Lecture 15

Optimization, Null-Finding, and Control

References

http://en.wikipedia.org/wiki/Optimization
http://en.wikipedia.org/wiki/Optimization_(mathematics)
http://en.wikipedia.org/wiki/Gradient_descent
http://en.wikipedia.org/wiki/Newton%27s_method
http://en.wikipedia.org/wiki/Secant_method
http://en.wikipedia.org/wiki/Broyden%27s_method
http://en.wikipedia.org/wiki/Constrained_optimization
http://en.wikipedia.org/wiki/Quadratic_programming
http://en.wikipedia.org/wiki/Linear_programming
http://www.mathworks.com/products/optimization/
http://en.wikipedia.org/wiki/Control_system
http://en.wikipedia.org/wiki/Control_system_(disambiguation)
http://www.mathworks.com/help/toolbox/control/

ROOT FINDING:

Roots of $\vec{F}(\vec{x})$, where $\vec{x} = (x_1, x_2 \dots x_n)$
$\vec{F} = (F_1, F_2 \dots F_n)$

$$\vec{F}(\vec{x}) = \vec{0} \quad , \text{ or}$$

$$\begin{cases} F_1(x_1, x_2, \dots x_n) = 0 \\ F_2(x_1, x_2, \dots x_n) = 0 \\ \vdots \\ F_n(x_1, x_2, \dots x_n) = 0 \end{cases}$$

n equations in n variables

Solution is unique, with a single root $\vec{x}$, when $\vec{F}$ is linear and full-rank.

In general, multiple solutions exist.

UNCONSTRAINED OPTIMIZATION :

Extrema of $u(\vec{x}) = u(x_1, x_2, \dots x_n)$:

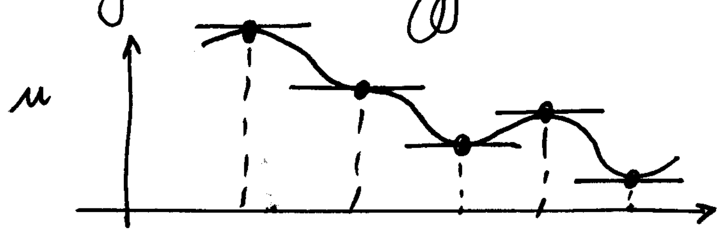
- $\max\limits_{x_1, \dots x_n} : u(x_1, x_2, \dots x_n)$
- $\min\limits_{x_1, \dots x_n} : u(x_1, x_2, \dots x_n) \quad \left(= \max\limits_{x_1, \dots x_n} \left[-u(x_1, \dots x_n) \right] \right)$

Let : $\vec{F} = \vec{\nabla} u$, GRADIENT of u : $F_i = \dfrac{\partial u}{\partial x_i}$

$\bar{\bar{H}}$, HESSIAN of u : $H_{ij} = \dfrac{\partial^2 u}{\partial x_i \partial x_j} = \dfrac{\partial F_i}{\partial x_j} = \dfrac{\partial F_j}{\partial x_i}$

- STATIONARY POINTS : $\vec{x}$ for which $\vec{F}(\vec{x}) = \vec{0}$

  Necessary but not sufficient condition for (local) extrema!



(in 1-D)

- LOCAL MAXIMA : stationary points for which

  $\bar{\bar{H}}(\vec{x})$ = NEGATIVE DEFINITE, or $\vec{\Delta x}^T \cdot \bar{\bar{H}} \cdot \vec{\Delta x} \lessgtr 0, \forall \vec{\Delta x}$

- LOCAL MINIMA : stationary points for which

  $\bar{\bar{H}}(\vec{x})$ = POSITIVE DEFINITE, or $\vec{\Delta x}^T \cdot \bar{\bar{H}} \cdot \vec{\Delta x} \geqslant 0, \forall \vec{\Delta x}$

Taylor series expansion around $\vec{x}$:

$$\mu(\vec{x}+\Delta\vec{x}) = \mu(\vec{x}) + \vec{F}\cdot\Delta\vec{x} + \frac{1}{2}\Delta\vec{x}^{T}\cdot\bar{\bar{H}}\cdot\Delta\vec{x} + \dots$$

$$\vec{F}\cdot\Delta\vec{x} = \sum_i \frac{\partial\mu}{\partial x_i}\cdot\Delta x_i = 0 \text{ for STATIONARY POINT}$$

$$\frac{1}{2}\Delta\vec{x}^{T}\cdot\bar{\bar{H}}\cdot\Delta\vec{x} = \frac{1}{2}\sum_i\sum_j \frac{\partial^2\mu}{\partial x_i\partial x_j}\Delta x_i\Delta x_j$$

≥ 0 for $\bar{\bar{H}}$ POS. DEF.

