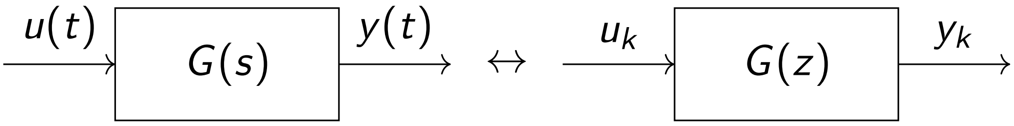
3F1 Signals and Systems: Handout 9

Response invariant continuous to discrete mappings and Digital Filtering: Part I (IIR design)

Jossy Sayir

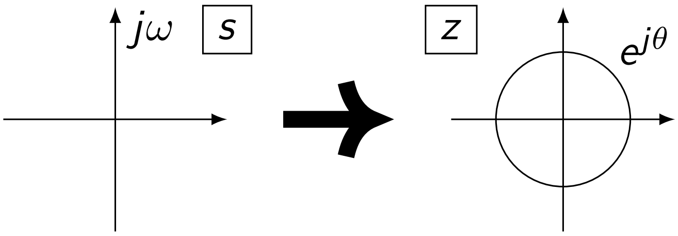
Michaelmas Term 2025

Recall: Mapping analog systems to discrete time systems



We can map analog designs to discrete time in two ways:

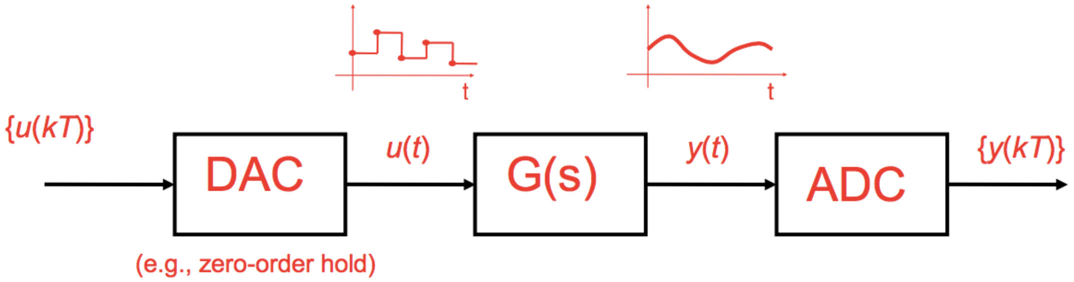
1. **algebraic transformations:** transform the Laplace s domain directly to the z domain (last lecture)



2. **response matching:** design a discrete time system whose response matches the impulse/step/ramp response of the analog system (this lecture)

Transfer function analysis of DAC and ADC interfaces

Hybrid: G(s) linear continuous system,
discrete input/output - before DAC/after ADC



{u(kT)}

DAC

(e.g., zero-order hold)

u(t)

G(s)

y(t)

ADC

{y(kT)}

- What is the transfer function $G(z)$ from $\{u_k\}_{k\geq 0}$ to $\{y_k\}_{k\geq 0}$? (does the transfer function exist?)
- It is **linear** and **time-invariant** thus it has a z-transfer function.
- How to find it? Take an **appropriate** input, find the output, take the ratio of the z transforms.

Transfer function analysis of DAC and ADC interfaces

{u(kT)}

DAC

u(t)

G(s)

y(t)

ADC

{y(kT)}

It is **linear** and **time-invariant** thus it has a z-transfer function.

For example, take $u(kT) = 1$ for all $k \geq 0$.

$$\Rightarrow \quad u(t) = 1 \qquad \forall t \geq 0$$

$$\Rightarrow \quad Y(s) = G(s)\frac{1}{s}$$

$$\Rightarrow \quad y(kT) = \mathcal{L}^{-1}\left(\frac{G(s)}{s}\right)_{t=kT\geq 0}$$

Since $\mathcal{Z}(u(kT)) = \frac{1}{1-z^{-1}}$ we get

$$G(z) = \frac{z-1}{z}\mathcal{Z}\left(\mathcal{L}^{-1}\left(\frac{G(s)}{s}\right)_{t=kT\geq 0}\right)$$

