Basics on Digital Signal Processing

Introduction

Vassilis Anastassopoulos
Electronics Laboratory, Physics Department, University of Patras
Outline of the Course

1. Introduction (sampling – quantization)
2. Signals and Systems
3. Z-Transform
4. The Discreet and the Fast Fourier Transform
5. Linear Filter Design
6. Noise
7. Median Filters
Analog & digital signals

**Analog**

*Continuous function* $V$ of *continuous* variable $t$ (time, space etc) : $V(t)$.

**Digital**

*Discrete function* $V_k$ of *discrete* sampling variable $t_k$, with $k =$ integer: $V_k = V(t_k)$.

Uniform (periodic) sampling. Sampling frequency $f_S = 1/ t_S$
Analog & digital systems

Digital lowpass filter

Analog lowpass filter

1. Discrete-time
2. Difference equation
   \[ y(n) = b y(n-1) + x(n) \]
3. \( z \)-plane (\( Z \)-transform used for analysis)

1. Continuous-time
2. Differential equation
   \[ v(t) = RC \frac{dv_0(t)}{dt} + v_0(t) \]
3. \( s \)-plane (Laplace transform used for analysis)
Digital vs analog processing

Digital Signal Processing (DSPing)

Advantages

• More flexible.
• Often easier system upgrade.
• Data easily stored -memory.
• Better control over accuracy requirements.
• Reproducibility.
• Linear phase
• No drift with time and temperature

Limitations

• A/D & signal processors speed: wide-band signals still difficult to treat (real-time systems).
• Finite word-length effect.
DSPing: aim & tools

Applications
- Predicting a system’s output.
- Implementing a certain processing task.
- Studying a certain signal.

Hardware
- General purpose processors (GPP), \(\mu\)-controllers.
- Digital Signal Processors (DSP).
- Programmable logic (PLD, FPGA).

Software
- Programming languages: Pascal, C / C++ ...
- "High level" languages: Matlab, Mathcad, Mathematica...
- Dedicated tools (ex: filter design s/w packages).

Fast
Faster
real-time DSPing
Related areas

Digital Signal Processing

Communication Theory
Numerical Analysis
Probability and Statistics
Analog Signal Processing

Analog Electronics  Digital Electronics  Decision Theory
Applications

- Space
  - Space photograph enhancement
  - Data compression
  - Intelligent sensory analysis by remote space probes

- Medical
  - Diagnostic imaging (CT, MRI, ultrasound, and others)
  - Electrocardiogram analysis
  - Medical image storage/retrieval

- Commercial
  - Image and sound compression for multimedia presentation
  - Movie special effects
  - Video conference calling

- Telephone
  - Voice and data compression
  - Echo reduction
  - Signal multiplexing
  - Filtering

- Military
  - Radar
  - Sonar
  - Ordnance guidance
  - Secure communication

- Industrial
  - Oil and mineral prospecting
  - Process monitoring & control
  - Nondestructive testing
  - CAD and design tools

- Scientific
  - Earthquake recording & analysis
  - Data acquisition
  - Spectral analysis
  - Simulation and modeling
Important digital signals

Unit Impulse or Unit Sample.
The most important signal for two reasons
\[ \delta(n) = 1 \text{ for } n = 0 \]
\[ \delta(n) = u(n) - u(n-1) \]

Unit Step
\[ u(n) = 1 \text{ for } n \geq 0 \]
\[ \delta(n) = u(n) - u(n-1) \]

Unit Ramp
\[ r(n) = nu(n) \]
General scheme

Sometimes steps missing
- Filter + A/D
  (ex: economics);
- D/A + filter
  (ex: digital output wanted).

Topics of this lecture.
Digital system implementation

KEY DECISION POINTS:
Analysis bandwidth, Dynamic range

1. Pass / stop bands.
2. Sampling rate.
3. No. of bits. Parameters.
4. Digital format.

What to use for processing?
AD/DA Conversion – General Scheme

antialias filter

Analog Filter → ADC → Digital Processing → DAC → Analog Filter

Filtered Analog Input → Digitized Input → Digitized Output → S/H Analog Output → Analog Output

reconstruction filter
AD Conversion - Details

- **Input filter**
- **ADC with sample and hold**
- **Digital processor**
- **DAC**
- **Output filter**