≤ 0 for $\bar{\bar{H}}$ NEG. DEF.

$\dots$ HIGHER ORDER TERMS: neglect for $\Delta\vec{x}$ "small"

$\Rightarrow$ local maxima $\quad \mu(\vec{x}+\Delta\vec{x}) \leq \mu(\vec{x})$ for $\bar{\bar{H}}$ NEG. DEF.

minima $\quad \geq$ POS. DEF.

Notes:
- $\bar{\bar{H}}$ is symmetric, and so all eigenvalues are REAL
- $\bar{\bar{H}}$ is POS. DEF. if all eigenvalues are POSITIVE

  NEG. NEGATIVE

GRADIENT ASCENT / DESCENT:

iteratively searches for MAX/MIN extrema of $u(\vec{x})$:

- Initial guess $\vec{x}_0$
- Iterate, $i \to i+1$:

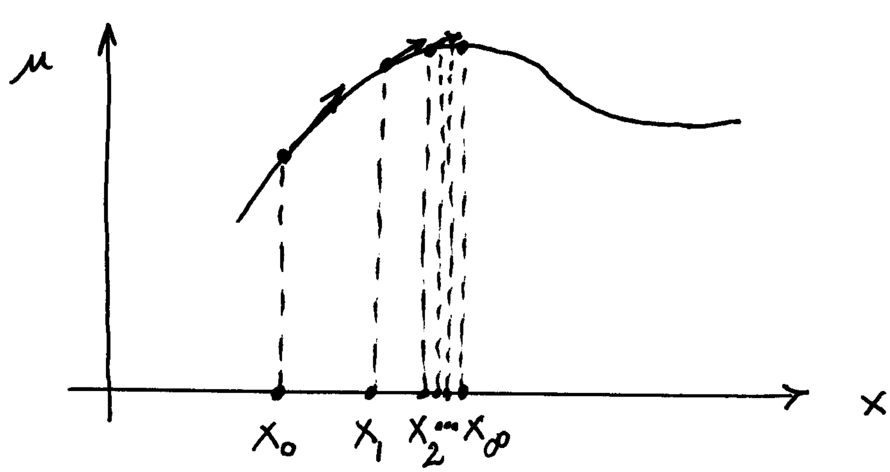
$$\vec{x}_{i+1} \Leftarrow \vec{x}_i + \mu \, \vec{F}(\vec{x}_i)$$

If μ is "sufficiently" small, the series $\vec{x}_i$ converges to a local maximum $(\mu > 0)$ or minimum $(\mu < 0)$ of $u(\vec{x})$:

$$u(\vec{x}_{i+1}) = u(\vec{x}_i + \mu \vec{F}(\vec{x}_i)) \approx u(\vec{x}_i) + \vec{F} \cdot (\mu \vec{F}) + \underbrace{\tfrac{1}{2}\, \mu \vec{F}^T \cdot \bar{\bar{H}} \cdot \mu \vec{F} + \dots}_{\text{NEGLECT}}$$

$$\Rightarrow u(\vec{x}_{i+1}) \approx u(\vec{x}_i) + \mu \|\vec{F}\|^2 \begin{array}{c} \geqslant \\ \leqslant \end{array} u(\vec{x}_i) \quad \text{for} \begin{array}{l} \mu > 0 \\ \mu < 0 \end{array} \text{ and small}$$

Equality: at convergence: $\vec{F} = 0$, and $\bar{\bar{H}}$ $\begin{array}{l}\text{POS.}\\ \text{NEG.}\end{array}$ DEF.



If μ is too large the series diverges...

