPeak`.

by `Lin` or `Asymp` to generate the true Bode plot, the linear approximation, or the asymptotic approximation, respectively.

`\MagSOFeaturePeak` `\MagSOFeaturePeak[⟨draw-options⟩]{⟨a1⟩}{⟨a0⟩}`

This entry describes 2 different macros of the form `\MagSOFeaturePeak` that take the coefficients  $a_1$  and  $a_0$  of a general second order system as inputs, and draw a resonant peak using the `\draw` TikZ macro. The **Feature** in the macro name should be replaced by either **Poles** or **Zeros** to generate a peak for poles and a valley for zeros, respectively. For example, the command

```
\begin{BodePlot}[xlabel={}] {0.1}{10}
  \addBodeComponentPlot[red,dashed,thick]{\MagSOPoles{0.2}{1}}
  \addBodeComponentPlot[black,thick]{\MagSOPolesLin{0.2}{1}}
  \MagSOPolesPeak[thick]{0.2}{1}
\end{BodePlot}
```

generates the plot in Figure 6.

`\TypeCSFeatureApprox` `\TypeCSFeatureApprox{⟨zeta⟩}{⟨omega-n⟩}`

This entry describes 12 different macros of the form `\TypeCSFeatureApprox` that take the damping ratio,  $\zeta$ , and the natural frequency,  $\omega_n$  of a canonical second order system as inputs. The **Type** in the macro name should be replaced by either **Mag** or **Ph** to generate a parametric function corresponding to the magnitude or the phase plot, respectively. The **Feature** in the macro name should be replaced by either **Poles** or **Zeros** to generate the Bode plot of  $G(s) = \frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2}$  or  $G(s) = s^2 + 2\zeta\omega_n s + \omega_n^2$ , respectively. The **Approx** in the macro name should either be removed, or it should be replaced by **Lin** or **Asymp** to generate the true Bode plot, the linear approximation, or the asymptotic approximation, respectively.

`\MagCSFeaturePeak` `\MagCSFeaturePeak[⟨draw-options⟩]{⟨zeta⟩}{⟨omega-n⟩}`

This entry describes 2 different macros of the form `\MagCSFeaturePeak` that take the damping ratio,  $\zeta$ , and the natural frequency,  $\omega_n$  of a canonical second order system as inputs, and draw a resonant peak using the `\draw` TikZ macro. The **Feature** in the macro name should be replaced by either **Poles** or **Zeros** to generate a peak for poles and a valley for zeros, respectively.

`\MagCCFeaturePeak` `\MagCCFeaturePeak[⟨draw-options⟩]{⟨real-part⟩}{⟨imaginary-part⟩}`

This entry describes 2 different macros of the form `\MagCCFeaturePeak` that take the real and imaginary parts of a pair of complex conjugate poles or zeros as



inputs, and draw a resonant peak using the `\draw` TikZ macro. The `Feature` in the macro name should be replaced by either `Poles` or `Zeros` to generate a peak for poles and a valley for zeros, respectively.

## 2.2 Nyquist plots

`\NyquistZPK` `\NyquistZPK` [ $\langle plot/\{opt\} \rangle$ ,  $\langle axes/\{opt\} \rangle$ ]  
 $\{z/\{zeros\} \}$ ,  $p/\{poles\} \}$ ,  $k/\{gain\} \}$ ,  $d/\{delay\} \}$   
 $\{min-freq\} \{max-freq\}$

Plots the Nyquist plot of a transfer function given in ZPK format with a thick red + marking the critical point (-1,0). The mandatory arguments are the same as `\BodeZPK`. Since there is only one plot in a Nyquist diagram, the `\typ` specifier in the optional argument tuples is not needed. As such, the supported optional argument tuples are `plot/\{opt\}`, which passes `\{opt\}` to `\addplot` and `axes/\{opt\}`, which passes `\{opt\}` to the `axis` environment. Asymptotic/linear approximations are not supported in Nyquist plots. If just `\{opt\}` is provided as the optional argument, it is interpreted as `plot/\{opt\}`. Arrows to indicate the direction of increasing  $\omega$  can be added by adding `\usetikzlibrary{decorations.markings}` and `\usetikzlibrary{arrows.meta}` to the preamble and then passing a tuple of the form

