1 Block Diagram Algebra

In series: $\Sigma = \Sigma_1\Sigma_2$
In parallel: $\Sigma = \Sigma_1 + \Sigma_2$
positive feedback: $\Sigma = \frac{\Sigma_1}{I-\Sigma_1\Sigma_2}$
negative feedback: $\Sigma = \frac{\Sigma_1}{I+\Sigma_1\Sigma_2}$

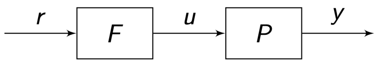
2 Modeling

2.1 Definitions

Inputs u(t): How does the outside world affect the system?
Outputs y(t): What do we observe about the system?
State/Memory x(t): How does the system change internally over time?
Disturbances d(t): External, unintentional effects on the system.
Parameters: System specifications that are fixed and do not change over time.

2.2 Basic Control Architectures

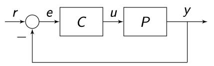
2.2.1 Feed-Forward



Feed-forward

Doesn't change it's dynamics but relies on **precise knowledge** of the plant.

2.2.2 Feedback loop



Feedback

Feedback control can **stabilize** unstable systems, reject external disturbances and **handle uncertainties** in the system.
But it can **introduce instability** and feed sensor noise into the system.

2.3 System Classification

Static / Memoryless

Output only depends on inputs at current time t.
Non-static Examples:

$$y(t) = u(t-1), \quad y(t) = \int_0^t u(\tau)\, d\tau$$

Note that for a static LTI the state vector x does not appear, hence matrices A, B, C are zero.

Causal

Output relies on past and current inputs but not on future inputs.
Non-causal systems **cannot be implemented** in the real world.
Non-causal Examples:

$$y(t) = u(t+2), \quad y(t) = \frac{d}{dt}u(t), \quad y(t) = \int_{-\infty}^t u(\tau)\, d\tau$$

For LTI: strictly causal means feedthrough D is zero.

Time-invariant

Shifting the output has the same effect as shifting the input.
Time-varying Examples:

$$y(t) = t \cdot u(t), \quad y(t) = u(\sin(t)), \quad y(t^2) = u(t)$$

Linearity

Linear systems satisfy following properties:

$$\Sigma(\alpha u_1 + \beta u_2) = \alpha\Sigma(u_1) + \beta\Sigma(u_2)$$
$$\Sigma(k \cdot u) = k \cdot \Sigma(u)$$

where Σ can be treated as a scalar:

$$\Sigma(u) = \Sigma \cdot u$$

Proper

Proper systems have TF where the denominator has a higher degree than the nominator.

Realizable

Realizable Systems need to be **proper** and **causal**.

2.4 LTI Systems

There is not many LTI systems in real world applications but a lot of real world systems can be approximated very well with LTI's.

2.4.1 State Space Model

$$\dot{x}(t) = f(x(t), u(t)) = Ax(t) + Bu(t)$$
$$y(t) = h(x(t), u(t)) = Cx(t) + Du(t)$$

2.4.2 Time Response

$$x(t) = e^{At}x_0 + \int_0^t e^{A(t-\tau)}Bu(\tau)\, d\tau$$
$$y(t) = Ce^{At}x_0 + C\int_0^t e^{A(t-\tau)}Bu(\tau)\, d\tau + Du(t)$$

Where $y(t)$ is composed y_{IC} (the first part) and y_F, its second part.
$Du(t)$ is called **feedthrough**.

2.4.3 Matrix Exponentials

The inverse of a 2×2 matrix A is computed as follows:

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad \Rightarrow \quad A^{-1} = \frac{1}{ad-bc}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

If A is in diagonal form, e^{At} can be simplified to:

$$\exp\left(\begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} t\right) = \begin{bmatrix} e^{\lambda_1 t} & 0 \\ 0 & e^{\lambda_2 t} \end{bmatrix}$$

If A is not diagonal but diagonalizable, e^{At} is computed using:

$$\det\left(\begin{bmatrix} a-\lambda & b \\ c & d-\lambda \end{bmatrix}\right) = (a-\lambda)(d-\lambda) - bc = 0$$
$$e^{At} = \text{diag}\left[e^{\lambda_1 t}, e^{\lambda_2 t}\right]$$