NEWTON'S METHOD :

A second-order null-finding / optimization method for faster convergence than first-order methods such as gradient ascent/descent:

$$\vec{F}(\vec{x}_{i+1}) = \vec{F}(\vec{x}_i + \underset{(\vec{x}_{i+1}-\vec{x}_i)}{\underset{\parallel}{\Delta\vec{x}_i}}) \approx \vec{F}(\vec{x}_i) + \bar{\bar{H}} \cdot \Delta\vec{x}_i + \ldots$$

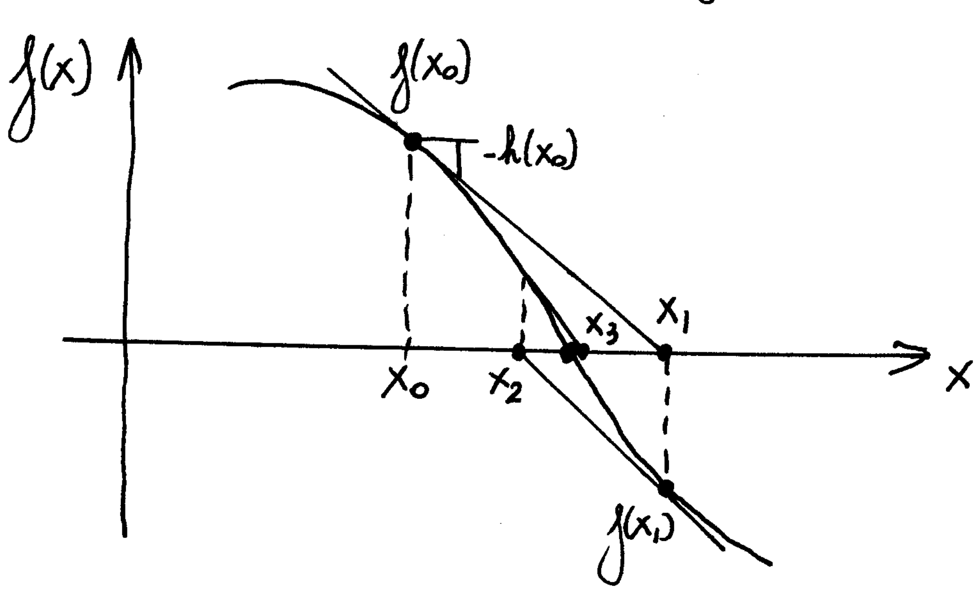
$= \vec{0}$ for STATIONARY POINT

NEGLECT for $\Delta\vec{x}_i$ "small"

$$\Rightarrow \vec{x}_{i+1} \leftarrow \vec{x}_i - \underbrace{\left(\bar{\bar{H}}(\vec{x}_i)\right)^{-1}}_{\text{inverse Hessian, rather than constant scalar } \mu} \cdot \vec{F}(\vec{x}_i)$$

e.g., in 1-D:

$$x_{i+1} \leftarrow x_i - \frac{1}{h(x_i)} \cdot f(x_i) \quad \text{where } h(x) = \frac{df(x)}{dx}$$



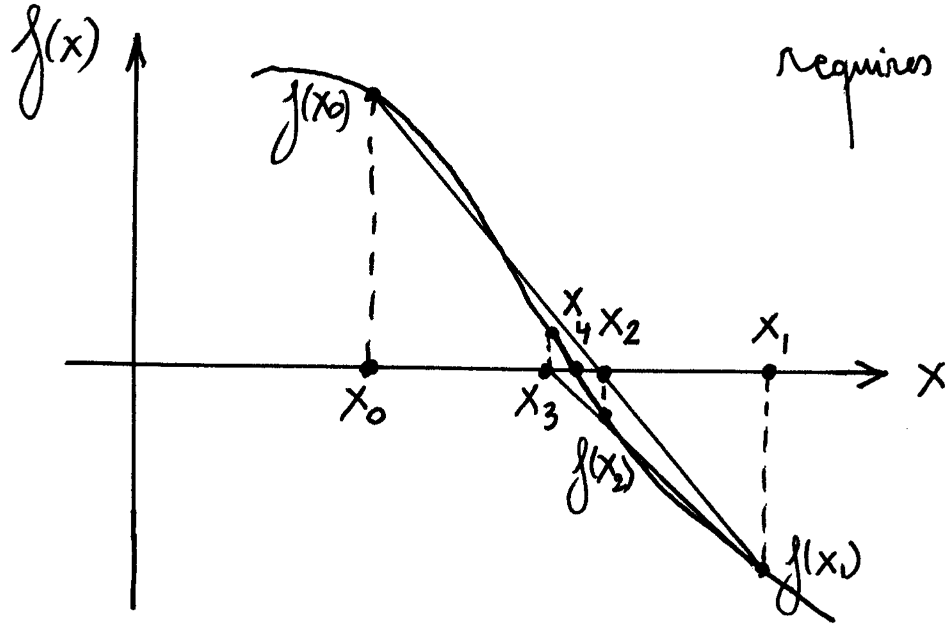
SECANT METHOD:

A second-order null-finding / optimization method with a linear approximation for $\bar{\bar{H}}$, and values for $\bar{F}$ only:

- In 1-D:

$$h(x_i) = \frac{df(x_i)}{dx_i} \approx \frac{f(x_i) - f(x_{i-1})}{x_i - x_{i-1}}$$

$$\Rightarrow \quad x_{i+1} \leftarrow x_i - \frac{x_i - x_{i-1}}{f(x_i) - f(x_{i-1})} \cdot f(x_i)$$



requires two starting points x_0, x_1

- In higher dimensions, n-D: more involved

Broyden - Fletcher - Goldfarb - Shanno (BFGS) method

CONSTRAINED OPTIMIZATION :

Find extrema (maxima) of $u(\vec{x})$ subject to equality and/or inequality constraints :

$$\max_{x_1, x_2 \ldots x_n} : \quad u(\vec{x}) \quad \text{subject to} : \begin{cases} v_i(\vec{x}) = c_i \ , & i = 1, \ldots p \\ w_j(\vec{x}) \leq d_j \ , & j = 1, \ldots q \end{cases}$$

When $u(\vec{x})$ is concave in $\vec{x}$, and $v_i(\vec{x})$ and $w_j(\vec{x})$ are linear in $\vec{x}$, then sufficient conditions for an optimum $\vec{x}$ are given by the Karush-Kuhn-Tucker (KKT) conditions:

- Stationarity : $\vec{\nabla}\left(u(\vec{x}) + \sum_{i=1}^{p} \lambda_i v_i(\vec{x}) + \sum_{j=1}^{q} \mu_j w_j(\vec{x}) \right) = \vec{0}$
- Feasibility & slackness :

$$v_i(\vec{x}) = c_i \ , \quad \forall i = 1, \ldots p$$

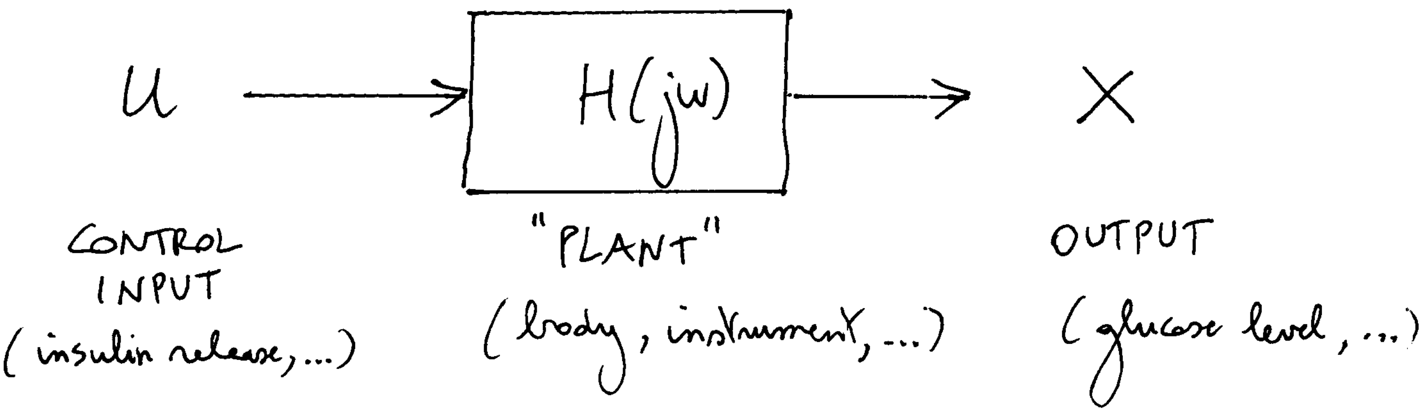
$$\left.\begin{array}{l} w_j(\vec{x}) \leq d_j \\ \mu_j \geq 0 \end{array}\right\} \text{ and } \ \mu_j \cdot (w_j(\vec{x}) - d_j) = 0, \ \forall j = 1, \ldots q$$