Transfer function analysis of DAC and ADC interfaces

$$G(s) = \frac{1}{s+1} \quad \text{(example)}$$

take any input

$$u(t) = 1 \;\Rightarrow\; U(s) = \frac{1}{s}$$

find the output

$$Y(s) = \frac{1}{s(s+1)} = \frac{1}{s} - \frac{1}{s+1} \quad \Rightarrow \quad y(kT) = \left(1 - e^{-t}\right)_{t=kT\geq 0}$$

take the ratio of z transforms

$$G(z) = Y(z)/U(z) = \frac{z-1}{z}\mathcal{Z}\left(1 - e^{-kT}\right)$$

$$= \frac{z-1}{z}\left(\frac{z}{z-1} - \frac{z}{z-e^{-T}}\right) = 1 - \frac{z-1}{z-e^{-T}}$$

Response matching: impulse invariance

$G_c(s)$ Laplace transform continuous-time filter. The impulse response of the corresponding *impulse invariance* digital filter G(z) (with sampling T) is equal to the impulse response of the $G_c(s)$ sampled at $t = kT$.

$$G(z) = \mathcal{Z}\left(\mathcal{L}^{-1}\left(G_c(s)\right)_{t=kT}\right)$$

- For bandlimited filters the digital filter frequency response will closely approximate the continuous-time frequency response.
- Preserves stability:
  $\Re(\beta) < 0 \rightarrow \int |e^{\beta t}| dt < \infty \rightarrow \sum |e^{\beta Tk}| < \infty.$

Example:

$$G_c(s) = \frac{\alpha}{s-\beta} \overset{\mathcal{L}^{-1}}{\rightarrow} \alpha e^{\beta t} \overset{\text{sample}}{\longrightarrow} \alpha e^{\beta Tk} \overset{\mathcal{Z}}{\rightarrow} \frac{\alpha}{1 - e^{\beta T} z^{-1}} = G(z)$$

This works well when a Laplace transfer function is known, but not to measure a response: D/A converter **cannot** convert δ_k to $\delta(t)$.

Response matching: step invariance

As seen previously in our example...

Take a continuous time filter/system with Laplace transfer function $G_c(s)$. The corresponding *step response invariance* digital filter (with sampling period T) is a digital filter whose step response is equal to the step response of the continuous time filter sampled at $t = kT$.

1. Find the step response of the continuous system
2. Sample at time $t = kT$ and take the z-transform
3. Multiply by $(z-1)/z$.

$$G_c(s) \overset{\text{step}}{\rightarrow} \frac{G_c(s)}{s} \overset{\mathcal{L}^{-1}}{\rightarrow} y(t) \overset{\text{sample}}{\rightarrow} y(kT) \overset{\mathcal{Z}}{\rightarrow} Y(z) \overset{\frac{1}{\text{step}}}{\rightarrow} \frac{z-1}{z} Y(z) = G(z)$$

$$G(z) = \frac{z-1}{z} \mathcal{Z}\left(\mathcal{L}^{-1}\left(\frac{G_c(s)}{s} \right)_{t=kT} \right)$$

Response matching: ramp invariance

Take a continuous time filter/system with Laplace transfer function $G_c(s)$. The corresponding *ramp response invariance* digital filter (with sampling period T) is a digital filter whose ramp response is equal to the ramp response of the continuous time filter sampled at $t = kT$.

$$G(z) = \frac{(z-1)^2}{Tz} \mathcal{Z}\left(\mathcal{L}^{-1}\left(\frac{G_c(s)}{s^2} \right)_{t=kT} \right)$$

Note: any waveform invariance can be considered. The digital filter will preserve the properties of the continuous filter response to that particular waveform.

Impulse vs. Step vs. Ramp invariance

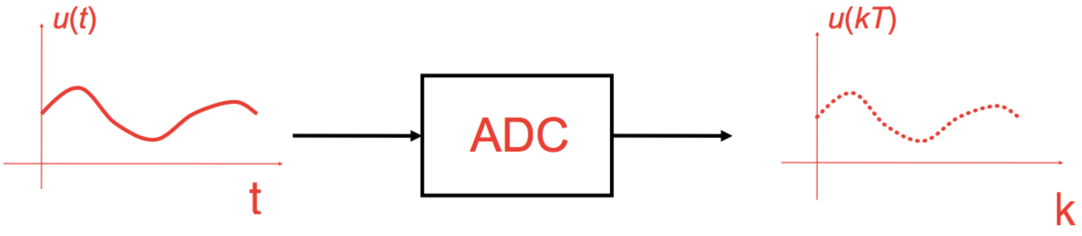
- When do we choose one over the other?
- This depends on the properties of the conversion from analog to digital (Digital to Analog Converters DAC and Analog to Digital Converter ADC) .
- You need to choose a response to an analog signal that will be converted to its digital equivalent by the ADC

Analog-to-digital converter (ADC)

Takes a continuous time signal $u(t)$, which is assumed to be continuous, and sample it to produce the number sequence $u(kT)$.

