

MIMO Toolbox
For Use with MATLAB

Oskar Vivero

About the toolbox

The MIMO Toolbox is a collection of MATLAB functions and a GUI. Its purpose is to complement the Control Toolbox for MATLAB with functions capable of handling the multivariable input-output scheme. All the results and examples except for example 1.1.2.1 were obtained with the MIMO Toolbox and were corroborated with the bibliography.

April, 2006

Installation

The installation is straightforward just copy the directory “Mimotools” and add the path to the MATLAB search path.

See path, in the MATLAB documentation for more information.

Requirements

The MIMO Toolbox was created in Matlab 7.1 (R14) SP3, and requires the Symbolic and Control Toolboxes.

Contact

Oskar Vivero

oskar.vivero@gmail.com

Contents

1. Theory behind the functions	1
1.1. SISO Systems	1
1.1.1. Feedback Basic Concepts	1
1.1.2. Nyquist's Stability Criterion	2
1.2. MIMO Systems	4
1.2.1. Poles and Zeros of a MIMO System	4
1.2.1.1. Smith-McMillan Transformation	4
1.2.2. Stability of MIMO Systems	7
1.2.2.1. Generalized Nyquist's Stability Criterion	7
1.2.3. Treating a MIMO System with SISO techniques	10
1.2.3.1. Coupling degree and pairings of inputs and outputs	10
1.2.3.2. Nyquist's Arrays and Gershgorin Bands	10
1.2.3.3. Relative Gain Array (RGA)	11
1.2.3.4. Individual Channel Design (ICD)	12
2. Function Reference	14
2.1. The Symbolic Transfer Function	14
2.2. Function Description	14
2.2.1. tf2sym	14
2.2.2. sym2tf	15
2.2.3. ss2sym	16
2.2.4. smform	16
2.2.5. rga	17
2.2.6. nyqmimo	17
2.2.7. m_circles	19
2.2.8. icdtool	19
2.2.9. gershband	20
2.2.10. arrowh	21
3. References	22

1. Theory behind the functions

The aim of this chapter is to introduce the MIMO control theories so that one can understand both, the algorithm behind each function and its proper use. This chapter is only intended to provide a brief description of such theories and it's recommended that the user refers to the bibliography listed at the end of this document.

1.1 SISO Systems

1.1.1 Feedback Basic Concepts

Assuming a lineal process that is time-invariant whose behavior is defined by lineal differential equations with constant coefficients:

$$\ddot{y} + a_1 \dot{y} + a_2 y = b_1 \dot{u} + b_2 u$$

where $y(t)$ is the output signal and $u(t)$ is the input signal, its possible to obtain a transfer function by applying the Laplace's Transform

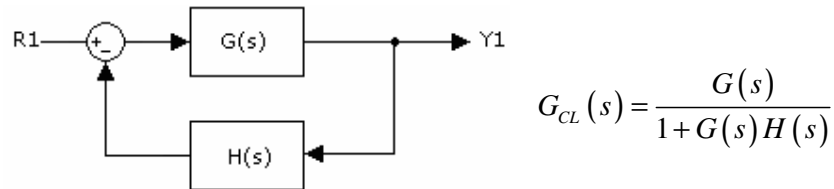
$$\frac{Y(s)}{U(s)} = G(s) = \frac{N(s)}{D(s)} = \frac{b_1 s + b_2}{s^2 + a_1 s + a_2}$$

If $s = s_z \Rightarrow G(s_z) = 0$ then s_z is defined as a zero.

If $s = s_p \Rightarrow G(s_p) = \infty$ then s_p is defined as a pole.

If $G(s)$ is rational, usually $D(s)$ determines the dynamic characteristics of the system, unless there exist cancellations between $N(s)$ and $D(s)$.

Let $H(s)$ and $G(s)$ be two transfer functions



The stability of the system in a closed-loop configuration is given by $1 + G(s)H(s)$ if and only if there is no cancellation of instabilities. For any design, it's possible to verify its stability by finding the singularities of $G_{CL}(s)$. If any of the singularities is located in $+\mathbb{R}^2$ or near the imaginary axis, it's almost impossible to determine the modifications needed on either $G(s)$ or $H(s)$ to avoid the location of the singularities. The Nyquist's stability criterion provides a tool for solving the problem

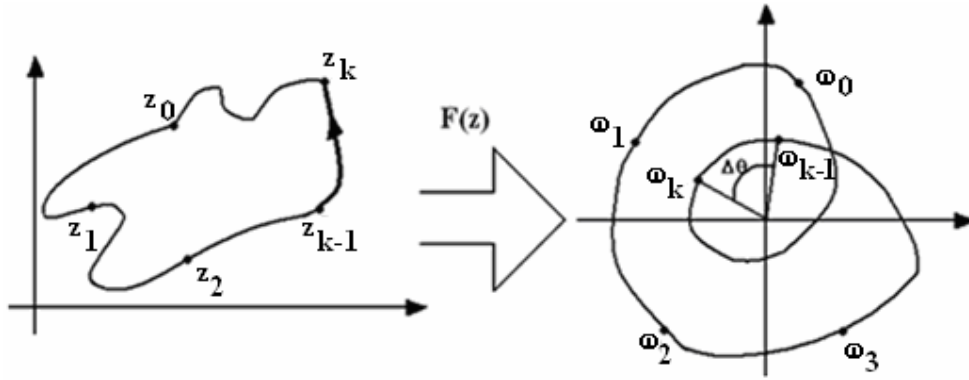
1.1.2 Nyquist's Stability Criterion

The system in a closed-loop configuration is stable if and only if the trajectory of the Nyquist diagram of $G(j\omega)H(j\omega)$ from $-\infty < \omega < \infty$ surrounds the point $(-1,0)$ in a counter-clockwise direction as much times as $G(s)H(s)$ has unstable poles.

Theorem – suppose that a function $f(z)$ is meromorphic in a simply connected domain D , and that C is a simple closed positively oriented contour in D such that $f(z)$ does not contain any singularities. Then

$$N = \frac{1}{i2\pi} \oint_C \frac{f'(z)}{f(z)} dz = Z_f - P_f$$

where N is the winding number, Z_f is the number of zeros inside the contour and P_f is the number of poles inside the contour.