Examples:

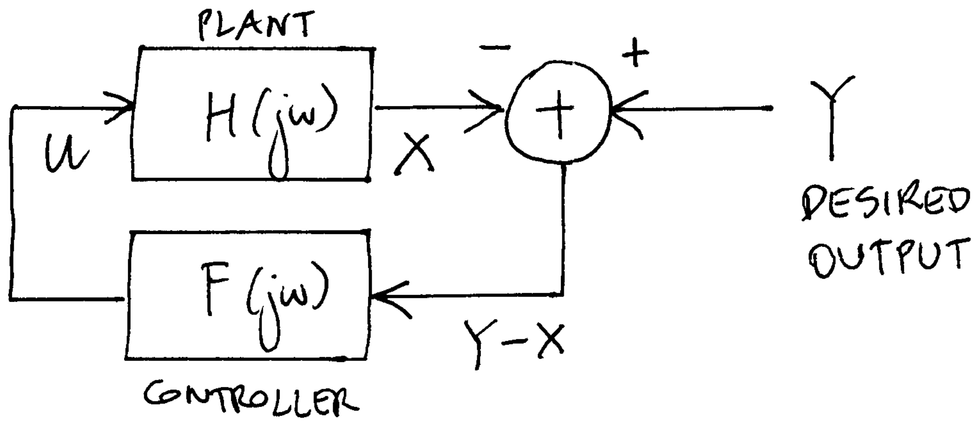
- $u(\vec{x}) = \vec{A} \cdot \vec{x}$ $\longrightarrow$ LINEAR PROGRAMMING
- $u(\vec{x}) = \frac{1}{2} \vec{x}^T \cdot \overline{\overline{Q}} \cdot \vec{x}$ $\longrightarrow$ QUADRATIC PROGRAMMING

See Matlab toolboxes

LINEAR CONTROL : A cursory introduction



A controller will attempt to control the input U to drive the output X towards a desired state Y :



$$\left.\begin{aligned} X(j\omega) &= H(j\omega)\,.\,U(j\omega) \\ U(j\omega) &= F(j\omega)\,.\,\big(Y(j\omega) - X(j\omega)\big) \end{aligned}\right\} \Rightarrow \quad X(j\omega) = \frac{F(j\omega)\,H(j\omega)}{1 + F(j\omega)H(j\omega)}\,Y(j\omega)$$

