Audio Signal Processing : II. Analogic signal/Digital Signal

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

Definition : For $s(t) \in L^1$

$$\hat{s}(\omega) = \int s(t) e^{-i\omega t} dt$$

Inversion Theorem :

$$s(t) = \frac{1}{2\pi} \int \hat{s}(\omega) e^{i\omega t} d\omega$$

Fourier transform + inversion : For $s(t) \in L^1$

$$s(t) \in L^1, \quad \hat{s}(\omega) = \int s(t) e^{-i\omega t} dt, \quad s(t) = \frac{1}{2\pi} \int \hat{s}(\omega) e^{i\omega t} d\omega$$

Remarks

1. $\hat{s}(\omega) in C^0$
2. $L^1 \cap L^2$ is dense in L^1 $\rightarrow$ extension to L^2

Fourier transform + inversion : For $s(t) \in L^2$

$$s(t) \in L^2, \quad \hat{s}(\omega) = \int s(t) e^{-i\omega t} dt, \quad s(t) = \frac{1}{2\pi} \int \hat{s}(\omega) e^{i\omega t} d\omega$$

Remarks (continued)

3. We could write symbolically (this is not rigorous since $e^{i\omega t} \notin L^2$)

$$\hat{s}(\omega) = < s, e^{i\omega t} >$$

And then write

$$s(t) = \frac{1}{2\pi} \int < s, e^{i\omega t} > e^{i\omega t} d\omega$$

Fourier transform + inversion : For $s(t) \in L^2$

$$s(t) \in L^2, \quad \hat{s}(\omega) = \int s(t) e^{-i\omega t} dt, \quad s(t) = \frac{1}{2\pi} \int \hat{s}(\omega) e^{i\omega t} d\omega$$

Remarks (continued)

4. $s(t)$ is real $\Longrightarrow \hat{s}(-\omega) = \hat{s}^*(\omega)$
   (no information in negative frequency)
5. The inverse Fourier transform can be seen as a decomposition on sums of sinusoids with
   - frequency $\omega/2\pi$
   - phase $arg(\hat{s}(\omega))$
   - amplitude $|\hat{s}(\omega)|$

Fourier transform + inversion : For $s(t) \in L^2$

$$s(t) \in L^2, \quad \hat{s}(\omega) = \int s(t) e^{-i\omega t} dt, \quad s(t) = \frac{1}{2\pi} \int \hat{s}(\omega) e^{i\omega t} d\omega$$

Remarks (continued)

6. Amplitude modulation : a multiplication of the signal by $e^{i\omega_0 t}$ leads to a translation by ω_0 (towards the "right") of its Fourier transform.

$$\widehat{s(t)e^{i\omega_0 t}}(\omega) = \hat{s}(\omega - \omega_0)$$

Fourier transform + inversion : For $s(t) \in L^2$

$$s(t) \in L^2, \quad \hat{s}(\omega) = \int s(t) e^{-i\omega t} dt, \quad s(t) = \frac{1}{2\pi} \int \hat{s}(\omega) e^{i\omega t} d\omega$$

Remarks (continued)

7. Derivation enhances high frequencies

$$\widehat{s'(t)}(\omega) = i\omega \hat{s}(\omega)$$

Consequently

$$\widehat{s^{(p)}(t)}(\omega) = (i\omega)^p \hat{s}(\omega)$$

Fourier transform + inversion : For $s(t) \in L^2$

$$s(t) \in L^2, \quad \hat{s}(\omega) = \int s(t) e^{-i\omega t} dt, \quad s(t) = \frac{1}{2\pi} \int \hat{s}(\omega) e^{i\omega t} d\omega$$

Remarks (continued)

8. Since $\widehat{s^{(p)}(t)}(\omega) = (i\omega)^p \hat{s}(\omega)$, there is a strong link between regularity of $s(t)$ and its energy at high frequencies.
   Actually
   $$|\hat{s}(\omega)| < \frac{K}{\omega^p + 1 + \epsilon} \Longrightarrow s \in C^p$$
9. Moreover
   $\hat{s}(\omega)$ has a compact support $\longrightarrow s \in C^\infty$