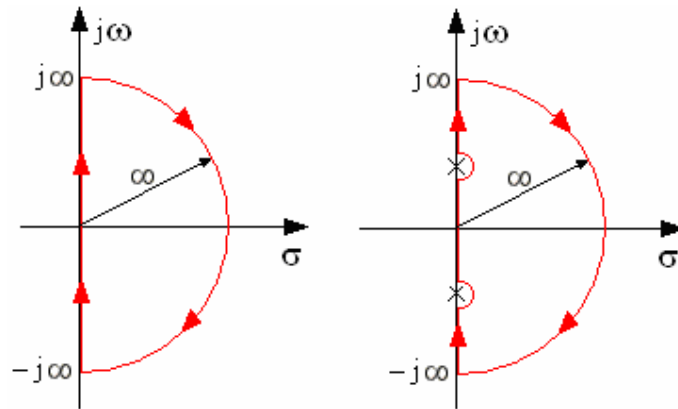
Example 1.1.2.1

The image of the circle of radius 2 centered at the origin under $f(z) = z^2 + z$ is the curve $g(x, y) = (4\cos(2t) + 2\cos(t), 4\sin(2t) + 2\sin(t))$. Note that the curve $g(x, y)$ winds up twice around the origin. We check this by computing

$$N = \frac{1}{i2\pi} \int_C \frac{f'(z)}{f(z)} dz ; \text{ Singularities at } z_0 = 0 \text{ and } z_1 = -1$$

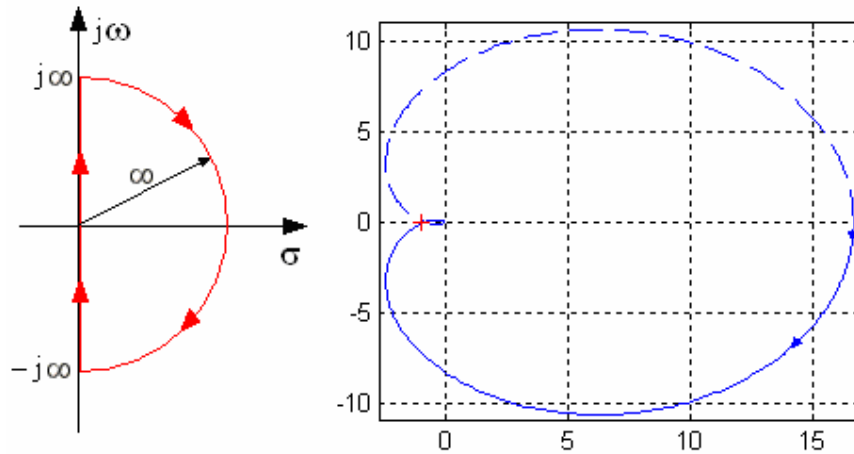
$$N = \text{Res}\left(\frac{2z+1}{z^2+z}\right)_{z_0} + \text{Res}\left(\frac{2z+1}{z^2+z}\right)_{z_1} = 2$$

Having defined N , it's important to define a useful contour for the stability analysis. It's possible to know from the root locus analysis and the time response that the unstable poles are at the right side of the S-plane. Since the zeros of the open loop system are the poles of the closed loop system, we'll focus on finding the unstable zeros through Nyquist frequency analysis. The contours that we'll consider are:



Example 1.1.2.2 – [4] pp. 620

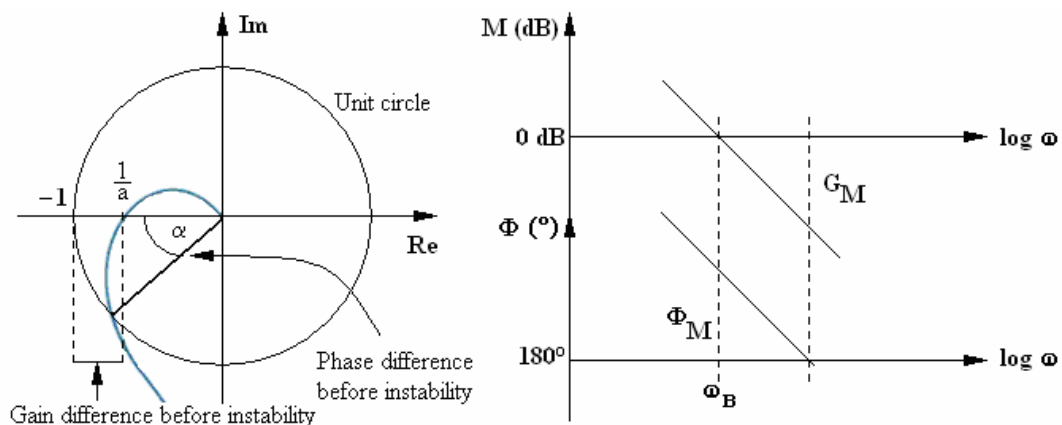
Draw the Nyquist contour and diagram of $G(s) = \frac{500}{(s+1)(s+3)(s+10)}$



Once the stability of a system has been defined through the Nyquist's diagram, the robustness of the system can be defined by two quantities, the gain and phase margins.

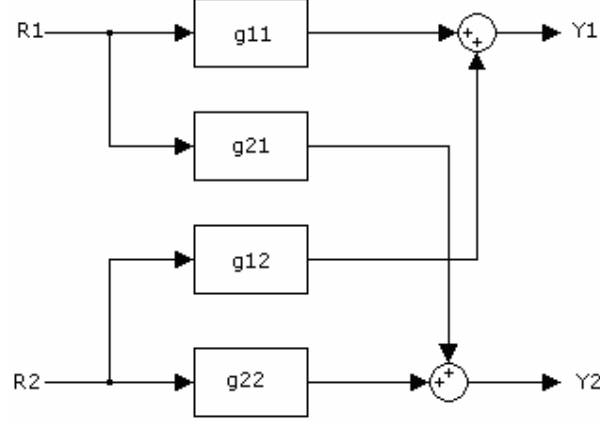
Gain margin (G_M) - the change of gain in open loop needed to obtain a phase shift at 180° that turns the system unstable.

Phase margin (Φ_M) - the phase shift in open loop needed to turn the system unstable with a unit gain.



1.2 MIMO Systems

The basic description of a multivariable system is through a transfer function matrix (TFM), whose elements $g_{i,j}$ represent the i -est output with the j -est input. The elements $g_{i,j}$ are individual transfer functions.



1.2.1 Poles and zeros of a MIMO system

The poles and zeros of multivariable systems can be defined in several (not all equivalent) ways, but the definitions that yield the most significant consequences are given by [1]:

The zeros of a transfer function matrix $H(s)$, are the roots of the (nonzero) numerator polynomials $\{\varepsilon_i(s)\}$ in the Smith-McMillan form of $H(s)$. The Smith-McMillan form allows us to give a physical interpretation of the zeros. If s_z is a zero, then the Smith-McMillan form of $H(s)$ will lose rank at $s = s_z$.

The poles of a transfer function matrix $H(s)$, are the roots of the denominator polynomials in the Smith-McMillan form of $H(s)$.