If A is in Jordan form, e^{At} can be simplified to:

$$\exp\left(\begin{bmatrix} \lambda & 1 \\ 0 & \lambda \end{bmatrix} t\right) = e^{\lambda t}\begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix}$$
$$\exp\left(\begin{bmatrix} \lambda & 1 & 0 \\ 0 & \lambda & 1 \\ 0 & 0 & \lambda \end{bmatrix} t\right) = e^{\lambda t}\begin{bmatrix} 1 & t & \frac{t^2}{2} \\ 0 & 1 & t \\ 0 & 0 & 1 \end{bmatrix}$$

2.4.4 Diagonalization

Compute eigenvector's and -values.
$T = (v_1, v_2, ..., v_n)$, $\tilde{A} = T^{-1}AT = diag(\lambda_1, \lambda_2, ..., \lambda_n)$
$\tilde{B} = T^{-1}B$, $\tilde{C} = CT$, $\tilde{D} = D$, $\tilde{x} = T^{-1}x$

2.4.5 Stability Conditions

Lyapunov stable: State will remain bounded for bounded initial condition & zero input. $Re(\lambda_i) \le 0 \quad \forall i$
Asymptotically stable: State converges to "zero" (stays constant) for bounded initial condition & zero input. $Re(\lambda_i) < 0 \quad \forall i$
BIBO stable: Output remains bounded for every bounded input.

2.4.6 Minimal LTI System

It is defined by **observability** $\mathcal{O}$ and **controllability** $\mathcal{C}$ which means that it is expressed in its simplest form while retaining all dynamic behavior and contains no unnecessary states. ($\mathcal{C}$ and $\mathcal{O}$ have full rank n)
For Minimal System **asymptotic stability = BIBO stability!**

2.5 Non-linear Systems

Hartman-Grobman Theorem

It states that if the approximated LTI system is asymptotically stable, the real, non-linear system is also locally stable. (Assuming you stay close to the equilibrium point)

2.5.1 Linearization

To linearize a non-linear system, we use the **Jacobian Linearization procedure** based on the *Taylor-Series expansion* around an equilibrium point (x_e, u_e), where $\dot{x} = f(x_e, u_e) = 0$.

$$\begin{cases} \dot{x}(t) = f(x(t), u(t)) \\ y(t) = g(x(t), u(t)) \\ x(t_0) = x_0 \end{cases} \Rightarrow \begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \\ x(0) = x_0 \end{cases}$$

$$A = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}_{(x_e, u_e)} \qquad B = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} \\ \vdots \\ \frac{\partial f_n}{\partial u} \end{bmatrix}_{(x_e, u_e)}$$

$$C = \begin{bmatrix} \frac{\partial g}{\partial x_1} & \cdots & \frac{\partial g}{\partial x_n} \end{bmatrix}_{(x_e, u_e)} \qquad D = \begin{bmatrix} \frac{\partial g}{\partial u} \end{bmatrix}_{(x_e, u_e)}$$

$A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{p \times n}$, $D \in \mathbb{R}^{p \times m}$
Where n, m and p are the dimensions of the state-, input- and output vector respectively.
Note: Since in this course we deal with SISO systems, the B and D matrices have 1 column.

3 Signal Analysis

Understand the input-output behaviour of a given system.

3.1 Output Response

$$\text{y(t)} = \underbrace{Ce^{At}\left(x(0) - (sI-A)^{-1}B\right)}_{\text{Transient response } (\to 0 \text{ if asymptotically stable})} + \underbrace{\left(C\left(sI-A\right)^{-1}B + D\right)e^{st}}_{\text{Steady-state response}}$$

Note: We're mostly looking at the steady-state response y_{ss} because the transient response disappears.

$$y_{ss}(t) = \left(C\left(sI-A\right)^{-1}B + D\right)e^{st} = G(s)e^{st}$$

3.2 Transfer Function

This $G(s)$ is called transfer function. It is a complex number and describes the input - steady-state-output relation of a system.