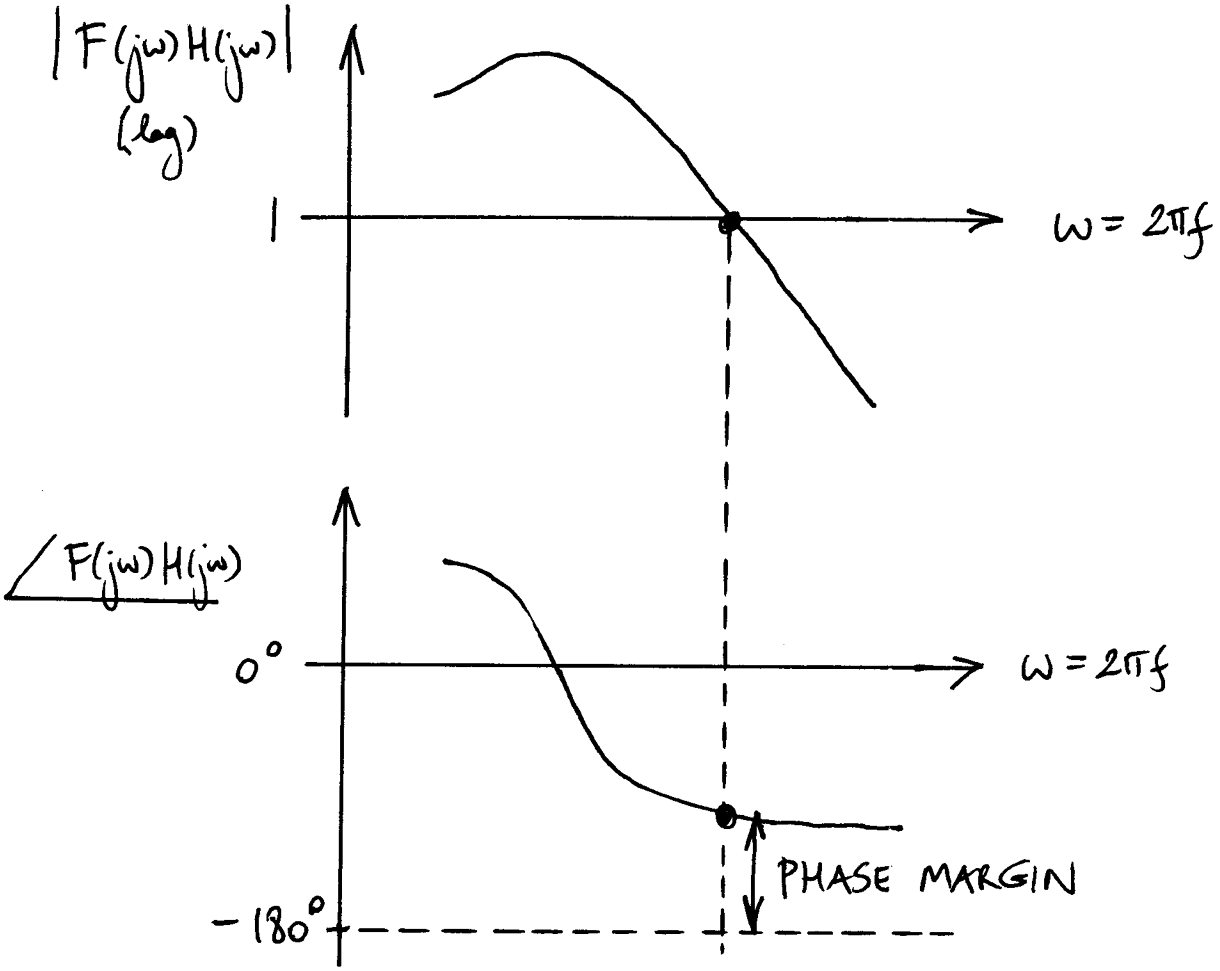
or $$X(j\omega) = \frac{1}{1 + \dfrac{1}{F(j\omega)\,H(j\omega)}}\,Y(j\omega) \to Y(j\omega) \text{ for } |FH| \gg 1$$

Considerations:

1. STABILITY: The control loop becomes unstable when the open loop gain $F(j\omega)H(j\omega)$ reaches -1.

→ ABSOLUTE STABILITY as long as the open loop phase $\angle F(j\omega)H(j\omega)$ is greater than $-180°$ (and less than $180°$) for all frequencies where the open loop gain $|F(j\omega)H(j\omega)|$ is greater than 1.