**Blocks:**
- **Lowpass filter**
- **Sample and hold**
- **Quantizer**
- **Encoder**

**Signal Flow:**
- $x(t)$ to input filter
- Output of input filter to ADC with sample and hold
- Output of ADC with sample and hold to digital processor
- Output of digital processor to DAC
- Output of DAC to output filter

**Analog to Digital Conversion:**
- $x(t)$ to analogue input
- Output of analogue input to lowpass filter
- Output of lowpass filter to sample and hold
- Output of sample and hold to quantizer
- Output of quantizer to encoder
- Output of encoder to digital output code

**Symbol:**
- $F_s$
Sampling

a. Analog frequency = 0.0 (i.e., DC)

b. Analog frequency = 0.09 of sampling rate

c. Analog frequency = 0.31 of sampling rate

d. Analog frequency = 0.95 of sampling rate
Sampling

How fast must we sample a continuous signal to preserve its info content?

Ex: train wheels in a movie.

25 frames (=samples) per second.

Train starts ➔ wheels ‘go’ clockwise.

Train accelerates ➔ wheels ‘go’ counter-clockwise.

Why?

Frequency misidentification due to low sampling frequency.
How fast do we have to instantly stare at the disk if it rotates with frequency 0.5 Hz?
**The sampling theorem**

A signal \( s(t) \) with maximum frequency \( f_{\text{MAX}} \) can be recovered if sampled at frequency \( f_s > 2 f_{\text{MAX}} \).

* Multiple proposers: Whittaker(s), Nyquist, Shannon, Kotel’nikov.

Naming gets confusing! Nyquist frequency (rate) \( f_N = 2 f_{\text{MAX}} \) or \( f_{\text{MAX}} \) or \( f_{\text{S,MIN}} \) or \( f_{\text{S,MIN}}/2 \)

---

**Example**

\[
s(t) = 3 \cdot \cos(50\pi t) + 10 \cdot \sin(300\pi t) - \cos(100\pi t)
\]

\( F_1 = 25 \text{ Hz}, F_2 = 150 \text{ Hz}, F_3 = 50 \text{ Hz} \)

Condition on \( f_s \)?

\( f_s > 300 \text{ Hz} \)
Sampling and Spectrum
1 Sampling low-pass signals

(a) Band-limited signal: frequencies in \([-B, B]\) \((f_{\text{MAX}} = B)\).

(b) Time sampling \(\rightarrow\) frequency repetition.
   \(f_S > 2B \rightarrow\) no aliasing.

(c) \(f_S \leq 2B \rightarrow\) aliasing!
   
   Aliasing: signal ambiguity in frequency domain.
Antialiasing filter

(a), (b) Out-of-band noise can alias into band of interest. Filter it before!

(c) Antialiasing filter

Passband: depends on bandwidth of interest.

Attenuation $A_{MIN}$: depends on
- ADC resolution (number of bits $N$).
  \[ A_{MIN}, \text{dB} \approx 6.02 N + 1.76 \]
- Out-of-band noise magnitude.

Other parameters: ripple, stopband frequency...
Under-sampling

Using spectral replications to reduce sampling frequency $f_S$ req’ments.

$$\frac{2 \cdot f_C + B}{m + 1} \leq f_S \leq \frac{2 \cdot f_C - B}{m}$$

$m \in \bigcap$, selected so that $f_S > 2B$

Example

$f_C = 20$ MHz, $B = 5$MHz

Without under-sampling $f_S > 40$ MHz.

With under-sampling $f_S = 22.5$ MHz (m=1);

= 17.5 MHz (m=2); = 11.66 MHz (m=3).

Advantages

➢ Slower ADCs / electronics needed.

➢ Simpler antialiasing filters.
Quantization and Coding

N Quantization Levels

Quantization Noise
SNR of ideal ADC

\[ \text{SNR}_{\text{ideal}} = 20 \cdot \log_{10} \left( \frac{\text{RMS(input)}}{\text{RMS(e}_q)} \right) \] (1)

Also called SQNR
(signal-to-quantisation-noise ratio)

\[ \text{RMS(input)} = \sqrt{\frac{1}{T} \int_{0}^{T} \left( \frac{V_{\text{FSR}}}{2} \cdot \sin(\omega t) \right)^2 dt} = \frac{V_{\text{FSR}}}{2\sqrt{2}} \]

\[ \text{RMS(e}_q) = \sqrt{\int_{-q/2}^{q/2} e_q^2 \cdot p(e_q) de_q} = \frac{q}{\sqrt{12}} = \frac{V_{\text{FSR}}}{2^N \cdot \sqrt{12}} \]

(sampling frequency \( f_S = 2 f_{\text{MAX}} \))

Assumptions

- Ideal ADC: only quantisation error \( e_q \) (\( p(e) \) constant, no stuck bits…)
- \( e_q \) uncorrelated with signal.
- ADC performance constant in time.