1.2.1.1 Smith – McMillan Transformation

Given a rational matrix $H(s)$

$$H(s) = \frac{N(s)}{d(s)}$$

where $d(s)$ is the monic least common multiple of the denominators of $H(s)$

Then $d(s)H(s) = N(s)$ is a polynomial matrix, so that we can write,

$$d(s)H(s) = N(s) = U_1(s)\Lambda(s)U_2(s)$$

where the $\{U_i(s)\}$ are unimodular matrices and $\Lambda(s)$ is in Smith form

$$U_1^{-1}(s)H(s)U_2^{-1}(s) = \frac{\Lambda(s)}{d(s)} = \text{diag} \left\{ \frac{\lambda_i(s)}{d(s)} \right\}$$

$$\frac{\lambda_i(s)}{d(s)} = \frac{\varepsilon_i(s)}{\psi_i(s)}$$

where $\{\varepsilon_i(s), \psi_i(s)\}$ are coprime, $i = 1, \dots, r$

and r is the (normal) rank of $H(s)$

Then we can write

$$H(s) = U_1(s)M(s)U_2(s)$$

where $M(s)$ is the Smith-McMillan transformation of $H(s)$ given by

$$M(s) = \begin{bmatrix} \text{diag} \left\{ \frac{\varepsilon_i(s)}{\psi_i(s)} \right\} & 0 \\ 0 & 0 \end{bmatrix}$$

Smith Form

For any $p \times m$ polynomial matrix $P(s)$ we can find elementary row and column operations, or corresponding unimodular matrices $\{U(s), V(s)\}$ such that

$$U(s)P(s)V(s) = \Lambda(s)$$

where

$$\Lambda(s) = \begin{bmatrix} \lambda_1(s) & & & 0 \\ & \ddots & & \vdots \\ & & \lambda_r(s) & 0 \\ 0 & \dots & 0 & 0 \end{bmatrix}$$

$r = \text{the (normal) rank of } P(s)$

and the $\{\lambda_i(s)\}$ are unique monic polynomials obeying a division property

$$\lambda_i(s) \mid \lambda_{i+1}(s), \quad i = 1, \dots, r-1$$

Moreover, if

$$\Delta_i(s) = \text{the gcd of all } i \times i \text{ minors of } P(s)$$

then

$$\lambda_i(s) = \frac{\Delta_i(s)}{\Delta_{i-1}(s)}, \quad \Delta_0(s) = 1$$

The matrix $\Lambda(s)$ is called the Smith form of $P(s)$

Example.1.2.1.1 – [1] pp. 444-446

Find the poles and zeros of

$$G(s) = \frac{1}{(s+1)^2 (s+2)^2} \begin{bmatrix} s & s(s+1)^2 \\ -s(s+1)^2 & -s(s+1)^2 \end{bmatrix}$$

Solution

Given

$$G(s) = \frac{1}{d(s)} N(s)$$

where

$$d(s) = lcm\{(s+1)^2 (s+2)^2\} = (s+1)^2 (s+2)^2$$

$$N(s) = \begin{bmatrix} s & s(s+1)^2 \\ -s(s+1)^2 & -s(s+1)^2 \end{bmatrix}$$

Finding $\Lambda(s)$ in Smith form:

$$\Delta_0(s) = 1$$

$$\Delta_1(s) = \gcd(s, s(s+1)^2, -s(s+1)^2, -s(s+1)^2) = s$$

$$\Delta_2(s) = \gcd(s^3 (s+1)^2 (s+2))$$

$$\varepsilon_1(s) = \lambda_1(s) = s$$

$$\varepsilon_2(s) = \lambda_2(s) = s^2 (s+1)^2 (s+2)$$

Therefore

$$M(s) = \begin{bmatrix} \text{diag} \left\{ \frac{\varepsilon_i(s)}{\psi_i(s)} \right\} & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \frac{s}{(s+1)^2 (s+2)^2} & 0 \\ 0 & \frac{s^2}{s+2} \end{bmatrix}$$

The poles are defined as

$$p(s) = \psi_1(s) \cdot \psi_2(s) = (s+1)^2 (s+2)^2 \cdot (s+2) = 0$$

$$p = [-1 \quad -1 \quad -2 \quad -2 \quad -2]$$

The zeros are defined as

$$z(s) = \varepsilon_1(s) \cdot \varepsilon_2(s) = s^3 = 0$$

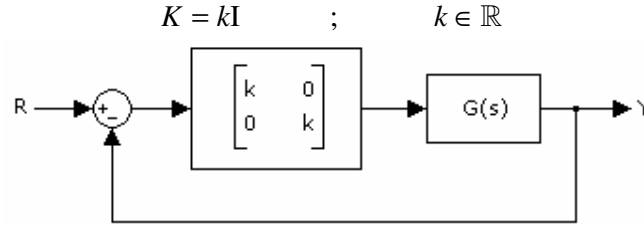
$$z = [0 \quad 0 \quad 0]$$

1.2.2 Stability of MIMO Systems

Having defined a tool in order to obtain the poles and zeros of a MIMO system, it's necessary to define if the system is stable. This can be achieved by the generalized Nyquist's stability criterion, which is an adaptation from the Nyquist's stability criterion for SISO systems.

1.2.2.1 Generalized Nyquist's Stability Criterion

Let $G(s)$ be a rational TFM, assuming that it has no cancellations between poles and zeros. Let K be a compensator with a negative feedback loop, that is



Let the $\det[I + KG(s)]$ have P poles and Z zeros in the right hand plane (RHP), then like in SISO systems

$$\Delta \arg \det[I + KG(s)] = -2\pi(Z - P)$$

where $\Delta \arg$ is the change of phase given by s along the Nyquist's contour. In order to obtain stability in a closed loop, it is required that the diagram mapped by $\det[I + KG(s)]$ surrounds the origin P times.

As a difference from the SISO case, the Nyquist diagram of each k of interest must be obtained.

If $\lambda_i(s)$ is an eigenvalue of $G(s)$, then $k\lambda_i(s)$ is an eigenvalue of $KG(s)$, and therefore, $1 + k\lambda_i(s)$ is an eigenvalue of $I + KG(s)$. Then

$$\det[I + KG(s)] = \prod_i [1 + k\lambda_i(s)]$$

and

$$\Delta \arg \det[I + KG(s)] = \sum_i \Delta \arg [1 + k\lambda_i(s)]$$

Therefore, the stability in a closed loop system can be inferred by the number of wind ups to the point $(-1,0)$ given by the Nyquist diagram of $k\lambda_i(s)$. The Nyquist diagram of $\lambda_i(s)$ are known as the characteristic graphs of $G(s)$.

Theorem – If $G(s)$ has right hand plane poles (RHPP), P , given by the Smith-McMillan transformation, then the closed loop with negative feedback is stable if and only if the characteristic graphs of $KG(s)$ surround the point $(-1,0)$ P times in a counter-clockwise direction, assuming that there was no cancellations of instabilities.

