Audio Signal Processing : III. Filtering

Emmanuel Bacry

bacry@ceremade.dauphine.fr
http://www.cmap.polytechnique.fr/~bacry

Framework :

$s(t)$ is a time continuous signal ($\simeq$ electrical tension)

$s(t)$ is real-valued

Combination of electronic resistance and elecctronic capacitor leads to

$$a_N \frac{d^N v(t)}{dt^N} + \ldots + a_1 \frac{dv(t)}{dt} + a_0 v(t) = b_M \frac{d^M u(t)}{dt^M} + \ldots + b_1 \frac{du(t)}{dt} + b_0 u(t)$$

where

- $u(t)$ is the input voltage
- $v(t)$ is the output voltage

and the $\{a_n]_{0 \leq n \leq N}$ and $\{b_m]_{0 \leq m \leq M}$ are reals

After Fourier transform

$$a_N \frac{d^N v(t)}{dt^N} + \ldots + a_1 \frac{dv(t)}{dt} + a_0 v(t) = b_M \frac{d^M u(t)}{dt^M} + \ldots + b_1 \frac{du(t)}{dt} + b_0 u(t)$$

gives

$$\begin{aligned} & a_N (i\omega)^N \hat{v}(\omega) + \ldots + a_1 (i\omega) \hat{v}(\omega) + a_0 \hat{v}(\omega) \\ & \quad = b_M (i\omega)^M \hat{u}(\omega) + \ldots + b_1 (i\omega) \hat{u}(\omega) \quad + b_0 \hat{u}(\omega) \end{aligned}$$

And

$$
\begin{aligned}
& a_N(i\omega)^N \hat{v}(\omega) + \ldots + a_1(i\omega)\hat{v}(\omega) + a_0\hat{v}(\omega) \\
& \qquad = b_M(i\omega)^M \hat{u}(\omega) + \ldots + b_1(i\omega)\hat{u}(\omega) \quad + b_0\hat{u}(\omega)
\end{aligned}
$$

can be rewritten

$$
\hat{v}(\omega)\sum_{n=0}^{N} a_n(i\omega)^n = \hat{u}(\omega)\sum_{m=0}^{M} b_n(i\omega)^n
$$

Consequently, the operator that associates $u(t)$ (input) to $v(t)$ (output) such that

$$\hat{v}(\omega) \sum_{n=0}^{N} a_n (i\omega)^n = \hat{u}(\omega) \sum_{m=0}^{M} b_m (i\omega)^n$$

is a time-invariant linear operator (i.e., a convolution) with an impulsional response $h(t)$ (i.e. the filter of the convolution) whose Fourier transform is

$$\hat{h}(\omega) = \frac{\hat{v}(\omega)}{\hat{u}(\omega)} = \frac{\sum_{n=0}^{N} a_n (i\omega)^n}{\sum_{m=0}^{M} b_n (i\omega)^n}$$

The considered electronic circuit corresponds to a filter $h(t)$ defined by

$$\hat{h}(\omega) = \frac{\sum_{m=0}^{M} b_m (i\omega)^m}{\sum_{n=0}^{N} a_n (i\omega)^n}$$

which we can rewrite

$$\hat{h}(\omega) = \frac{N(i\omega)}{D(i\omega)}$$

where N and D are two polynomials with real coefficients.

N and D : polynomials with real coefficients :

$$\hat{h}(\omega) = \frac{N(i\omega)}{D(i\omega)}$$

Theorem : The filter $h(t)$ is causal and stable iff

(i) $\delta N < \delta D$

(ii) All the solutions of $D(z) = 0$ are such that $\Re(z) < 0$

The problem : How to design an electronic circuit which corresponds to a fixed $|\hat{h}(\omega)|^2$ (we do not care about the phase) ?

Rephrasing the problem : We choose a positive-valued function $H(\omega)$, and we want to find two polynomials $N(z)$ and $D(z)$ such that