T is the sampling time.

ADC also termed *sampler*

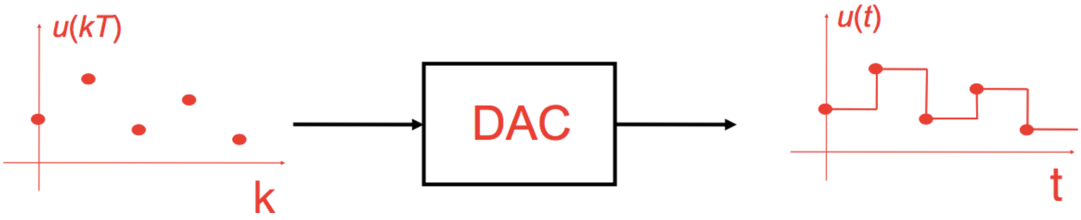


Digital-to-analog converter (DAC)

Take the number sequence $u(kT)$ and produces a continuous time signal u(t).

Zero order hold:

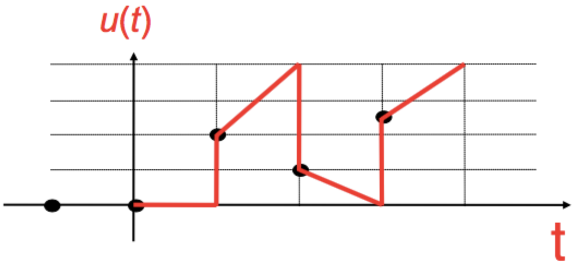
$$u(t) = u(kT) \qquad kT \leq t < (k+1)T$$



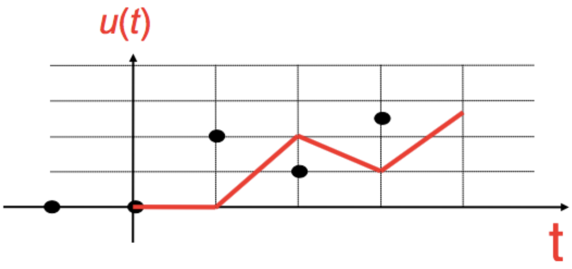
Digital-to-analog converter (DAC)

First order hold:

Linear extrapolation through the last two discrete inputs



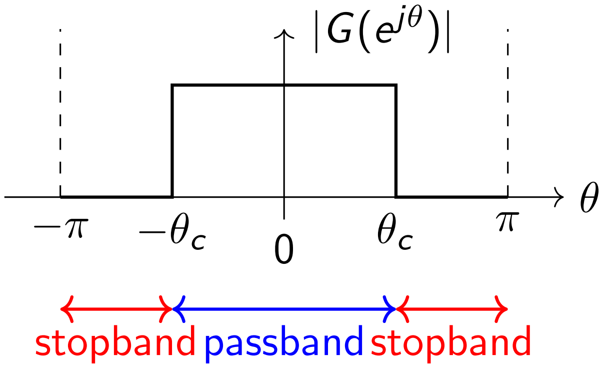
Linear interpolation of last two discrete inputs



Digital Filtering

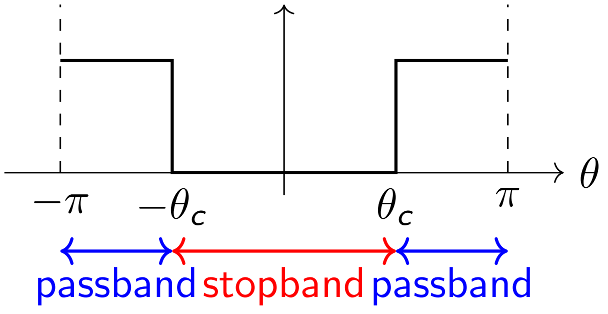
Part I: Design of IIR filters

Desired frequency responses



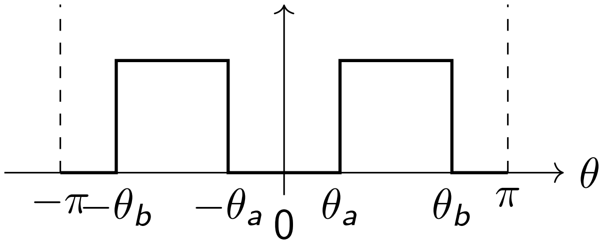
Lowpass:

$$G(e^{j\theta}) = \begin{cases} 1 & |\theta| \leq \theta_c \\ 0 & |\theta| > \theta_c \end{cases}$$



