Signals and systems
M3070 - E305A

M3070: Applied mathematics and tools
E3050: Signaux, systèmes et télécommunications

Franky De Bruyne

F.DeBruyne@ecam.be

ECAM
Institut supérieur industriel
Haute école Léonard de Vinci

2017

4. Laplace transform

5. Fourier frequency analysis

8. Z-transform

9. Discrete-time Fourier frequency analysis

1. Introduction

Practicalities

- Slides in English
- Course (and exams) in French
- Exams
- Typos
- Matlab

Reference

Luis F. Chaparro, “Signals and Systems using Matlab”, Academic Press, Elsevier, 2011.

The Scientist and Engineer’s Guide to Digital Signal Processing by Steven W. Smith, Ph.D.
http://www.dspguide.com/pdfbook.htm

Signal

Signal

A signal is a formal description of a phenomenon evolving over time or space. It is the information carrier emitted by a source and intended for a receiver; it is the vehicle of intelligence in systems.

Classification according to:

- **origin**: audio, video, speech, image, communication, geophysical, sonar, radar, medical and musical signals, etc.
- **dimension**: real (or complex) valued function of one or more real variables.
- **description type**: signals that can be modeled exactly by a mathematical formula are known as **deterministic** signals. Deterministic signals are not always adequate to model real-world situations. **Stochastic** (random) signals, on the other hand, cannot be described by a mathematical equation; they are modeled in probabilistic terms.

Analog and digital signals

- Most natural signals are **analog** in nature:
  - these signals are continuous functions of time and/or space.
  - Historically, before the advent of widespread digital technology, analog signal processing was the only method by which to manipulate a signal, e.g. telephone, loud speaker.
- Since that time, as computers and software became more advanced, digital signal processing has become the method of choice. **Analog to Digital Converters** (ADC) perform periodic **sampling** and **quantisation** of the input signal. The result is a sequence of **digital**[1] values that have been converted from a continuous-time and continuous-amplitude analog signal to a discrete-time and discrete-amplitude digital signal, i.e.
  - the signals are defined only at given discrete sampling instants
  - and only take a finite number of discrete values.

[1] The adjective "digital" derives from *digitus*, Latin for finger: it captures the idea that the signal is represented by a sequence of integer numbers.

Digital Signal Processing (DSP)

Objects: digital signals, i.e.

- sequences of integer numbers,
- digitized discrete-time signals sampled from a time continuous analog signal,
- variations of a physical quantity that provides information on the status of a physical system.

Operations:

- analysis: understand the information content of the signal,
- processing: modify this information.

Wikipedia

Digital Signal Processing (DSP) is the mathematical manipulation of an information signal to modify or improve it in some way.

It is characterized by the representation of discrete-time, discrete frequency, or other discrete domain signals by a sequence of numbers or symbols and the processing of these signals.

Historical background

Roots of DSP: 1960s and 1970s

- Digital computers/processors first becomes available.
- Fast Fourier Transform (FFT) algorithm and description how to implement it conveniently on a computer: "An algorithm for the machine calculation of complex Fourier series", Cooley & Tukey, 1965.
- Before this time, signal processing is analog, numerical algorithms are available but not recognised as being useful because of the lack of computational power.
- Computers are expensive during this era, and DSP is limited to only a few critical applications. Pioneering efforts are made in four key areas:
  - **radar and sonar**: national security of United States
  - **oil exploration**: financial motivation
  - **space exploration**: irreplaceable data
  - **medical imaging**: saving lives

Historical background

Personal Computer revolution: 1980s and 1990s

- Rather than being motivated by military and government needs, DSP is suddenly driven by the commercial marketplace.
- First commercial Digital Signal Processors (DSPs).
- DSP reaches the public in such products as: mobile telephones, compact disc players, and electronic voice mail.

Historical background

Towards embedded intelligence:

- The technologies underlying digital signal processing (ADC - DAC - computing and communications devices) have ever increasing capabilities and declining costs. A multitude of commercial objects have been equipped with them.
- The paradigm **central intelligence unit** connected to peripheral units is gradually abandoned in favour of **distributed embedded systems** with distributed intelligence resulting in networked systems of embedded computers whose functional components are nearly invisible to end users.
- Systems have the potential to alter radically the way in which people interact with their environment by linking a range of devices and sensors that will allow information to be collected, shared, and processed in unprecedented ways.
- Today, DSP is a basic skill needed by scientists and engineers in many fields.

Advantages of DSP

Digital systems have a number of key advantages over their analog counterparts:

- **simplicity**: difference or recursive equations
- **flexibility**: programmable and re-programmable
- **power**: more complex computations, e.g. nonlinear operations
- **cost and congestion**: ever decreasing cost and size
- **robustness to noise**: digital circuits are less affected by noise
- **precision and stability**: quantisation noise has to be taken into consideration, insensibility to temperature and system age
- **memory**: information storage is easy and less memory space is required

Advantages of DSP

DSP has produced revolutionary changes in a number of fields:

- **classical signal processing**: filtering, fast transforms (FFT, wavelets), signal analysers and generators
- **telecommunication**: modulation and demodulation, adaptive equalising, echo cancellation
- **image and speech processing**: compression, analysis, voice recognition and synthesis, pattern recognition
- **radar and sonar**: echo location, target identification
- **medical applications**: Digital Signal Processing for EEG[2] and Image processing for MRI[3]
- **control**: filtering, estimators, control algorithms, optimisation
- **asset management**: performance monitoring, condition monitoring

[2] Electro-EncephaloGraphy

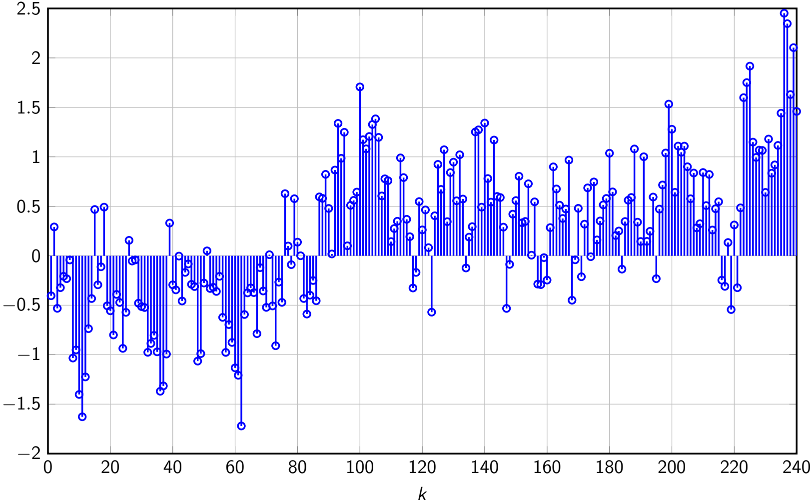
[3] Magnetic Resonance Imaging

Spectral analysis example[4]

- Suppose one would like to analyse sounds recorded in the ocean.
- A hydrophone is used to amplify and record the signal.
- An analog low pass filter is used to remove spectral components above 80Hz. The signal is sampled at 160Hz.
- The spectrum of the signal is subsequently analysed.

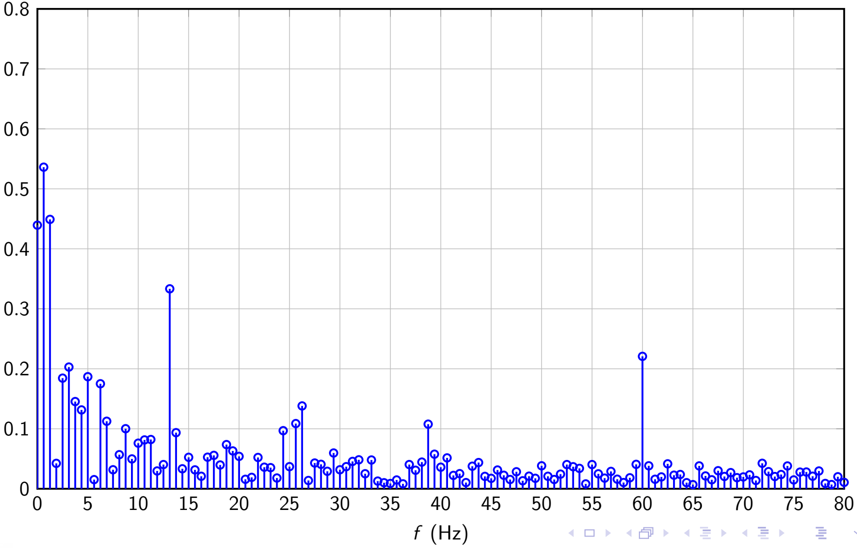
[4] GELE2511, Chapitre 7: Transformée de Fourier discrète, Gabriel Cormier, Université de Moncton.

Spectral analysis example: preprocessed signal



240 data points, i.e. 1.5 seconds of recording

Spectral analysis example: signal spectrum



Spectral analysis example

Let us analyse the spectral content:

- If the peaks are ignored, the spectrum is relatively constant between 20Hz and 70Hz; this is white Gaussian noise.
- White Gaussian noise draws its name from the flat spectrum over frequency.
- White Gaussian noise can be caused from many sources, e.g. microphone, ocean.

Spectral analysis example

Let us analyse the spectral content (continued):

- At low frequencies, the noise level increases rapidly in what appears to be a $\frac{1}{f^{\alpha}}$ relation: this is flicker noise or pink noise[5].
- Flicker noise seems ever present in physical systems (mechanical and electronic devices) and life science.
- Although many sources of noise are well understood, the origin of flicker noises remains, in general, a mystery in spite of their remarkably widespread occurrence in nature.

[5] bruit de scintillement

Spectral analysis example

Let us analyse the spectral content (continued): peaks

- 60Hz[6] peak: as most appliances and devices draw their power from the power grid, power line related noises are most prominent in our living environment.

En general, one can also observe:

- harmonics (150Hz, 250Hz, etc.) of the main power frequency,
- higher frequency pollutions emitted by switching power adapters (generally from 10kHz to 1MHz),
- frequencies related to mechanical devices, e.g. a faulty bearing will causes vibrations[7] that depend on the rotational speed and the bearing characteristics.

[6] 50Hz in Europe

[7] Vibration based condition monitoring

Spectral analysis example

Let us analyse the spectral content (continued): peaks

- An important peak can be distinguished around 13Hz with smaller peaks around 26Hz and 39Hz, i.e. second and third order harmonics of the first peak.
- The peaks could have been caused by a three blade propeller rotating at approximately 4.33 turns/second.
- This technique lies on the basis of passive sonar.

2. Continuous-time signals

Definition

Continuous signal

A continuous signal $x(.)$ can be thought of as a real or complex valued function of an independent variable.

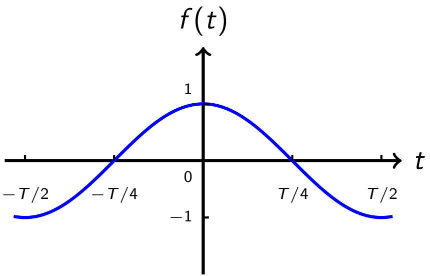
The independent variable is often time t, i.e. this results in a continuous-time signal $x(t)$.

The terms **continuous-time** and **analog** are used interchangeably for these signals.

Properties: parity

Even function

A function is **even** if $\forall t \in \mathbb{R}$: $f(-t) = f(t)$.



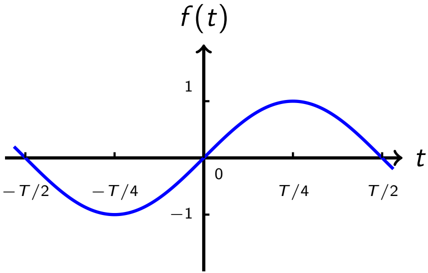
Graphical properties

The graph of an even function is **symmetric around the vertical axis**.
A graph is symmetric around the vertical axis if the point (-t, $f(t)$) lies on the graph whenever $(t, f(t))$ does.

Properties: parity

Odd function

A function is **odd** if $\forall t \in \mathbb{R}$: $f(-t) = -f(t)$.



Graphical properties

The graph of an odd function is **symmetric around the origin**.
A graph is symmetric around the origin if the point $(t, f(t))$ lies on the graph whenever $(-t, -f(t))$ does.

Properties: even and odd decomposition

Even and odd decomposition

A continuous-time signal $x(t)$ is representable as a sum of an even component and an odd component:

$$\begin{aligned} x(t) &= \underbrace{\frac{1}{2}(x(t)+x(-t))}_{x_e(t)} + \underbrace{\frac{1}{2}(x(t)-x(-t))}_{x_o(t)} \\ &= x_e(t) + x_o(t) \end{aligned}$$

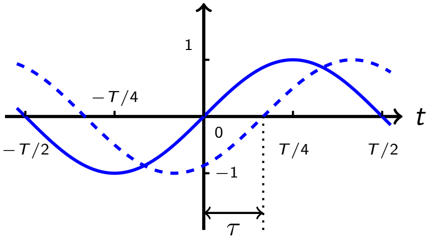
By construction, $x_e(t)$ and $x_o(t)$ are, respectively, even and odd.

Operations: time shifting

Delayed or shifted function (time shifting)

A function $g(t)$ is the original function $f(t)$ delayed of shifted by τ if $\forall t \in \mathbb{R}$, $g(t) = D_\tau f(t) = f(t-\tau)$. D_τ is the delay operator. The output of D_τ is the input signal delayed by τ seconds.

$f(t), g(t) = f(t-\tau)$ with $\tau > 0$



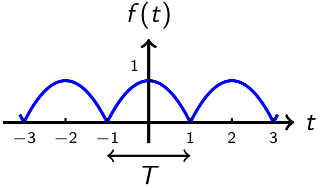
Properties

- If $\tau > 0$, $g(t)$ **lags** $f(t)$.
- If $\tau < 0$, $g(t)$ **leads** $f(t)$.

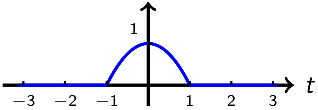
Properties: periodicity

Periodic function

A **periodic** function $f(t)$ with period T is the infinite repetition of the same motif $f_T(t)$ defined on an interval T. The motif $f_T(t)$ is often the restriction of function $g(t)$ on the interval T.



$$f_T(t) = \begin{cases} g(t) = 1 - t^2, \; t \in [-1, 1] \\ 0, \; \text{elsewhere} \end{cases}$$



Properties

Periodic signals are defined by the relation

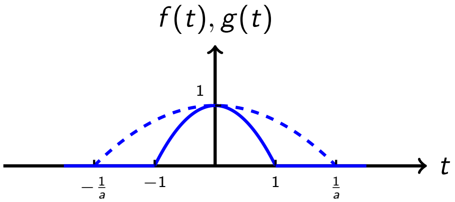
$$f(t) = \sum_{k=-\infty}^{\infty} f_T(t - kT), \quad k \in \mathbb{Z}$$

$$\Rightarrow \forall t \in \mathbb{R}, k \in \mathbb{Z}, f(t) = f(t + kT)$$

Operations: time scaling

Time scaling (expansion or contraction)

The time scaling operation transforms the function $f(t)$ into the function $g(t) = f(at)$ with $a \in \mathbb{R}^{0+}$. Equivalently, $g(t/a) = f(t)$.



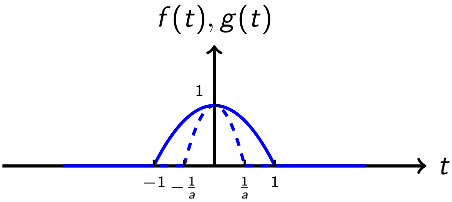
Expansion

$$g(t) = f(at), 0 < a < 1$$

Operations: time scaling

Time scaling (expansion or contraction)

The time scaling operation transforms the function $f(t)$ into the function $g(t) = f(at)$ with $a \in \mathbb{R}^{0+}$. Equivalently, $g(t/a) = f(t)$.



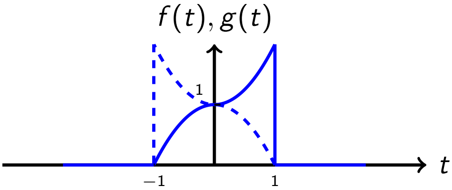
Compression

$$g(t) = f(at),\ a > 1$$

Operations: time reversal

Time reversal

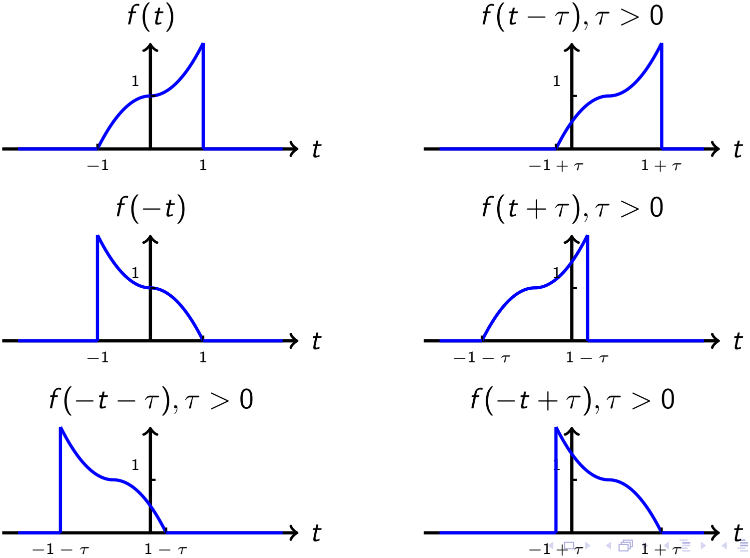
The time reversal or reflection operation transforms the function $f(t)$ into the function $g(t) = f(-t)$.



Use

The time reversal operation will be used when computing the convolution integral.

Operations: combinations



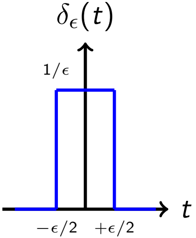
Operations: remarks

- Whenever we combine the delaying or advancing with reflection, delaying and advancing are swapped.
- Thus, $x(-t+1)$ is $x(t)$ reflected and delayed, or shifted to the right, by 1. The value $x(0)$ of the original signal is found in $x(-t+1)$ at $t=1$.
- Likewise, $x(-t-1)$ is $x(t)$ reflected and advanced, or shifted to the left by 1. The value $x(0)$ of the original signal is found in $x(-t-1)$ at $t=-1$.

- It is important to understand that **advancing** or **reflecting cannot be implemented in real time**, i.e. that is as the signal is being processed.
- Delays can be implemented in real time.
- Advancing and reflection require that the signal be saved or recorded.

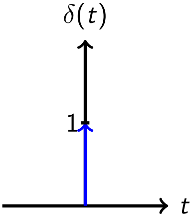
Dirac impulse

Consider a rectangular pulse of duration ϵ and unit area:



$$\delta_\epsilon(t) = \begin{cases} \frac{1}{\epsilon} & -\epsilon/2 \le t \le \epsilon/2 \\ 0 & t < -\epsilon/2 \text{ and } t > \epsilon/2 \end{cases}$$

Extremely narrow unit area pulse $\delta_\epsilon(t)$ when $\epsilon \to 0$.



Dirac impulse or distribution:

$$\delta(t) = \lim_{\epsilon \to 0} \delta_\epsilon(t)$$

Dirac impulse properties

Properties

- $\delta(t) = 0$ for $t \neq 0$,
- $\delta(t)$ is **not** defined for $t = 0$,
- $\int_{-a}^{a} \delta(t)dt = 1$ for $a > 0$.

