Filters

slide set # 7

Alessandro Falaschi

Signal Processing and Information Theory
Bioinformatics Degree, Sapienza University of Roma

Didactic material on TeoriadeiSegnali.it

May 2023

Content index

Overwiev

Summary of the presentation

Now let's delve into the physical nature of the filtering devices. Analog filters are first briefly introduced, since the terminology is still valid to describe different classes of numeric filters, and the result of their synthesis technique can be converted into the zeta transform of the transfer function used to describe a numeric filter

Digital filters are a class of filters operating according to a computational architecture rather than a circuit one, and can represent the effect of natural causes as well as being defined by man. The transversal filter or FIR is then introduced, both from the point of view of analysis and synthesis, highlighting some implementation aspects

The squared frequency response of a *first order* FIR is discussed in more detail, as it is the basis of how FT Infrared Spectrometry works; then also the first order IIR is discussed together with its applications.

At this point, numerical filters are illustrated, describing the synthesis of the coefficients for an FIR filter, as well as the description of a numerical filter using finite differences, arriving to describe its transfer function in terms of zeta transform, and to represent its canonical architecture. Finally, the design of numerical filters obtained by applying a change of variable to the transfer function resulting from the design of analog filters is mentioned

Now we talk about...

What analog filters are made of

whether they are designed or natural

Man-made analog filters operate on time-continuous electrical signals, and are made using a variety of techniques, such as

- circuits made up of capacitors, inductors and resistances (RLC, or *lumped constants*), called *passive filters* as they do not require power. The difficulty of producing small components has led to realize their function through

- *operational amplifiers* that give rise (up to about 1 MHz) to *active filters*, while
- at higher frequencies *crystal*, *electromechanical*, *waveguide* and *microstrip* filters can also be implemented

But analog filters are not only those produced by man: the behavior of any physical system, be it mechanical, electrical, acoustic (for example a concert hall), hydraulic, pneumatic, can be considered as a filter

Analog filters as ratio of polynomials

after which, Fourier leverages Laplace

In all cases it is possible to describe the input-output behaviour $y(t) = f\{x(t)\}$ of the system by means of differential equations, whose *Laplace transform* gives the expression of a *transfer function* of the type

$$H(s) = \frac{Y(s)}{X(s)} = \frac{\sum_{i=0}^{M} a_i s^i}{\sum_{j=0}^{N} b_j s^j} \qquad M < N \tag{1}$$

which is a function of the *complex variable* $s = \sigma + j\Omega$: setting now $s = j2\pi f$ we obtain the *frequency response* $H(f) = {}^{Y(f)}/_{X(f)} = H(s = j2\pi f)$

- Indeed if $s = j\pi 2f$ the definition of the *Laplace* transform $H(s) = \int_{-\infty}^{\infty} h(t)\, e^{-st} dt$ becomes identical to the *Fourier* one, and is equivalent to calculating $H(s)$ along the imaginary axis. This is valid only if the filter is *stable*, which in the Laplace domain translates into the need for the poles of $H(s)$ to be to *the left* of imaginary axis

The degree N of the denominator of (1) is equal to the number of its complex roots (or *poles*) and defines *the order* of the filter, i.e. it measures its constructive complexity, as well as *the rate* of variation of $H(f)$

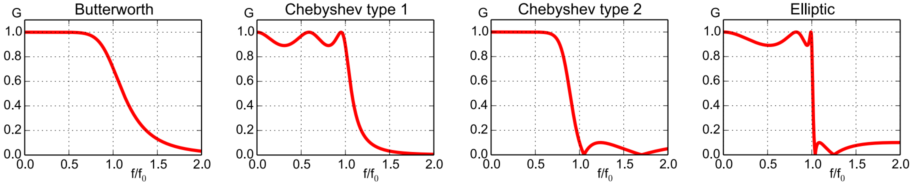
Now we talk about...