Looking for "simple" sound transformation L:

- L : linear operator
- L : translation invariant, i.e., $L(s(.-t_0))(t) = L(s(.))(t-t_0)$

We can write

$$s(t) = \int s(u)\delta(t-u)$$

Thus

$$L(s)(t) = \int s(u)L(\delta)(t-u)du$$

Setting $h = L(\delta)$ the impulse response of L, we get

$$L(s)(t) = \int s(u)h(t-u)du = s \star h(t)$$

$\Longrightarrow$ That leads to convolution operators

$$s \star h = \int s(u)h(t-u)du = \int s(t-u)h(u)du$$

Three very important properties of convolution

$$L(s) = s \star h = \int s(u)h(t-u)du$$

- **Causality** : $h(u) = 0$ for $u < 0$
- **Stability** (i.e., s bounded $\Longrightarrow s \star h$ bounded) : $h \in L^1$
- for all ω, the function of t : $e^{i\omega t}$ is an **eigen vector** of the convolution operator associated to the eigen value $\hat{h}(\omega)$

$$L(e^{i\omega t}) = \int e^{i\omega(t-u)}h(u)du = e^{i\omega t}\int e^{-i\omega u}h(u)du = \hat{h}(\omega)e^{i\omega t}$$

Thus since $s(t) = \frac{1}{2\pi}\int \hat{s}(\omega)e^{i\omega t}dt$, we get

$$\begin{aligned} L(s)(t) &= s \star h = \frac{1}{2\pi}\int \hat{s}(\omega)L(e^{i\omega t})dt \\ &= \frac{1}{2\pi}\int \hat{s}(\omega)\hat{h}(\omega)e^{i\omega t}dt \end{aligned}$$

Since $L(s)(t) = \frac{1}{2\pi}\int \hat{L(s)}(\omega)e^{i\omega t}dt$,

by identification, one gets the **convolution theorem**

$$\widehat{s \star h}(\omega) = \hat{s}(\omega)\hat{h}(\omega)$$

$$\widehat{s \star h}(\omega) = \hat{s}(\omega)\hat{h}(\omega)$$

Thus a convolution can be seen as a **filtering** process (more on that later)

Three "classic" filter categories

- low-pass filter (ex : $h_{\omega_0}(t) = \sin(\omega_0 t)/\pi t$)
- band-pass filter
- high-pass filter

In order to be able to manipulate an audio signal on a computer, we need to **sample** it

$$\{s(t)\}_t \longrightarrow \{s(nT)\}_n,$$

where

- T is the sampling period
- $F_s = \frac{2\pi}{T}$ is the **sampling frequency**

"No loss" $\Longrightarrow$ we want to be able to go back $s(nT) \longrightarrow s(t)$

Any intuition ?

$$
\begin{aligned}
s(nT) &= \frac{1}{2\pi} \int \hat{s}(\omega) e^{i\omega nT} d\omega \\
&= \frac{1}{2\pi} \sum_k \int_{\frac{2\pi k}{T}}^{\frac{2\pi(k+1)}{T}} \hat{s}(\omega) e^{i\omega nT} d\omega \\
&= \frac{1}{2\pi} \sum_k \int_0^{\frac{2\pi}{T}} \hat{s}(\omega + \frac{2\pi k}{T}) e^{i\omega nT} d\omega \\
&= \frac{1}{2\pi} \int_0^{\frac{2\pi}{T}} \sum_k \left(\hat{s}(\omega + \frac{2\pi k}{T}) \right) e^{i\omega nT} d\omega
\end{aligned}
$$

We set the function $\tilde{\hat{s}}(\omega) = \sum_k \left(\hat{s}(\omega + \frac{2\pi k}{T}) \right)$, we have

$$
\tilde{\hat{s}}(\omega) = T \sum_n s(nT) e^{-i\omega nT}
$$

How do we go back from $\tilde{\hat{s}}(\omega)$ to $s(t)$?

$$\tilde{\hat{s}}(\omega) = T \sum_n s(nT) e^{-i\omega nT}$$

How do we go back from $\tilde{\hat{s}}(\omega)$ to $s(t)$?