Dirac impulse: sifting property[8]

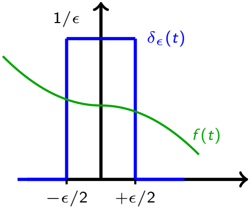
If $f(t)$ is a continuous signal in $t = 0$, then

$$f(t)\,\delta(t) = f(0)\,\delta(t).$$

In particular

$$\int_{-a}^{a} f(t)\,\delta(t)dt = f(0) \text{ for } 0 < a \leq \infty.$$

To see this, approximate $\delta(t)$ by $\delta_\epsilon(t)$



$f(t) \approx f(0)$ for $-\epsilon/2 \leq t \leq \epsilon/2$ and ϵ small

[8] propriété de localisation; to sift: passer au crible

Dirac impulse: sifting property[8]

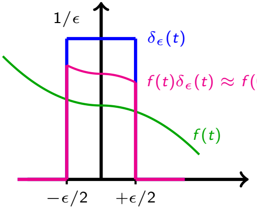
If $f(t)$ is a continuous signal in $t = 0$, then

$$f(t)\,\delta(t) = f(0)\,\delta(t).$$

In particular

$$\int_{-a}^{a} f(t)\,\delta(t)dt = f(0) \text{ for } 0 < a \leq \infty.$$

To see this, approximate $\delta(t)$ by $\delta_\epsilon(t)$



$f(t) \approx f(0)$ for $-\epsilon/2 \leq t \leq \epsilon/2$ and ϵ small

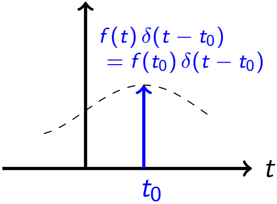
[8] propriété de localisation; to sift: passer au crible

Dirac impulse: generalisation of the sifting property

Generalisation of the sifting property

$$\boxed{f(t)\,\delta(t-t_0) = f(t_0)\,\delta(t-t_0),}$$

i.e. a Dirac impulse of weight $f(t_0)$ in $t = t_0$.



Schematic representation

The Dirac impulse is represented by a line surmounted by an arrow. The height of the arrow is usually used to specify the value of the weight, which gives the area under the curve.

Dirac impulse: generalisation of the sifting property

Generalisation of the sifting property

If $f(t)$ is a continuous signal in $t = 0$, then

$$\boxed{f(t_0) = \int_{-\infty}^{\infty} f(t)\,\delta(t - t_0)dt = \int_{-\infty}^{\infty} f(t + t_0)\,\delta(t)dt}$$

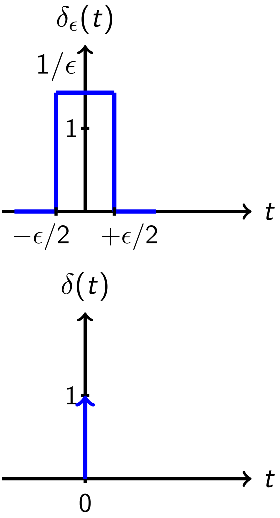
Generic representation of a continuous signal

$$\boxed{f(t) = \int_{-\infty}^{\infty} f(\tau)\,\delta(t - \tau)d\tau}$$

Proof:

$$\begin{aligned}\int_{-\infty}^{\infty} f(\tau)\,\delta(t - \tau)d\tau &= \int_{-\infty}^{\infty} f(t)\,\delta(t - \tau)d\tau \\ &= f(t)\int_{-\infty}^{\infty} \delta(t - \tau)d\tau = f(t)\end{aligned}$$

Unit step or Heaviside function



Unit step or Heaviside function

Definition

The unit step or Heaviside function is the integral of the Dirac impulse, i.e.

$$
\begin{aligned}
u(t) &= \int_{-\infty}^{t} \delta(\tau) d\tau \\
&= \lim_{\epsilon \to 0} u_\epsilon(t) = \lim_{\epsilon \to 0} \left(\int_{-\infty}^{t} \delta_\epsilon(\tau) d\tau \right)
\end{aligned}
$$

$$
u_\epsilon(t) = \begin{cases} 0 & t < -\epsilon/2 \\ \frac{1}{\epsilon}\left(t + \frac{\epsilon}{2}\right) & -\epsilon/2 \leq t \leq \epsilon/2 \\ 1 & t > \epsilon/2 \end{cases}
$$

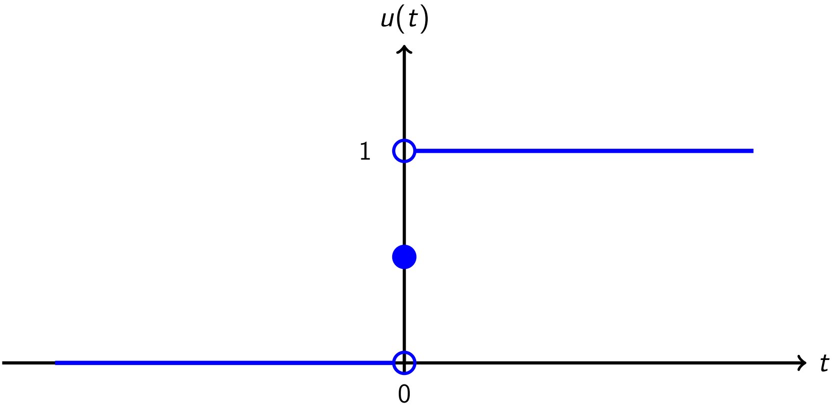
Unit step or Heaviside function

Properties

- $u(t) = \lim_{\epsilon \to 0} u_\epsilon(t) = \begin{cases} 0 & t < 0 \\ \frac{1}{2} & t = 0 \\ 1 & t > 0 \end{cases}$
- The derivative of $u(t)$ is zero over $\mathbb{R}^0$.
- The derivative of $u(t)$ is a Dirac impulse of weight 1 centered in $t = 0$, i.e.
$$\frac{du(t)}{dt} = \delta(t).$$
- Useful property: multiplying an ordinary function $x(t)$ by the step function $u(t)$ changes it into a **causal**[9] function; e.g. if $x(t) = \sin t$ then $\sin t\, u(t)$ is causal.

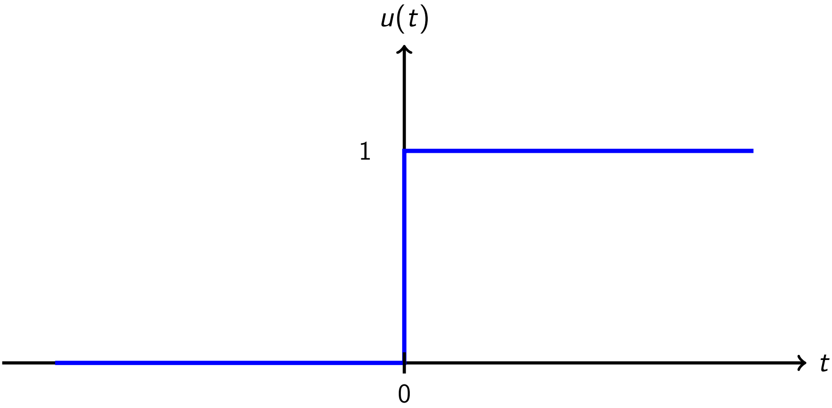
[9] causal functions take the value 0 when $t < 0$.

Unit step: "mathematically correct" representation



Heaviside function is a piecewise constant function

Unit step: "oscilloscope" representation

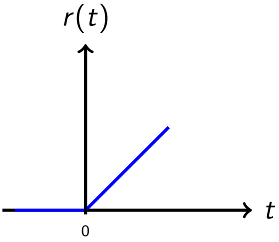


Heaviside function as it would appear on an oscilloscope

Unit step: remarks

- Since $u(t)$ is not a continuous function, it jumps from 0 to 1 instantaneously around $t = 0$, from the calculus point of view it should not have a derivative. That $\delta(t)$ is its derivative must be taken with suspicion, which makes the $\delta(t)$ signal also suspicious. Such signals can, however, be formally defined using the theory of distributions.
- Signals with jump discontinuities can be represented as the sum of a continuous signal and unit-step signals at the discontinuities. This is useful in computing the derivative of these signals.

Unit ramp



Definition

The ramp signal $r(t)$ is defined as

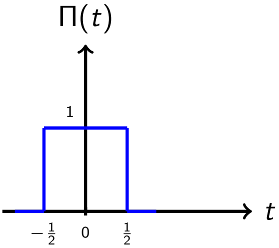
$$r(t) = \int_{-\infty}^{t} u(\tau) d\tau = t\, u(t).$$

Properties

Its relation to the unit-step and the unit-impulse signals is

$$\frac{dr(t)}{dt} = u(t), \quad \frac{d^2 r(t)}{dt^2} = \delta(t).$$

Rectangular pulse window function



Definition

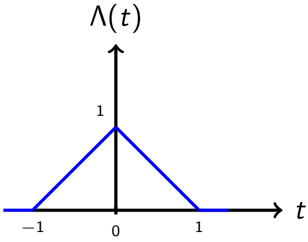
The rectangular pulse window function $\Pi(t)$ is defined as

$$\begin{aligned}\Pi(t) &= u(t + {}^1\!/_2) - u(t - {}^1\!/_2) \\ &= \begin{cases} 1 & |t| < {}^1\!/_2 \\ {}^1\!/_2 & |t| = {}^1\!/_2 \\ 0 & |t| > {}^1\!/_2 \end{cases}\end{aligned}$$

Properties

- The area is $\int_{-\infty}^{\infty} \Pi(t)dt = 1$.
- Window of length T, amplitude A and centered in $t = \tau$: $A\,\Pi(\frac{t-\tau}{T})$

Triangular pulse window function



Definition

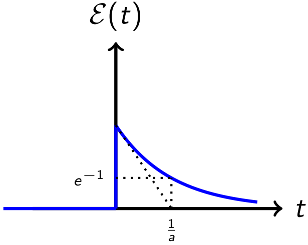
The triangular pulse window function $\Lambda(t)$ is defined as

$$\begin{aligned}\Lambda(t) &= (t+1)\,u(t+1) - 2\,t\,u(t) \\ &\quad +(t-1)\,u(t-1) \\ &= \begin{cases} 1-|t| & |t| \leq 1 \\ 0 & |t| > 1 \end{cases}\end{aligned}$$

Properties

- The area is $\int_{-\infty}^{\infty} \Lambda(t)dt = 1$.
- Window of length $2T$, amplitude A and centered in $t=\tau$: $A\,\Lambda(\frac{t-\tau}{T})$

Exponentiel pulse window function



Definition

The exponential pulse window function $\mathcal{E}(t)$ is defined as

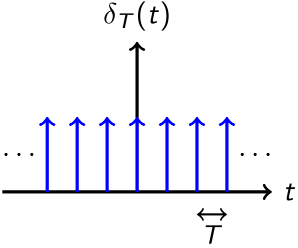
$$\mathcal{E}(t) = e^{-a\,t}u(t) \text{ for } a > 0.$$

Properties

- The area is $\int_{-\infty}^{\infty} \mathcal{E}(t)dt = \frac{1}{a}$.
- The exponential pulse window function ($a > 0$) can be used to dampen[10] a signal.

[10] amortir

Dirac comb[11]



Definition

The Dirac comb, denoted $\delta_T(t)$, is an infinite series of Dirac delta functions spaced at intervals of T, i.e.

$$\delta_T(t) = \sum_{k=-\infty}^{\infty} \delta(t - kT), \quad k \in \mathbb{Z}.$$

[11] peigne de Dirac

Total energy

Total energy

In signal processing, the energy of a continuous-time signal $x(t)$ is defined as

$$E = \int_{-\infty}^{\infty} |x(t)|^2 \, dt.$$

In the formulas for energy (and power) we are considering the possibility that the signals might be complex and so we are squaring its magnitude. If the signal being considered is real, this simply is equivalent to squaring the signal.

Average power

Average power

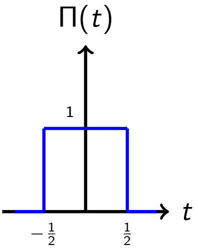
In signal processing, the average total power of a continuous-time signal $x(t)$ is defined as

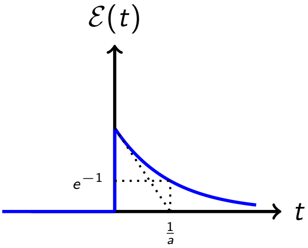
$$P = \lim_{T \to \infty} \frac{1}{T} \int_{-T/2}^{T/2} |x(t)|^2 dt.$$

For a periodic signal, P is computed over a period T

$$P = \frac{1}{T} \int_{T} |x(t)|^2 \, dt.$$

Finite energy, or square integrable signals





Finite energy signals

$$E = \int_{-\infty}^{\infty} |x(t)|^2 dt < \infty$$

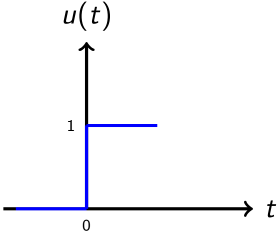
Remarks

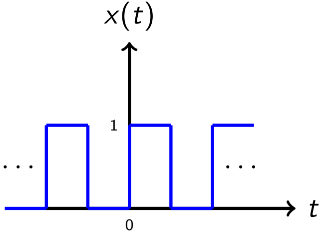
- A finite-energy signal has zero power. Indeed, if the energy of the signal is some constant $E < \infty$, then

$$P = \lim_{T \to \infty} \frac{E}{T} = 0.$$

- These signals are often called square integrable.

Finite power signals





Finite (average total) power signals

$$P = \lim_{T \to \infty} \frac{1}{T} \int_{-T/2}^{T/2} |x(t)|^2 dt < \infty$$

Remark

Finite power signals often have an infinite total energy, e.g. periodic signals.

Average power: sinusoidal case

The average total power of a sinusoidal signal is

$$x(t) = V_{max} \sin\left(\frac{2\pi t}{T}\right)$$

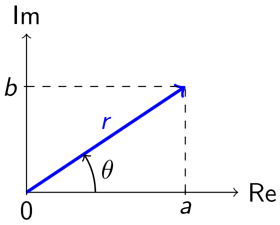
$$\begin{aligned} P = \frac{1}{T}\int_T |x(t)|^2 dt &= \frac{1}{T}\int_0^T V_{max}^2 \sin^2\left(\frac{2\pi t}{T}\right) dt \\ &= \frac{V_{max}^2}{2T}\int_0^T \left(1 - \cos\left(\frac{4\pi t}{T}\right)\right) dt \\ &= \frac{V_{max}^2}{2T}\left[t - \frac{\sin\left(\frac{4\pi t}{T}\right)}{\frac{4\pi}{T}}\right]_0^T = \frac{V_{max}^2}{2} \end{aligned}$$

This yields the root mean square value or "rms" value[12], i.e.

$$V_{rms} = \sqrt{P} = \frac{V_{max}}{\sqrt{2}}.$$

[12] valeur efficace

Complex numbers



$$r = |z| = \sqrt{a^2 + b^2}$$

$$\theta = \tan^{-1}\left(\frac{b}{a}\right)$$

- **Cartesian form**: $z = a + j\,b,\ j = \sqrt{-1}$
- **Polar form**: $z = r\,e^{j\theta}$
- **Euler identity**:

$$e^{j\theta} = \cos(\theta) + j\,sin(\theta)$$

- Note that

$$\cos(\theta) = \frac{1}{2}\left(e^{j\theta} + e^{-j\theta}\right)$$

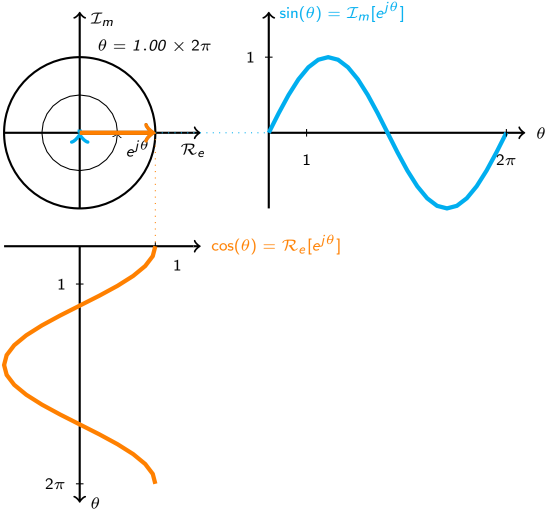
$$\sin(\theta) = \frac{1}{2j}\left(e^{j\theta} - e^{-j\theta}\right)$$

- **Complex conjugate**: $z^\star = a - jb$

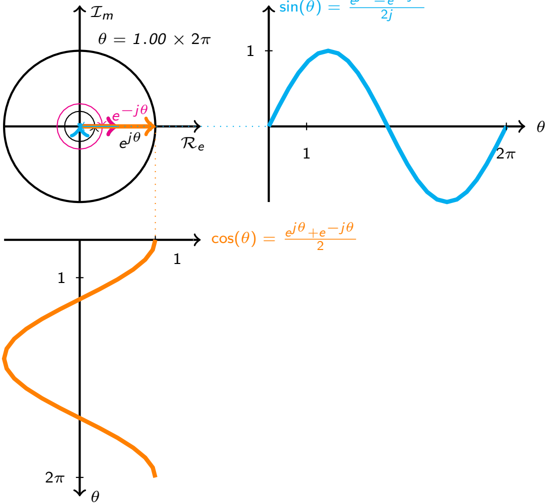
$$zz^\star = |z|^2 = a^2 + b^2$$

- Note that $e^{j\pi} = -1,\ e^{j\frac{\pi}{2}} = j$.

Sine and cosine: projection on imaginary and real axis



Sine and cosine: vector difference and sum



Complex exponentials

Complex exponentials

A complex exponential is a signal of the form

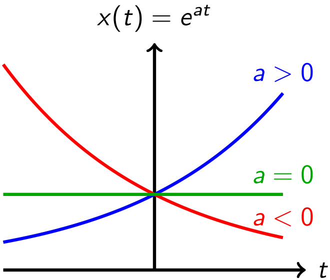
$$x(t) = Ae^{at}, \quad -\infty < t < \infty$$

where $A = |A|\, e^{j\phi}$ and $a = r + j\omega_0$ are **complex numbers**, respectively in polar and Cartesian form.

It follows that

$$\begin{aligned} x(t) &= Ae^{at}, \\ &= |A|\, e^{j\phi} e^{(r+j\omega_0)t}, \\ &= |A|\, e^{rt} e^{j(\omega_0 t+\phi)}, \\ &= |A|\, e^{rt} \left[\cos(\omega_0 t + \phi) + j \sin(\omega_0 t + \phi)\right]. \end{aligned}$$

Real exponentials[13]



Suppose that A and a are **real** then

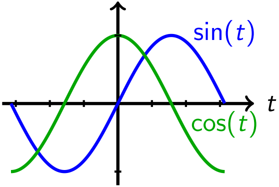
$$x(t) = A\, e^{at} = A\, e^{rt}, \quad -\infty < t < \infty$$

is

- exponentially **decreasing** if $a < 0$,
- exponentially **increasing** if $a > 0$.

[13] A is real $\Rightarrow A = |A|, \phi = 0$ and a is real $\Rightarrow a = r, \omega_0 = 0$

Purely imaginary exponentials[14]



Suppose that A is **real** and a is **purely imaginary**, i.e. $a = j\omega_0$, then

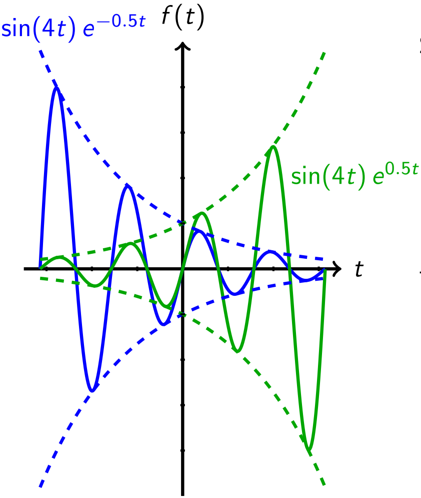
$$\begin{aligned} x(t) &= A\,e^{at} = A\,e^{j\omega_0 t} \\ &= A\,[\cos(\omega_0 t) + j\sin(\omega_0 t)], \\ & \qquad -\infty < t < \infty. \end{aligned}$$

Using the Euler identity, the following signals are obtained

- $A\cos(\omega_0 t) = \frac{A}{2}[e^{j\omega_0 t} + e^{-j\omega_0 t}] = \mathcal{R}_e[x(t)]$,
- $A\sin(\omega_0 t) = \frac{A}{2j}[e^{j\omega_0 t} - e^{-j\omega_0 t}] = \mathcal{I}_m[x(t)]$.