`plot/\{postaction=decorate,decoration={markings,`  
`mark=between positions 0.1 and 0.9 step 5em with`  
`\arrow{Stealth [length=2mm, blue]}\}`

**Caution:** with a high number of samples, adding arrows in this way may cause the error message ! Dimension too big.

For example, the command

`\NyquistZPK`[`plot/\{red,thick,samples=2000\}`,`axes/\{blue,thick\}`]  
 $\{z/\{0,-0.1,-0.5\},\{-0.1,0.5\} \}$ ,  $p/\{\{-0.5,-10\},\{-0.5,10\} \}$ ,  $k/10$   
 $\{-30\} \{30\}$

generates the Nyquist plot in Figure 7.

`\NyquistTF` `\NyquistTF` [ $\langle plot/\{opt\} \rangle$ ,  $\langle axes/\{opt\} \rangle$ ]  
 $\{num/\{coeffs\} \}$ ,  $den/\{coeffs\} \}$ ,  $d/\{delay\} \}$   
 $\{min-freq\} \{max-freq\}$

Nyquist plot of a transfer function given in TF format. Same mandatory arguments as `\BodeTF` and same optional arguments as `\NyquistZPK`. For example, the command

`\NyquistTF`[`plot/\{green,thick,samples=500,postaction=decorate,`  
`decoration={markings,`  
`mark=between positions 0.1 and 0.9 step 5em`  
`with{\arrow{Stealth[length=2mm, blue]}\}`  
 $\{num/\{10,2,2.6,0\},den/\{1,1,100.25\} \}$   
 $\{-30\} \{30\}$

generates the Nyquist plot in Figure 8.

`NyquistPlot` `\begin{NyquistPlot}` [ $\langle axis-options \rangle$ ]  $\{min-frequency\} \{max-frequency\}$   
`\addNyquist...`  
`\end{NyquistPlot}`

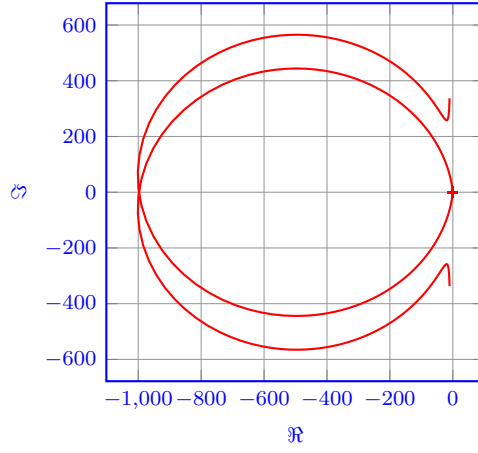


Figure 7: Output of the `\NyquistZPK` macro.

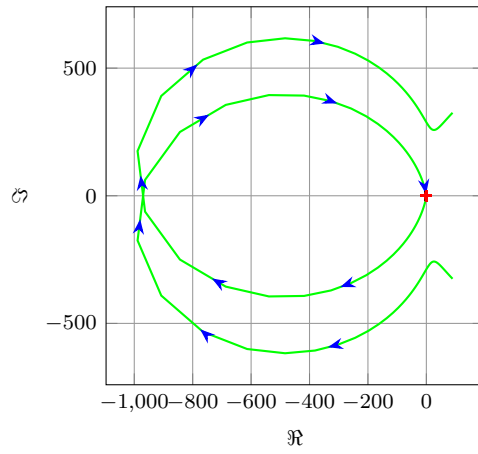


Figure 8: Output of the `\NyquistTF` macro with direction arrows. Increasing the number of samples can cause `decorations.markings` to throw errors.

The `NyquistPlot` environment works in conjunction with the parametric function generator macros `\addNyquistZPKPlot` and `\addNyquistTFPlot`. If supplied, `axis-options` are passed directly to the `axis` environment and the frequency limits are translated to the x-axis limits and the domain of the `axis` environment.

`\addNyquistZPKPlot`      `\addNyquistZPKPlot[⟨plot-options⟩]`  
                            $\{ \langle z/\langle zeros \rangle \rangle, p/\langle poles \rangle \rangle, k/\langle gain \rangle \rangle, d/\langle delay \rangle \}$

Generates a two parametric functions for the magnitude and the phase a transfer function in ZPK form. The generated magnitude and phase parametric functions are converted to real and imaginary part parametric functions and passed to the `\addplot` macro. This macro can be used inside any `axis` environment as long as a domain for the x-axis is supplied through either the `plot-options` interface or directly in the optional argument of the container `axis` environment. Use with the `NyquistPlot` environment supplied with this package is recommended. The mandatory argument is the same as `\BodeZPK`.