1. **Analog filters**
   - Components and polinomials
   - Filter classes and design template
   - Filters outcome
2. **Digital filters**
   - Key assumptions
   - Finite impulse response or FIR
   - First order FIR
   - First order infinite impulse response (IIR) filter
3. **Numeric filters**
   - FIR synthesis starting from the continuous time description
   - Zeta transform and filtering
   - Synthesis of an IIR filter starting from an analog filter

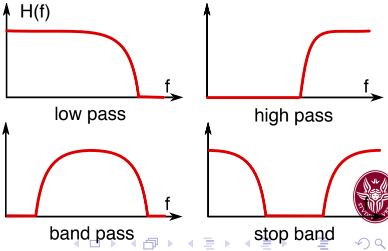
Bessel, Butterworth, Chebyschev or Elliptic?

Restricting the type of the polynomials appearing in $H(s)$ (i.e. the position allowed for its poles) we obtain a frequency response whose shape is typical of that type of polymomial

Here a *low-pass* filter type is depicted:



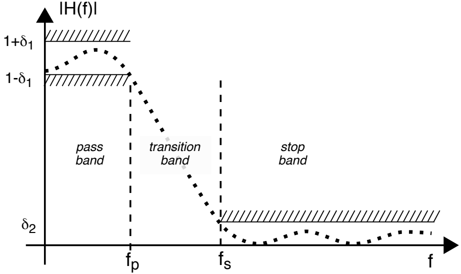
But keep in mind that specifications can be given to get a different kind of frequency response, such as those shown on the side



Tolerance scheme mask

The effective values of the coefficients of the polynomials of $H(s)$ are calculated through an optimization procedure, fed by a template mask which (for a low pass filter) imposes the values

- a *passband* $f < f_p$ which indicates the frequency region to pass;
- the percentage value δ_1 within which $H(f)$ can oscillate in the passband;
- a *suppressed band* $f > f_s$ in which the input frequency components must be attenuated by at least δ_2% with respect to those of the pass band;

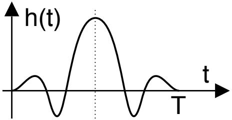


- a *transition band* $f_s - f_p$ in which the frequency response varies;
- whether or not the *phase linearity* property is required for the filter
  - using Bessel polynomials phase distortion can be avoided, at the expense of a wider transition band

But the degree N of the $H(s)$ denominator has to be big enough!

Phase linearity

There is a sufficient condition to ensure that a filter has a linear phase response: this happens for an impulse response with *even symmetry* with respect to half of its duration T, i.e.



$$h(t)|_{0<t<T/2} = h(T-t)$$

Proof: given that, consider a second filter with impulse response $g(t) = h(t + T/2)$ obtained by anticipating $h(t)$ by half duration:

- $g(t)$ has *even* symmetry respect to $t = 0$, and therefore its transform is a *real* $G(f)$
- now revert the time translation, and obtain $H(f) = G(f)\, e^{-j2\pi f \frac{T}{2}}$, i.e. a frequency response with a linear phase $\varphi(f) = -\pi f T$

The condition described is easily achievable by means of a *transversal filter*, see page 17

Now we talk about...

Simpler filter cells in cascade

Once the coefficients are known, $H(s)$ (1) is rewritten in factored form

$$H(s) = \prod_{j=1}^{N/2} H_j(s)$$

with each term $H_j(s)$

- having a pair of conjugated poles and
- corresponding to a second order resonant circuit or *circuit cell*, obtaining the maximum for $|H_j(j2\pi f)|^2$ at the resonant frequency f_0^j.

By placing the different cells *in cascade*, an $H(f)$ is obtained which satisfies the requirements established by the tolerance scheme.

While the Bessel, Butterworth and Chebyshev-I filters only have poles, the Chebyshev-II and Elliptical filters possess also zeros.

A polynomial $P(s) = \sum_{j=0}^{N} b_j s^j$ is zeroed for the N values $s = \beta_j$, known as *zeros* of $P(s)$. The same polynomial can therefore be written as $P(s) = \prod_{j=1}^{N} (s - \beta_j)$, or by grouping the zeros in pairs (possibly conjugated) in order to obtain a development with second degree factors of the type $P(s) = \prod_{j=1}^{N/2} \left(s^2 + c_j s + d_j\right)$ to which, if N is odd, a first degree factor must be added.