The simplest case is to suppose that s is supported by $]-\pi/T, \pi/T[=]-F_s/2, F_s/2[$

If it is not the case : aliasing

$$\begin{aligned}
\hat{s}(\omega) &= 1_{]-\frac{\pi}{2},\frac{\pi}{2}[}(\omega)\tilde{\hat{s}}(\omega) \\
&= T 1_{]-\frac{\pi}{2},\frac{\pi}{2}[}(\omega) \sum_n s(nT) e^{-i\omega nT} \\
&= T \sum_n s(nT) 1_{]-\frac{\pi}{2},\frac{\pi}{2}[} e^{-i\omega nT}
\end{aligned}$$

Thus

$$s(t) = T \sum_n s(nT) h_{\frac{\pi}{T}}(t - nT) = \sum_n s(nT) sinc\left(\frac{\pi}{T}(t - nT)\right)$$

The **Shannon theorem**

If the support of $\hat{s}(\omega)$ is included in $]-F_s/2, F_s/2[$ then the "go back" is possible through

$$s(t) = T \sum_n s(nT) h_{\frac{\pi}{T}}(t - nT)$$

Remarks :

- Discretization $\longleftrightarrow$ Periodization
- Preprocessing low pass filter : Beware aliasing (which filter ?)
- Which sampling frequency ?

Framework

$s[n]$ is a real-valued discrete time audio signal.

$s[n]$ can be seen as a "continuous-time" signal (isomorphism)

$$\{s[n]\}_n \longleftrightarrow f(t) = \sum_n s[n]\delta(t-n)$$

Since

$$\hat{f}(\omega) = \sum_n s[n]e^{-in\omega}$$

(a 2π-periodic function), that leads to the "natural" definition

Fourier Transform of a discrete-time signal

$$\hat{s}(e^{i\omega}) = \sum_n s[n]e^{-in\omega}$$

Again : **Looking for "simple" sound transformation L:**

- L : linear operator
- L : translation invariant, i.e., $L(s[. - n_0])[n] = L(s[.])[n - n_0]$

$\Longrightarrow$ That leads to convolution operators :

$$L(s)[n] = s \star h[n] = \sum_k s[k]h[n-k] = \sum_k s[n-k]h[k]$$

where h is the impulsional response of L

Two very important properties of convolution

$$L(s) = s \star h[n] = \sum_k s[k]h[n-k]$$

- **Causality** : $h[n] = 0$ for $n < 0$
- **Stability** (i.e., s bounded $\longrightarrow s \star h$ bounded) : $h \in l^1$

Definition of the Z-transform

If $s[n]$ is a time-discrete signal, its Z-transform is a function of a complex variable (Z) defined by

$$\hat{S}(Z) = \sum_n s[n] Z^{-n}$$

Remarks

- "Equivalent" to the Laplace transform for time-continuous functions
- **The convolution theorem reads :**

$$\widehat{S \star H}(Z) = \hat{S}(Z)\hat{H}(Z)$$

- We get $\hat{s}(e^{i\omega}) = \hat{S}(Z)$ and $s[n] = \frac{1}{2\pi}\int_0^{2\pi} \hat{s}(e^{i\omega}) e^{in\omega} d\omega$
- $\Longrightarrow$ **Filtering**

- What does a low-pass filter look like ?
- What does a band-pass filter look like ?
- What does a high-pass filter look like ?