Highpass:

$$G(e^{j\theta}) = \begin{cases} 0 & |\theta| \leq \theta_c \\ 1 & |\theta| > \theta_c \end{cases}$$

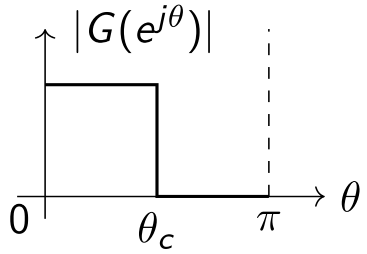


Bandpass:

$$G(e^{j\theta}) = \begin{cases} 1 & \theta_a \leq |\theta| \leq \theta_b \\ 0 & \text{otherwise} \end{cases}$$

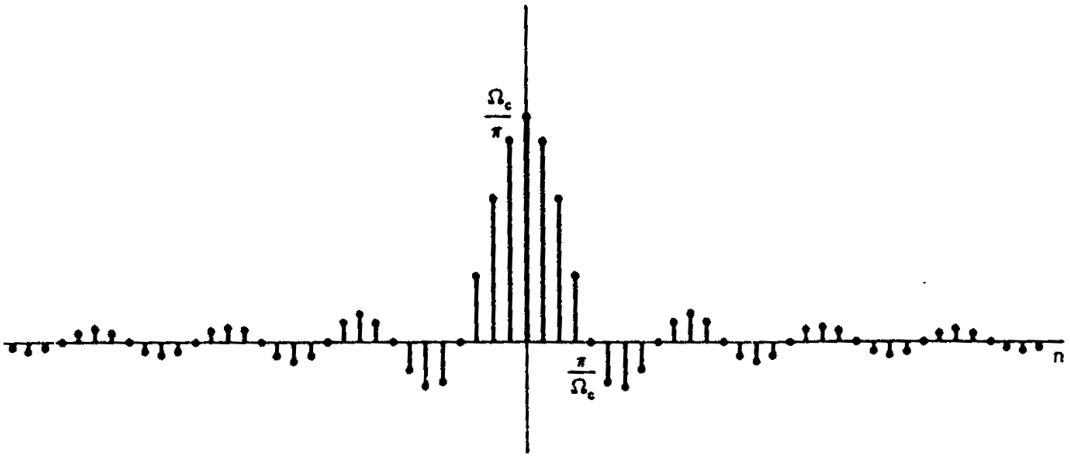
Ideal lowpass delta response

Inverse DTFT as seen in Examples Paper 1 (Question 2 (b)):



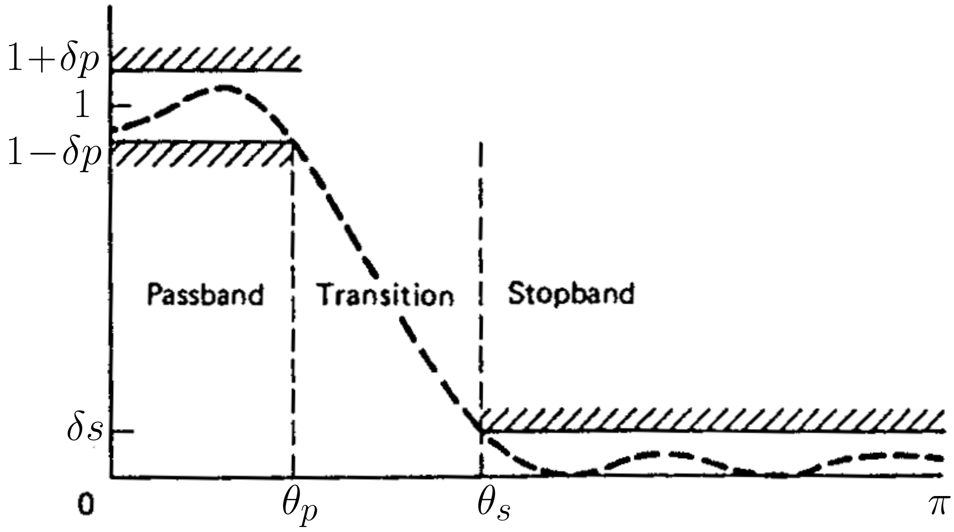
$$g_k = \frac{1}{2\pi}\int_{-\pi}^{\pi} G(e^{j\theta})e^{j\theta k}\,d\theta = \frac{1}{2\pi}\int_{-\theta_c}^{\theta_c} e^{j\theta k}\,d\theta$$