[14] A is real $\Rightarrow A = |A|, \phi = 0$ and a is purely imaginary $\Rightarrow a = j\omega_0, r = 0$

Complex exponentials



Suppose that A and a are **complex**.
Let us study the signal

$$\begin{aligned} f(t) &= \mathcal{I}_m[x(t)], \\ &= |A| e^{rt} \sin(\omega_0 t + \phi), -\infty < t < \infty. \end{aligned}$$

Remember $A = |A|\, e^{j\phi}$ and $a = r + j\omega_0$.

This signal is

- an exponentially **decreasing** sinusoid if $r = \mathcal{R}_e[a] < 0$,
- an exponentially **increasing** sinusoid if $r = \mathcal{R}_e[a] > 0$.

Complex exponentials

The complex exponential with $A = |A|\, e^{j\phi}$ and $a = r + j\omega_0$ can be written as

$$x(t) = Ae^{at} = |A|\, e^{rt}\left[\cos(\omega_0 t + \phi) + j\sin(\omega_0 t + \phi)\right].$$

Note that, in order to construct a **real exponential modulated by a sine or cosine**, one needs **2 complex exponentials**:

$$\begin{aligned} |A|\, e^{rt}\, \cos(\omega_0 t + \phi) &= \frac{1}{2}\left(Ae^{at} + A^\star e^{a^\star t}\right) \\ |A|\, e^{rt}\, \sin(\omega_0 t + \phi) &= \frac{1}{2j}\left(Ae^{at} - A^\star e^{a^\star t}\right) \end{aligned}$$

3. Continuous-time systems

Linear time invariant systems
Representations
Convolution integral
Stability

Continuous-time system

Definition

A continuous-time system is a system in which the signals at its input and output are continuous-time signals.

Mathematically we represent it as an operator (transformation) H that converts an input signal $x(t)$ into an output signal $y(t)$, i.e.

$$\underset{\text{Input}}{x(t)} \xrightarrow{H} \underset{\text{Output}}{y(t)} = S[x(t)].$$

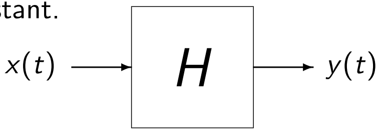
Input-output description

Definition

The **input-output description** of a system[15] gives a mathematical relation between the input and the output of a system.

Knowledge of the internal structure of the system is unavailable. The only access to the system is by means of input terminals and output terminals.

In order to leave aside the initial conditions, we assume that system is relaxed[16] at time t_0, i.e. no energy is stored in that system at that instant.



[15] The system is described by the operator H. The symbol H is often used to make the link with the impulse response $h(t)$ of the system introduced later in the course. Often the symbol P is used instead to refer to a "process".

[16] le système est au repos.

Linearity

Definition

A relaxed system is **linear** if and only if

$$\mathbf{y} = P\left(\alpha_1 \mathbf{u_1} + \alpha_2 \mathbf{u_2}\right) = \alpha_1\, P(\mathbf{u_1}) + \alpha_2\, P(\mathbf{u_2})$$

for any inputs $\mathbf{u_1}$ and $\mathbf{u_2}$ and any real numbers α_1 and α_2. Otherwise the system is **nonlinear**.

A linear system satisfies the **principle of superposition**.

Sinusoidal fidelity

If the input to a linear system is a sinusoidal wave, the output will also be a sinusoidal wave, and at exactly the same frequency as the input: only the amplitude and the phase will be modified, i.e.

$$A \sin(\omega_0 t + \phi) \longrightarrow \boxed{H} \longrightarrow B(\omega_0) \sin(\omega_0 t + \Phi(\omega_0)).$$

Causality

Definition

A system is **causal** or **non-anticipatory** if the output of the system at time t does not depend on the input applied after time t.

- The output at time t only depends on the input applied before and at time t.
- The past affects the future, but not conversely.
- If a relaxed system is causal, its input output relation can be written as

$$\mathbf{y}(t) = P\,\mathbf{u}_{(-\infty,\mathbf{t}]}$$

for all t in $(-\infty, \infty)$.

Time invariance

Definition

A system is **time invariant** or **stationary** if the characteristics of the system do not change with time.

An experiment conducted at time t will yield the same result an hour later, the day after or a year later.

A relaxed system is time invariant if and only if

$$\begin{aligned} H\, D_\tau\, x(t) &= D_\tau\, H\, x(t) \\ H\, x(t-\tau) &= D_\tau\, y(t) = y(t-\tau) \end{aligned}$$

where D_τ is the delay or shift operator. Otherwise the system is **time-varying**.

Systems by differential equations

Most continuous-time dynamic systems with lumped parameters are represented by linear ordinary differential equations with constant coefficients.

Definition (1)

Given a dynamic system represented by a linear differential equation with constant coefficients,

$$a_0 y(t) + a_1 \frac{dy(t)}{dt} + \ldots + a_n \frac{d^n y(t)}{dt^n} = b_0 x(t) + b_1 \frac{dx(t)}{dt} + \ldots + b_m \frac{d^m x(t)}{dt^m}$$

with n initial conditions $y(0)$ and ${}^{d^k y(t)}/_{dt^k}\big|_{t=0}$ for $k = 1, \ldots, n-1$ and inputs $x(t) = 0$ for $t < 0$.

Systems by differential equations

Definition (2)

The complete response of the system for $t \geq 0$ has two components:

- The zero-state response[17], $y_{zs}(t)$, due exclusively to the input $x(t)$ as the initial conditions are zero.
- The zero-input response[18], $y_{zi}(t)$, due exclusively to the initial conditions as the input $x(t)$ is zero.

So that

$$y(t) = y_{zs}(t) + y_{zi}(t).$$

Thus, when the initial conditions are zero (system is relaxed), then $y(t)$ depends exclusively on the input (i.e., $y(t) = y_{zs}(t)$), and the system is Linear and Time Invariant or LTI.

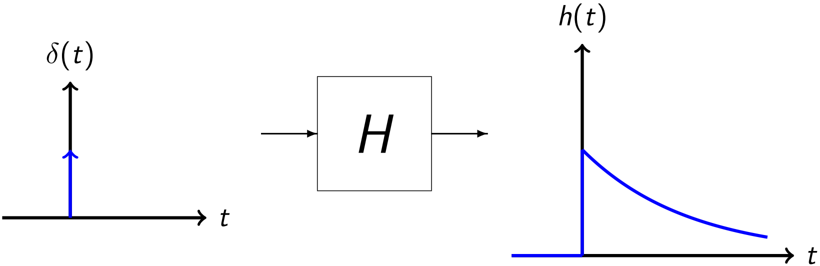
[17] la réponse **forcée**

[18] la réponse **libre**

Impulse response

Impulse response

The impulse response[19] $h(t)$ of a LTI system is its response to a Dirac impulse $\delta(t)$ given that initial conditions are zero, i.e. the system is relaxed.

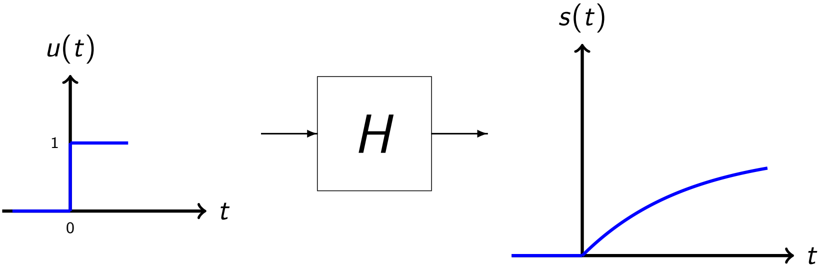


[19] Réponse impulsionnelle

Unit step response

Unit step response

The step response[20] $s(t)$ of a LTI system is its response to a unit step input $u(t)$ given that initial conditions are zero, i.e. the system is relaxed.



[20] Réponse indicielle ou réponse à un échelon unitaire

Convolution integral[21]

Definition

The response of an LTI system represented by its impulse response $h(t)$ to any signal $x(t)$ is the convolution integral

$$\begin{aligned} y(t) &= \int_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau = \int_{-\infty}^{\infty} x(t-\tau)h(\tau)d\tau \\ &= x(t) * h(t) = h(t) * x(t) \end{aligned}$$

where the symbol $*$ is used to denote the convolution operation.

We will see subsequently that the convolution integral is often evaluated using the Laplace transform.

[21] produit de convolution

Convolution integral

Reminder: generic representation of a continuous-time signal

$x(t) = \int_{-\infty}^{\infty} x(\tau)\,\delta(t-\tau)d\tau$

- **Hypothesis**: $h(t)$ is the impulse response of a relaxed system: the response to a $\delta(t)$ input Diract impulse is $h(t)$.
- **Time invariance**: the response to $\delta(t-\tau)$ is $h(t-\tau)$.
- **Linearity**: the response to $x(\tau)\,\delta(t-\tau)$ is $x(\tau)\,h(t-\tau)$ given that $x(\tau)$ is independent of time t.
- **Superposition**: the response to $x(t) = \int_{-\infty}^{\infty} x(\tau)\,\delta(t-\tau)d\tau$ is:

$$y(t) = \int_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau = \int_{-\infty}^{\infty} x(t-\sigma)h(\sigma)d\sigma.$$

The last equality is obtained using the substitution $\sigma = t - \tau$.

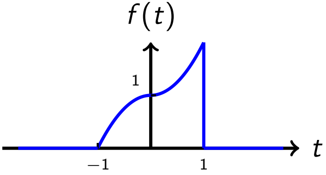
Convolution integral: remarks

- The impulse response is fundamental in the characterization of linear time-invariant systems.
- Any system characterized by the convolution integral is linear and time invariant by construction. The convolution integral is a general representation of LTI systems, given that it was obtained from a generic representation of the input signal.
- A system represented by a linear differential equation with constant coefficients and no initial conditions, or input, before $t = 0$ is LTI. Thus, one should be able to represent that system by a convolution integral after finding its impulse response $h(t)$.

Convolution integral: support

Definition

According to the dimension of their support, signals can be of finite or of infinite support. The support of a signal $x(t)$ can be understood as the smallest time interval of the signal outside of which the signal is always zero, i.e. $x(t) = 0$.



Support of $f(t)$: $[-1, 1]$

Length of the support of $f(t)$: 2

Property

The length of the support of $y(t) = x(t) * h(t)$ is equal to the sum of the lengths of the supports of $x(t)$ and $h(t)$.

Convolution integral: causality

Causality

An LTI system is **causal** if

$$h(t) = 0 \text{ for } t < 0.$$

The response of such a system to a causal input $x(t)$, i.e. $x(t) = 0$ for $t < 0$, is

$$y(t) = \int_0^t x(\tau)h(t-\tau)d\tau.$$

Convolution integral: causality

Convolution integral: general case

$y(t) = \int_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau$

- $y(t) = \int_{-\infty}^{t} x(\tau)h(t-\tau)d\tau + \int_{t}^{\infty} x(\tau)h(t-\tau)d\tau$.
- **Causality of the system**: $y(t) = \int_{-\infty}^{t} x(\tau)h(t-\tau)d\tau$ given that $h(t-\tau) = 0$ for $\tau > t$.
- **Causality of the input**: $y(t) = \int_{0}^{t} x(\tau)h(t-\tau)d\tau$ given that $x(\tau) = 0$ for $\tau < 0$.

Convolution integral: properties

Properties

- **Commutativity**: $\boxed{f(t) * g(t) = g(t) * f(t)}$
- **Distributivity**:

$$\boxed{[f(t) + g(t)] * h(t) = f(t) * h(t) + g(t) * h(t)}$$

- **Associativity**:

$$\boxed{[f(t) * g(t)] * h(t) = f(t) * [g(t) * h(t)]}$$

- **Identity element**: $\boxed{f(t) * \delta(t) = f(t)}$

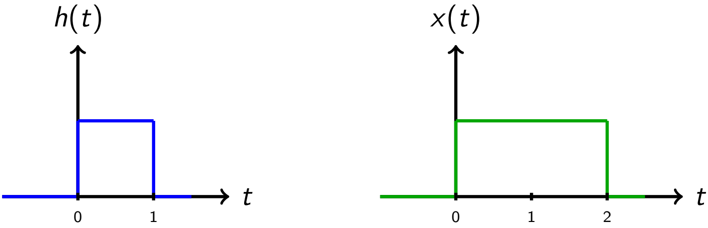
Convolution integral: visual explanations

Visual explanation of convolution for causal signals

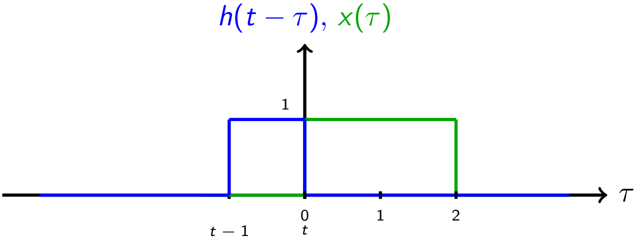
- Express $x(.)$ and $h(.)$ in terms of a dummy variable τ.
- Reflect the function $h(\tau) \longrightarrow h(-\tau)$.
- Add a time-offset t, which allows $h(t-\tau)$ to slide along the τ-axis.
- Start t at 0 and slide it all the way to ∞. Wherever the functions $x(\tau)$ and $h(t-\tau)$ intersect, find the integral from 0 to t of their product.

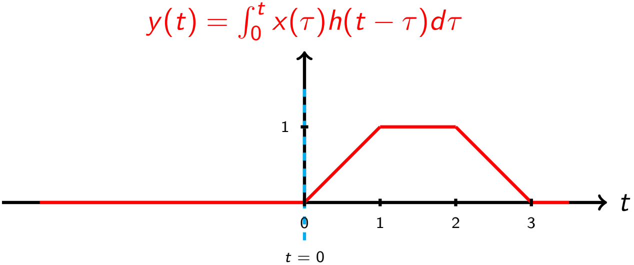
Convolution integral: visual example

- Impulse response: $h(t) = u(t) - u(t-1)$
- System input: $x(t) = u(t) - u(t-2)$

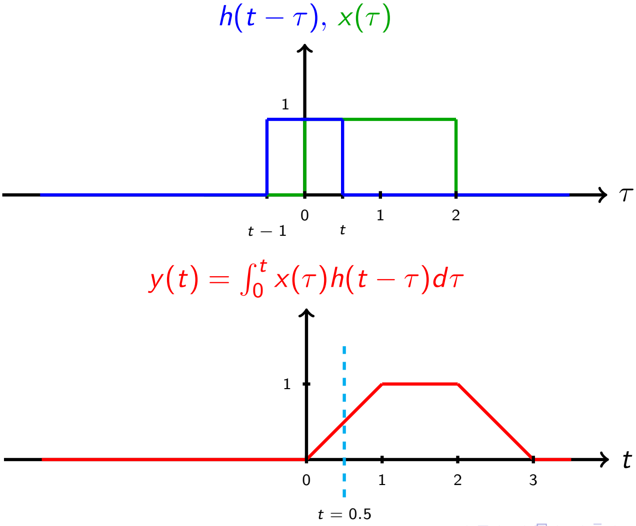


Convolution integral: visual example, $t = 0$



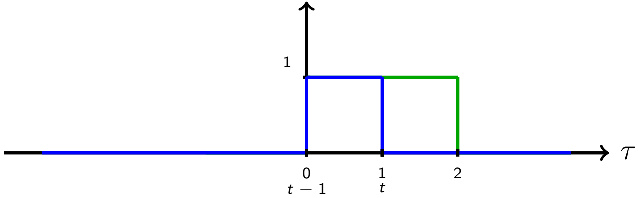


Convolution integral: visual example, $t = 0.5$

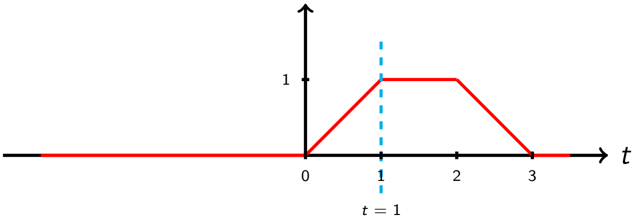


Convolution integral: visual example, $t = 1$

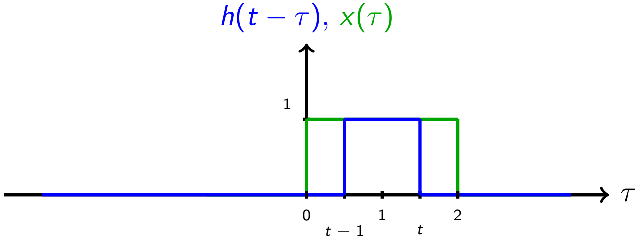
$h(t-\tau)$, $x(\tau)$

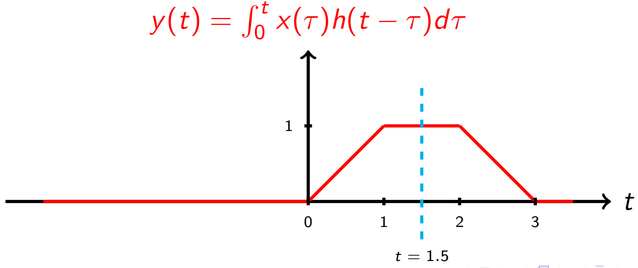


$y(t) = \int_0^t x(\tau)h(t-\tau)d\tau$

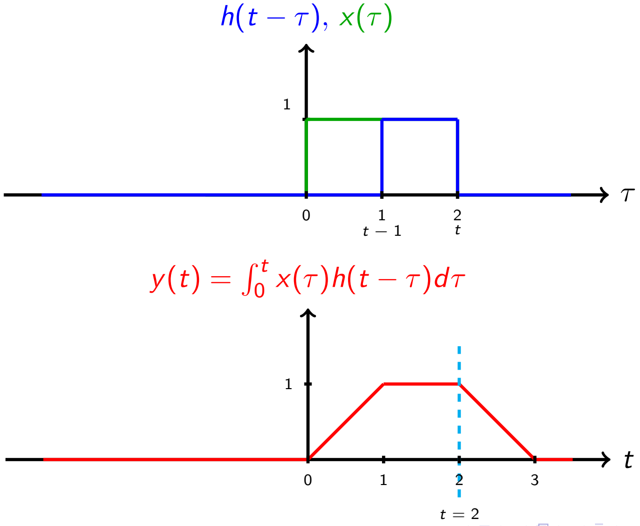


Convolution integral: visual example, $t = 1.5$

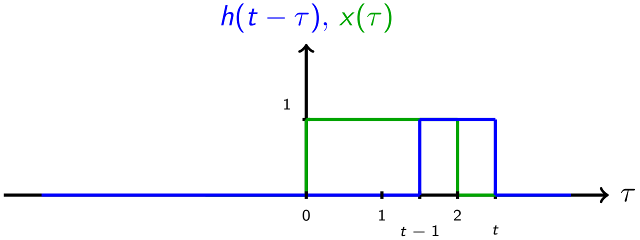


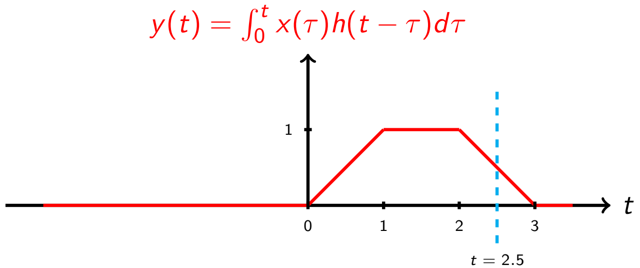


Convolution integral: visual example, $t = 2$

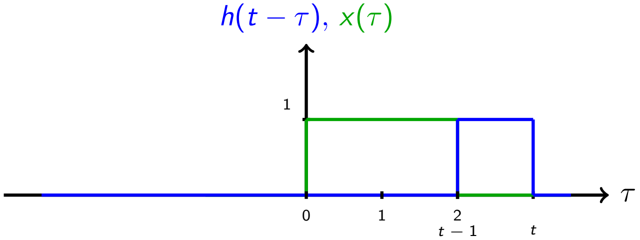


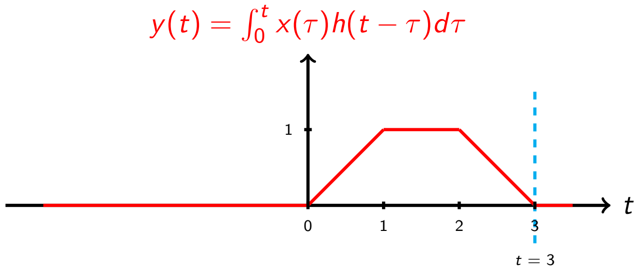
Convolution integral: visual example, $t = 2.5$



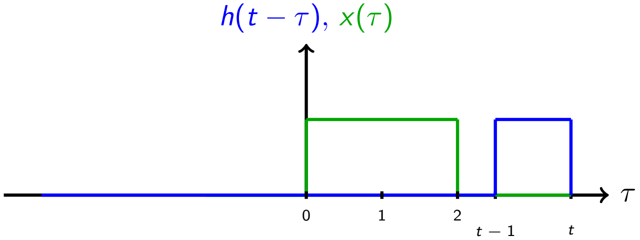


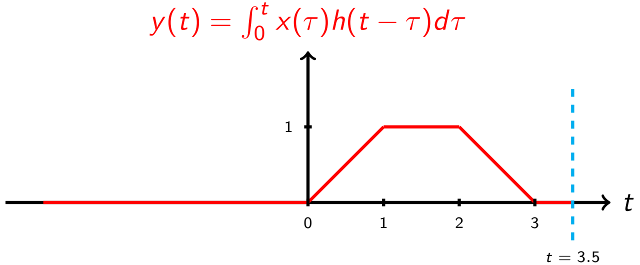
Convolution integral: visual example, $t = 3$





Convolution integral: visual example, $t = 3.5$





BIBO Stability

BIBO Stability

Bounded Input Bounded Output (BIBO) stability[22] establishes that for a bounded (i.e. a well-behaved) input $x(t)$, the output of a BIBO stable system $y(t)$ is also bounded. This means that if there is a finite bound $M < \infty$ such that $|x(t)| < M$ (you can think of it as an envelope $[-M, M]$ inside which the input is in) the output is also bounded.