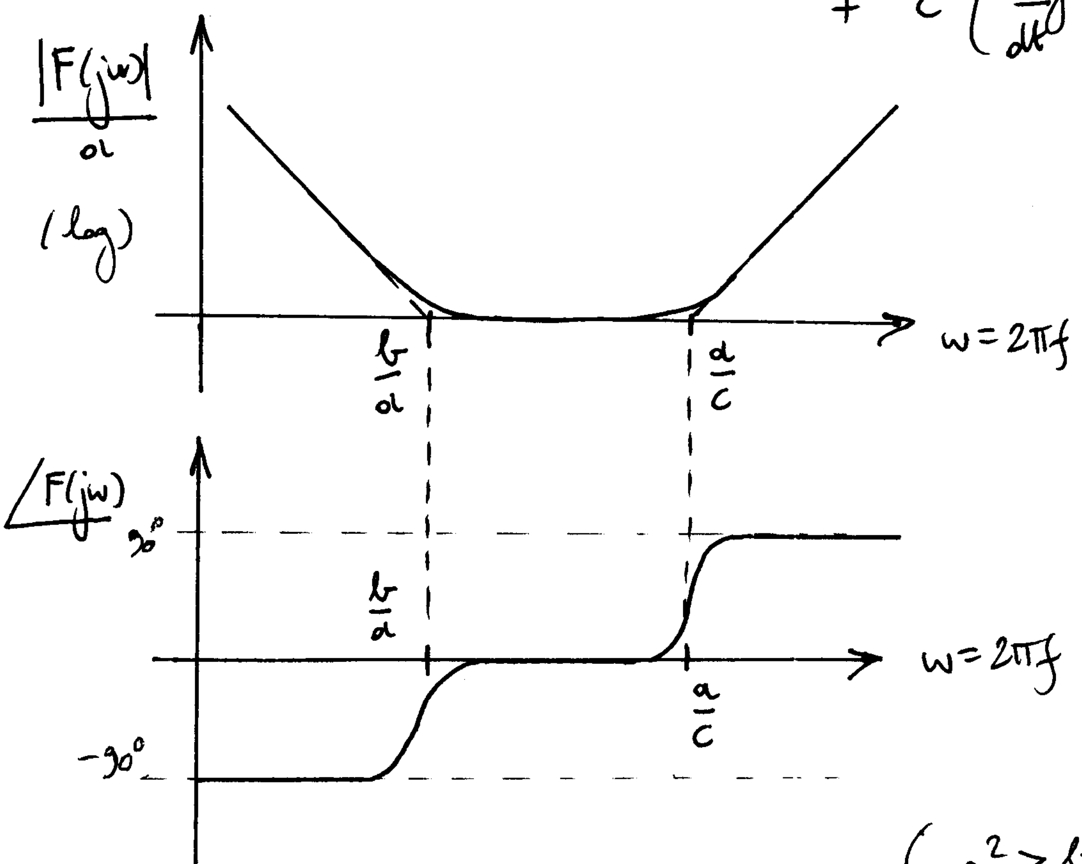
→ PHASE MARGIN needs to be positive: (and ideally $\geq 60°$) to minimize ringing



2. CONTROLLER DESIGN: A universal design that often works is the P.I.D. (proportional - integral - differential) controller:

$$F(j\omega) = \underset{\text{"P"}}{a} + \underset{\text{"I"}}{b\,\frac{1}{j\omega}} + \underset{\text{"D"}}{c\,j\omega} \quad , \text{ or}$$

$$u(t) = a\left(y(t) - x(t)\right) + b\int_{-\infty}^{t} \left(y(\theta) - x(\theta)\right) d\theta + c\left(\frac{dy}{dt} - \frac{dx}{dt}\right)$$



$(a^2 > bc)$

3. DISCRETE TIME SYSTEMS: Same principles, using z-transforms rather than Fourier / Laplace

$$z = e^{sT} = e^{j\omega T}$$ where T is time step

(unit time advance) ($\frac{1}{T}$ is sampling rate)

$$u(t) \rightarrow u[n] = u(nT)$$
$$x(t) \rightarrow x[n] = x(nT)$$
$$y(t) \rightarrow y[n] = y(nT)$$

4. NONLINEAR SYSTEMS: Complex!

Sometimes a Taylor expansion of H (and F) around a known stable state (or limit cycle) of U, X and Y will work.

$$U = U_0 + \tilde{U}(j\omega) \qquad \text{with } |\tilde{U}| \ll |U_0|$$
$$X = X_0 + \tilde{X}(j\omega) \qquad |\tilde{X}| \ll |X_0|$$

$$X = H(U) = H(U_0 + \tilde{U}) \approx X_0 + \left.\frac{\partial H}{\partial U}\right|_{U_0} \cdot \tilde{U}$$

(nonlinear dynamical system)

$$\Rightarrow \quad \tilde{X}(j\omega) \approx \tilde{H}(j\omega) \cdot \tilde{U}(j\omega) \quad \text{with } \tilde{H}(j\omega) = \left.\frac{\partial H}{\partial U}\right|_{U_0}$$

where $\frac{\partial}{\partial t} \rightarrow j\omega$