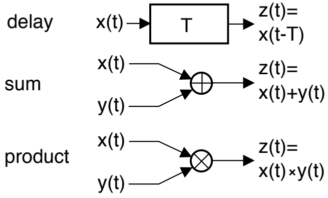
From analog to numeric filters

- although what has been discussed may seem directly applicable only to the case of analog signals
- at page 35 we will show how these results can also be used for the case of numerical signal processing
- since there are techniques
    - based on variable changes that *map* the left half plane of the variable s inside the *unit circle* of the z-plane
- which allow to pass from one $H(s)$ to one $H(z)$, and from it to a realization of the filter in *numerical* form

Now we talk about...

No circuits, but a computation scheme

Digital filters are a class of filters defined by a *computational* rather than a circuit scheme, expressed only in terms of the elementary processing units *product, sum and delay*, producing filtering effects on signals in transit.



They can represent a model of *natural causes*

- for example, the acoustics of a room (or of a theater) are the result of the various reflections of sounds on walls and furniture, each reflection more or less attenuated, and with different propagation delay between source and listener. But this also applies to radio waves (be it TV, WIFI or telephones)
- or you can also design a specific architecture to combine the *product, sum and delay* elements and achieve the desired effect

For the class of digital filters

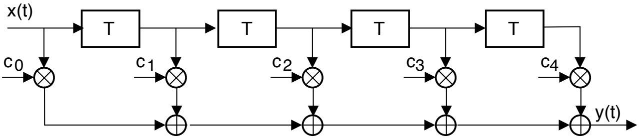
- the convolution integral *is reduced to a sum*, and
- although the following analysis is based on *continous time* signals, they can be implemented by operating directly on the *samples* of the signals, which can be done via software or hardware

Now we talk about...

1. Analog filters
   - Components and polinomials
   - Filter classes and design template
   - Filters outcome
2. Digital filters
   - Key assumptions
   - Finite impulse response or FIR
   - First order FIR
   - First order infinite impulse response (IIR) filter
3. Numeric filters
   - FIR synthesis starting from the continuous time description
   - Zeta transform and filtering
   - Synthesis of an IIR filter starting from an analog filter

Transversal filter

All of its delay elements take the same T value, and the filter is named *transversal* after its computation scheme



whose effect can be written as

$$y(t) = \sum_{n=0}^{N} c_n \, x(t - nT) \tag{2}$$

in which the coefficients c_n are often named *taps* because they *modulate* the amount of the input delayed by nT

If a Dirac pulse $\delta(t - nT)$ is put at the input, we will observe an impulse response

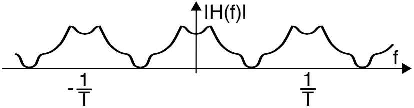
$$h(t) = \sum_{n=0}^{N} c_n \, \delta(t - nT)$$

so that for any other input the result of the convolution is that given by (2)

Periodic frequency response

- The number of taps N is *the order* of the filter, related to its complexity and to the slope of its frequency response
- since N is a finite numer, the filter has a *finite* (length) *impulse response*, or FIR
- **analysis**: as $h(t)$ takes the form of a sampled signal $h^{\bullet}(t)$, its Fourier transform is periodic in frequency with period $F = \frac{1}{T}$:

$$H(f) = \mathcal{F}\{h(t)\} = \sum_{n=0}^{N} c_n \, \mathrm{e}^{-j2\pi fnT}$$

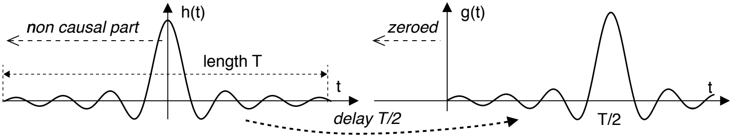


- so that we can also write $H(f) = \sum_{n=-\infty}^{\infty} G\left(f - \frac{n}{T}\right)$ and (**synthesis**) compute the values of the taps by applying the Fourier series expansion:

$$c_n = T \int_{-1/2T}^{1/2T} G(f) \, \mathrm{e}^{j2\pi fnT} df$$

FIR implementation

- depending on $G(f)$, coefficients c_n can remain non-negligible for high values of n, so that the series will have to be truncated (that is, windowed), consequently modifying the resulting $H(f)$
- now you have a finite set $c_{-N/2}, \cdots, c_{-N/2}, \cdots, c_{N/2}$ of N samples, and you have to *rigth shift* them in order to have a causal filter