- N and D have real coefficients
- The filter $\frac{N(i\omega)}{D(\omega)}$ is causal and stable, i.e.,
  - (i) $\delta N < \delta D$
  - (ii) All the solutions of $D(z) = 0$ are such that $\Re(z) < 0$
- One has

$$H(\omega) = \frac{|N(i\omega)|^2}{|D(\omega)|^2}$$

Theorem If a positive valued function $H(\omega)$ satisfies

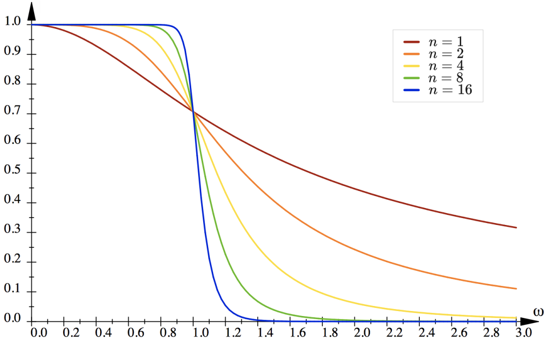
- H is a rationnal fraction in $i\omega$ with real coefficients, of the form
$$H(\omega) = \frac{P(i\omega)}{Q(i\omega)}$$
with $\delta P < \delta Q$
- The poles of H (i.e., the roots of $Q(z)$) are such that $\Re(z) \neq 0$

Then we can design an electronic circuit corresponding to a (stable and causal) filter $h(t)$ such that

$$|\hat{h}(\omega)|^2 = H(\omega)$$

Application The Butterworth low-pass filters $h_{\omega_0,n}$

$$|\hat{h}_{\omega_0,n}(\omega)|^2 = H_n(\omega) = \frac{1}{1 + (\omega/\omega_0)^{2n}}$$



What about the phase ?

- Linear phase filters
- Minimum phase filters

Framework :

$s[n]$ is a time-discrete signal ($=$ sampling of an analog signal)

$s[n]$ is real-valued

Causal filtering in discrete time :

$$h * s[n] = \sum_{k \geq 0} h[k] s[n-k]$$

We want to have a finite number of operations !

$\Longrightarrow$ we need to have $h[k]$ to be compact support

Definition : A Moving Average filtering (MA)

- $f[n]$ (resp. $g[n]$) is the input (resp. output) signal
- $g[n] = h * f[n]$

$$g[n] = \sum_{k=0}^{M} b_k f[n-k]$$

And in Fourier space

$$\hat{g}(e^{i\omega}) = \hat{b}(e^{i\omega})\hat{f}(e^{i\omega})$$

with

$$\hat{b}(e^{i\omega}) = \sum_{k=0}^{M} b_k e^{-i\omega}$$

In that case, the filter h is simply given by

$$\hat{h}(e^{i\omega}) = \hat{b}(e^{i\omega})$$

A Moving Average filtering (MA)

$$g[n] = h * f[n]$$

with

$$\hat{h}(e^{i\omega}) = \hat{b}(e^{i\omega}) = \sum_{k=0}^{M} b_k e^{-ki\omega}$$

This is a Finite-Impulse-Response (FIR) filter (i.e., b is compact support).

$\Longrightarrow$ **If we want to implement "sharp band filters, one need large support which will induce ... long computations.**

How could we implement a (causal) Infinite-Impulse-Response (IIR) filter with a finite number of operations ?

Definition : An Auto-Regressive filtering (AR)

- $f[n]$ (resp. $g[n]$) is the input (resp. output) signal
- $g[n] = h * f[n]$

$$g[n] = f[n] - \sum_{k=1}^{N} a_k g[n-k], \quad \text{with } a_0 = 1$$

or equivalently

$$\sum_{k=0}^{N} a_k g[n-k] = f[n]$$

Then

$$\hat{h}(e^{i\omega}) = \frac{1}{\hat{a}(e^{i\omega})} = \frac{1}{\sum_{k=0}^{N} a_k e^{-ik\omega}}$$

Definition : An Auto-Regressive filtering (AR)

$$g[n] = h * f[n]$$

with

$$\sum_{k=0}^{N} a_k g[n-k] = f[n]$$

And

$$\hat{h}(e^{i\omega}) = \frac{1}{\hat{a}(e^{i\omega})} = \frac{1}{\sum_{k=0}^{N} a_k e^{-ik\omega}}$$

$\Longrightarrow$ ***h*** **is an IIR filter that can be immplemented with a finite number of operations !**

Definition : An ARMA filter (= AR+MA)

$$g[n] = h * f[n]$$

with

$$\sum_{k=0}^{N} a_k g[n-k] = \sum_{k=0}^{M} b_k f[n-k] \quad \text{with } a_0 = 1$$

Thus

$$\hat{h}(e^{i\omega}) = \frac{\hat{b}(e^{i\omega})}{\hat{a}(e^{i\omega})} = \frac{\sum_{k=0}^{M} b_k e^{-ik\omega}}{\sum_{k=0}^{N} a_k e^{-ik\omega}}$$