Let's discuss some filtering examples

1. The Z^{-1} operator
2. The $1 + Z^{-1}$ operator
3. The $1 - Z^{-1}$ operator
4. The $\frac{1}{1-Z^{-1}}$ operator

In order to be able to manipulate the Fourier transform of an audio digital signal on a computer, we need to **sample** its Fourier transform on $[0, 2\pi]$

$$\{s(e^{i\omega})\}_\omega \longrightarrow \{s(e^{i\frac{2\pi k}{N}})\}_{0\leq k<N}$$

where

- We sample using N frequencies $\{\omega_k = \frac{2\pi k}{N}\}_{0\leq k<N}$

"No loss" $\Longrightarrow$ we want to be able to go back :

$$\{s(e^{i\frac{2\pi k}{N}})\}_{0\leq k<N} \longrightarrow \{s(e^{i\omega})\}_\omega$$

Discretization of the frequency space $\Longrightarrow$ Periodisation of the time space

The theorem (equivalent to Shannon theorem) :

If the support of $s[n]$ is included in $[0, N[$ (or equivalently N-periodic) then the "go back" is possible through

$$s(e^{i\omega}) = \frac{1}{N} \sum_{k=0}^{N-1} \hat{s}(e^{i\frac{2\pi k}{N}}) \hat{h}_N(\omega - \frac{2\pi k}{N})$$

with $h_N[n] = 1_{[0,N[}[n]$ and

$$\hat{h}_N(e^{i\omega}) = \frac{\sin(\frac{N\omega}{2})}{\sin(\frac{\omega}{2})} e^{-i\frac{(N-1)\omega}{2}}$$

Framework

$s[n]$ is a real-valued disrete-time audio signal of finite support size N (or alternatively, N-periodic)

$s[n]$ a real-valued signal with support $[0, N]$. We just apply the Fourier transform formula for discrete-time signals :

$$s(e^{i\omega}) = \sum_{n=0}^{N-1} s[n] e^{-in\omega}$$

and we sample it using the previous frequency sampling $\omega_k = \frac{2\pi k}{N}$

$$s(e^{i\omega_k}) = \sum_{n=0}^{N-1} s[n] e^{-i\frac{2\pi n}{k}}$$

The Discrete Fourier Transform (Definition)

$$s[k] = \sum_{n=0}^{N-1} s[n] e^{-i\frac{2\pi n}{k}}$$

The Discrete Fourier Transform (Definition)

$$\hat{s}[k] = \sum_{n=0}^{N-1} s[n] e^{-i\frac{2\pi n}{k}}$$

The Inverse of the Discrete Fourier Transform

$$s[n] = \frac{1}{N} \sum_{k=0}^{N-1} \hat{s}[k] e^{i\frac{2\pi n}{k}}$$

The FFT(W) : a fast algorithm in $O(N \log_2 N)$

We want to find the linear transformations (i.e., the "convolutions") that are

- invariant by time translation
  ($\Rightarrow$ No way if s has a finite support !)
- that satisfies the convolution theorem ($\widehat{s?h}[k] = \hat{s}[k]\hat{h}[k]$)

The right framework : $s[n]$ and $h[n]$ are N-periodic

$s[n]$ and $h[n]$ are N-periodic

The circular convolution (Definition)

$$L(s) = s \circledast h[n] = \sum_{k=0}^{N-1} s[k]h[n-k]$$

$s[n]$ and $h[n]$ are N-periodic
The circular convolution (Definition)

$$L(s) = s \circledast h[n] = \sum_{k=0}^{N-1} s[k]h[n-k]$$

for all k, the function $n \rightarrow e^{i2\pi kn/N}$ is an **eigen vector** of the convolution operator associated to the eigen value $\hat{h}[k]$

$$L(e^{i2\pi kn/N}) = \hat{h}[k]e^{i2\pi kn/N}$$

$\Longrightarrow$ **The convolution Theorem**

$$\widehat{s \circledast h}[k] = \hat{s}[k]\hat{h}[k]$$

$s[n]$ and $h[n]$ are N-periodic

The circular convolution (Definition)

$$L(s) = s \circledast h[n] = \sum_{k=0} N - 1 s[k]h[n-k]$$

The convolution Theorem

$$\widehat{s \circledast h}[k] = \hat{s}[k]\hat{h}[k]$$

What the hell are we going to do with that ?