[22] En français, on utilise parfois la notion de stabilité EBSB, c.-à-d. à une Entrée Bornée correspond une Sortie Bornée.

BIBO Stability

BIBO stability condition in the time domain

An **LTI** system is BIBO stable **if and only if**

$$\int_{-\infty}^{\infty} |h(\tau)|\, d\tau < \infty,$$

i.e. its impulse response is absolutely integrable.

A **causal LTI** system with an absolutely integrable impulse response is BIBO stable **if and only if**

$$\int_{0}^{\infty} |h(\tau)|\, d\tau < \infty.$$

A simpler way, using the Laplace transform, to test the BIBO stability of a system is given later.

BIBO Stability: sufficiency

For an LTI system with impulse response $h(t)$ to be BIBO stable, its response to a bounded input $x(t)$,

$$y(t) = \int_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau = \int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau$$

needs to be bounded. This implies

$$\begin{aligned}
|y(t)| &= \left|\int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau\right| \\
&\leq \int_{-\infty}^{\infty} |h(\tau)||x(t-\tau)|d\tau \\
&\leq \int_{-\infty}^{\infty} |h(\tau)|\, M\, d\tau = M\int_{-\infty}^{\infty} |h(\tau)|d\tau < \infty
\end{aligned}$$

i.e. $h(t)$ needs to be absolutely integrable[23].

[23] Here we have only proved the sufficiency of the stability condition. It is also possible to prove its necessity although this is more complicated.

4. Laplace transform

Two-sided Laplace transform

Definition

The two-sided Laplace transform of a continuous signal $f(t)$ is defined as

$$F(s) = \mathcal{L}[f(t)] = \int_{-\infty}^{\infty} f(t)e^{-st}dt, \quad s \in \text{ROC}$$

where $s = \sigma + j\omega$, with ω is the frequency (pulsation) expressed in rad/sec and σ is a damping factor.

ROC stands for the **Region Of Convergence**, i.e. the region where the integral exists.

Inverse Laplace transform

Definition

The inverse Laplace transform is given by

$$f(t) = \mathcal{L}^{-1}[F(s)] = \frac{1}{2\pi j} \int_{\sigma - j\infty}^{\sigma + j\infty} F(s) e^{st} ds, \quad s \in \text{ROC}$$

The inverse Laplace transform in the previous equation can be understood as the representation of $f(t)$ by an infinite summation of complex exponentials with weights $F(s)$.

The computation of the inverse Laplace transform using this equation would require complex integration. Algebraic methods will be used later to find the inverse Laplace transform, thus avoiding the complex integration.

Two-sided Laplace transform: remarks

- The Laplace transform $F(s)$ provides a representation of $f(t)$ in the s-domain, which in turn can be converted back into the original time-domain function in a one-to-one manner using the region of convergence. Thus,

$$F(s) \quad s \in \mathrm{ROC} \iff f(t)$$

- If $f(t) = h(t)$, the impulse response of an LTI system, then $H(s)$ is called the system or transfer function[24] of the system and it characterizes the system in the s-domain just like $h(t)$ does in the time-domain.
- If $f(t)$ is a signal, then $F(s)$ is its Laplace transform.

[24] The notion of a transfer function will be explained in more detail later.

Region of convergence

The values of σ for which the integral converges define the ROC. The pulsation ω does **not** affect the ROC.

$$\text{ROC} = \left\{ s = \sigma + j\omega \text{ such that } \int_{-\infty}^{\infty} |f(t)e^{-\sigma t}| dt < \infty \right\}$$

In order for the Laplace transform of $f(t)$ to exist, one needs

$$\begin{aligned} \left| \int_{-\infty}^{\infty} f(t)e^{-st} dt \right| &= \left| \int_{-\infty}^{\infty} \left(f(t)e^{-\sigma t} \right) e^{-j\omega t} dt \right|, \\ &\leq \int_{-\infty}^{\infty} |f(t)e^{-\sigma t}||e^{-j\omega t}| dt, \\ &= \int_{-\infty}^{\infty} |f(t)e^{-\sigma t}| dt < \infty, \end{aligned}$$

i.e. $f(t)e^{-\sigma t}$ needs to be absolutely integrable[25]. This is often possible by an appropriate choice of σ even if $f(t)$ is not itself absolutely integrable.

[25] Here we have only proved the sufficiency of this condition. It is also possible to prove its necessity.

Region of convergence

Conditions on σ for $\int_{-\infty}^{\infty} |f(t)e^{-\sigma t}|\, dt$ to converge ?

Suppose there exists N such that

- $|f(t)| < Ne^{\alpha t}, \ \forall t > 0, \alpha \in \mathbb{R}$
- $|f(t)| < Ne^{\beta t}, \ \forall t < 0, \beta \in \mathbb{R}$

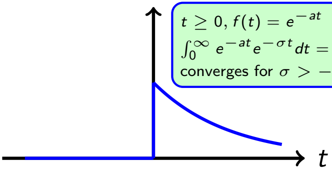
Then $F(s)$ converges for $\alpha < \sigma < \beta$.

$$\begin{aligned}
\int_{-\infty}^{\infty} |f(t)e^{-\sigma t}| dt &= \int_{-\infty}^{0} |f(t)e^{-\sigma t}|\, dt + \int_{0}^{\infty} |f(t)e^{-\sigma t}|\, dt \\
&< \int_{-\infty}^{0} |Ne^{\beta t}e^{-\sigma t}|\, dt + \int_{0}^{\infty} |Ne^{\alpha t}e^{-\sigma t}|\, dt \\
&< N\left[\frac{1}{\beta-\sigma} e^{(\beta-\sigma)t}\Big|_{-\infty}^{0} + \frac{1}{\alpha-\sigma} e^{(\alpha-\sigma)t}\Big|_{0}^{\infty} \right]
\end{aligned}$$

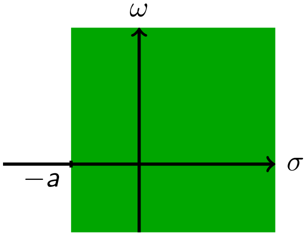
The integral converges if $(\beta - \sigma) > 0$ and $(\alpha - \sigma) < 0$, i.e. $\alpha < \sigma < \beta$.

Region of convergence: example

$$f(t) = e^{-at}u(t), a > 0$$



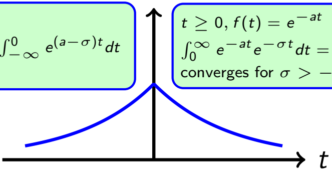
$t \geq 0, f(t) = e^{-at}$
$\int_0^\infty e^{-at}e^{-\sigma t}dt = \int_0^\infty e^{-(a+\sigma)t}dt$
converges for $\sigma > -a$

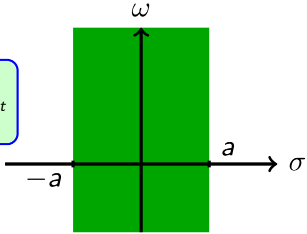


$$f(t) = e^{-a|t|} = e^{at}u(-t) + e^{-at}u(t), a > 0$$

$t \leq 0, f(t) = e^{at}$
$\int_{-\infty}^0 e^{at}e^{-\sigma t}dt = \int_{-\infty}^0 e^{(a-\sigma)t}dt$
converges for $\sigma < a$

$t \geq 0, f(t) = e^{-at}$
$\int_0^\infty e^{-at}e^{-\sigma t}dt = \int_0^\infty e^{-(a+\sigma)t}dt$
converges for $\sigma > -a$





One-sided Laplace transform

Definition

The one-sided Laplace transform of a continuous signal $f(t)$ is defined as

$$F(s) = \mathcal{L}[f(t)u(t)] = \int_{0^-}^{\infty} f(t)e^{-st}dt, \;\; s \in \mathrm{ROC}$$

where $f(t)$ is either causal or non-causal and made into a causal function by multiplication by $u(t)$.

Causal and anti-causal decomposition

- A non-causal function can be decomposed as

$$
\begin{aligned}
f(t) &= f_{ac}(t) + f_c(t) \\
&= f(t)u(-t) + f(t)u(t)
\end{aligned}
$$

- At time $t = 0$, $u(0) = 0.5$ insures $f(0) = f_{ac}(0) + f_c(0)$.
- The two-sided Laplace transform of $f(t)$ is

$$
\begin{aligned}
F(s) &= \int_{-\infty}^{0} f(t)u(-t)e^{-st}dt + \int_{0}^{\infty} f(t)u(t)e^{-st}dt \\
&= \int_{0}^{\infty} f(-t)u(t)e^{st}dt + \int_{0}^{\infty} f(t)u(t)e^{-st}dt \\
&= \mathcal{L}[f(-t)u(t)]_{(-s)} + \mathcal{L}[f(t)u(t)] = \mathcal{L}[f_{ac}(-t)]_{(-s)} + \mathcal{L}[f_c(t)]
\end{aligned}
$$

- The two-sided Laplace transform can be computed using only the one-sided transform with an ROC the intersection of the ROCs of the causal and the anti-causal Laplace transforms.

One-sided Laplace transform: properties

- The one-sided Laplace transform is of significance given that most of the applications deal with causal systems and signals.
- Notice that when $f(t)$ is causal, the two-sided and the one-sided Laplace transforms of $f(t)$ coincide.
- The lower limit of the integral in the one-sided Laplace transform is set to $0^- = 0 - \epsilon$, which corresponds to a value on the left side of zero for an infinitesimal value ϵ.
- The reason for this is to make sure that the Laplace transform of an impulse function $\delta(t)$ is defined. For any other signal this limit can be taken as zero with no effect on the transform.
- As we will see, the advantage of the one-sided Laplace transform is that it can be used in the solution of differential equations with initial conditions at $t = 0$.

Dirac impulse

The Laplace transform of a Dirac impulse $\delta(t)$ is

$$\mathcal{L}[\delta(t)] = \int_{0^-}^{\infty} \delta(t) e^{-st} dt = \int_{0^-}^{\infty} \delta(t) e^{-s0} dt = \int_{0^-}^{\infty} \delta(t) dt = 1.$$

There are no restrictions on the region of convergence, i.e. the ROC is the whole s-plane.

Unit step

The Laplace transform of a unit step $u(t)$ is

$$\begin{aligned}\mathcal{L}[u(t)] &= \int_0^\infty u(t)e^{-st}dt = \int_0^\infty e^{-st}dt = \int_0^\infty e^{-\sigma t}e^{-j\omega t}dt \\ &= \left.\frac{e^{-st}}{-s}\right|_0^\infty = \frac{1}{s}.\end{aligned}$$

The integral converges if $\mathcal{R}_e[s] = \sigma > 0$. The ROC is the open right half s-plane.

Unit ramp

The Laplace transform of a unit ramp $r(t) = tu(t)$ is

$$\begin{aligned}\mathcal{L}[r(t)] &= \int_{-\infty}^{\infty} tu(t)e^{-st}dt = \int_{0}^{\infty} te^{-st}dt \\ &= \left.\frac{-te^{-st}}{s}\right|_0^{\infty} + \int_0^{\infty} \frac{e^{-st}}{s}dt = 0 - \left.\frac{e^{-st}}{s^2}\right|_0^{\infty} = \frac{1}{s^2}\end{aligned}$$

The integral[26] converges if $\mathcal{R}_e[s] = \sigma > 0$. The ROC is the open right half s-plane.

[26] integration by parts !

Real exponential (1)

The Laplace transform of a real exponential $e^{at}u(t)$:

$$\begin{aligned}\mathcal{L}[e^{at}u(t)] &= \int_{-\infty}^{\infty} e^{at}u(t)e^{-st}dt = \int_{0}^{\infty} e^{-(s-a)t}dt \\ &= \frac{-1}{s-a}\, e^{-(s-a)t}\Big|_0^{\infty} = \frac{1}{s-a}.\end{aligned}$$

The integral converges if $\mathcal{R}_e[s] = \sigma > a$. The ROC is the open half plane to the right of the axis $\mathcal{R}_e[s] = a$.

The transform is identical for $a \in \mathbb{C}$ with ROC $\mathcal{R}_e[s] = \sigma > \mathcal{R}_e[a]$.

Real exponential (2)

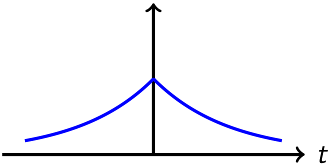
The one-sided Laplace transform of a real exponential $\frac{K}{T}e^{-\frac{t}{T}}u(t)$ is

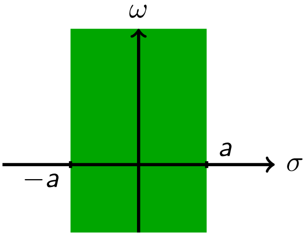
$$\begin{aligned}\mathcal{L}[\tfrac{K}{T}e^{-\frac{t}{T}}u(t)] &= \frac{K}{T}\int_0^\infty e^{-\frac{t}{T}}u(t)e^{-st}dt = \frac{K}{T}\int_0^\infty e^{-(Ts+1)\frac{t}{T}}dt \\ &= \frac{-K}{Ts+1}\,e^{-(Ts+1)\frac{t}{T}}\Big|_0^\infty = \frac{K}{Ts+1}.\end{aligned}$$

The integral converges if $\mathcal{R}_e[s] = \sigma > -\frac{1}{T}$. The ROC is the open half plane to the right of the axis $\mathcal{R}_e[s] = -\frac{1}{T}$.

Non-causal function: example

$f(t) = e^{-a|t|}, a > 0$





The function $f(t)$ can be decomposed in its causal and anti-causal components, i.e.

$$f(t) = e^{-a|t|} = e^{at}u(-t) + e^{-at}u(t) = f_{ac}(t) + f_c(t).$$

The Laplace transform $F(s)$ can be written as

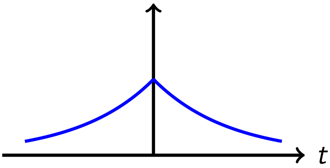
$$F(s) = \mathcal{L}[f(t)] = \mathcal{L}[f_{ac}(-t)]_{(-s)} + \mathcal{L}[f_c(t)].$$

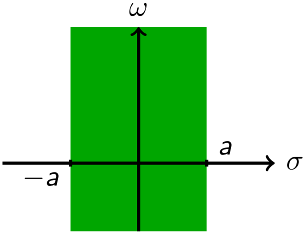
This yields

$$\mathcal{L}[e^{-a|t|}] = \mathcal{L}[e^{-at}u(t)]_{(-s)} + \mathcal{L}[e^{-at}u(t)].$$

Non-causal function: example

$f(t) = e^{-a|t|}, a > 0$





The Laplace transform $e^{-at}u(t)$ is $\mathcal{L}[e^{-at}u(t)] = \frac{1}{s+a}$.

The Laplace transform $F(s)$ can therefore be written as

$$\begin{aligned} F(s) &= \mathcal{L}[e^{-at}u(t)]_{(-s)} + \mathcal{L}[e^{-at}u(t)], \\ &= \frac{1}{-s+a} + \frac{1}{s+a} = \frac{-2a}{s^2-a^2} = F_{ac}(s) + F_c(s). \end{aligned}$$

Poles and zeros

Poles and zeros

Suppose $F(s) = \mathcal{L}[f(t)] = \frac{N(s)}{D(s)}$ is a rational function,

- A **zero** of $F(s)$ is a value of s for which $F(s) = 0$.
- A **pole** of $F(s)$ is a value of s for which "$F(s) = \infty$".

Typically, $N(s)$ and $D(s)$ are polynomials in s.

A **zero** of $F(s)$ is a value of s for which $N(s) = 0$.
A **pole** of $F(s)$ is a value of s for which $D(s) = 0$.

The poles and zeros of $F(s)$ can be complex[27].

Remark: By definition, no poles are included in the ROC.

[27] They come in complex conjugate pairs as we will consider $N(s)$ and $D(s)$ with real coefficients.

Region of convergence and poles

If $\{\sigma_i\}$ are the real parts of the poles of $F(s) = \mathcal{L}[f(t)]$ then:

- If $f(t)$ has **finite support**, i.e. $f(t) = 0$ for $t < t_1$ and $t > t_2$ and $t_1 < t_2$, then
  ROC = whole s-plane.
- If $f(t)$ is **causal**, i.e. $f(t) = 0$ for $t < 0$, then
  $\text{ROC} = \{(\sigma, \omega) : \sigma > \max\{\sigma_i\}, -\infty < \omega < \infty\}$.
- If $f(t)$ is **anti-causal**, i.e. $f(t) = 0$ for $t > 0$ the
  $\text{ROC} = \{(\sigma, \omega) : \sigma < \min\{\sigma_i\}, -\infty < \omega < \infty\}$.
- If $f(t)$ is **non-causal**, i.e. $f(t) = f_c(t) + f_{ac}(t)$ then
  $\text{ROC} = \text{ROC}_c \cap \text{ROC}_{ac}$.

Complex exponentials

The Laplace transform of the complex exponentials $e^{j(\omega_0 t+\theta)}u(t)$ is

$$\begin{aligned}
\mathcal{L}[e^{j(\omega_0 t+\theta)}u(t)] &= \int_0^\infty e^{j(\omega_0 t+\theta)}e^{-st}dt = e^{j\theta}\int_0^\infty e^{-(s-j\omega_0)t}dt, \\
&= \frac{-e^{j\theta}}{s-j\omega_0}\, e^{-(s-j\omega_0)t}\Big|_0^\infty = \frac{e^{j\theta}}{s-j\omega_0}.
\end{aligned}$$

The integral converge if $\mathcal{R}_e[s] = \sigma > 0$. The region of convergence is the open half plane s-plane.