Input(t) = \( \frac{1}{2} V_{\text{FSR}} \sin(\omega t) \).

\( p(e) \)
quantisation error probability density

\( e_q \)
Error value

\( \frac{1}{q} \)

Error value

\( -\frac{q}{2} \)

\( \frac{q}{2} \)
SNR of ideal ADC

Substituting in (1):

\[ \text{SNR}_{\text{ideal}} = 6.02 \cdot N + 1.76 \text{[dB]} \]  

One additional bit \( \Rightarrow \) SNR increased by 6 dB

Real SNR lower because:
- Real signals have noise.
- Forcing input to full scale unwise.
- Real ADCs have additional noise (aperture jitter, non-linearities etc).

Actually (2) needs correction factor depending on ratio between sampling freq & Nyquist freq. Processing gain due to oversampling.
Coding - Conventional
Coding – Flash AD

![Flash AD Diagram](image-url)
DAC process

![Diagram showing the DAC process](image-url)
The oversampling process takes apart the images of the signal band.

When the sampling rate increases (4 times) the quantization noise spreads over a larger region. The quantization noise power in the signal band is 4 times smaller.

Spectrum at the output of a noise shaping quantizer loop compared to those obtained from Nyquist and Oversampling converters.
A discreet-time system is a device or algorithm that operates on an input sequence according to some computational procedure.

It may be:
- A general purpose computer
- A microprocessor
- Dedicated hardware
- A combination of all these
System Properties
- linear
- Time Invariant
- Stable
- Causal

\[ y(n) = \sum_{k=0}^{N} a_k x(n - k) \]  

Convolution
Linear Systems - Convolution

\[ x(n) \quad h(n) \quad y(n) \]

\[ 5 + 7 - 1 = 11 \text{ terms} \]
Linear Systems - Convolution

5+7-1=11 terms
General Linear Structure

\[ y(n) = \sum_{k=0}^{M} a_k x(n-k) - \sum_{k=1}^{L} b_k y(n-k) \]
Simple Examples

\[ x(n) \xrightarrow{T_s} x(n-1) \xrightarrow{T_s} x(n-2) \xrightarrow{T_s} x(n-3) \xrightarrow{T_s} x(n-4) \]

\[ 0.2 \quad 0.2 \quad 0.2 \quad 0.2 \]

\[ \alpha. \]

\[ y(n-1) \xrightarrow{T_s} y(n) = \]

\[ \beta. \]

\[ x(n) \xrightarrow{+} y(n) = \]

\[ \gamma. \]

\[ x(n) \xrightarrow{T_s} x(n-1) \]

\[ y(n) = \]

\[ \]
**Linearity – Superposition – Frequency Preservation**

**Principle of Superposition**

\[ x_1(n) \] \[ -H\] \[ \rightarrow \] \[ y_1(n) \]

\[ x_2(n) \] \[ -H\] \[ \rightarrow \] \[ y_2(n) \]

\[ ax_1(n) + bx_2(n) \] \[ -H\] \[ \rightarrow \] \[ ay_1(n) + by_2(n) \]

**Principle of Superposition \(\Rightarrow\) Frequency Preservation**

\[ x_1(n) \] \[ -x^2\] \[ \rightarrow \] \[ x_1^2(n) \]

\[ x_2(n) \] \[ -x^2\] \[ \rightarrow \] \[ x_2^2(n) \]

\[ x_1(n) + x_2(n) \] \[ -x^2\] \[ \rightarrow \] \[ x_1^2(n) + x_2^2(n) + 2x_1(n)x_2(n) \]

**Non-linear**

If \( y(n) = x^2(n) \) then for \( x(n) = \sin(n\omega) \) \( y(n) = \sin^2(n\omega) = 0.5 + 0.5\cos(2n\omega) \)
The END

Have a nice Weekend

Back on Tuesday