An ARMA filter (= AR+MA)

$$\hat{h}(e^{i\omega}) = \frac{\sum_{k=0}^{M} b_k e^{-ik\omega}}{\sum_{k=0}^{N} a_k e^{-ik\omega}}$$

Thus, the Z transform is

$$\hat{h}(Z) = \frac{\sum_{k=0}^{M} b_k Z^{-k}}{\sum_{k=0}^{N} a_k Z^{-k}}$$

Let's remember that in general

$$h(Z) = \sum_n h[n] Z^{-n}$$

and the convergence domain is a ring $C = \{Z,\ \rho_1 < |Z| < \rho_2\}$

Thus

- **Causality**

$$\hat{h}(Z) = \sum_{n>0} h[n] Z^{-n}$$

$\Longrightarrow$ The convergence domain is of the form $C = \{\rho_1 < |Z|\}$

- **Stability**

$$\sum_n |h[n]| < +\infty$$

$\Longrightarrow$ The convergence domain is such that $1 \in C$

An ARMA filter (= AR+MA)

$$\hat{h}(Z) = \frac{\sum_{k=0}^{M} b_k Z^{-k}}{\sum_{k=0}^{N} a_k Z^{-k}} = \frac{N(Z)}{D(Z)}$$

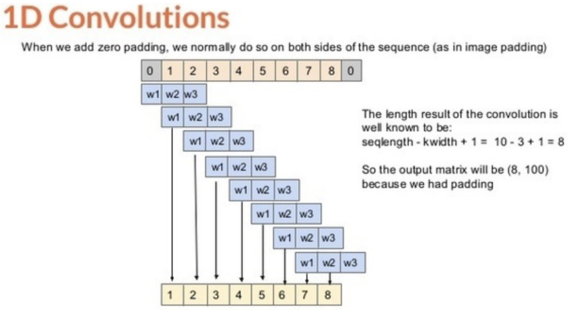
where N and D are polynomials with real coefficients.

Thus

- **Causality** : Always ! (by definition)
- **Stability** : $1 \in C$

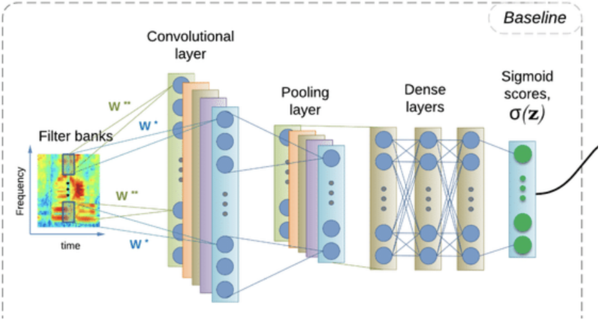
$\Longrightarrow$ C is of the form $C = \{Z,\ |Z| > \rho\}$ with $\rho > 1$
$\Longrightarrow$ the roots of $D(Z) = 0$ verify $Z < 1$

A MA filter $\simeq$ first layer of a 1D-CNN

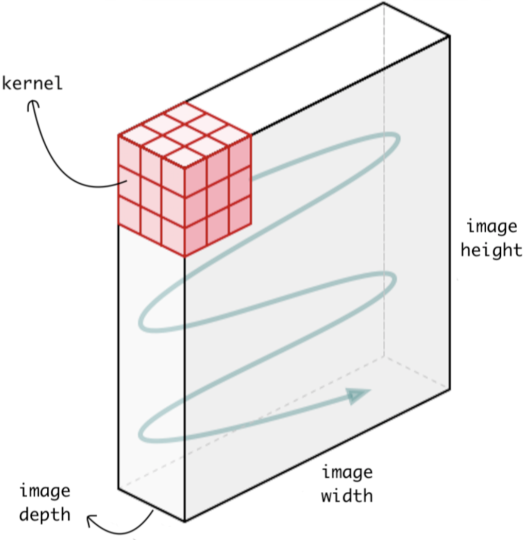


The kernel size, number of filters are hyperparameters once again.