Example 1.2.2.1 – [2] pp. 61

Find the values of $k > 0$ in order to keep $G(s)$ from becoming unstable

$$G(s) = \frac{1}{1.25(s+1)(s+2)} \begin{bmatrix} s-1 & s \\ -6 & s-2 \end{bmatrix}$$

Solution

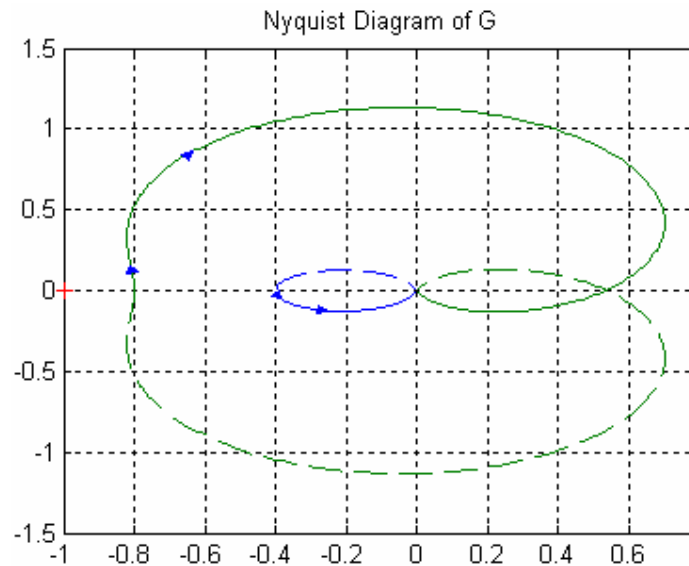
It's easy to verify that the open loop poles of $G(s)$ are $P = [-1 \ -2]$, therefore, in order to maintain the closed loop system stable, we must ensure that the number of surroundings of the characteristic graphs of $G(s)$ is equal to zero.

Assuming $k = 1$, then

$$\Lambda = \det[\lambda I - KG(s)] = 0$$

$$\Lambda = \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} \frac{2(2s-3+\sqrt{-24s+1})}{5(s^2+3s+2)} \\ \frac{2(2s-3-\sqrt{-24s+1})}{5(s^2+3s+2)} \end{bmatrix}$$

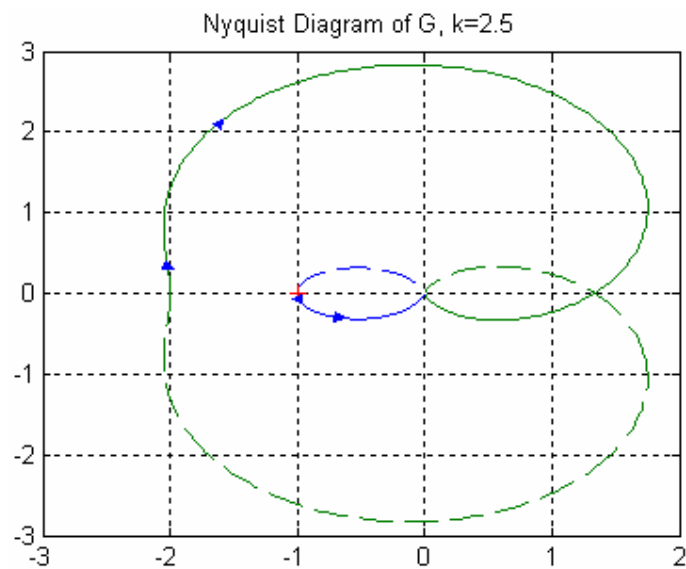
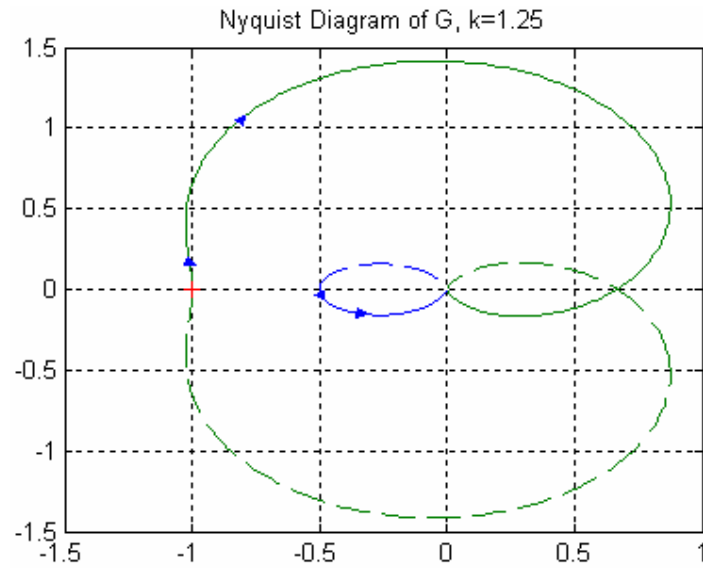
From the eigenvalues of $G(s)$, its possible to obtain the Generalized Nyquist Diagram when $k = 1$



From the diagram it's possible to obtain the critical points in which we can calculate the values of k , in order to keep the system stable. Knowing that the critical points are in -0.8 and -0.4 , therefore

$$k < \left(\frac{1}{0.8} = 1.25 \right), k > \left(\frac{1}{0.4} = 2.5 \right)$$

This can be proven by setting the values of k equal to 1.25 and 2.5 and obtaining the Nyquist Diagram.



1.2.3 Treating a MIMO system with SISO techniques

1.2.3.1 Coupling degree and pairing of inputs and outputs

Given the coupling degree of a MIMO system, it's possible to apply certain SISO techniques for designing controllers. The degree of coupling can be found by several techniques, such as the Nyquist's arrays and the Gershgorin bands, the relative gain array (RGA), and the individual channel design (ICD). In some cases, it is possible to cross couple inputs and outputs in order to obtain a less coupled system.

1.2.3.2 Nyquist's Arrays and Gershgorin bands

The Nyquist's array of a TFM $G(s)$, is an array of graphs, where the i, j -est graph is the Nyquist Diagram of $g_{i,j}$.

Gershgorin Theorem – let Z a $n \times n$ complex matrix. Then, the eigenvalues of Z fall on the union of circles with center at $z_{i,i}$ with radius

$$\sum_{\substack{j=1 \\ j \neq i}}^m |z_{i,j}|$$

and on the circles with center at $z_{j,j}$ with radius

$$\sum_{\substack{i=1 \\ i \neq j}}^m |z_{i,j}|$$

Gershgorin Bands

Over the Nyquist's diagram of $g_{i,i}(s)$, on each point, we super impose a circle of radius

$$\sum_{\substack{i=1 \\ i \neq j}}^m |g_{i,j}(j\omega)| \quad \text{or} \quad \sum_{\substack{i=1 \\ i \neq j}}^m |g_{j,i}(j\omega)|$$

The bands obtained in this way are known as the Gershgorin Bands.