`\addNyquistTFPlot`      `\addNyquistTFPlot[⟨plot-options⟩]`  
                            $\{ \langle num/\langle coeffs \rangle \rangle, den/\langle coeffs \rangle \rangle, d/\langle delay \rangle \}$

Similar to `\addNyquistZPKPlot`, with a transfer function input in the TF form.

## 2.3 Nichols charts

`\NicholsZPK`      `\NicholsZPK[⟨plot/⟨opt⟩⟩, axes/⟨opt⟩⟩]`  
                            $\{ \langle z/\langle zeros \rangle \rangle, p/\langle poles \rangle \rangle, k/\langle gain \rangle \rangle, d/\langle delay \rangle \}$   
                            $\{ \langle min-freq \rangle \} \{ \langle max-freq \rangle \}$

Nichols chart of a transfer function given in ZPK format. Same arguments as `\NyquistZPK`.

`\NicholsTF`      `\NicholsTF[⟨plot/⟨opt⟩⟩, axes/⟨opt⟩⟩]`  
                            $\{ \langle num/\langle coeffs \rangle \rangle, den/\langle coeffs \rangle \rangle, d/\langle delay \rangle \}$   
                            $\{ \langle min-freq \rangle \} \{ \langle max-freq \rangle \}$

Nichols chart of a transfer function given in TF format. Same arguments as `\NyquistTF`. For example, the command

```
\NicholsTF[plot/{green,thick,samples=2000}]
{num/{10,2,2.6,0},den/{1,1,100.25},d/0.01}
{0.001}{100}
```

generates the Nichols chart in Figure 9.

`NicholsChart`      `\begin{NicholsChart}[⟨axis-options⟩]{⟨min-frequency⟩}{⟨max-frequency⟩}`  
                           `\addNichols...`  
                           `\end{NicholsChart}`

The `NicholsChart` environment works in conjunction with the parametric function generator macros `\addNicholsZPKChart` and `\addNicholsTFChart`. If supplied, `axis-options` are passed directly to the `axis` environment and the frequency limits are translated to the x-axis limits and the domain of the `axis` environment.

`\addNicholsZPKChart`      `\addNicholsZPKChart[⟨plot-options⟩]`  
                            $\{ \langle z/\langle zeros \rangle \rangle, p/\langle poles \rangle \rangle, k/\langle gain \rangle \rangle, d/\langle delay \rangle \}$

Generates a two parametric functions for the magnitude and the phase a transfer function in ZPK form. The generated magnitude and phase parametric functions

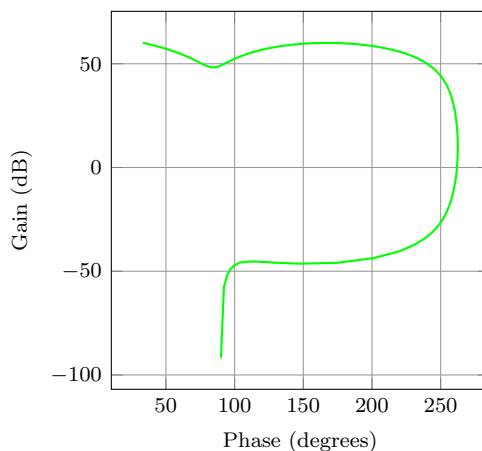


Figure 9: Output of the `\NyquistZPK` macro.

are passed to the `\addplot` macro. This macro can be used inside any `axis` environment as long as a domain for the x-axis is supplied through either the `plot-options` interface or directly in the optional argument of the container `axis` environment. Use with the `NicholsChart` environment supplied with this package is recommended. The mandatory argument is the same as `\BodeZPK`.

`\addNicholsTFChart`      `\addNicholsTFChart[<plot-options>]`  
                               `{<num/<coeffs><coeffs><delay>`

Similar to `\addNicholsZPKChart`, with a transfer function input in the TF form.

## Change History

v1.0  
   General: Initial release . . . . . 1