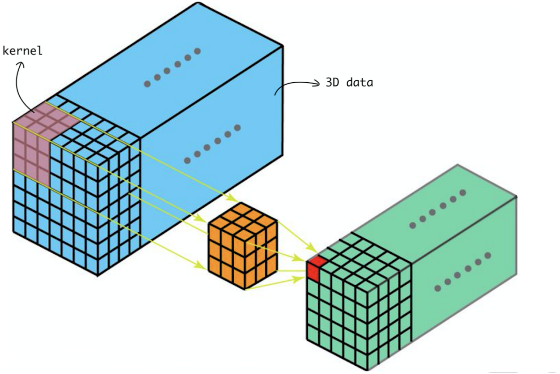
A MA filter $\simeq$ first layer of a 1D-CNN .. In Action !



A MA filter $\simeq$ first layer of a 2D-CNN

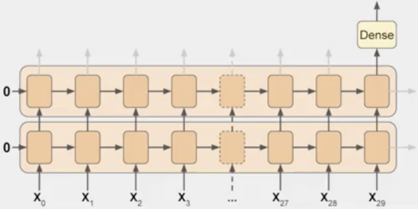


A MA filter $\simeq$ first layer of a 3D-CNN



A ARMA filter $\simeq$ first layer of a SimpleRNN

```
model = keras.models.Sequential([
  keras.layers.SimpleRNN(20, return_sequences=True,
                         input_shape=[None, 1]),
  keras.layers.SimpleRNN(20),
  keras.layers.Dense(1)
])
```



The problem : How to design a discrete time filter which corresponds to a fixed $|\hat{h}(e^{i\omega})|^2$ (we do not care about the phase) ?

Theorem If a positive valued function $H(e^{i\omega})$ satisfies

- H is a rationnal fraction in $e^{i\omega}$ with real coefficients, of the form

$$H(e^{i\omega}) = \frac{P(e^{i\omega})}{Q(e^{i\omega})}$$

  with $\delta P < \delta Q$

- The poles of $H(Z)$ (i.e., the roots of $Q(Z)$) are such that $|Z| < 1 >$

Then we can design a discrete time ARMA filter corresponding to a (stable and causal) filter $h[n]$ such that

$$|\hat{h}(e^{i\omega})|^2 = H(e^{i\omega})$$

The simplest ("interesting") ARMA filter is an AR(2) :

$$\hat{h}(Z) = \frac{b_0}{1 + a_1 Z^{-1} + a_2 Z^{-2}}$$

Thus There are two poles $\rho e^{i\omega_0}$ and $\rho e^{-i\omega_0}$

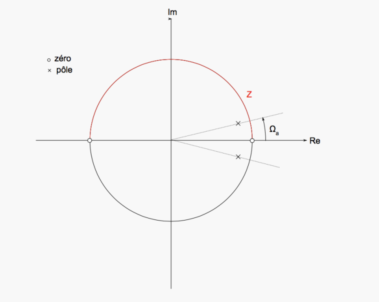
$$\longrightarrow$$

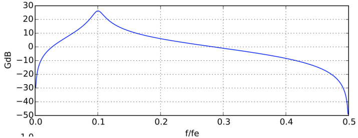
this is an IIR band-pass filter !

- ω_0 the resonnance frequency
- ρ allows to tune the band-width
- b_0 is the amplittude

A classical bandwidth filter AR(2) MA(2)

$$\hat{h}(Z) = \frac{(Z - 1(Z+1))}{1 + a_1 Z^{-1} + a_2 Z^{-2}}$$

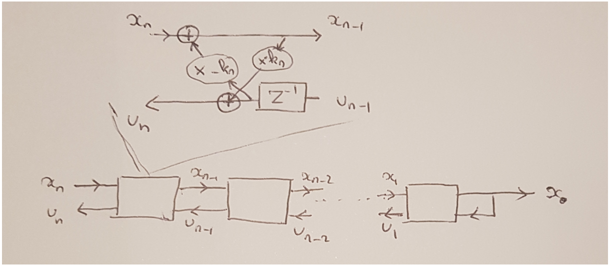




$$\hat{h}(Z) = \frac{b_0}{1 + a_1 Z^{-1} + a_2 Z^{-2}}$$

$\Longrightarrow$ Interpolation of parameters is possible keeping stability !

Percussive sound synthesis



- Input : x_N
- Output : x_0

$$
\begin{aligned}
\hat{x}_{n-1}(Z) &= -k_n Z^{-1} \hat{u}_{n-1}(Z) + \hat{x}_n(Z) \\
\hat{u}_n(Z) &= k_n \hat{x}_{n-1}(Z) + Z^{-1} \hat{u}_{n-1}(Z)
\end{aligned}
$$