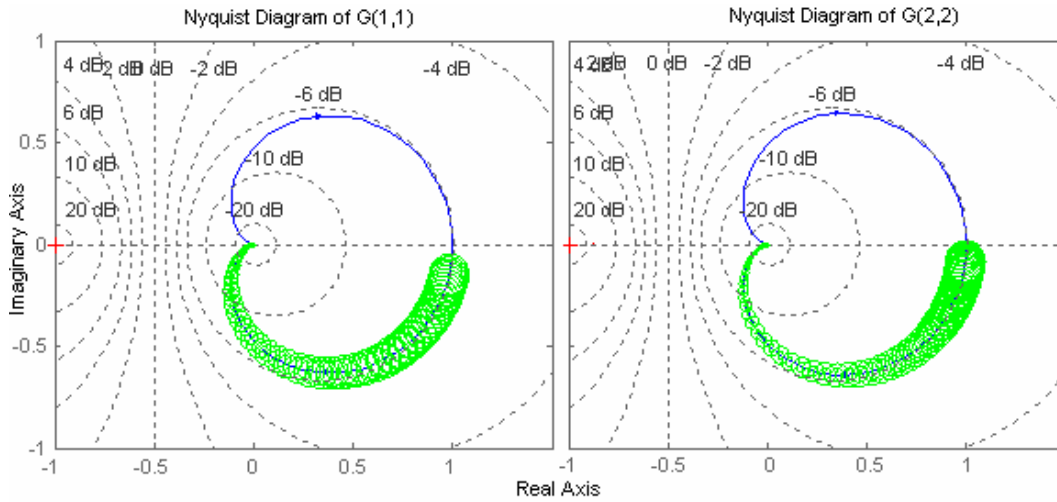
By the Gershgorin theorem it's possible to know that the bands trap the unions of the Nyquist diagram. More over, it's possible to demonstrate that the bands occupy different regions, therefore there will be as many Nyquist's diagrams trapped in a region as many Gershgorin bands are there.

Then, by counting the number of wind ups that the Gershgorin bands do around the point $(-1,0)$, its possible to determine the stability of the MIMO system.

If the Gershgorin bands are thin and exclude the origin, it is said that $G(s)$ is diagonally dominant which can be interpreted as a decoupled system.

Example 1.2.3.1– [3] pp. 657
Find the Gershgorin bands of

$$G(s) = \begin{bmatrix} \frac{2}{s^2 + 3s + 2} & \frac{0.1}{s + 1} \\ \frac{0.1}{s^2 + 2s + 1} & \frac{6}{s^2 + 5s + 6} \end{bmatrix}$$



From the graphs above, it's possible to observe that the system is stable and highly decoupled. Therefore, it can be managed as two independent SISO systems with small disturbances due to the small coupling degree.

1.2.3.3 Relative Gain Array (RGA)

To measure the degree of coupling or interaction in a system, the concept of relative gain array can be used. For an arbitrary $n \times n$ matrix A , it is defined as

$$\text{RGA}(A) = A * (A^{-1})^T$$

The RGA matrix has a number of interesting properties

- The sum of the elements of any row or column is always 1
- RGA is independent of any scaling
- The sum of the absolute values of all elements in $\text{RGA}(A)$ is a good measure of A 's true condition number, i.e., the best condition number that can be achieved in the family $D_1 A D_2$, where D_i are diagonal matrices.
- Permutation of the rows or columns of A leads to permutations of the corresponding rows or columns of $\text{RGA}(A)$.

Clearly RGA of a diagonal matrix is the unit matrix. The deviation of RGA from the unit matrix can be taken as a measure of the coupling degree between $y = Ax$.

For a $n \times m$ matrix, the RGA is defined as

$$\text{RGA}(A) = A \cdot (A^\dagger)^T$$

where A^\dagger denotes the pseudo inverse of A .

Example 1.2.3.2– [3] pp. 657

Let $G(s)$ be

$$G(s) = \begin{bmatrix} \frac{2}{s^2 + 3s + 2} & \frac{0.1}{s + 1} \\ \frac{0.1}{s^2 + 2s + 1} & \frac{6}{s^2 + 5s + 6} \end{bmatrix}$$

$$\text{RGA}(G(s)) = \begin{bmatrix} 1.0101 & -0.0101 \\ -0.0101 & 1.0101 \end{bmatrix}$$

Since $\text{RGA}(G(s)) \rightarrow I$, it's possible to observe that the system is highly decoupled. This corroborates the result given by the Gershgorin bands in example 1.2.3.1.

1.2.3.4 Individual Channel Design (ICD)

As a difference from RGA and the Gershgorin Bands which are useful tools for calculating the cross-coupling of a system, the Individual Channel Design is a framework in which Bode/Nyquist techniques can be applied directly to the channels not only when cross-coupling is weak but in all circumstances, including when cross-coupling is strong. The multivariable system is decomposed into an equivalent set of SISO systems. Each SISO system is the open-loop channel transmittance between output $Y_i(s)$ and input $R_i(s)$ with all internal loops closed. [6]

Let $G(s)$ be a 2×2 system with a controller of the type $K(s) = \text{diag}[k_i(s)]$ then the relationships of $\frac{Y_i(s)}{R_i(s)}$ can be established as

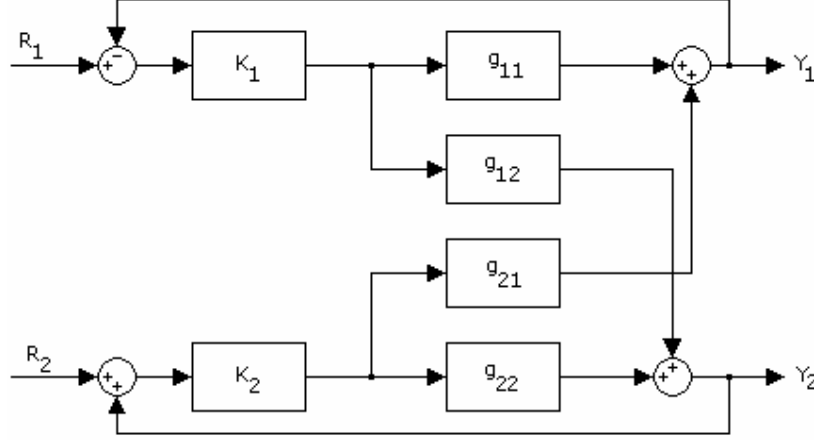
$$C_1 = \frac{Y_1(s)}{R_1(s)} = k_1 g_{11} (1 - \gamma h_2)$$

$$C_2 = \frac{Y_2(s)}{R_2(s)} = k_2 g_{22} (1 - \gamma h_1)$$

where $C_i(s)$ is the individual channel, $\gamma(s)$ is the multivariable structure function (MSF) and $h_i(s)$ is the subsystem i , which are respectively defined as:

$$\gamma(s) = \frac{g_{12}g_{21}}{g_{11}g_{22}}$$

$$h_i(s) = \frac{k_i g_{ii}}{1 + k_i g_{ii}}$$



The MSF $\gamma(s)$ is of great importance inside the ICD analysis framework, since it is capable of [6]

- Determining the dynamical characteristics between each input and each output
- It has an interpretation in the frequency domain
- Its magnitude quantifies the amount of coupling between the channels
- It can determine the transmittance zeros of the system from $1 - \gamma(s)$
- $\gamma(s) = 1$ determines the non-minimum phase conditions
- Its closeness to the point $(1,0)$ it's a key point in determining the robustness of the system

Robustness Conditions

In order to obtain a design that provides a channel that is robust and stable, the following conditions should be satisfied

1. $\gamma(s)$ should not be close to the point $(1,0)$ for all ω
2. $\gamma(s)h_i(s)$ shall be robust
3. $k_i(s)g_{ii}(s)$ shall be robust

The interaction between the discarded inputs and outputs can be observed from

$$Y_i(s) = \frac{g_{ij}(s)}{g_{jj}(s)} h_j(s) \cdot \frac{1}{1 + C_i(s)} \cdot R_j(s)$$

2. Function Reference

The aim of this chapter is to give a brief description of the functions used in the MIMO toolbox. For the users that are new to the Matlab environment it is recommended to review the getting started documentation.