The framework

- $s[n]$ supported by $[0, N[$
- $h[n]$ supported by $[0, N[$

The problem Can I design a fast (i.e., $O(N \log_2 N)$) algorithm to compute

$$s \star h[n] = \sum_k s[k]h[n-k]$$

?

The framework

- $s[n]$ supported by $[0, N[$
- $h[n]$ supported by $[0, N[$

$\Longrightarrow s \star h[n]$ is supported by $[0, 2N[$

New framework

- We define $\tilde{s}[n]$ a $2N$-periodic signal such that

$$
\begin{aligned}
\tilde{s}[n] &= s[n] \quad , \forall n \in [0, N[\\
\tilde{s}[n] &= 0 \quad , \forall n \in [N, 2N[
\end{aligned}
$$

- We define in the same way $\tilde{h}[n]$ a $2N$-periodic signal

New framework

- We define $\tilde{s}[n]$ a $2N$-periodic signal such that

$$\begin{aligned} \tilde{s}[n] &= s[n] \quad, \forall n \in [0, N[\\ \tilde{s}[n] &= 0 \quad, \forall n \in [N, 2N[\end{aligned}$$

- We define in the same way $\tilde{h}[n]$ a $2N$-periodic signal

Theorem

$$s \star h[n] = \tilde{s} \circledast \tilde{h}[n], \quad \forall n \in [0, 2N[$$

We did it ! The complexity is the one of the FFT $O(N \log_2 N)$

But often $s[n]$ (the audio signal) has a larger size (N) than the size (M) of $h[n]$ (the filter)

In that case, can't we do better than $O(N \log_2 N)$?

The framework

- $s[n]$ supported by $[0, N[$
- $h[n]$ supported by $[0, M[$ with $M << N$

We can write

$$s \star h[n] = \sum_{i=0}^{N/M} s_i \star h[n]$$

where $s_i[n] = s[n]1_{[iM,(i+1)M]}$

That corresponds to a complexity of $O(N \log_2 M)$

The principle

$s[n]$ is a continuous value $\longrightarrow$ we need to bound it and discretize it
In practice : Two steps

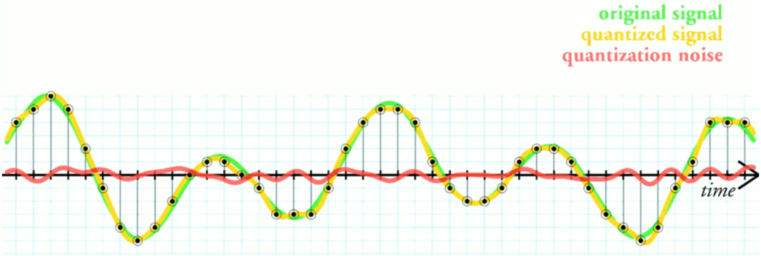
- clipping
- discretization

Discretization : two characteristics

- number of bits used for each value (8,12,16,24, ...)
- the quantification type
  - uniform (linear) $\Rightarrow$ pb in low-amplitude zones if there are high-amplitude zones
  - log

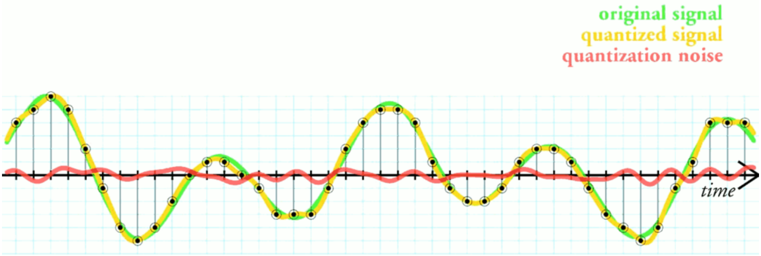
$\Longrightarrow$ Quantization noise

- $s[n]$: the original signal
- $\tilde{s}[n]$: the quantized signal
- Quantization noise : $s[n] - \tilde{s}[n]$



Ouups, it looks like we have a problem here ?