- the previous step adds a linear phase contribution the the resultant $H(f)$, but if you chose an even real $H(f)$, your c_n will be even real, and the filter will have a linear phase

A completely numerical FIR

- if you intends to implement the FIR numerically by working on samples x_n of $x(t)$, and obtain the corresponding samples y_n (and not $y(t)$ for any t), then
  - we are referring to a sampling frequency $f_c = \frac{1}{T}$
  - the series of input values to the delays blocks constitutes the memory (or state) of the filter
  - for the output samples y_n the expression of the *discrete convolution* formula must holds, i.e.

$$y_n = T \cdot \sum_{k=-\infty}^{\infty} h_k x_{n-k}$$

- in order to have FIR
  - the last summation is limited to $k = 0, \cdots N$ and
  - (by comparison) we obtain

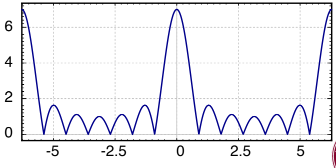
$$c_n = T \cdot h_n$$

The *moving average* filter

It is one of the simplest forms of FIR filter, and has coefficients c_n all equal to $\frac{1}{N+1}$, so that it performs a *moving average* (or MA) as an arithmetic mean of the last input values, or

$$y_k = \frac{1}{N+1} \sum_{n=0}^{N} x_{k-n}$$

- is the method commonly used to *smooth* discrete time series, such as temperatures or stock prices
- is the equivalent of a continuous-time filter with $h(t) = rect_{(N+1)T_c}(t)$ and has a low-pass effect
- it's z-transform is equal to $H_{MA}(z) = \sum_{n=0}^{N-1} z^{-n} = \frac{1-z^{-N}}{1-z^{-1}}$
- the corresponding $H(\omega)$ is
  $$|H_{MA}(\omega)| = \left| \frac{\sin(\omega N/2)}{\sin(\omega/2)} \right|$$
- for instance, setting $N = 7$ the graph of $|H_{MA}(\omega)|$ is obtained as shown on the side

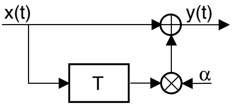


Now we talk about...

First order transversal filter

With $N = 1$ the transversal filter can be redrawn as shown, having set $c_0 = 1$ and $c_1 = \alpha$. It's impulse response is

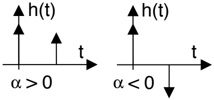
$$h(t) = \delta(t) + \alpha\delta(t - T)$$



so that the frequency response is equal to

$$H(f) = 1 + \alpha e^{-j2\pi fT}$$

and is *a model of an echo* between source and receiver

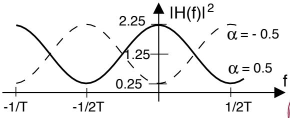


Its power gain $|H(f)|^2$ is

$$\begin{aligned} |H(f)|^2 &= (\Re\{H(f)\})^2 + (\Im\{H(f)\})^2 = (1 + \alpha\cos 2\pi fT)^2 + (\alpha \sin 2\pi fT)^2 = \\ &= 1 + 2\alpha\cos 2\pi fT + \alpha^2\left(\cos^2 2\pi fT + \sin^2 2\pi fT\right) = \\ &= 1 + \alpha^2 + 2\alpha\cos 2\pi fT \end{aligned}$$

whose plot is shown for $\alpha = \pm 0.5$.