Using Euler's identity, one obtains

$$\begin{aligned}
\mathcal{L}[\cos(\omega_0 t+\theta)u(t)] &= 0.5\left(\mathcal{L}[e^{j(\omega_0 t+\theta)}u(t)] + \mathcal{L}[e^{-j(\omega_0 t+\theta)}u(t)]\right), \\
&= 0.5\,\frac{e^{j\theta}(s+j\omega_0) + e^{-j\theta}(s-j\omega_0)}{s^2+\omega_0^2}, \\
&= \frac{s\cos(\theta) - \omega_0\sin(\theta)}{s^2+\omega_0^2}.
\end{aligned}$$

Complex exponentials

The Laplace transform of the complex exponentials $e^{j(\omega_0 t+\theta)}u(t)$ is

$$\mathcal{L}[\cos(\omega_0 t + \theta)u(t)] = \frac{s\cos(\theta) - \omega_0 \sin(\theta)}{s^2 + \omega_0^2}.$$

Now if we let $\theta = 0$ and $\theta = -\frac{\pi}{2}$ in the above equation we obtain, respectively,

- $\mathcal{L}[\cos(\omega_0 t)u(t)] = \dfrac{s}{s^2 + \omega_0^2}$ and
- $\mathcal{L}[\sin(\omega_0 t)u(t)] = \dfrac{\omega_0}{s^2 + \omega_0^2}$.

with region of convergence the open half plane s-plane ($\sigma > 0$).

Linearity

Linearity

For signals $f(t)$ and $g(t)$ with Laplace transforms

- $\mathcal{L}[f(t)] = F(s)$,
- $\mathcal{L}[g(t)] = G(s)$ and
- and constants α and $\beta \in \mathbb{C}$

we have

$$\mathcal{L}[\alpha f(t) + \beta g(t)] = \alpha\, \mathcal{L}[f(t)] + \beta\, \mathcal{L}[g(t)] = \alpha F(s) + \beta G(s)$$

with ROC the intersection of the regions of convergence $\mathcal{L}[f(t)]$ and $\mathcal{L}[g(t)]$.

Time shifting

Time shifting

For a function $f(t)$ with Laplace transform $\mathcal{L}[f(t)] = F(s)$, one has

$$\begin{aligned}\mathcal{L}[f(t-\tau)] &= \int_{-\infty}^{\infty} f(t-\tau)e^{-st}dt = e^{-s\tau}\int_{-\infty}^{\infty} f(\bar{t})e^{-s\bar{t}}d\bar{t} \\ &= e^{-s\tau}\mathcal{L}[f(t)] = e^{-s\tau}F(s)\end{aligned}$$

with the same region of convergence as $\mathcal{L}[f(t)]$.

We have used the substitution $\bar{t} = t - \tau$.

Time shifting

Suppose $f(t)$ is causal, periodic, i.e. the infinite repetition of the motif $f_T(t)$, a restriction of the function $g(t)$ on the interval T, i.e.

$$f(t) = f_T(t) + f_T(t-T)u(t-T) + f_T(t-2T)u(t-2T) + \cdots$$

with

$$f_T(t) = \begin{cases} g(t), & t \in [0 \quad T] \\ 0 & \text{elsewhere} \end{cases}$$

Then[28]

$$\begin{aligned} \mathcal{L}[f(t)] &= F_T(s)\left[1 + e^{-sT} + e^{-s2T} + \cdots\right] \\ &= F_T(s)\left(\frac{1}{1-e^{-sT}}\right) \end{aligned}$$

with $F_T(s) = \int_0^\infty f_T(t)e^{-st}dt = \int_0^T g(t)e^{-st}dt.$

[28] Geometric series: $\sum_{i=0}^{\infty} x^i = \frac{1}{1-x}$

Frequency shifting

Frequency shifting

For a function $f(t)$ with Laplace transform $\mathcal{L}[f(t)] = F(s)$ and $\lambda \in \mathbb{C}$,

$$\begin{aligned} F(s+\lambda) &= \int_{-\infty}^{\infty} f(t) e^{-(s+\lambda)t} dt = \int_{-\infty}^{\infty} \left(f(t) e^{-\lambda t} \right) e^{-st} dt \\ &= \mathcal{L}[f(t) e^{-\lambda t}] \end{aligned}$$

with ROC the region of convergence of $\mathcal{L}[f(t)]$ shifted $\mathcal{R}_e[\lambda]$ to the left if $\mathcal{R}_e[\lambda] > 0$ and to the right if $\mathcal{R}_e[\lambda] < 0$.

Frequency shifting

The frequency shifting property can be used to computed new transforms, i.e.

- $\mathcal{L}[te^{-at}u(t)] = \dfrac{1}{(s+a)^2}$
- $\mathcal{L}[e^{rt}\sin(\omega_0 t)\,u(t)] = \dfrac{\omega_0}{((s-r)^2+\omega_0^2)} = \dfrac{\omega_0}{(s-r-j\omega_0)(s-r+j\omega_0)}$
- $\mathcal{L}[e^{rt}\cos(\omega_0 t)\,u(t)] = \dfrac{s-r}{((s-r)^2+\omega_0^2)} = \dfrac{s-r}{(s-r-j\omega_0)(s-r+j\omega_0)}$

Differentiation

Derivatives of $f(t)$

For a signal $f(t)$ with Laplace transform $\mathcal{L}[f(t)u(t)] = F(s)$,

$$\mathcal{L}\left[\frac{df(t)}{dt}u(t)\right] = sF(s) - f(0^-).$$

In general, if $f^{(N)}(t)$ denotes an Nth-order derivative of a function $f(t)$ that has a Laplace transform $F(s)$, we have

$$\mathcal{L}[f^{(N)}u(t)] = s^N F(s) - \sum_{k=0}^{N-1} f^{(k)}(0^-)s^{N-1-k}$$

where $f^{(k)}(t) = \dfrac{d^k f(t)}{dt^k}, k > 0$ and $f^{(0)}(t) \triangleq f(t)$.

Differentiation

Derivatives of $f(t)$

For a signal $f(t)$ with Laplace transform $\mathcal{L}[f(t)u(t)] = F(s)$,

$$\mathcal{L}[f'(t)u(t)] = sF(s) - f(0).$$

This property can be used to find the Laplace transform of the successive derivatives of $f(t)$, i.e.

$$\begin{aligned}\mathcal{L}[f''(t)u(t)] = s\mathcal{L}[f'(t)u(t)] - f'(0) &= s(sF(s) - f(0)) - f'(0) \\ &= s^2F(s) - sf(0) - f'(0).\end{aligned}$$

Differentiation

It is now possible to compute the transform $f(t) = t^n\, u(t)$ knowing that

- $f^{(n)}(t) = n!\, u(t)$ and
- $f^{(m)}(0) = 0$ for $m < n$.

We have that

$$\mathcal{L}[f^{(n)}(t)\, u(t)] = s^n \mathcal{L}[t^n\, u(t)] = \mathcal{L}[n!\, u(t)] = \frac{n!}{s}.$$

This yields

$$\mathcal{L}[t^n\, u(t)] = \frac{n!}{s^{n+1}}.$$

This results is consistent with the results obtained earlier for a unit step and a unit ramp, i.e. for $n = 0$ and $n = 1$.

Differentiation

Derivatives of $F(s)$

For a function $f(t)$ with Laplace transform $\mathcal{L}[f(t)] = F(s)$,

$$\frac{dF(s)}{ds} = \int_{-\infty}^{\infty} f(t)\frac{d(e^{-st})}{ds}dt = \int_{-\infty}^{\infty} (-t\,f(t))e^{-st}dt = -\mathcal{L}[t\,f(t)],$$

i.e.

$$\frac{dF(s)}{ds} = -\mathcal{L}[t\,f(t)].$$

In general,

$$\frac{d^nF(s)}{ds^n} = (-1)^n\,\mathcal{L}[t^n f(t)].$$

Differentiation

This property can be used to computed new transforms, i.e.

- $\mathcal{L}[t^n e^{-at} u(t)] = (-1)^n \frac{d^n}{ds^n}\left(\frac{1}{s+a}\right) = \frac{n!}{(s+a)^{n+1}}$
- $\mathcal{L}[t \sin(\omega_0 t) u(t)] = -\frac{d}{ds}\left(\frac{\omega_0}{(s^2+\omega_0^2)}\right) = \frac{2\,\omega_0 s}{(s^2+\omega_0^2)^2}$
- $\mathcal{L}[t \cos(\omega_0 t) u(t)] = -\frac{d}{ds}\left(\frac{s}{(s^2+\omega_0^2)}\right) = \frac{s^2-\omega_0^2}{(s^2+\omega_0^2)^2}$

Integration

Integration of $f(t)$

The Laplace transform of the integral of a causal signal $f(t)$ is given by

$$\mathcal{L}\left[\left(\int_{0^-}^{t} f(\bar{t})d\bar{t}\right) u(t)\right] = \frac{F(s)}{s}$$

where $\mathcal{L}[f(t)u(t)] = F(s)$.

Integration of $F(s)$

For a function $f(t)$ with Laplace transform $\mathcal{L}[f(t)] = F(s)$, we have

$$\int_{s}^{\infty} F(u)du = \mathcal{L}\left[\frac{f(t)}{t}\right].$$

Time scaling

Time scaling

If the time scale[29] of a signal $f(t)$ is contracted (expanded), the frequency scale of the Laplace transform $F(s)$ is expanded (contracted).

For a function $f(t)$ with Laplace transform $\mathcal{L}[f(t)] = F(s)$, we have

$$\mathcal{L}[f(at)] = \int_{-\infty}^{\infty} f(at)e^{-st}dt = \frac{1}{a}\int_{-\infty}^{\infty} f(\bar{t})e^{-\frac{s}{a}\bar{t}}d\bar{t} = \frac{1}{a}F\left(\frac{s}{a}\right).$$

[29] $a \neq 0$!

Convolution integral

Convolution integral

The Laplace transform of the convolution integral of a causal signal $x(t)$, with Laplace transform $X(s)$, and a causal impulse response $h(t)$, with Laplace transform $H(s)$, is given by

$$\mathcal{L}[(h * x)(t)] = H(s)X(s).$$

Corollary

The response $y(t)$ of a LTI system with causal impulse response $h(t)$ to a causal input $x(t)$ is given by

$$y(t) = \mathcal{L}^{-1}[Y(s)] = \mathcal{L}^{-1}[H(s)X(s)] = [h * x](t).$$

Once $Y(s)$ is found, $y(t)$ is computed by means of the inverse Laplace transform. This typically involves a partial fraction expansion, i.e. decomposing the proper rational function into a sum of rational components of which the inverse transform can be found directly.

Transfer function

Transfer function

The transfer function $H(s) = \mathcal{L}[h(t)]$, the Laplace transform of the impulse response $h(t)$ of a LTI system can be expressed as the ratio

$$H(s) = \frac{\mathcal{L}[y(t)]}{\mathcal{L}[x(t)]} = \frac{\mathcal{L}[\text{output signal}]}{\mathcal{L}[\text{input signal}]} \; (= P(s))^{30}$$

This function is called **transfer function** as it transfers the Laplace transform of the input to the output.

The transfer function characterizes the system by means of its poles and zeros.

Thus it becomes a very important tool for the analysis and synthesis of (control) systems.

[30] We will sometimes use $P(s)$ to represent the process by its transfer function. The downside is that it is no clear anymore from the notations that the transfer function $P(s)$ is the Laplace transform of the impulse response of the system.

Transfer function and transform of a signal

Note that $^1/_s$ represents

- the **Laplace transform of a step input** signal $x(t) = u(t)$, i.e.

$$\mathcal{L}[x(t)] = \mathcal{L}[u(t)] = \frac{1}{s},$$

- the **transfer function of an integrating process**, i.e. the Laplace transform of the inpulse response of an integrating system $h(t) = u(t)$. We have that

$$H(s) = \mathcal{L}[h(t)] = \mathcal{L}[u(t)] = \frac{1}{s}.$$

Initial value theorem

From the Laplace transform $Y(s)$, it is possible to establish the initial value of $y(t)$ in the vicinity of $t = 0^+$.

Initial value theorem

Supposing that

- $y(t)$ and its derivatives have Laplace transforms,
- the limit $y(0^+) = \lim_{t\to 0^+} y(t)$ exists,

one has

$$y(0^+) = \lim_{s\to\infty} sY(s).$$

Final value theorem

From the Laplace transform $Y(s)$, it is possible to establish the steady-state value of $y(t)$ when $t \to \infty$.

Final value theorem

Assuming $\mathcal{R}_e[\text{poles}(sY(s))] < 0$, one has

$$\lim_{t\to\infty} y(t) = \lim_{s\to 0} sY(s).$$

Initial and final value theorems: application

Consider

- a lead-lag system $H(s) = K\,\dfrac{T_1\,s+1}{T_2\,s+1}$ and
- and unit step $x(t)$, i.e. $X(s) = \frac{1}{s}$.

The Laplace transform of the response is

$$Y(s) = K\,\frac{T_1\,s+1}{s(T_2\,s+1)}.$$

We can use the initial and final value theorems to get an idea of the response in time domain, i.e.

- **Initial value theorem**: response in the vicinity of $t = 0^+$ is $y(0^+) = \lim_{s\to\infty} sY(s) = K\,\frac{T_1}{T_2}$.
- **Final value theorem**: steady-state response is $y(\infty) = \lim_{s\to 0} sY(s) = K$.

Stability in the Laplace domain

Reminder:

- The region of convergence is of the Laplace transform of the impulse response of a system, i.e. its transfer function, is

$$\text{ROC} = \left\{ s = \sigma + j\omega \text{ such that } \int_{-\infty}^{\infty} |h(t)e^{-\sigma t}|dt < \infty \right\}.$$

- An LTI system is BIBO stable if and only if

$$\int_{-\infty}^{\infty} |h(t)|dt < \infty,$$

i.e. the Laplace transform of its impulse response converges on the imaginary axis.

Stability in the Laplace domain

Stability in the Laplace domain

The system is stable if it is representable by a transfer function $H(s)$ with region of convergence which **includes the imaginary axis**.

This condition is valid for causal, anti-causal and non-causal systems.

For a causal system, the condition implies that all the poles of $H(s)$ are strictly in the open left half s-plane.

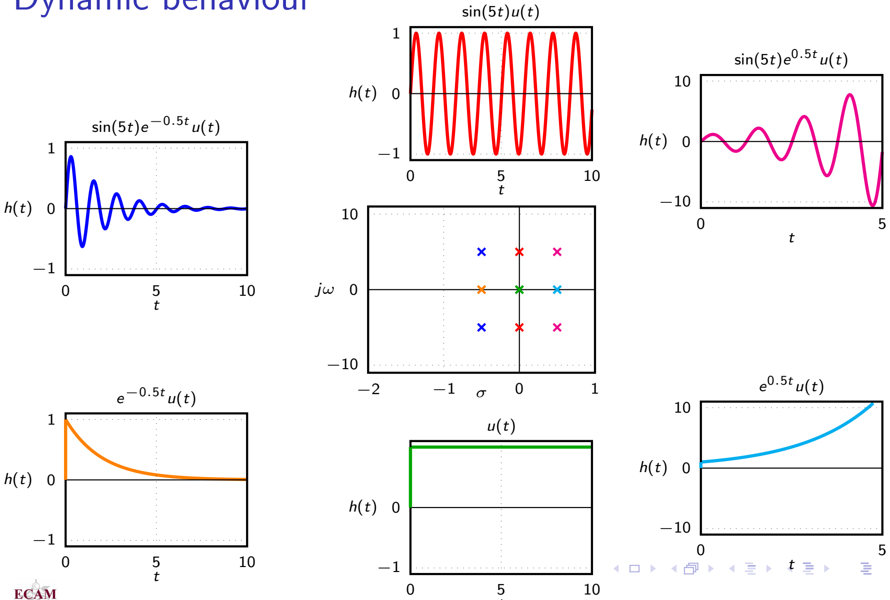
Stability in the Laplace domain

Control engineering joke

A group of Polish tourists is flying on a small airplane through the Grand Canyon on a sightseeing tour. The tour guide anounces: "On the right of the airplane, you can see the famous Bright Angle Falls". The tourists leap out of their seats and crowd to the windows on the right side. This causes a dynamic imbalance, and the plane violently rolls to the side and crashes into the canyon wall. All aboard are lost.

The moral to this episode is: always keep your poles to the left side of the plane.

Dynamic behaviour



General ideas

Assume the signal we wish to find has a rational Laplace transform, i.e.

$$F(s) = \frac{N(s)}{D(s)}$$

where $N(s)$ and $D(s)$ are polynomials in s with real-valued coefficients.

In order for the partial fraction expansion to be possible, it is required that $F(s)$ be proper rational, which means that the degree of the numerator polynomial $N(s)$ is less than that of the denominator polynomial $D(s)$.

If $F(s)$ is not proper, then we need to do long division until we obtain a proper rational function, i.e.

$$F(s) = g_0 + g_1 s + \cdots + g_m s^m + \frac{B(s)}{D(s)}$$

where the degree of $B(s)$ is now less than that of $D(s)$ so that we can perform partial expansion.

General ideas

- The poles of $F(s)$ provide the basic characteristics of the $f(t)$. If $F(s) = H(s)$, the poles provide the essential characteristics of the system described by the transfer $H(s)$.
- If $N(s)$ and $D(s)$ are polynomials in s with real coefficients, then the zeros and poles of $F(s)$ are real and/or complex conjugate pairs, and can be simple or multiple.
- In the inverse, $u(t)$ should be included since the result of the inverse is causal, i.e. the function $u(t)$ is an integral part of the inverse.
- The basic idea of the partial expansion is to decompose proper rational functions into a sum of rational components of which the inverse transform can be found directly in tables.

Partial fraction expansion: simple real poles

Simple real poles

If $F(s)$ is a proper rational function

$$F(s) = \frac{N(s)}{D(s)} = \frac{N(s)}{\Pi_k(s + p_k)}$$

where the $\{p_k\}$ are simple real poles of $F(s)$, its partial fraction expansion and its inverse are given by

$$F(s) = \sum_k \frac{A_k}{s + p_k} \Longleftrightarrow f(t) = \sum_k A_k e^{-p_k t} u(t)$$

where the expansion coefficients A_k are computed as

$$A_k = F(s)(s + p_k)|_{s=-p_k}$$

Partial fraction expansion: simple complex conjugate poles

Simple complex conjugate poles

The partial fraction expansion of a proper rational function[31]

$$F(s) = \frac{N(s)}{(s+r)^2 + \omega_0^2} = \frac{N(s)}{(s+r-j\omega_0)(s+r+j\omega_0)}$$

with complex conjugate poles $\{s_{1,2} = -r \pm j\omega_0\}$ is given by

$$F(s) = \frac{A}{(s+r-j\omega_0)} + \frac{A^\star}{(s+r+j\omega_0)}$$

$$\text{where } A = F(s)(s+r-j\omega_0)|_{s=-r+j\omega_0} = |A|e^{j\phi}$$

so that the inverse is the function

$$f(t) = 2|A|e^{-rt}\cos(\omega_0 t + \phi)u(t).$$

[31] The form used above is obtained by completing the squares !

Partial fraction expansion: simple complex conjugate poles

Simple complex conjugate poles

An equivalent partial fraction expansion[32] consists in expressing the numerator $N(s)$ of the proper rational function $F(s)$ as $N(s) = a + b(s+r)$, for some constants a and b, i.e.

$$F(s) = \frac{a + b(s+r)}{(s+r)^2 + \omega_0^2} = \frac{a}{\omega_0} \frac{\omega_0}{(s+r)^2 + \omega_0^2} + b \frac{(s+r)}{(s+r)^2 + \omega_0^2}.$$

The inverse Laplace transform is a sum of a sine and a cosine multiplied by a decaying exponential, i.e.

$$f(t) = e^{-rt} \left[\frac{a}{\omega_0} \sin \omega_0 t + b \cos \omega_0 t \right] u(t).$$

Note that this form is equivalent to the one in the previous slide.