$$G(s) = C(sI-A)^{-1}B + D$$
$$G(s) = \frac{Y(S)}{U(S)}$$

The impulse response $h(t)$ defines our system because of its Laplace transform being the transfer function:

$$\mathcal{L}\{h(t)\} = H(s) = G(s)$$

3.2.1 Canonical Form

$$G(s) = \frac{b_{n-1}s^{n-1} + b_{n-2}s^{n-2} + \cdots + b_0}{s^n + a_{n-1}s^{n-1} + \cdots + a_0} + d$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & \cdots \\ 0 & 0 & 1 & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \\ 0 & 0 & 0 & 1 & -a_0 \\ -a_1 & -a_2 & \cdots & -a_{n-1} & 0 \end{bmatrix},$$

$$B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}, \quad C = \begin{bmatrix} b_0 & \ldots & b_{n-1} \end{bmatrix}, \quad D = [d]$$

3.2.2 Other Forms

The transfer function for the **Root Locus** analysis is:

$$G(s) = k_{rl}\frac{(s-z_1)(s-z_2)\cdots(s-z_n)}{(s-p_1)(s-p_2)\cdots(s-p_m)}$$

The transfer function for the **Bode Plot** analysis is:

$$G(s) = \frac{k_{Bode}}{s^q}\frac{\left(\frac{s}{-z_1}+1\right)\left(\frac{s}{-z_2}+1\right)}{\left(\frac{s}{-p_1}+1\right)}$$

The **Partial Fraction Expansion** can be written as:

$$G(s) = \frac{r_1}{s-p_1} + \frac{r_2}{s-p_2} + \cdots + \frac{r_n}{s-p_n} + r_0$$

Residues of non-repeated poles p_i $\rightarrow$ **cover-up** method:

$$r_i = \lim_{s \to p_i}(s-p_i)G(s)$$

For repeated poles of order m:

$$r_i = \frac{1}{(m-1)!}\lim_{s \to p_i}\frac{\partial^{m-1}}{\partial s^{m-1}}\left((s-p_i)^m\right)G(s)$$

3.2.3 Graphical computation

Phase and Magnitude of an example input $s = 2jw$ can be computed by looking at the complex plane. By inspecting the poles and zeros of the TF G(s).
Magnitude: $|G(2j\omega)| = \frac{\Pi|Z_i|}{\Pi|P_i|}$
Phase: $\angle G(2j\omega) = \sum\angle Z_i - \sum\angle P_i$
Where P_i, Z_i are vectors from the corresponding pole or zero to the input s (here $= 2j\omega$).

3.3 Poles & Zeros

The roots of the denominator $D(s)$ of a TF are called **poles** and analogously the roots of the nominator $N(s)$ **zero's**.
Stable pole: $Re(p_i) < 0 \rightarrow e^{p_i t}$
Unstable pole: $Re(p_i) > 0$, if one pole is unstable, the whole system is unstable!
In General: Stable poles closer to the origin are dominant.
Mnm phase zero: $Re(z_i) < 0$ causes overshoot.
Non-mnm phase zero: $Re(z_i) > 0$ causes undershoot.
$|\mathbf{z}| \propto$ over-/undershoot.
Note: In case of pole-zero cancellation stability information is lost! Dangerous since perfect mathematical cancellation is not how the real system would behave (due approximation, noise, etc.).

3.4 Feedback Systems

Open Loop

Open-loop gain $L(s)$ is the transfer function without the feedback connection.

Complementary Sensitivity/Closed-loop TF

It is the closed-loop transfer function $T(s)$ (**noise rejection**) with gain k. $T(s) = \frac{kL(s)}{1+kL(s)}$, $\quad |S(s)| \to 1$

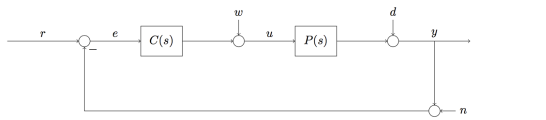
Characteristic equation: $1 + kL(s) = 0$

Sensitivity Function

SF Also called **Disturbance Rejection**.

$S(s) = \frac{k}{1+kL(s)}$, $\quad |S(s)| \to 0$

Closed-Loop Dynamics



$Y(s) = S(s)\cdot(D(s)+P(s)\cdot W(s)) + T(s)\cdot(R(s)-N(s))$

Cascaded Control (SIMO)

Example: Adaptive cruise control. Key requirement is for the inner loop to have a bigger bandwidth (rule of thumb: 10 times bigger) than the outer loop.