2.1 The Symbolic Transfer Function

The Control Toolbox in Matlab posses an object class that describes a transfer function. This model representation is numerical and it's sensitive to floating point errors due to arithmetic operations such as inversion. Also, from the LTI definition, the transfer function class cannot handle nonlinearities such as square roots, trigonometric functions, etc, since it's only a rate in polynomial representation. Given such problems, a symbolic conversion for transfer function has been developed, and it's key to some of the functions inside the MIMO toolbox. This conversion enables the user to handle the transfer function and its operations as an algebraic problem, simplifying the computational difficulties that may arise for the users without a computer science background.

2.2 Function Description

2.2.1 tf2sym

Syntax

```
G_sym = tf2sym(G)
```

Description

tf2sym performs the conversion from a numeric to a symbolic representation of a transfer function. The function is capable to deal with both SISO and MIMO models. To be able to differentiate the type of transfer function, the symbolic transfer function is represented with the Laplace operator 'p' instead of 's'.

Example 1

```
>> G=tf([1 2 3],[1 5 6 7])
>> G_sym=tf2sym(G);
>> pretty(G_sym)
```

$$\frac{p^3 + 2p^2 + 3p}{p^4 + 5p^3 + 6p^2 + 7p}$$

Example 2

```
>> g11=tf([1 2],[1 2 1]);
g12=tf([1 -1],[1 5 6]);
g21=tf([1 -1],[1 3 2]);
g22=tf([1 2],[1 1]);
G=[g11 g12; g21 g22];
g=tf2sym(G);
pretty(g)
```

$$\begin{bmatrix} \frac{2 + p}{(p + 1)^2} & \frac{-1 + p}{(p + 3)(2 + p)} \\ \frac{-1 + p}{(2 + p)(p + 1)} & \frac{2 + p}{p + 1} \end{bmatrix}$$

2.2.2 sym2tf

Syntax

```
G = sym2tf(G_sym)
```

Description

`sym2tf` performs the conversion from a symbolic to a numeric representation of a transfer function. As `tf2sym`, this function is capable of dealing with both SISO and MIMO models. This conversion cannot handle nonlinearities since the numeric transfer function class 'tf' is not capable of dealing with the nonlinearities.

Example 1

```
>> syms p
>> G_sym = (p^2 + 2*p + 3)/(p^3 + 5*p^2 + 6*p + 7);
>> G=sym2tf(G_sym)
```

```
Transfer function:
      s^2 + 2 s + 3
      -----
s^3 + 5 s^2 + 6 s + 7
```

Example 2

```
>> g11=(p + 2)/(p^2 + 2*p + 1);
    g12=(p - 1)/(p^2 + 5*p + 6);
    g21=(p - 1)/(p^2 + 3*p + 2);
    g22=(p + 2)/(p + 1);
    g=[g11 g12; g21 g22];
    G=sym2tf(g)
```

```
Transfer function from input 1 to output...
```

```
      s + 2
#1:  -----
      s^2 + 2 s + 1
```

```
      s - 1
#2:  -----
      s^2 + 3 s + 2
```

```
Transfer function from input 2 to output...
```

```
      s - 1
#1:  -----
      s^2 + 5 s + 6
```

```
      s + 2
#2:  -----
      s + 1
```

2.2.3 ss2sym

Syntax

```
G = ss2sym(A,B,C,D)
```

Description

ss2sym performs the conversion from a state space representation to a symbolic transfer function.

Example

```
>> A = [0 1 0 0 0; 0 0 1 0 0; -2 -5 -4 0 0; 0 0 0 0 1; 0 0 0 -3 -4];
      B = [0 0; 0 0; 1 0; 0 0; 0 1];
      C = [1 2 1 9 3; 14 9 1 1 1];
      D = [0 1; 0 0];
      g=ss2sym(A,B,C,D);
      pretty(g)
```

```
[      1      p + 4]
[ -----]
[ 2 + p      p + 1]
[      ]
[ p + 7      1 ]
[-----]
[      2      p + 3]
[ (p + 1)      ]
```

2.2.4 smform

Syntax

```
[M,poles,zeros] = smform(G)
```

Description

[M,poles,zeros] = smform computes the Smith-McMillan transformation, the poles and zeros of a MIMO TFM. The algorithm obeys directly the theory specified on the section 1.2.2.1 of this document by establishing an active link with the Maple kernel.

Example

```
>> g11=tf([1 1],[1 3 2]);
      g12=tf([1 4],[1 1]);
      g21=tf([1 7],[1 2 1]);
      g22=tf([1 2],[1 5 6]);
      G=[g11 g12 ; g21 g22];
      [M,poles,zeros]=smform(G);
```

```
>> pretty(M)
```

```
[      1      ]
[-----]
[      2      ]
[ (p + 3) (p + 1) (2 + p) ]
[      ]
[      4      3      2      ]
[      p  + 15 p  + 86 p  + 203 p + 167 ]
[      0      ]
[-----]
[      p + 1      ]
```

```
>> poles

poles =

    -3.0000
    -2.0000
    -1.0000
    -1.0000 + 0.0000i
    -1.0000 - 0.0000i

>> zeros

zeros =

    -5.3101 + 2.5439i
    -5.3101 - 2.5439i
    -2.1899 + 0.1460i
    -2.1899 - 0.1460i
```

2.2.5 rga

Syntax

A=rga(G)

Description

rga computes the Relative Gain Array of a MIMO TFM. The algorithm obeys directly the theory specified on the section 1.2.3.3

Example

```
>> g11=tf([1 2],[1 2 1]);
    g12=tf([1 -1],[1 5 6]);
    g21=tf([1 -1],[1 3 2]);
    g22=tf([1 2],[1 1]);
    G=[g11 g12; g21 g22];
    A=rga(G)
```

```
A =

    1.0213    -0.0213
   -0.0213     1.0213
```

2.2.6 nyqmimo

Syntax

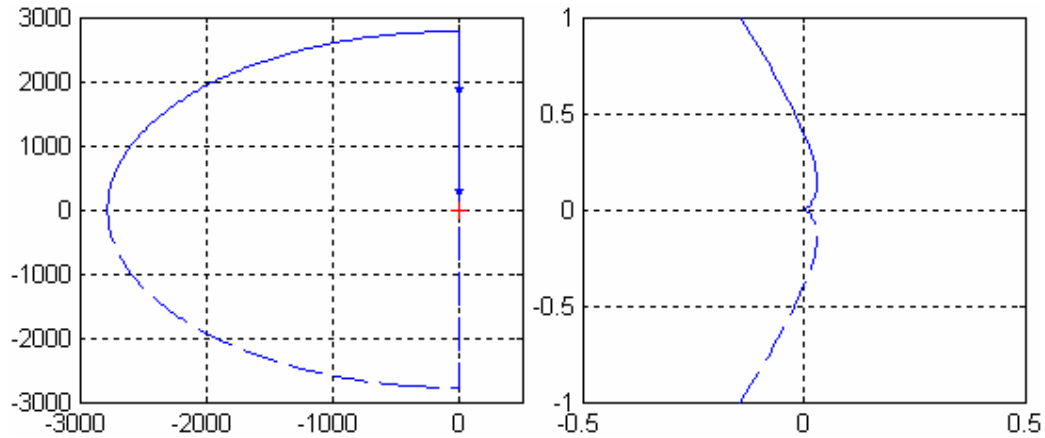
nyqmimo(G)