$$= \left[\frac{e^{j\theta k}}{2j\pi k}\right]_{-\theta_c}^{\theta_c} = \frac{\sin(\theta_c k)}{\pi k} = \frac{\theta_c}{\pi}\,\mathrm{sinc}(\theta_c k)$$



- It is non-causal and has a two-sided infinite delta response.
- Not realisable as a real-time linear system ⟶ approximate!

Realizable filter specs



A typical filter specification must specify maximum permissible deviations from the ideal:

- band edge frequencies or corner frequencies
- a maximum passband ripple
- a minimum stopband attenuation

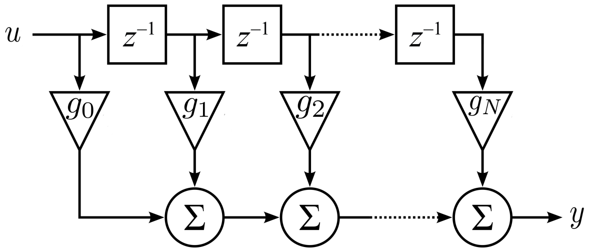
Example: $\delta p = 0.06$ gives peak-to-peak passband ripple $\simeq 1dB$; $\delta s = 0.01$ gives a minimum stopband attenuation $= 40dB$.

FIR (Finite Impulse Response) filters

- FIR filters are simple

$$G(z) = \sum_{k=0}^{N} g_k z^{-k} \quad \rightarrow \quad y_n = \sum_{k=0}^{N} g_k u_{n-k}$$

- Feedforward or non-recursive



- Realized efficiently in hardware (FFT)
- Inherently stable: $G(z) = \dfrac{\sum_{k=0}^{N} g_k z^{N-k}}{z^N}$, all poles in 0.

IIR (Infinite Impulse Response) Filters

- Feedback: the output "goes back" to the filter:

$$y_k = -a_{n-1} y_{k-1} + \cdots - a_0 y_{k-n} + b_m u_{k-n+m-1} + \cdots + b_0 u_{k-n}$$

- They have finite polynomial representation:

$$A(z)Y(z) = B(z)U(z)$$

$$A(z) = z^n + a_{n-1} z^{n-1} + \cdots + a_0$$
$$B(z) = b_m z^m + b_{m-1} z^{m-1} + \cdots + b_0$$

but infinite impulse response:

$$G(z) = \frac{B(z)}{A(z)} = \sum_{k=0}^{\infty} g_k z^{-k}$$

- They typically meet a given set of specifications with a much lower filter order than FIR filters.

FIR vs IIR

FIR filters

- Simple, stable, feedforward, fast implementation (FFT)
- Can have exactly linear phase $G(e^{j\theta}) = |G(e^{j\theta})|e^{-j\theta}$
- Design methods are generally linear (easy, efficient, next lecture)
- Higher filter order than IIR filters to achieve a given level of performance.
- Often greater delay than for an equal performance IIR filter.

IIR filters

- Can be unstable, use feedback
- Design methods: (i) direct optimization of the transfer function (not covered), (ii) generation of a digital filter from an analogue prototype (this lecture).
- Lower filter order than FIR filters to achieve a given level of performance.
- Shorter delay than for an equal performance FIR filter.