- $s[n]$: the original signal
- $\tilde{s}[n]$: the quantized signal
- Quantization noise : $s[n] - \tilde{s}[n]$



In low-amplitude zones the quantization noise is highly correlated to the signal itself. This is a real problem !

In low-amplitude zones the quantization noise is highly correlated to the signal itself. This is a real problem !

Ideally : we would like the quantization noise to be independant of the audio signal, e.g., we would like the quantization noise to be a white noise (with a density that does not depend on the density of the signal itself)

The framework

- We consider that the audio signal is a stationnary stochastic process $X[n]$
- Q is a linear quantifier of step q

$$Q : [iq - \frac{q}{2}, iq + \frac{q}{2}] \to x - iq, \quad \forall i$$

- The quantization errror

$$\tilde{X}[n] = X[n] - Q(X)$$

The (first-order) problem

What preprocessing can we make to $X[n]$ so that the density of $\tilde{X}[n]$ is always uniform (i.e., does not depend on the one of $X[n]$) ?

Let $p_X(x)$ the density of the law of $X[n]$ and $p_{\tilde{X}}(\tilde{x})$ the one of $\tilde{X}$
Since

$$Prob(\tilde{X} = \tilde{x}) = 1_{]-\frac{q}{2},\frac{q}{2}[}(\tilde{x}) \sum_k Prob(X = \tilde{x} + kq)$$

one gets

$$p_{\tilde{X}}(\tilde{x}) = 1_{]-\frac{q}{2},\frac{q}{2}[}(\tilde{x}) \sum_k p_X(\tilde{x} + kq)$$

Thus we would like to have $\sum_k p_X(\tilde{x} + kq) = \frac{1}{q}$
Since $\sum_k p_X(\tilde{x} + kq) = p_X \star \sum_k \delta(\tilde{x} + kq)$, by Fourier transform we get (using the Poisson formula)

$$\hat{p}_X(\omega)\frac{2\pi}{q}\sum_k \delta(\omega + \frac{2\pi k}{q}) = \frac{2\pi}{q}\delta(\omega)$$

Theorem

The density of $\tilde{X}[n]$ is uniform (on $[-q/2, q/2[$) iff

$$\hat{p}_X\left(\frac{2\pi k}{q}\right) = 0, \quad \forall k \neq 0$$

Theorem

The density of $(\tilde{X}[n], \tilde{X}[n+m])$ is uniform (on $[-q/2, q/2[$) iff

$$\hat{p}_{X[n],X[n+m]}\left(\frac{2\pi k}{q}, \frac{2\pi l}{q}\right) = 0, \quad \forall (k,l) \neq (0,0)$$

What preprocessing should we make one $X[n]$?

Theorem

The density of $\tilde{X}[n]$ is uniform (on $[-q/2, q/2[$) iff

$$\hat{p}_X\left(\frac{2\pi k}{q}\right) = 0, \quad \forall k \neq 0$$

Preprocessing : Dithering

$$X[n] \longrightarrow X[n] + W[n]$$

where $W[n]$ is a uniform white noise.

Preprocessing : Dithering

$$X[n] \longrightarrow X[n] + W[n]$$

where $W[n]$ is a uniform white noise.

Did we really solve our problem ?

Nope : $X + W - Q(X + W)$ is white but not $X - Q(X + W)$

$\Longrightarrow$ What we did corresponds to substractive dithering
$(X - (Q(X + W) - W)$ is white)

Solution ?

- Oversampling technique
- $\sigma - \Delta$ technique
- ...

In practice ?