Description

nyqmimo computes the Nyquist's Diagram based on the type of transfer function at the input. nyqmimo distinguishes between transfer functions with singularities at the origin, proper, strictly proper and strictly improper transfer functions, based on that structure, it maps the Nyquist's Diagram according to the Nyquist's Contour. nyqmimo is capable of computing the Nyquist's Diagram for SISO and the Generalized Nyquist's Diagram for MIMO systems.

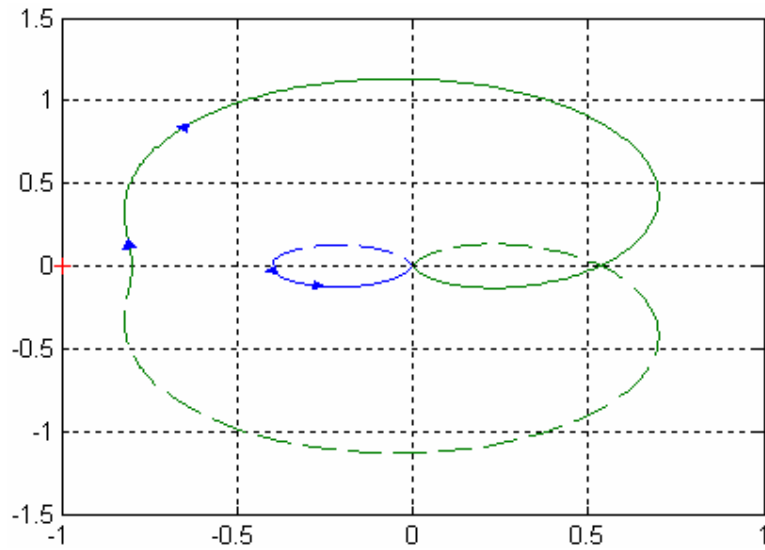
Example 1

```
>> G=tf([1 25],[1 5 3 -9 0]);
nyqmimo(G)
```



Example 2

```
>> den=1.25*conv([1 1],[1 2]);
g11=tf([1 -1],den);
g12=tf([1 0],den);
g21=tf([-6],den);
g22=tf([1 -2],den);
G=[g11 g12;g21 g22];
nyqmimo(G)
```



2.2.7 m_circles

Syntax

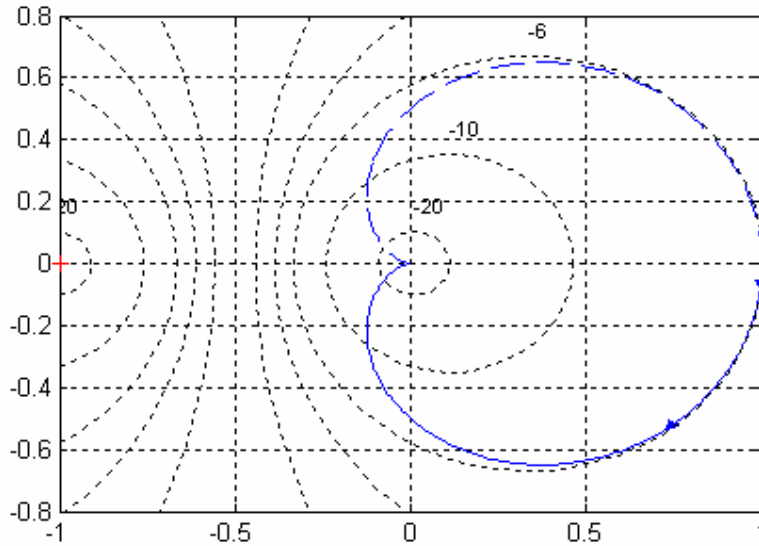
m_circles

Description

m_circles superimposes the M circles of magnitude -2dB to 20dB. This function is useful to measure the gain margin on a Nyquist's Diagram.

Example

```
>> G=tf(1,[1 2 1]);
>> nyqmimo(G)
Verifying for singularities on the origin...
Setting frequency range...
Mapping...
Plotting...
>> m_circles
```

**2.2.8 icdtool****Syntax**

```
icdtool(G)
```

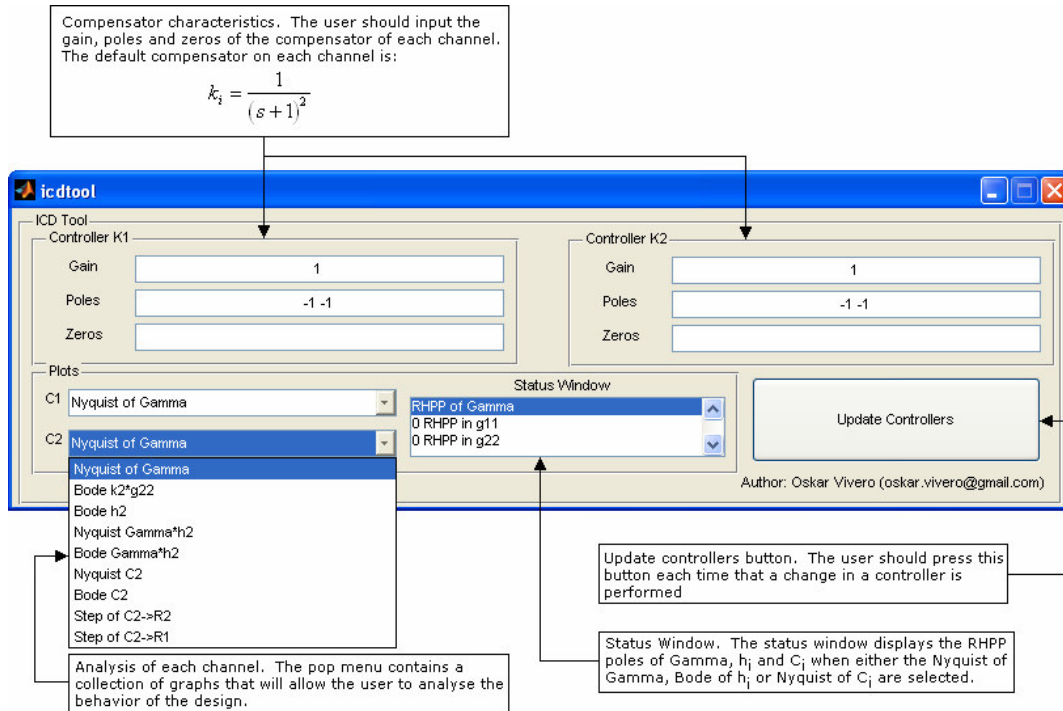
Description

`icdtool` is a GUI designed to help the user in the task of designing controllers for a 2×2 MIMO system with under the Individual Channel

Design Scheme.