[32] This is often easier than to work with complex numbers !

Partial fraction expansion: multiple real poles

Multiple real poles

If $F(s)$ is a proper rational function with multiple real poles, the partial fraction expansion is

$$F(s) = \frac{N(s)}{(s+\alpha)^r} = \frac{c_1}{(s+\alpha)} + \frac{c_2}{(s+\alpha)^2} + \cdots + \frac{c_r}{(s+\alpha)^r}$$

and the inverse Laplace transform is

$$f(t) = [c_r \frac{t^{r-1}}{(r-1)!} + \cdots + c_2 t + c_1] e^{-\alpha t} u(t)$$

with

$$\begin{aligned} c_r &= F(s)(s+\alpha)^r|_{s=-\alpha} \\ c_{r-k} &= \frac{1}{k!} \frac{d^k}{ds^k} [F(s)(s+\alpha)^r]\Big|_{s=-\alpha}, \quad k = 1, \cdots, r-1 \end{aligned}$$

Partial fraction expansion: example

Find the inverse Laplace transform of the function

$$F(s) = \frac{4}{s^3 + 2s^2 + 4s}.$$

Completing the squares, one obtains

$$F(s) = \frac{4}{s(s^2 + 2s + 4)} = \frac{4}{s((s+1)^2 + 3)}.$$

We will perform a partial fraction expansion

$$F(s) = \frac{A}{s} + \frac{Bs + C}{(s+1)^2 + 3}.$$

We have that

$$A = F(s)s|_{s=0} = 1.$$

Bringing back to same denominator, one obtains the following constraint

$$N(s) = 4 = A(s^2 + 2s + 4) + Bs^2 + Cs$$

yielding $B = -1$ and $C = -2$.

Partial fraction expansion: example

We have thus obtained the partial fraction expansion

$$F(s) = \frac{1}{s} - \frac{s+2}{(s+1)^2+3}.$$

Equivalently, one obtains

$$F(s) = \frac{1}{s} - \frac{s+1}{(s+1)^2+3} - \frac{1}{\sqrt{3}}\frac{\sqrt{3}}{(s+1)^2+3}.$$

The inverse Laplace transform is therefore

$$f(t) = \left[1 - e^{-t}\left(\cos(\sqrt{3}t) + \frac{1}{\sqrt{3}}\sin(\sqrt{3}t)\right)\right]u(t).$$

Analysis of LTI systems

The complete response $y(t)$ of a system represented by an nth-order linear differential equation with constant coefficients,

$$\sum_{k=0}^{n} a_k\, y^{(k)}(t) = \sum_{l=0}^{m} b_l\, x^{(l)}(t)$$

where $x(t)$ is the input and $y(t)$ is the output of the system, and initial conditions

$$\{y^{(k)}(0),\ 0 \leq k \leq n-1\},$$

is obtained by inverting the Laplace transform

$$Y(s) = \frac{B(s)}{A(s)}\, X(s) + \frac{1}{A(s)}\, I(s)$$

where $Y(s) = \mathcal{L}[y(t)]$ and $X(s) = \mathcal{L}[x(t)]$.

Here we have used the notation $f^{(k)}(t) = \frac{d^k f(t)}{dt^k}$.

Analysis of LTI systems

The complete response $y(t)$ is obtained by inverting the Laplace transform

$$Y(s) = \frac{B(s)}{A(s)} X(s) + \frac{1}{A(s)} I(s)$$

where $Y(s) = \mathcal{L}[y(t)]$ and $X(s) = \mathcal{L}[x(t)]$, and

$$A(s) = \sum_{k=0}^{n} a_k s^k, \quad B(s) = \sum_{l=0}^{m} b_l\, s^l \text{ and}$$

$$I(s) = \sum_{k=1}^{n} a_k \left(\sum_{m=0}^{k-1} s^{k-m-1} y^{(m)}(0) \right)$$

depends on the initial conditions.

Analysis of LTI systems

The complete response $y(t)$ is obtained by inverting the Laplace transform

$$Y(s) = \frac{B(s)}{A(s)} X(s) + \frac{1}{A(s)} I(s)$$

which gives

$$y(t) = y_{zs}(t) + y_{zi}(t)$$

where

- $y_{zs}(t) = \mathcal{L}^{-1}[H(s)X(s)]$ is the **zero-state response**,
- $y_{zi}(t) = \mathcal{L}^{-1}[H_i(s)I(s)]$ is the **zero-input response**, with

$$H(s) = \frac{B(s)}{A(s)} \text{ et } H_i(s) = \frac{1}{A(s)}.$$

The associated transfer function is $H(s)$.

Transient and steady-state responses

Consider a **stable** system with the following response in the Laplace domain

$$Y(s) = H(s)\,X(s) = \frac{B(s)}{A(s)}\,X(s) \ \text{(zero initial conditions)}.$$

Expanding $H(s)$ and $X(s)$ into partial fractions [33]

$$H(s) = \frac{B(s)}{A(s)} = \frac{B(s)}{a_n(s+p_1)\cdots(s+p_n)}$$

$$X(s) = \frac{N(s)}{D(s)} = \frac{N(s)}{d_n(s+q_1)\cdots(s+q_l)}$$

yields

$$Y(s) = \sum_{i=1}^{n} \frac{c_i}{s+p_i} + \sum_{j=1}^{l} \frac{k_j}{s+q_j}.$$

[33] For convenience, we suppose that all poles are **simple**. The result generalises to multiple and/or complex poles.

Transient and steady-state responses

The inverse Laplace transform of a **stable** system yields

$$y(t) = \underbrace{\sum_{i=1}^{n} c_i e^{-p_i t}}_{\text{Transient response}} + \underbrace{\sum_{j=1}^{l} k_j e^{-q_j t}}_{\text{Steady-state response}} .$$

- The **natural** response only depends on the natural modes of the system. As the system is stable, the natural response decays and it disappears when $t \to \infty$. It is then called a **transient** response.
- Assuming that the system is stable, then the system response in the long run is determined by its steady state response only.

  The **steady-state** response then depends on the input signal only.

Steady-state response: **general case**

- If the poles (simple or multiple, real or complex) of the Laplace transform of the output, $Y(s)$, of an LTI system are in the open left half s-plane (i.e., no poles on the $j\omega$-axis), the steady-state response is zero.
- Simple complex conjugate poles and a simple real pole at the origin of the s-plane cause a steady-state response:
  - If the pole of $Y(s)$ is $s = 0$ we know that its inverse transform is of the form $Au(t)$.
  - If the poles are complex conjugates $\pm j\omega_0$ the corresponding inverse transform is a sinusoid.
- However, multiple poles on the $j\omega$-axis, or any poles in the right half s-plane will give inverses that grow as $t \to \infty$.

Frequency response

There is a tight link between the transfer function and the frequency response of a system, defined as the steady-state response of the system to a sinusoidal input signal.

Consider $X(s) = \frac{\omega}{s^2+\omega^2}$. The response of the system is

$$Y(s) = H(s)\,\frac{\omega}{s^2+\omega^2} = \frac{B(s)}{A(s)}\,\frac{\omega}{s^2+\omega^2} = \frac{B(s)}{A(s)}\,\frac{\omega}{(s+j\omega)(s-j\omega)}$$

Expanding into partial fractions and assuming a **stable** system, one obtains

$$Y(s) = \underbrace{\frac{c_1}{(s+j\omega)} + \frac{c_2}{(s-j\omega)}}_{\text{steady-state response}} + \underbrace{\frac{\overline{B}(s)}{A(s)}}_{\text{transient response}}$$

$$c_1 = \frac{B(s)}{A(s)}\,\frac{\omega}{(s-j\omega)}\bigg|_{s=-j\omega} = -\frac{H(-j\omega)}{2j},\quad c_2 = \frac{B(s)}{A(s)}\,\frac{\omega}{(s+j\omega)}\bigg|_{s=+j\omega} = \frac{H(j\omega)}{2j}$$

Frequency response

The steady-state response to a sinusoidal input is

$$\begin{aligned} y_{ss}(t) &= \mathcal{L}^{-1}[-\frac{H(-j\omega)}{2j(s+j\omega)} + \frac{H(j\omega)}{2j(s-j\omega)}] \\ &= -\frac{H(-j\omega)}{2j}e^{-j\omega t} + \frac{H(j\omega)}{2j}e^{j\omega t} \end{aligned}$$

Writing $H(j\omega)$ in polar form $H(j\omega) = |H(\omega)|e^{j\theta(\omega)}$ yields

$$\begin{aligned} y_{ss}(t) &= \frac{|H(\omega)|}{2j}(e^{j(\theta(\omega)+\omega t)} - e^{-j(\theta(\omega)+\omega t)}), \\ &= |H(\omega)|\sin(\omega t + \theta(\omega)). \end{aligned}$$

Frequency response

The steady-state response to a sinusoidal input is

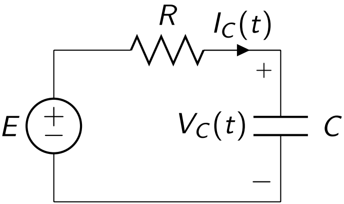
$$y_{ss}(t) = |H(\omega)| \sin(\omega t + \theta(\omega)) \quad \text{where}$$

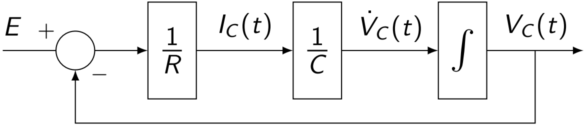
- $|H(\omega)|$ is the magnitude as a function of ω (modulus of $H(j\omega)$),
- $\theta(\omega)$ is the phase shift as a function of ω (argument of $H(j\omega)$).

The linear system acts as a frequency filter. Varying the pulsation ω from 0 to infinity, one obtains an infinite number of sinusoidal input steady-state responses that can be represented graphically:

- **Bode diagram**: $\{\omega, |H(\omega)|, \theta(\omega)\}$, i.e. the gain and phase shift as a function of frequency using logarithmic or semi-logarithmic scales.
- **Nyquist diagram**: $\{\mathcal{R}_e[H(j\omega)], \mathcal{I}_m[H(j\omega)]\}$, i.e. in Cartesian coordinates, the real part of $H(j\omega)$ versus the imaginary part of $H(j\omega)$ using frequency as a parameter in the plot.
- **Black-Nichols diagram**: $\{|H(\omega)|, \theta(\omega)\}$, i.e. in Cartesian coordinates, the gain versus the phase shift using frequency as a parameter in the plot.

First order RC system





$$I_C(t) = \frac{E - V_C(t)}{R} = C\,\frac{dV_C(t)}{dt} = C\dot{V}_C(t)$$

Self-regulation, i.e. stability, is obvious from the block diagram: if we keep the voltage E constant, the capacitor voltage will settle out at a constant voltage $V_C = E$.

First order system: Laplace domain

First order RC system:

$$\frac{dV_C(t)}{dt} = \frac{1}{RC}\left(E - V_C(t)\right) \Longleftrightarrow RC\,\frac{dV_C(t)}{dt} + V_C(t) = E.$$

In general, a **first order system** is described by the **first order differential equation**:

$$\frac{dy}{dt} = \frac{1}{T}\left(K_P\,x - y\right) \Longleftrightarrow T\,\frac{dy}{dt} + y = K_P\,x.$$

In the Laplace domain assuming zero initial conditions, one has

$$sY(s) = \frac{1}{T}\left(K_P\,X(s) - Y(s)\right) \Rightarrow \frac{Y(s)}{X(s)} = \frac{K_P}{Ts+1}.$$

First order system: Laplace domain

$$Y(s) = \frac{K_P}{Ts+1} X(s) = H(s)X(s)$$

Note that

$$\begin{aligned} H(j\omega) = \left.\frac{K_P}{Ts+1}\right|_{s=j\omega} &= \frac{K_P}{j\omega T + 1} = K_P \frac{1 - j\omega T}{1 + \omega^2 T^2} \\ &= K_P \frac{1}{1+\omega^2 T^2} - j\, K_P \frac{\omega T}{1 + \omega^2 T^2} \\ &= \mathcal{R}_e[H(j\omega)] + j\, \mathcal{I}_m[H(j\omega)]. \end{aligned}$$

Therefore

$$|H(j\omega)| = \left| \frac{K_P}{j\omega T + 1} \right| = \sqrt{\mathcal{R}_e^2[P(j\omega)] + \mathcal{I}_m^2[P(j\omega)])} = \frac{K_P}{\sqrt{1+\omega^2 T^2}},$$

$$\arg[H(j\omega)] = \arctan\left(\frac{\mathcal{I}_m[P(j\omega)]}{\mathcal{R}_e[P(j\omega)]} \right) = -\arctan(\omega T).$$

First order system: Laplace domain[34]

One has

$$|H(j\omega)| = \frac{K_P}{\sqrt{1+\omega^2 T^2}} \quad \text{and} \quad \arg[H(j\omega)] = -\arctan(\omega T).$$

Suppose $\omega << \frac{1}{T} = \omega_c$. Then

$$|H(j\omega)| \approx K_P \text{ and } \arg[H(j\omega)] \approx 0.$$

Suppose $\omega >> \frac{1}{T} = \omega_c$. Then

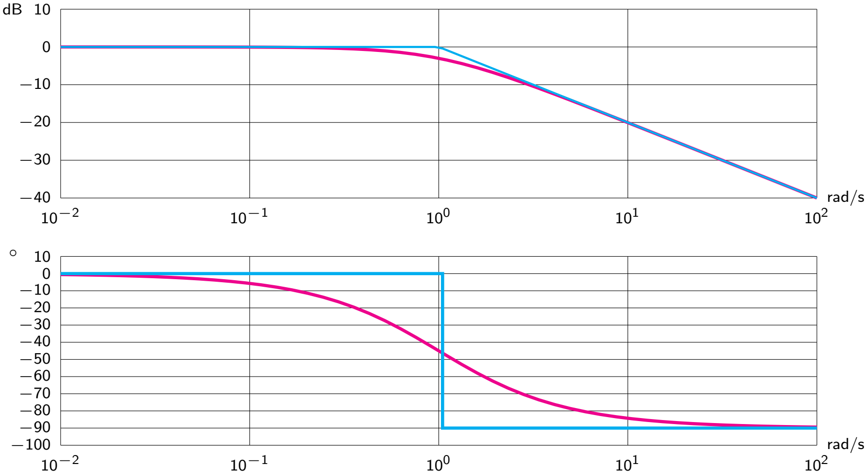
$$|H(j\omega)| \approx \frac{K_P}{\omega T} \text{ and } \arg[H(j\omega)] \approx -90^\circ.$$

Suppose $\omega = \frac{1}{T} = \omega_c$. Then

$$|H(j\omega)| = \frac{K_P}{\sqrt{2}} \text{ and } \arg[H(j\omega)] = -45^\circ.$$

[34] For a stable system, i.e. $T > 0$.

First order system: Bode diagram[35]



[35] $K_P = 1,\ T = 1$

Dead-time

Time domain:

$$y(t) = x(t - \theta)$$

Laplace domain:

$$Y(s) = e^{-\theta s} X(s) = H(s)X(s)$$

Note that

$$e^{-\theta s}\Big|_{s=j\omega} = e^{-j\omega\theta} = \cos(\omega\theta) - j\sin(\omega\theta) = \mathcal{R}_e[H(j\omega)] + j\,\mathcal{I}_m[H(j\omega)].$$

Therefore

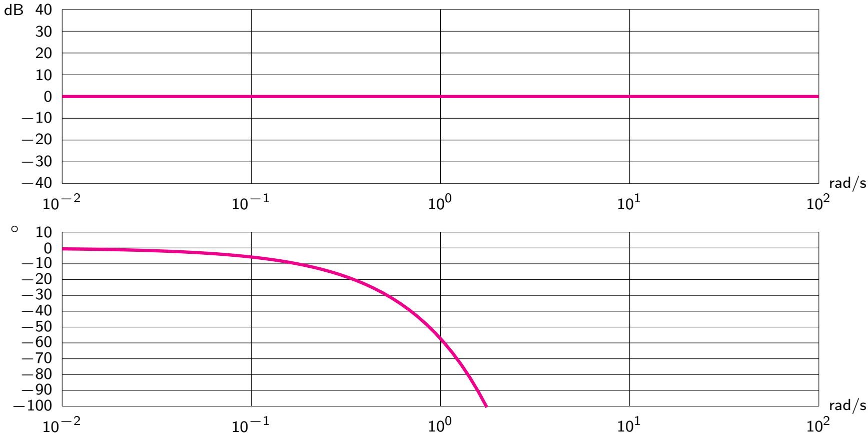
$$|P(j\omega)| = |e^{-j\omega\theta}| = \sqrt{\mathcal{R}_e^2[H(j\omega)] + \mathcal{I}_m^2[H(j\omega)]} = 1,$$

$$\arg[P(j\omega)] = \arctan\left(\frac{\mathcal{I}_m[P(j\omega)]}{\mathcal{R}_e[P(j\omega)]}\right) = -\omega\theta.$$

Note that in the previous formula, phase is expressed in radians !