Note that in the range $|f| < \frac{1}{2T}$ the $|H(f)|^2$ can behave either as a low-pass or an high-pass, depending on the sign of α



Differentiator and comb filters

Differentiator

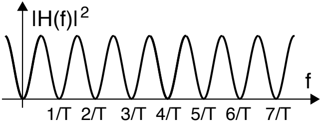
If $\alpha = -1$ we obtain a *numerical differentiator*, since in this case the output sequence

$$y_n = x_n - x_{n-1}$$

is the *finite difference* of the input

Comb filter

Still in case that $\alpha = -1$ and with a continuous-time input signal with bandwidth $W \gg {}^{1}/_{T}$ the filter is able to remove a *periodic component* of period T, since in this case



$$|H(f)|^2 = 0 \quad \text{with} \quad f = {}^{n}/_{T}$$

i.e. in correspondence of the harmonics of ${}^{1}/_{T}$

Note also that if $\alpha = 1$ the shape of $|H(f)|^2$ is still a *comb*, but it *shifts* by ${}^{1}/_{2T}$, with the effect of *not removing* the continous component, and canceling the frequency ${}^{1}/_{2T}$ together with its *odd* harmonics

Now we talk about...

A matter of recursion

It implements a somewhat *complementary* architecture to that of a FIR, and the *infinite* duration of $h(t)$ arises as a consequence of a recursion feature of the filter, making the output values dependent on *previous output values*

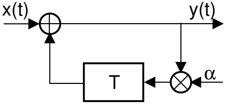
The simplest form of such a filter obey to the relation

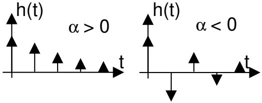
$$y(t) = x(t) + \alpha y(t - T)$$

so that when $x(t) = \delta(t)$ we obtain

$$h(t) = \sum_{n=0}^{\infty} \alpha^n \delta(t - nT)$$

in accordance with the computational scheme shown





Hence, it is easy to derive $H(f) = \sum_{n=0}^{\infty} \alpha^n e^{-j2\pi f n T}$ which converges to a more compact expression by applying the equality of the geometric series $\sum_{n=0}^{\infty} \beta^n = \frac{1}{1-\beta}$, which holds for $|\beta| < 1$, i.e.

$$H(f) = \sum_{n=0}^{\infty} (\alpha e^{-j2\pi f T})^n = \frac{1}{1 - \alpha e^{-j2\pi f T}}$$

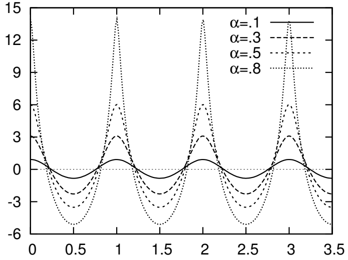
If $|\alpha| > 1$ the filter becomes *unstable*, since infinitesimal inputs produce an exponentially increasing output

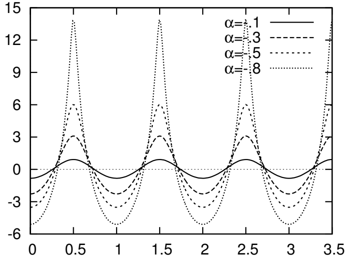
Frequency response of the first order IIR filter

With steps similar to those already seen for the FIR we obtain

$$|H(f)|^2 = \frac{1}{(1-\alpha\cos 2\pi fT)^2 + (\alpha\sin 2\pi fT)^2} = \frac{1}{1+\alpha^2 - 2\alpha\cos 2\pi fT}$$

whose graph is shown (in dB) below for $T = 1$ and α positive (left) or negative (right)



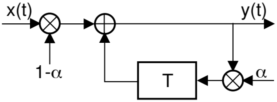


As evident, if $|\alpha| \to 1$ then $|H(f)|^2$ becomes very tight around frequencies that are spaced by of $1/T$, the *resonant* frequencies

Integrator and exponential smoothing

Integrator If $\alpha = 1$ then the filter behaves like a a *perfect integrator* which, for example, produces an output ramp, if there is an input step function