By default, `icdtool` calculates $\gamma(s)$, when just loaded, `icdtool` will not load the default controller. The controller will only be loaded and all the proper calculations performed after the user has clicked on the UPDATE CONTROLLERS button. If the user doesn't know entirely the behavior of the system, it is recommended that they load a unitary gain proportional controller, in that way, the user will be able to analyze both, $\gamma(s)$ and the diagonal elements of the system, $g_{1,1}(s)$, $g_{2,2}(s)$. It is possible to load a proportional controller by clearing the poles and zeros entries.

MIMO Toolbox



2.2.9 gershband

Syntax

gershband(G) - Gershgorin bands of G

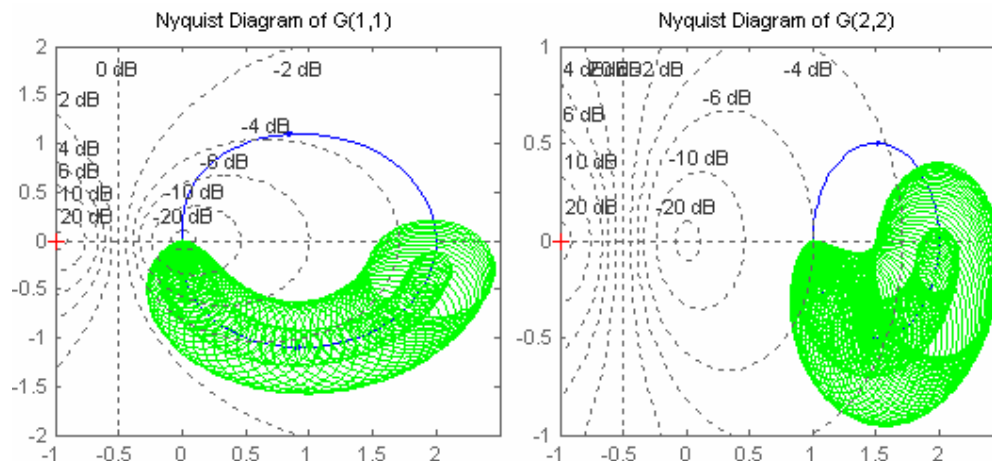
gershband(G,'v') - Gershgorin bands and Nyquist Array of G

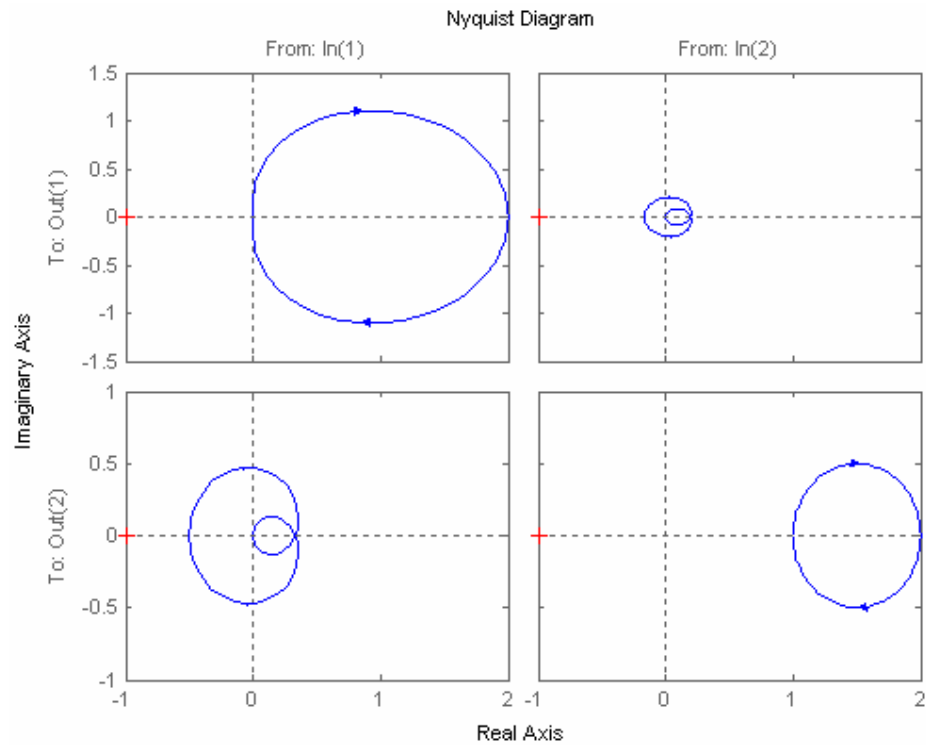
Description

gershband computes the Gershgorin Bands of a $n \times n$ MIMO system along with the Nyquist's Array.

Example

```
>> g11=tf([1 2],[1 2 1]);      g12=tf([1 -1],[1 5 6]);
    g21=tf([1 -1],[1 3 2]);      g22=tf([1 2],[1 1]);
    G=[g11 g12; g21 g22];
    gershband(G,'v');
```





2.2.10 arrowh

Syntax

`Arrowh(x,y,color,size,location)`

Description

`arrowh` draws a solid arrowhead in the current plot

Author: Florian Knorn

Email: florian.knorn@student.uni-magdeburg.de , florian@knorn.org

Homepage: <http://www.florian-knorn.com/>

3. References

- [1] Kailath, Thomas, "Linear Systems", Prentice Hall, 1980, pp. 390-392, 443-450
- [2] Maciejowski, J.M., "Multivariable Feedback Design", Addison-Wesley, 1989, pp. 37-71
- [3] Goodwin, Graham C., Graebe, Stefan F., Salgado, Mario E., "Control System Design", Prentice Hall, 2001, pp. 653-671
- [4] Nise, Norman S., "Sistemas de Control para Ingeniería", CECSA, 2002, pp. 614-635
- [5] Ogata, Katsuhiko, "Ingeniería de Control Moderna", Prentice-Hall, 2003, 523-566
- [6] Carlos E. Ugalde-Loo, Eduardo Licéaga-Castro, Jesús Licéaga-Castro, "2x2 Individual Channel Design MATLAB® Toolbox" Proceedings of the 44th IEEE CDC, pp. 7603-7608
- [7] J. Liceaga, E. Liceaga and L. Amézquita, "Multivariable Gyroscope Control by Individual Channel Design", *Control Systems Society CCA*, pp. 110-115, 2005
- [8] E. Liceaga-Castro and J. Liceaga-Castro, "Submarine depth control by individual channel design", *Proceedings of the 37th IEEE CDC*, e, pp. 3183-3188, 1998
- [9] J. Licéaga-Castro, C. Ramírez-España and E. Licéaga-Castro, "GPC control design for a Temperature and Humidity Prototype using ICD análisis", Submitted for publication.