3.5 Root Locus Method

Displays how the poles move with varying gain k.

Limitations: It can only be used for finite-dimensional systems and rational functions.

Sketching Rules:

(n, m are the number of openloop poles, zeros respectively.)

- the RL is symmetric about its real axis.
- all points on the real axis are on the root locus.
- all points on the real axis to the left of an **odd**/**even** number of poles or zeros are on the **positive**/**negative** k root locus.
- closed loop poles approach the open loop poles as $k \to 0$. Start drawing from open loop poles ($\times$)
- closed loop poles approach the open loop zeros ($\circ$) as $k \to \infty$.
- if $n > m$ the excess pole(s) will go to infinty at angle(s) $\boldsymbol{\delta_i}$. Two poles on the real axis meet in breakaway points and leave perpendicular to the real axis.
- breakouts from poles/zeros go to/come from infinity.
- the origin of asymptotes σ_a is where all the asymptotes intercept the real axis in one point.
- if there is at least two lines going to infinity, the sum of all roots (real parts) is constant. (good for intuition)

$$\delta_i = \frac{\pi(2i+1)}{n-m}, \text{if } k_{rl} > 0, i = \{0, 1, ..., n-m-1\}$$

$$\delta_i = \frac{2i\cdot\pi}{n-m}, \text{ if } k_{rl} < 0$$

$$\sigma_a = \frac{1}{n-m}\left(\sum Re(p_i) - \sum Re(z_i)\right)$$

$$\boldsymbol{L(0) = 1/k_{crit}}$$

3.6 System Specifications

3.6.1 1st Order system

Settling Time: $T_{d\%} = \tau\log(100/d), \tau = -\frac{1}{Re(p)}$

3.6.2 2nd Order System

$\omega_n = \sqrt{\sigma^2+\omega^2} \quad |\sigma| = \zeta\omega_n \quad \varphi = \arctan\left(\frac{\sigma}{\omega}\right)$

Damping ratio: $\zeta = \sin(|\varphi|)$

Time to peak: $T_p = \frac{\pi}{\omega}$

Peak overshoot ratio: $M_p = e^{\frac{\sigma\pi}{\omega}}, \zeta^2 = \frac{(\ln M_p)^2}{\pi^2+(\ln M_p)^2}$

Rise time: $T_{100\%} = \frac{\pi/2-\varphi}{\omega} \approx \frac{\pi}{2\omega_n}$

Settling time: $T_{d\%} = \frac{1}{|\sigma|}log\left(\frac{d}{100}\right)$

Note: Dominant pole is the one with the least negative real part (slowest decay). If you cancel out your non-dominant pole(s) p, you have to divide the gain by p.

3.6.3 Steady state error summary

e_{ss}	$q=0$	$q=1$	$q=2$
Type 0	$\frac{1}{1+k_{\text{Bode}}}$	∞	∞
Type 1	0	$\frac{1}{k_{\text{Bode}}}$	∞
Type 2	0	0	$\frac{1}{k_{\text{Bode}}}$

q : Unit ramp order Type : # integrators

$$\boldsymbol{e_{ss} = \lim_{s\to 0} G(s)}$$

This shows that if you want to reach a zero steady state you need $q+1$ integrators (Type).

Gain selection table:

	proportional	integral	derivative
ss-error	$\downarrow$	$\to 0$	n.a.
oscillations	$\uparrow$	$(\uparrow)$	$\downarrow$
sens. to noise	$\uparrow$	n.a.	$\uparrow$
stab. margins	$(\downarrow)$	$\downarrow$	$\uparrow$

3.7 Frequency Response

3.7.1 Bode Plot

$$|G(j\omega)|_{\text{dB}} = 20\log_{10}|G(j\omega)|$$

Inverting $G(j\omega)$ results in reflection about the horizontal axis.

Bode's Law: If the slope of Magnitude curve $|G(j\omega)|$ is c·20db/decade for more than one decade, then the phase will be approximately c·90°.