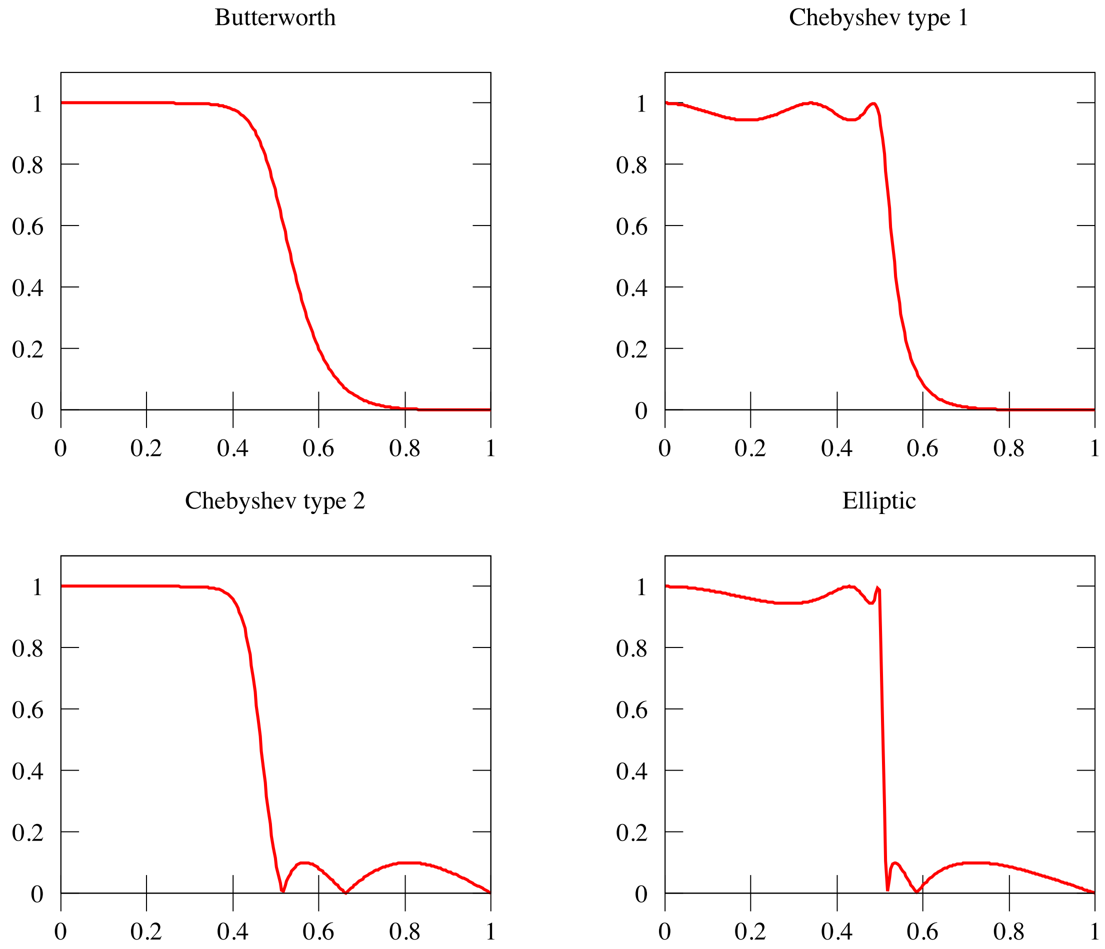
IIR design

Use the continuous to discrete mappings from Handout 8.

1. translate the filter specs on θ (where appropriate) to analog specs in ω
2. design an analog filter (we will see a few techniques)
3. either use **algebraic transforms** (forward difference, backward difference or Tustin transform) to obtain a discrete-time filter
4. alternatively, use **response matching** (e.g. impulse invariant mapping)
5. check design obtained, adjust if necessary, verify stability