Dead-time: Bode diagram[36]

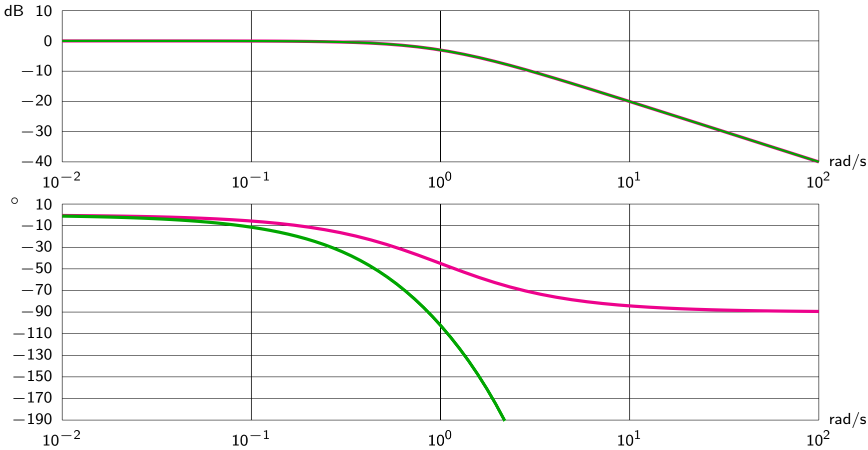
$$Y(s) = e^{-\theta s}X(s)$$



[36] $\theta = 1$

First order system with delay[37]: Bode diagram

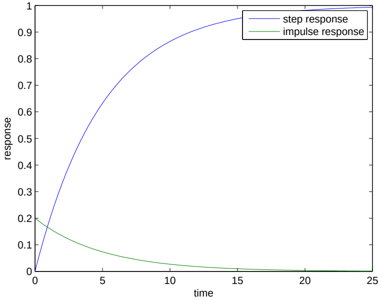
$$Y(s) = \frac{K_P}{T\,s+1}\,e^{-\theta s}\,X(s)$$

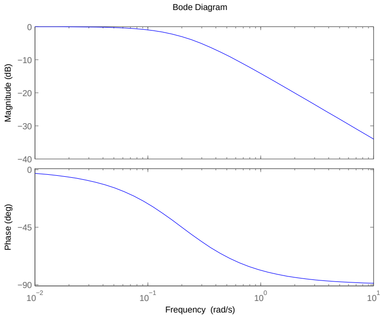


[37] $K_P = 1$, $T = 1$, $\theta = 0$ and $\theta = 1$

Step and impulse responses

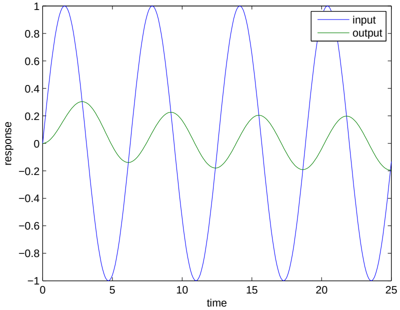
```
% pkg load control % uncomment if running octave with control toolbox
num = [0 1];
den = [5 1]; % Transfer function H(s) = 1/(5s + 1)
sys = tf(num,den);
t = 0:0.1:25;
y1 = step(sys,t);
y2 = impulse(sys,t);
figure(1), plot(t,[y1 y2])
legend('step response','impulse response')
xlabel('time'); ylabel('response')
figure(2), bode(sys)
```





Simulated responses

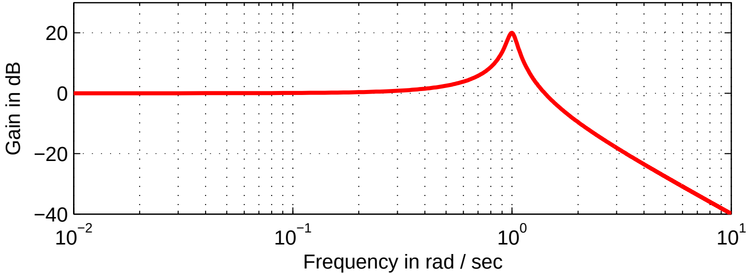
```
% pkg load control % uncomment if running octave with control toolbox

num = [0 1];
den = [5 1];
sys = tf(num,den); % Transfer function H(s) = 1/(5s + 1)
t = 0:0.1:25;
u = sin(t);
y = lsim(sys,u,t);
plot(t,[u; y'])
legend('input','output')
xlabel('time'); ylabel('response')
```

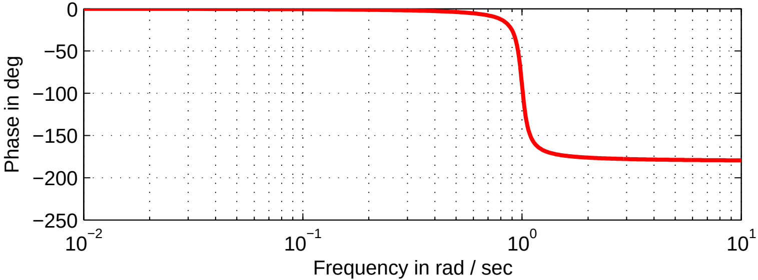


Bode and Nyquist plots

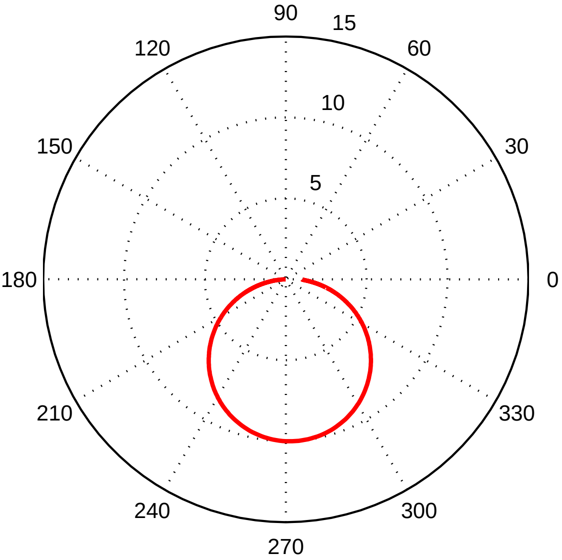
```
% pkg load control % uncomment if running octave with control toolbox

num = [0 0 1]; den = [1 0.1 1];  % Transfer function H(s) = 1/(s^2 + 0.1s + 1)
sys = tf(num,den);
w = 0:0.001:10;
[amplitude,phase] = bode(sys,w);
amplitude = reshape(amplitude,[length(amplitude) 1]);
phase = reshape(phase,[length(phase) 1]);

% Bode diagram
figure(1)
subplot(211)
semilogx(w,20*log10(amplitude),'r','Linewidth',2);
grid on; axis([0.01 10 -40 30]);
ylabel('Gain in dB'); xlabel('Frequency in rad / sec');
subplot(212)
semilogx(w,phase ,'r','Linewidth',2);
grid on; axis([0.01 10 -250 0]);
ylabel('Phase in deg'); xlabel('Frequency in rad / sec');

% Nyquist diagram
figure(2)
h1 = polar(phase*(pi/180), amplitude );
set(h1 ,'color','r','linewidth',2);
grid
```

Bode plots


20
0
−20
−40
Gain in dB
10^{-2}
10^{-1}
10^{0}
10^{1}
Frequency in rad / sec


0
−50
−100
−150
−200
−250
Phase in deg
10^{-2}
10^{-1}
10^{0}
10^{1}
Frequency in rad / sec

Nyquist plots



Partial fractions

Note the partial expansion of

$$P(s) = \frac{5s+3}{s^3+3s^2-4} = \frac{5s+3}{(s-1)(s+2)^2}$$
$$= \frac{-\frac{8}{9}}{s+2} + \frac{\frac{7}{3}}{(s+2)^2} + \frac{\frac{8}{9}}{s-1}$$

```
num = [5 3];          % 5s + 3
den = [1 3 0 -4];     % s^3 + 3s^2 - 4
[R,P] = residue(num,den)
```

```
R =

   -0.8889
    2.3333
    0.8889

P =

    -2
    -2
     1
```

Laplace transforms

```
% pkg load symbolic % uncomment for use with octave with symbolic toolbox
syms t
f = exp(-t/2) + sin(t);
laplace(f)

syms s
F = 1/(s^2 - 1);
ilaplace(F)
```

```
ans =

1/(s + 1/2) + 1/(s^2 + 1)

ans =

exp(t)/2 - exp(-t)/2
```

One-sided Laplace transforms		
$f(t)$	$\mathcal{L}[f(t)u(t)]$	ROC
$\delta(t)$	1	$\mathbb{C}$
1	$1/s$	$\mathcal{R}_e[s] > 0$
e^{-at}	$\dfrac{1}{s+a}$	$\mathcal{R}_e[s] > -a$
t^n	$\dfrac{n!}{s^{n+1}}$	$\mathcal{R}_e[s] > 0$
$\sin(at)$	$\dfrac{a}{s^2+a^2}$	$\mathcal{R}_e[s] > 0$
$\cos(at)$	$\dfrac{s}{s^2+a^2}$	$\mathcal{R}_e[s] > 0$

One-sided Laplace transforms

$f(t)$	$\mathcal{L}[f(t)u(t)]$	ROC
$t^n e^{-at}$	$\frac{n!}{(s+a)^{n+1}}$	$\mathcal{R}_e[s] > -a$
$(1 - e^{-at})$	$\frac{a}{s(s+a)}$	$\mathcal{R}_e[s] > 0$
$e^{-\sigma t} \sin at$	$\frac{a}{(s+\sigma)^2 + a^2}$	$\mathcal{R}_e[s] > -\mathcal{R}_e[\sigma]$
$e^{-\sigma t} \cos at$	$\frac{s+\sigma}{(s+\sigma)^2 + a^2}$	$\mathcal{R}_e[s] > -\mathcal{R}_e[\sigma]$

Basic properties of one-sided Laplace transforms		
Properties	$f(t)$	$F(s)$
Causal functions and constants	$\alpha f(t),\ \beta g(t)$	$\alpha F(s),\ \beta G(s)$
Linearity	$\alpha f(t) + \beta g(t)$	$\alpha F(s) + \beta G(s)$
Time shifting	$f(t-\tau)$	$e^{-\tau s} F(s)$
Frequency shifting	$e^{-\alpha t} f(t)$	$F(s+\alpha)$
Multiplication by t	$tf(t)$	$-\dfrac{dF(s)}{ds}$

Basic properties of one-sided Laplace transforms		
Properties	$f(t)$	$F(s)$
Convolution	$f(t) * g(t)$	$F(s)G(s)$
Derivative	$\frac{df(t)}{dt}$	$sF(s) - f(0^-)$
Integral	$\int_{0^-}^{t} f(\bar{t})d\bar{t}$	$\frac{F(s)}{s}$
Expansion / contraction	$f(at), (a \neq 0)$	$\frac{1}{a}F\left(\frac{s}{a}\right)$

Basic properties of one-sided Laplace transforms	
Properties	$f(t)$ and $F(s)$
Initial value	$f(0^+) = \lim_{s\to\infty} sF(s)$
Final value f	$\lim_{t\to\infty} f(t) = \lim_{s\to 0} sF(s)$

5. Fourier frequency analysis

Notations

- Scalar: $x = 1$
- Vector: $\mathbf{x} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$
- Unless specified otherwise, $\mathbf{x}$ is a column vector
- Unless specified otherwise, $\mathbf{x}^T$ is a row vector
- Matrix: $\mathbf{X} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$

Projection onto a subspace of $\mathbb{R}^m$

Inner product in $\mathbb{R}^m$

The inner, scalar or dot product of two vectors $\mathbf{x} = [x_1 \ \cdots \ x_m]^T$ and $\mathbf{y} = [y_1 \ \cdots \ y_m]^T \in \mathbb{R}^m$ is defined as

$$\mathbf{x}.\mathbf{y} = \sum_{i=1}^{m} x_i \, y_i = \mathbf{x}^T \mathbf{y}.$$

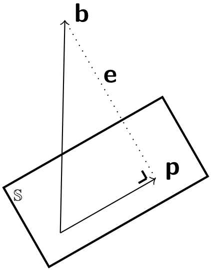
In particular, $\|\mathbf{x}\|^2 = \mathbf{x}.\mathbf{x}$ where $\|\mathbf{x}\|$ is the norm or length of $\mathbf{x}$.

Orthogonal vectors in $\mathbb{R}^m$

The vector $\mathbf{x}$ and $\mathbf{y}$ are orthogonal if their inner product, i.e.

$$\mathbf{x}.\mathbf{y} = 0.$$

Projection onto a subspace of $\mathbb{R}^m$



Consider n linearly independent column vectors

$$\mathbf{a}_1, \cdots, \mathbf{a}_n \in \mathbb{R}^m.$$

These vectors span an n dimensional subspace $\mathbb{S}$ of $\mathbb{R}^m$. They therefore form a basis of the subspace $\mathbb{S}$.

Consider a vector $\mathbf{b} \in \mathbb{R}^m$.

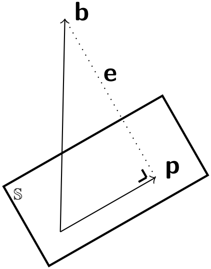
Find the vector $\mathbf{p} \in \mathbb{S}$ closest to the vector $\mathbf{b}$ in the least square sense. The vector p is the projection of b onto the subspace $\mathbb{S}$.

By definition, the vector $\mathbf{p} \in \mathbb{S}$ is a linear combination of the vectors $\mathbf{a}_i$, i.e.

$$\mathbf{p} = \hat{x}_1\,\mathbf{a}_1 + \cdots + \hat{x}_n\,\mathbf{a}_n = \mathbf{A}\,\hat{\mathbf{x}}$$

where $\mathbf{A} = [\mathbf{a}_1 \ \cdots \ \mathbf{a}_n] \in \mathbb{R}^{m \times n}$ and $\hat{\mathbf{x}} = [\hat{x}_1 \ \cdots \ \hat{x}_n]^T$. The coefficients $\hat{x}_i \in \mathbb{R}$ have to be determined.

Projection onto a subspace of $\mathbb{R}^m$



By definition, the vector $\mathbf{p} \in \mathbb{S}$ is a linear combination of the vectors $\mathbf{a}_i$, i.e.

$$\mathbf{p} = \hat{x}_1\,\mathbf{a}_1 + \cdots + \hat{x}_n\,\mathbf{a}_n = \mathbf{A}\,\hat{\mathbf{x}}$$

where $\mathbf{A} = [\mathbf{a}_1 \;\cdots\; \mathbf{a}_n] \in \mathbb{R}^{m\times n}$ and $\hat{\mathbf{x}} = [\hat{x}_1 \;\cdots\; \hat{x}_n]^T$. The coefficients $\hat{x}_i \in \mathbb{R}$ have to be determined.

The error $\mathbf{e} = \mathbf{b} - \mathbf{p} = \mathbf{b} - \mathbf{A}\,\hat{\mathbf{x}}$ is orthogonal to $\mathbf{p}$

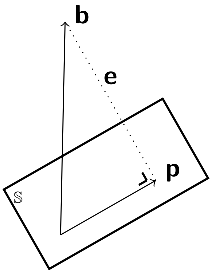
$$\mathbf{p}^T\mathbf{e} = \hat{\mathbf{x}}^T\mathbf{A}^T(\mathbf{b} - \mathbf{A}\,\hat{\mathbf{x}}) = 0.$$

This implies

$$\mathbf{A}^T\mathbf{b} = \mathbf{A}^T\mathbf{A}\,\hat{\mathbf{x}},\;\; \hat{\mathbf{x}} = [\mathbf{A}^T\mathbf{A}]^{-1}\mathbf{A}^T\mathbf{b}$$

$$\text{and } \mathbf{p} = \mathbf{A}\,\hat{\mathbf{x}} = \mathbf{A}[\mathbf{A}^T\mathbf{A}]^{-1}\mathbf{A}^T\mathbf{b}.$$

Projection onto a subspace of $\mathbb{R}^m$



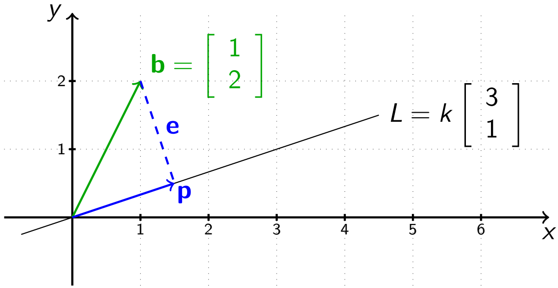
Suppose the vectors $\{\mathbf{a}_i\}$ form an orthonormal basis of $\mathbb{S}$, i.e.

$$\mathbf{a}_i \cdot \mathbf{a}_j = \mathbf{a}_i^T \mathbf{a}_j = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \Longrightarrow \mathbf{A}^T \mathbf{A} = \mathbf{I}.$$

The coordinates of $\mathbf{p}$ in the basis of $\mathbb{S}$ become

$$\hat{\mathbf{x}} = \mathbf{A}^T \mathbf{b} \text{ or } \hat{x}_i = \mathbf{a}_i^T \mathbf{b} = \mathbf{a}_i \cdot \mathbf{b}.$$

Projection on a line in $\mathbb{R}^2$



Projection of $\mathbf{b} \in \mathbb{R}^2$ on the line L, 1 dimensional subspace of $\mathbb{R}^2$. An orthonormal basis of L is $\mathbf{a} = [3/\sqrt{10}, 1/\sqrt{10}]^T$. The coordinates of the projection $\mathbf{p}$ in the orthonormal basis are

$$\hat{x} = \mathbf{a}^T\mathbf{b} = \mathbf{a} \cdot \mathbf{b} = 5/\sqrt{10} = \sqrt{10}/2 \Longrightarrow \mathbf{p} = \hat{x}\,\mathbf{a} = [3/2, 1/2]^T.$$

Objective

How can one represent a function $f(t)$ on an interval of length T using n individual basis functions

$$\{\phi_1(t), \cdots, \phi_n(t)\} \text{ ?}$$

We will examine the approximative representation of the function $f(t)$ as a linear combination of the basis functions, i.e.

$$f(t) \approx \sum_{i=1}^{n} c_i \, \phi_i(t).$$

Once the basis functions $\phi_i(t)$ have been selected, the task becomes **finding** the expansion coefficients c_i.

Signal space $\mathbb{L}_2(T)$

Signal space $\mathbb{L}_2(T)$

The set of real or complex valued functions of time defined on an interval of length T that are square integrable form the **signal space** $\mathbb{L}_2(T)$ with norm

$$\|s(t)\|^2 = \int_T s(t)s^\star(t)dt = \int_T |s(t)|^2 dt.$$

Inner product on $\mathbb{L}_2(T)$

The inner product of two function $s(t)$ and $r(t) \in \mathbb{L}_2(T)$ is

$$< s, r > \; = \int_T s(t)r^\star(t)dt, \text{ in particular } < s, s > \; = \|s(t)\|^2.$$

Signal space $\mathbb{L}_2(T)$

Orthogonal functions

Two functions $s(t)$ and $r(t)$ are **orthogonal** if their inner product is zero, i.e.

$$< s, r > \; = \int_T s(t) r^\star(t) dt = 0.$$

Orthogonal basis of a subspace of $\mathbb{L}_2(T)$

The functions $\{\phi_i(t)\} \in \mathbb{L}_2(T)$ $(i = 1, \cdots, n)$ form an **orthogonal basis** of a subspace $\mathbb{F} \subset \mathbb{L}_2(T)$ if

$$< \phi_i, \phi_j > \begin{cases} = 0, & \text{for } i \neq j, \\ \neq 0, & \text{for } i = j. \end{cases}$$

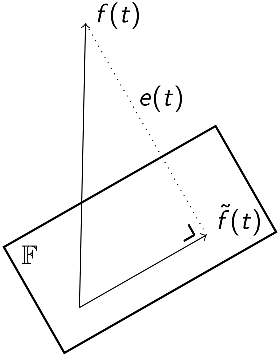
Moreover, if $< \phi_i, \phi_i > \; = 1$ for $i = 1, \cdots, n$ then the base is **orthonormal**.

Projection on a subspace of $\mathbb{L}_2(T)$

Given n functions $\{\phi_1(t), \cdots, \phi_n(t)\}$ forming a basis of the subspace $\mathbb{F} \subset \mathbb{L}_2(T)$.

Consider a function $f(t) \in \mathbb{L}_2(T)$.

Find the function $\tilde{f}(t) \in \mathbb{F}$ closest to $f(t)$ in the least square sense. The function $\tilde{f}(t)$ is in fact the projection of $f(t)$ on $\mathbb{F} \subset \mathbb{L}_2(T)$.



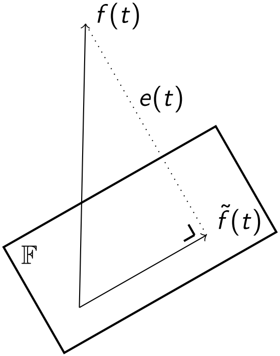
The function $\tilde{f}(t) \in \mathbb{F}$ is a linear combination of the vectors $\{\phi_i(t)\}$, i.e. $\tilde{f}(t) = \sum_{i=1}^{n} c_i\, \phi_i(t)$ where the coefficients $c_i \in \mathbb{C}$ $(i = 1, \cdots, n)$ need to be determined.

The approximation error $e(t) = f(t) - \tilde{f}(t)$ is orthogonal to $\tilde{f}(t)$. We have that

$$\begin{aligned} < e, \tilde{f} > &= < f - \tilde{f}, \tilde{f} >, \\ &= < f, \tilde{f} > - < \tilde{f}, \tilde{f} >, \\ &= 0 \end{aligned}$$

$$\Rightarrow < f, \tilde{f} > = < \tilde{f}, \tilde{f} > .$$

Projection on a subspace of $\mathbb{L}_2(T)$



The error $e(t) = f(t) - \tilde{f}(t)$ is orthogonal to $\tilde{f}(t) = \sum_{i=1}^{n} c_i\, \phi_i(t)$. We have that

$$
\begin{aligned}
< f, \tilde{f} > &= < \tilde{f}, \tilde{f} > \\
\sum_{i=1}^{n} c_i < f, \phi_i > &= \sum_{i=1}^{n} c_i < \phi_i, \tilde{f} > \cdot
\end{aligned}
$$

This implies

$$
< f, \phi_i > \; = \; < \phi_i, \tilde{f} > \; = \sum_{j=1}^{n} c_j < \phi_i, \phi_j > \cdot
$$

Conclusion

$$\boxed{< f, \phi_i > \; = \sum_{j=1}^{n} c_j < \phi_i, \phi_j >}$$

There are n complex equations with n complex unknowns. It is therefore possible to determine the coefficients $c_i \in \mathbb{C}$.