Exponential moving average or **EMA** A variant of the first order IIR is obtained by writing the input-output relationship as $y(t) = \alpha y(t - T) + (1 - \alpha) x(t)$ which corresponds to

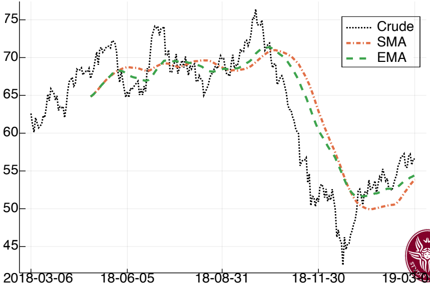


$$h(t) = (1 - \alpha) \sum_{n=0}^{\infty} \alpha^n \delta(t - nT) \quad \text{and} \quad |H(f)|^2 = \frac{(1-\alpha)^2}{1+\alpha^2-2\alpha \cos 2\pi f T}$$

with unity gain at zero frequency

Setting $0 < \alpha < 1$ the filter behaves like a low pass, and calculates an *exponentially weighted average* used as a method to smooth the data

For a *mean number* of the samples equal to that of a FIR MA, EMA is *more reactive* to sudden trend changes, and this characteristic makes it preferred, for instance, in the financial markets



Now we talk about...

1. **Analog filters**
   - Components and polinomials
   - Filter classes and design template
   - Filters outcome
2. **Digital filters**
   - Key assumptions
   - Finite impulse response or FIR
   - First order FIR
   - First order infinite impulse response (IIR) filter
3. **Numeric filters**
   - FIR synthesis starting from the continuous time description
   - Zeta transform and filtering
   - Synthesis of an IIR filter starting from an analog filter

Looking for a sampled impulse response

In the case of a completely numerical implementation of a FIR filter, a first approach can be to synthesize the transversal filter and then use the tap values in the discrete convolution formula

$$y_k = \sum_{n=-\infty}^{\infty} h_n x_{k-n}$$

to be computed either as is, or in batch mode. But let us report about alternatives

Windowing of the impulse response The values of h_n are obtained by sampling a *windowed* version $h_w(t) = h(t) \cdot w(t)$ of the desired $h(t)$, but this produces an $H_w(f) = H(f) * W(f)$, and therefore the window function $w(t)$ must be carefully selected

Uniform oscillation in frequency or **equiripple** Is an iterative technique ('72) which *minimizes* the *maximum approximation error* between the desired values for $|H(f)|$ and the values obtainable by expressing $|H(f)|$ as a combination of *Chebyshev polynomials*. The result is $\hat{H}(f)$ with reduced oscillations at all frequencies, and the h_n coefficients are obtained by means of an IDFT of the sequence H_n obtained by sampling $\hat{H}(f)$

Now we talk about...

1. **Analog filters**
   - Components and polinomials
   - Filter classes and design template
   - Filters outcome
2. **Digital filters**
   - Key assumptions
   - Finite impulse response or FIR
   - First order FIR
   - First order infinite impulse response (IIR) filter
3. **Numeric filters**
   - FIR synthesis starting from the continuous time description
   - Zeta transform and filtering
   - Synthesis of an IIR filter starting from an analog filter

General expression of a numeric filter
and of its Zeta transform

The *recursive calculation scheme* can be associated with the FIR component, allowing to implement filters with reduced complexity and good frequency selectivity, to the detriment of the lack of phase linearity characteristic

In the most general terms the input-output relationship for an IIR filter can be written as

$$y(t) = \sum_{k=1}^{N} b_k y(t - kT) + \sum_{k=0}^{M} a_k x(t - kT)$$

which means that the output values depend on a finite number of N past *output* values, as well as $M + 1$ input values. Its zeta transform can written as

$$Y(z) = \sum_{k=1}^{N} b_k Y(z) z^{-k} + \sum_{k=0}^{M} a_k X(z) z^{-k}$$

or

$$Y(z)\left(1 - \sum_{k=1}^{N} b_k z^{-k}\right) = X(z) \sum_{k=0}^{M} a_k z^{-k}$$