Note: response matching may be better suited for FIR design

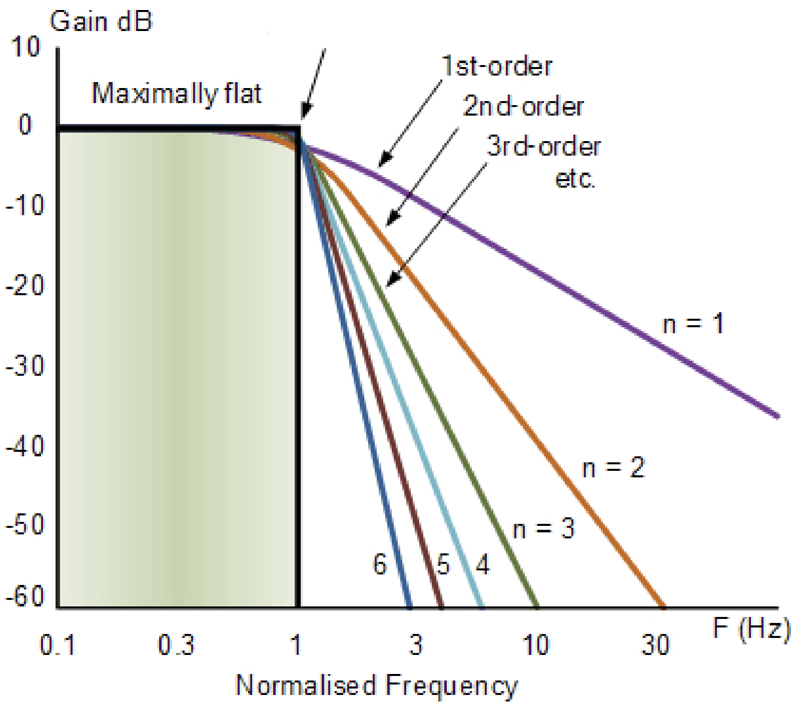
Classical analog prototypes



Butterworth filter

*N*th-order lowpass, $G_c(s)$ satisfies

$$G_c(s)G_c(-s) = \frac{1}{1 + \left(\frac{s}{j\omega_c}\right)^{2N}}$$

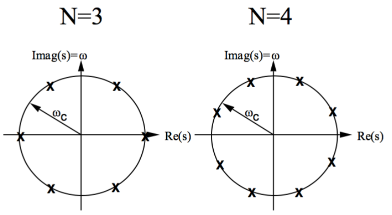


Butterworth filter

Nth-order lowpass, $G_c(s)$ satisfies

$$G_c(s)G_c(-s) = \frac{1}{1 + \left(\frac{s}{j\omega_c}\right)^{2N}}$$

- Unit DC gain, $G_c(j0) = 1$.
- -3dB cutoff frequency at $s = j\omega_c$.
- $G_c(s)G_c(-s)$ poles satisfies $\left(\frac{s}{j\omega c}\right)^{2N} = -1$, i.e. $s = j\omega_c e^{\frac{j(2k+1)\pi}{2N}}$



- If p_i is a root of $G_c(s)$ then $-p_i$ is a root of $G_c(-s)$. Thus the poles of $G_c(s)$ are those roots lying in the left half plane (stability), so that
$$G_c(s) = \prod_{i=1}^{P} \frac{1}{s + p_i}$$
- `scypi.signal.butter(N,Wn)` designs analog and digital Butterworth filters.

Band transformations

Prototypes are typically lowpass. Standard transformation can be used to convert a lowpass prototype into other types.

Assuming a lowpass prototype with cutoff at 1:

- Lowpass to Lowpass:
set $s = \frac{\bar{s}}{\omega_c}$ to change the cutoff frequency to ω_c.
- Lowpass to Highpass:
set $s = \frac{\omega_c}{\bar{s}}$ to get highpass with cutoff frequency at ω_c.
- Lowpass to Bandpass:
set $s = \frac{\bar{s}^2 + \omega_l \omega_u}{\bar{s}(\omega_u - \omega_l)}$ to get bandpass with lower cutoff at ω_l and upper cutoff at ω_u.
- Lowpass to Bandstop:
set $s = \frac{\bar{s}(\omega_u - \omega_l)}{\bar{s}^2 + \omega_l \omega_u}$ to get bandstop with lower cutoff at ω_l and upper cutoff at ω_u.

Band transformation example

- Lowpass to Lowpass by $s = \frac{\bar{s}}{\omega_c}$

$$\frac{1}{s+1} \quad \rightarrow \quad \frac{1}{\frac{\bar{s}}{\omega_c}+1} = \frac{\omega_c}{\bar{s}+\omega_c}$$

  after transformation cutoff at ω_c, $|G(j0)| = 1$, $|G(j\infty)| = 0$.

- Lowpass to Highpass by $s = \frac{\omega_c}{\bar{s}}$

$$\frac{1}{s+1} \quad \rightarrow \quad \frac{1}{\frac{\omega_c}{\bar{s}}+1} = \frac{\bar{s}}{\bar{s}+\omega_c}$$

  after transformation cutoff at ω_c, $|G(j0)| = 0$, $|G(j\infty)| = 1$.

- try the others...

Course outline

- you can now do questions 1-3 and 5-9 in Examples Paper 2
- next Tuesday we will complete digital filtering with FIR filtering (and you'll be able to do Question 4 after that)
- 3 lectures on the Discrete Fourier Transform (DFT) (covering Questions 10-12 in Examples Paper 2)
- final 3 lectures on continous time random processes (Examples Paper 3)