Bode plot table

Element type	Formally	Magnitude	Phase
single stable pole	$Re(p_i) \leq 0$	$-20\frac{dB}{dec}$	$-90°$
single unstable pole	$Re(p_i) > 0$	$-20\frac{dB}{dec}$	$+90°$
cmplx stable pole	$Re(p_i) \leq 0$	$-40\frac{dB}{dec}$	$-180°$
cmplx unstable pole	$Re(p_i) > 0$	$-40\frac{dB}{dec}$	$+180°$
single mnm phs zero	$Re(z_i) \leq 0$	$+20\frac{dB}{dec}$	$+90°$
single non-mnm phs zero	$Re(z_i) > 0$	$+20\frac{dB}{dec}$	$-90°$
cmplx non-mnm phs zero	$Re(p_i) > 0$	$+40\frac{dB}{dec}$	-180°
cmplx mnm phs zero	$Re(p_i) \leq 0$	$+40\frac{dB}{dec}$	+180°
positive constant	$k \geq 0$	$0\frac{dB}{dec}$	$0°$
negative constant	$k < 0$	$0\frac{dB}{dec}$	$\pm 180°$
differentiator	$\tau \cdot s$	$+20\frac{dB}{dec}$	$+90°$
integrator	$\frac{1}{\tau\cdot s}$	$-20\frac{dB}{dec}$	$-90°$
time delay	$e^{s\cdot\tau}$	$0\frac{dB}{dec}$	$-\omega\cdot\tau$

Bode plot identification: 1. Magnitude slope when $\omega \to 0$ implies integrator/differentiator.

2. Magnitude comes in from 0 at bode gain (unless 1. applies). 3. Resonance spikes only occur for complex poles.

3.7.2 Nyquist Plot

Nyquist Theorem

Nyquist contour is the path enclosing the complete right half of the complex plane CW.

$$\mathbf{N = Z - P}$$

N: #Clockwise Encirclements of $\frac{-1}{k}$

Z: #Unstable closed-loop poles

P: #Unstable open-loop poles

Poles on the imaginary axis: If moving CCW around the poles in the contour, close the plot CW and vice versa.

Nyquist Condition

If the open loop is stable, the Nyquist plot should NOT encircle the $\frac{-1}{k}$ point, in order for the closed loop to be stable.

Gain & Phase margins

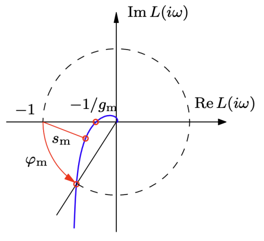
Measurements that tell you how close your system is to instability.

Note: Here the point is -1 due to the fact that $L(s)$ has gain 1.

$\phi_m = \phi(\omega_c) + 180°$

$= \angle L(j\omega_c) + 180°$

$g_m = 1/|L(j\omega_{pc})|$



$$|L(j\omega)| < \frac{1}{k}, \text{ whenever } \angle L(j\omega) = 180°$$

Bode plot: Magnitude has to be under 0db when phase crosses -180°.

Phase has to be above the -180° line when Magnitude is 0db.

Note: Only valid if the open loop is stable and non-mnm phase, otherwise it has to be double checked by other criterions.

3.7.3 Frequency Domain Specifications

Namely **disturbance** and **noise rejection**.

- Disturbances and commands act at 'low' frequencies (under 10Hz). $\to$ make $|S(j\omega)| \ll 1$ @low-freq
- Noise is typically conceived in high frequencies (over 100Hz).$\to$ $|T(j\omega)| \ll 1$ @high-freq

Bode plot obstacle course:

Low-freq obstacle $W_1(j\omega)$: $|L(j\omega)| > |W_1(j\omega)|$

High-freq obstacle $W_2(j\omega)$: $|L(j\omega)| < |W_2(j\omega)|^{-1}$

Closed-loop bandwidth

ω such that $|T(j\omega)| > 1/\sqrt{2}$.

Which means that the output can track the commands to within a factor of ≈ 0.7.

The (open-loop) **crossover freq** ω_c ($|L(j\omega_c) = 1|$) is approximately the bandwidth if phase margin is about 90°.