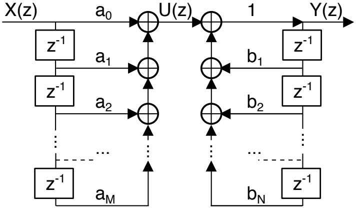
so that

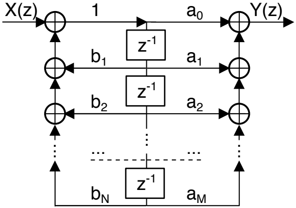
$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{M} a_k z^{-k}}{1 - \sum_{k=1}^{N} b_k z^{-k}}$$

looks (very) like the H(s) found for the transfer function of an analog filter

Canonical architecture of a numerical filter

- The previous expression for $H(z)$ can be considered as a complex function of the complex variable z, which corresponds to a frequency response $H(\omega)$ by making z move on the unit circle (i.e. setting $z = e^{j\omega}$), with a graph depending on the position of the poles and zeros of $H(z)$
- For the filter to be stable, the poles of $H(z)$ must fall *within the unit circle*
- Moreover we can rewrite the relation $y(t) = \sum_{k=1}^{N} b_k y(t - kT) + \sum_{k=0}^{M} a_k x(t - kT)$ as $y_n = \sum_{k=1}^{N} b_k y_{n-k} + u_n$ by setting $u_n = \sum_{k=0}^{M} a_k x_{n-k}$ in order to obtain the computational architecture shown on the left (z^{-1} blocks represent a unitary delay), highlighting the FIR and IIR components





- Considering now the commutativity of the filtering operations we arrive at the architecture shown on the right, in which the delay blocks (i.e. the memory of the filter) are put *in common*

Now we talk about...

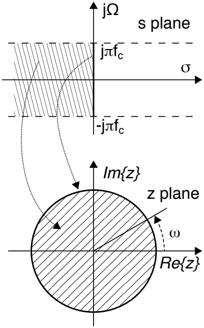
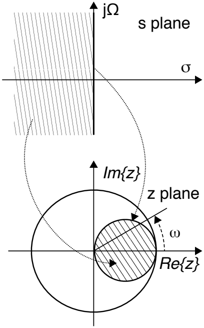
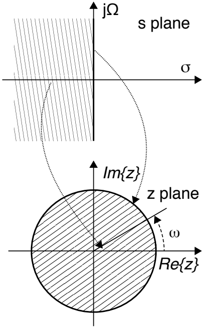
1. **Analog filters**
   - Components and polinomials
   - Filter classes and design template
   - Filters outcome
2. **Digital filters**
   - Key assumptions
   - Finite impulse response or FIR
   - First order FIR
   - First order infinite impulse response (IIR) filter
3. **Numeric filters**
   - FIR synthesis starting from the continuous time description
   - Zeta transform and filtering
   - Synthesis of an IIR filter starting from an analog filter

Finding the proper transformation

- Things starts with the transfer function $H_a(s) = \frac{\sum_{m=0}^{M} a_m s^m}{\sum_{n=0}^{N} b_n s^n}$ of the *analog filter*, the design of which is based on well-established and efficient methods
- Then, a *change of variable* is performed between the independent variables $s = \sigma + j\Omega$ and $z = \rho e^{j\omega}$, thus obtaing a *new rational function* $H(z)$ whose coefficients are then used for the implementation of the numerical filter
- The change of variable must meet some requirements, such as
  - mapping of the left half-plane of the variable s *within* the unit circle of the z plane
    - this preserves *stability*
  - map the imaginary axis of the s-plane to the unit circle in the z plane
    - this preserves the appearance of the *frequency response*
- as a result, poles and zeros *are also mapped* in between the two domains

The following slide summarizes the solutions adopted, see the book for a discussion

Use the most suitable for you

method	*invariance of $h(t)$, poles and zeros*	*difference equation*	*bilinear transform or Tustin's*
relation	$z = e^{sT}$	$s = \frac{1-z^{-1}}{T}$	$s = \frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}}$
mapping			
weaknesses	aliasing	it can only be used at low frequencies	distortion of the frequency axis