3.7.4 Loop shaping

Loop shaping is the task of designing a dynamic compensator $C(s)$, that guides the given TF of the physical system $P(s)$, such that $L(s) = C(s)\cdot P(s)$ satisfies all system requirements ('moves smoothly through the obstacle course').

For this task you have 4 tools to choose from: Prop. Gain k, Differentiator $1/s$, Lead-compensator, Lag-compensator.

Proportional Gain k

With the proportional gain you can simply shift the magnitude-plot up or down by either increasing or decreasing it. The phase **stays unchanged**. Useful to increase band-width ($k\uparrow$) or reduce steady-state error ($k\downarrow$)

Watch out for instability with too big gains!

Integrator $1/s$

Is mostly only used for eliminating steady-state error. (see Section 2.6.3)

Lead Compensator $|L(j\omega|\uparrow)$

$$C_{\text{lead}} = \frac{s/a+1}{s/b+1} = \frac{b}{a}\frac{s+a}{s+b}, \qquad a < b$$

The lead compensator induces an **up-ramp** in magnitude and a **bump-up** in phase around $\sqrt{ab}$ (i.e. first the zero than the pole). Multiply k by a/b, thus magnitude is not affected at high frequencies. Else if ω_c shouldn't be changed use: $|L(j\omega_c| = 1 = 0dB$ **Typical use:** Increase phase margin.

$\varphi_{max} = 2\arctan\left(\sqrt{\frac{b}{a}}\right) - 90°$ (max phase increase)

Lag Compensator $|L(j\omega|\downarrow)$

The lag compensator (same as 'lead', except $b < a$) induces a **down-ramp** in magnitude and a **bump-down** in phase. (i.e. pole comes first than the zero) **Typical use:** Improve command tracking/disturbance rejection.

$\varphi_{min} = 2\arctan\left(\sqrt{\frac{b}{a}}\right) - 90°$, magnitude increase $= a/b$

Step-by-step loop shaping approach

- 1. Secure 0 steady-state error ($\to$ integrators).
- 2. Adjust gain such that the magnitude plot evades the low-freq obstacle W_1 (left side bode plot).
- 3. Work with poles and zeros to reach bandwidth, phase-margin and noise rejection W_2.

3.8 Non-Mnm-Zeros and Unstable Poles

$P(s) = P_{\text{mp}}(s)D(s)$, $\quad |D(s)| = 1$ and $|P_{\text{mp}}(s)| = |P(s)|$

Where in $P_{\text{mp}}(s)$ you substitute the poles/zeros in the right half plane with its symmetric in the left half plane.

3.9 Time delays

TF of a time delay: e^{-sT}. It is non-rational but linear.

$$\left|e^{-j\omega T}\right| = 1, \qquad \angle e^{-j\omega T} = -\omega T$$

Or in words, the time delay does not affect the magnitude but it 'eats up' the phase margin ϕ_m

$$\phi_{m,T} = \phi_{m,0} - \omega_c T$$

Padé approximation: $e^{-sT} \approx \frac{2/T-s}{2/T+s}$ $\qquad \boldsymbol{2\pi\text{Hz} = rad/s}$

3.10 Nonlinearities

Given a non-linearity we check stability by setting boundaries. Thus the Nyquist criterion becomes a bit more complecated. We differentiate between:]

Necessary Condition: Nyquist plot encircles the range $[-1/k_1, -1/k_2]$ on the Re-axis the required ammount.

Sufficient condition (circle): Ny-plot encircles a circle which reaches from $[-1/k_1, -1/k_2]$.

Describing function

$N(A) = \frac{b_1(A)/a_1(A)}{A} = \frac{1}{\pi A}\int_{-\pi}^{\pi} y(t)\sin(\omega t)cos(\omega t)d(\omega t).$

For odd / even non-linearities.

With this describing function $N(A)$ you can now approximate the non linear system with a TF $L'(A,s)$:

$$L'(A,s) = N(A)\cdot L(s)$$

Which is dependent on the Amplitude A of the input signal and s.

Limit cycles

$$A = -AN(A)L(j\omega), \quad L(j\omega) = -1/N(A)$$

Which means that they intersect in the complex plane. Stability then can be determined sectionwise, by counting the encirclements!

dB	-20	0	1	5	20	40	60
	0.1	1	1.12	1.77	10	100	1000