Projection on a subspace of $\mathbb{L}_2(T)$

Orthogonal basis

It is advantageous to work with an orthogonal basis as

$$< \phi_i, \phi_j > \; = 0, \;\; \text{for } i \neq j.$$

The approximation equation becomes $< f, \phi_i > \; = c_i < \phi_i, \phi_i >$, i.e.

$$\boxed{c_i = \frac{< f, \phi_i >}{< \phi_i, \phi_i >}.}$$

Orthonormal basis

If the basis is orthonormal, we obtain $\boxed{c_i = \; < f, \phi_i > \cdot}$

Projection on a subspace of $\mathbb{L}_2(T)$

Given n functions $\{\phi_1(t), \cdots, \phi_n(t)\}$ forming an **orthonormal basis** of the subspace $\mathbb{F} \subset \mathbb{L}_2(T)$, i.e.

$$< \phi_i, \phi_j > = \int_T \phi_i(t)\phi_j^\star(t)dt = \begin{cases} 0, & \text{for } i \neq j, \\ 1, & \text{for } i = j. \end{cases}$$

Consider a function $f(t) \in \mathbb{L}_2(T)$. We are interested in $\tilde{f}(t) \in \mathbb{F}$ closest to $f(t)$ in the mean square sense. The function $\tilde{f}(t)$ is the projection of $f(t)$ on $\mathbb{F} \subset \mathbb{L}_2(T)$ and can be written as

$$\tilde{f}(t) = \sum_{i=1}^{n} c_i \, \phi_i(t)$$

with

$$\begin{aligned} c_i &= < f, \phi_i >, \\ &= \int_T f(t)\phi_i^\star(t)dt. \end{aligned}$$

Approximation error

It is possible to compute the approximation error e

$$
\begin{aligned}
\|e\|^2 = \, < e, e > \quad &= \quad < f - \tilde{f}, f - \tilde{f} > \\
&= \quad < f - \tilde{f}, f > - < f - \tilde{f}, \tilde{f} > \\
&= \quad < f - \tilde{f}, f >
\end{aligned}
$$

knowing that $< f - \tilde{f}, \tilde{f} > \; = \; < e, \tilde{f} > \; = 0.$ This yields

$$
\|e\|^2 = \; < f, f > - < \tilde{f}, f > \; = \; \|f\|^2 - < \tilde{f}, \tilde{f} >
$$

given that $< \tilde{f}, f - \tilde{f} > \; = 0 \; \Rightarrow < \tilde{f}, f > \; = \; < \tilde{f}, \tilde{f} > \cdot$
Finally, one obtains

$$
\boxed{\|e\|^2 = \|f\|^2 - \sum_{i=1}^{n} \sum_{j=1}^{n} c_i c_j^\star < \phi_i, \phi_j > \cdot}
$$

Approximation error

Orthogonal basis

$$\boxed{\|e\|^2 = \|f\|^2 - \sum_{i=1}^{n} |c_i|^2 < \phi_i, \phi_i >}$$

Orthonormal basis

$$\boxed{\|e\|^2 = \|f\|^2 - \sum_{i=1}^{n} |c_i|^2}$$

Parseval's identity

If the number basis functions n increases, one might expect that the approximation error $e \to 0$. This yields

$$\|f\|^2 = \int_T |f(t)|^2 dt = \sum_{i=1}^{n} \sum_{j=1}^{n} c_i c_j^\star < \phi_i, \phi_j > \cdot$$

This is **Parseval's identity**.

With an **orthogonal** basis, this yields

$$\int_T |f(t)|^2 dt = \sum_{i=1}^{n} |c_i|^2 < \phi_i, \phi_i > \cdot$$

With an **orthonormal** basis, one has

$$\boxed{\int_T |f(t)|^2 dt = \sum_{i=1}^{n} |c_i|^2.}$$

Choice of the basis function

- There exist a multitude of orthogonal functions: Legendre and Laguerre polynomials, etc.
- The choice of the type of basis function depends on the nature of the problem that is considered.
- It might even be possible that a non orthogonal basis is best suited to represent a given function.
- In the sequel, the basis that is considered is the **Fourier basis, i.e. complex exponentials**.

Fourier basis

Fourier basis

The complex Fourier basis is composed of the complex exponentials

$$F_k(t) = e^{jk\omega_0 t}, \; k \text{ integer and } \omega_0 = \frac{2\pi}{T}.$$

The Fourier basis functions are orthonormal over a period T, i.e.

$$< F_k, F_l > = \frac{1}{T} \int_{t_0}^{t_0+T} F_k(t) F_l^{\star}(t) dt = \begin{cases} 1, & k = l, \\ 0, & k \neq l. \end{cases}$$

Fourier series representation

Fourier series representation

The Fourier series representation of a periodic signal $f(t)$, of period T, is given by an infinite sum of weighted complex exponentials with frequencies multiples of the signal's fundamental frequency ω_0, i.e.

$$f(t) = \sum_{k=-\infty}^{\infty} c_k \, F_k(t) = \sum_{k=-\infty}^{\infty} c_k \, e^{jk\omega_0 t}, \;\; \omega_0 = \frac{2\pi}{T}$$

where the **Fourier coefficients** are found according to

$$c_k \; = \; < f, F_k > \; = \; \frac{1}{T} \int_{t_0}^{t_0+T} f(t) F_k^\star dt = \frac{1}{T} \int_{t_0}^{t_0+T} f(t) e^{-jk\omega_0 t} dt$$

for $k = 0, \pm 1, \pm 2, \cdots$ and any t_0. The Fourier coefficients c_k are **complex** and can be obtained from any period of $f(t)$.

Convergence of Fourier series

Convergence

The Fourier series of a piecewise smooth (continuous or discontinuous) periodic signal $f(t)$ converges for all values of t.

The mathematician **Dirichlet** showed that for the Fourier series to converge[38] to the periodic signal $f(t)$, the signal should satisfy the following sufficient (not necessary) conditions over a period:

- be absolutely integrable,
- have a finite number of maxima and minima,
- have a finite number of discontinuities.

[38] We can speak of pointwise convergence.

Convergence of Fourier series

Convergence

The infinite series equals $f(t)$ at every continuity point and equals the average

$$0.5(f(t_0^+) + f(t_0^-))$$

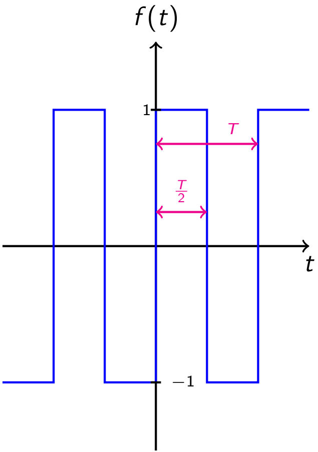
of the right limit $f(t_0^+)$ and the left limit $f(t_0^-)$ at every discontinuity point t_0. Here

$$f(t_0^+) = \lim_{\substack{t \to t_0 \\ >}} f(t) \text{ and } f(t_0^-) = \lim_{\substack{t \to t_0 \\ <}} f(t).$$

If $f(t)$ is continuous everywhere, the series converges absolutely and uniformly.

Although the Fourier series converges to the arithmetic average at discontinuities, it can be observed that there is some ringing before and after the discontinuity points. This is called the Gibbs phenomenon.

Fourier series: example



The Fourier coefficients c_k are

$$\begin{aligned}
c_k &= \frac{1}{T}\int_0^T f(t)e^{-jk\omega_0 t}dt \\
&= \frac{1}{T}\left(\int_0^{\frac{T}{2}} 1e^{-j2k\pi\frac{t}{T}}dt + \int_{\frac{T}{2}}^{T} -1e^{-j2k\pi\frac{t}{T}}dt\right) \\
&= -\frac{1}{T}\frac{T}{j2\pi k}\left([e^{-j2k\pi\frac{t}{T}}]_0^{\frac{T}{2}} - [e^{-j2k\pi\frac{t}{T}}]_{\frac{T}{2}}^{T}\right) \\
&= -\frac{1}{j2\pi k}\left(e^{-jk\pi} - 1 - \underbrace{e^{-j2k\pi}}_{=1} + e^{-jk\pi}\right) \\
&= -\frac{1}{j\pi k}\left(e^{-jk\pi} - 1\right)
\end{aligned}$$

$$\boxed{c_k = 0 \text{ for } k \text{ even or zero and } c_k = \frac{2}{j\pi k} \text{ for } k \text{ odd.}}$$

Fourier series: example

$$\begin{aligned} f(t) &= \sum_{k=-\infty}^{\infty} c_k \, e^{j2k\pi \frac{t}{T}}, \, c_k = \begin{cases} 0, & \text{k even or zero} \\ \dfrac{2}{j\pi k}, & \text{k odd} \end{cases} \\ &= \frac{2}{j\pi} \left(\cdots - \frac{1}{5} e^{-5j2\pi \frac{t}{T}} - \frac{1}{3} e^{-3j2\pi \frac{t}{T}} - e^{-j2\pi \frac{t}{T}} \right. \\ &\qquad \left. e^{j2\pi \frac{t}{T}} + \frac{1}{3} e^{3j2\pi \frac{t}{T}} + \frac{1}{5} e^{5j2\pi \frac{t}{T}} + \cdots \right). \end{aligned}$$

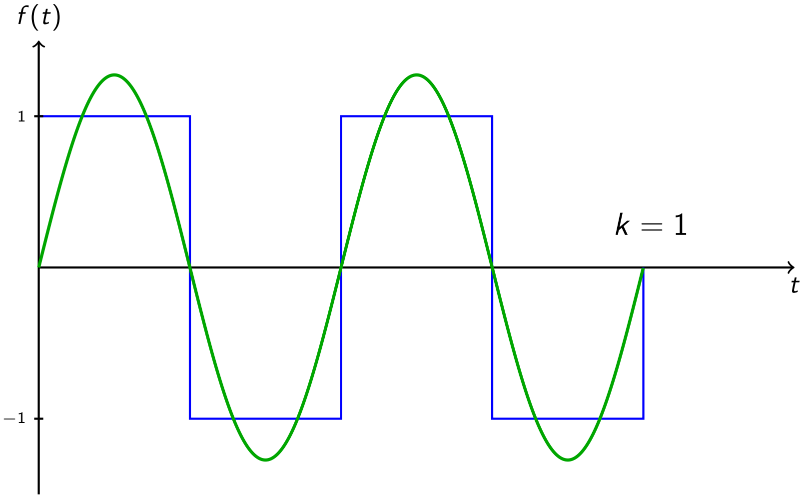
Note that

$$\frac{1}{k} \left(e^{kj2\pi \frac{t}{T}} - e^{-kj2\pi \frac{t}{T}} \right) = \frac{2j}{k} \sin(2k\pi \frac{t}{T}).$$

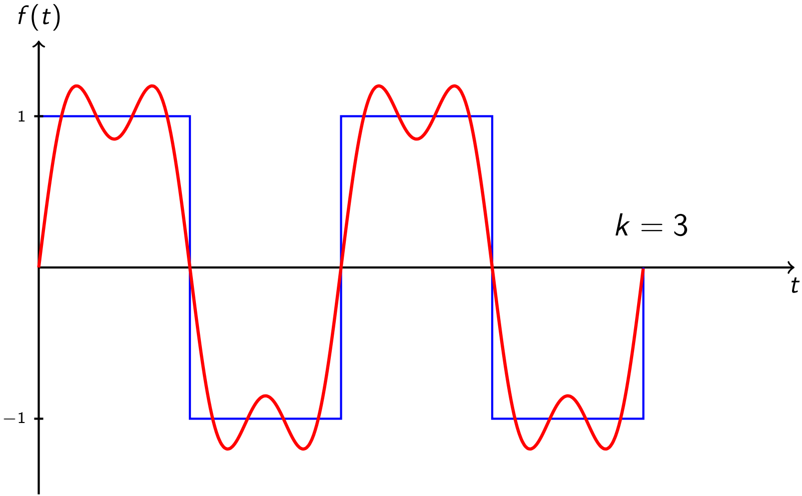
Therefore

$$\boxed{f(t) = \sum_{\substack{k>0 \\ \text{odd}}}^{\infty} \frac{4}{k\pi} \sin\left(2k\pi \frac{t}{T}\right).}$$

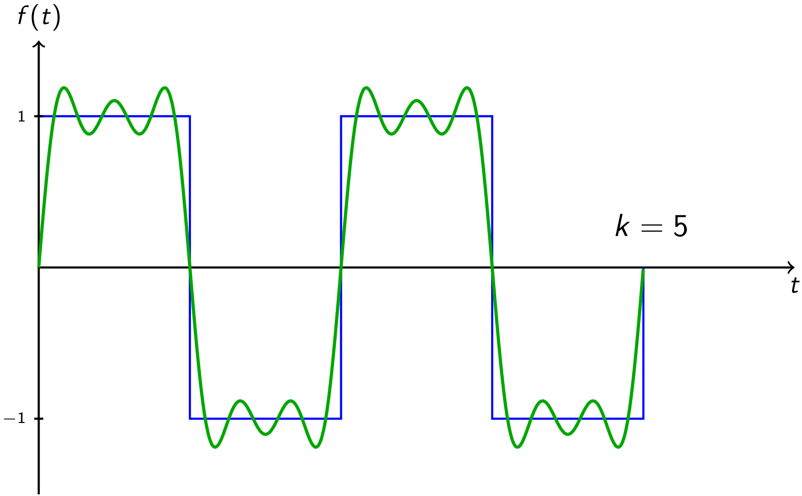
Fourier series representation: example



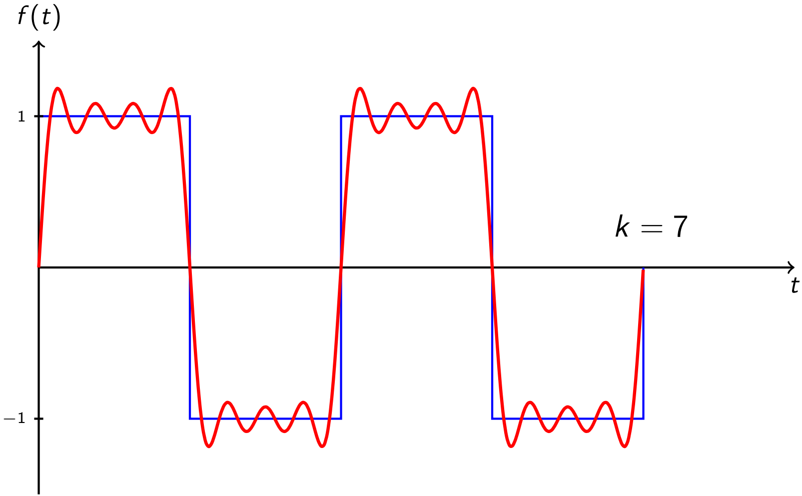
Fourier series representation: example



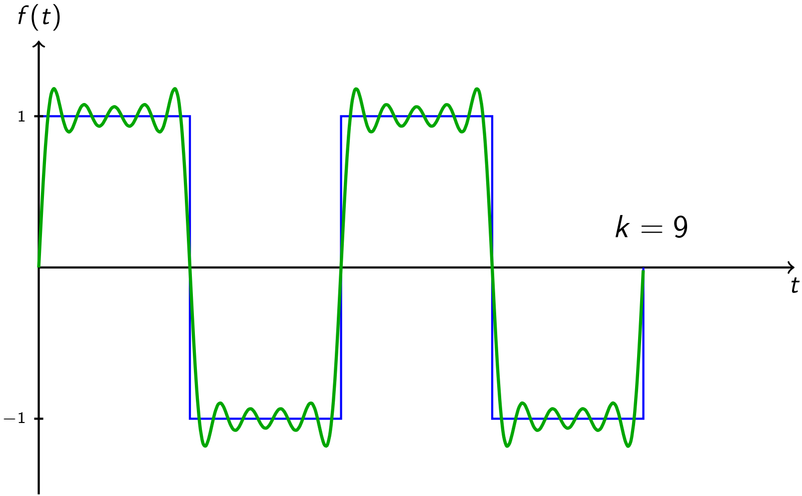
Fourier series representation: example



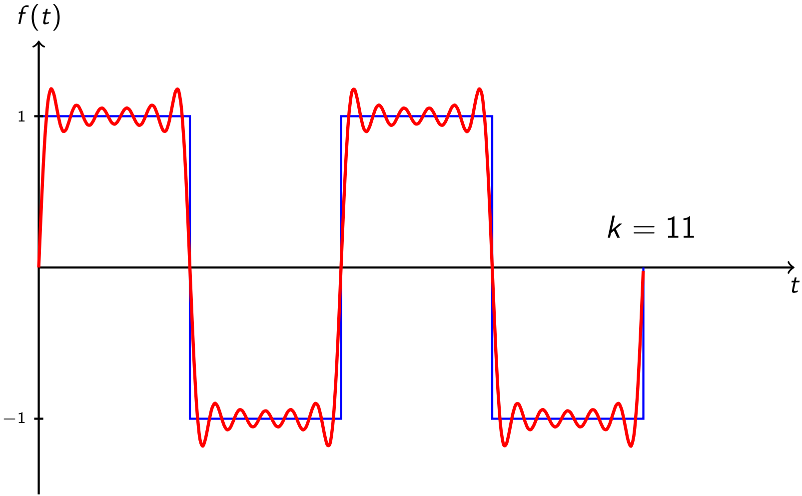
Fourier series representation: example



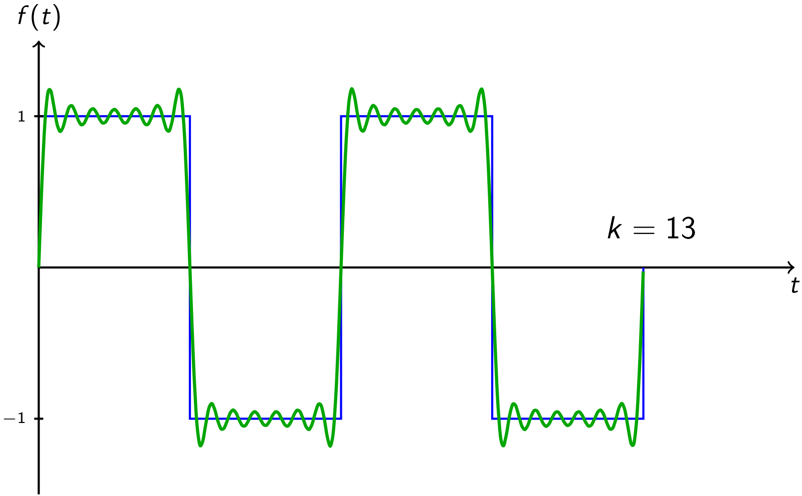
Fourier series representation: example



Fourier series representation: example



Fourier series representation: example



Fourier series representation

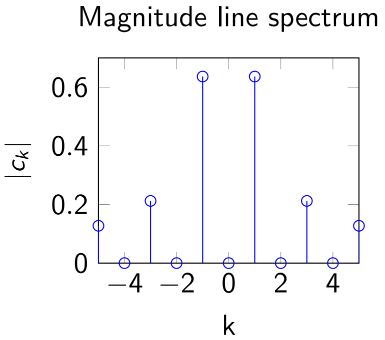
Representation

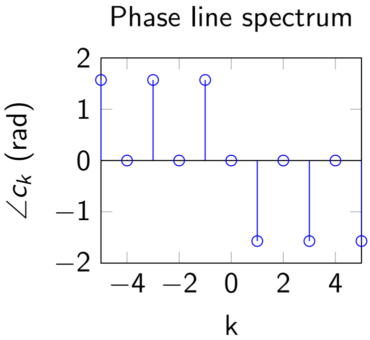
A periodic signal $f(t)$ of period T, is represented in the frequency domain by its

- **magnitude line spectrum**: magnitude $|c_k|$ versus $k\omega_0$,
- **phase line spectrum**: argument $\angle c_k$ versus $k\omega_0$.

The power line spectrum $|c_k|^2$ versus $k\omega_0$ of $f(t)$ displays the distribution of the power of the signal over frequency.

Fourier series representation: example





Fourier series representation

Properties of the Fourier series

For a signal $f(t)$ of period T, one can always write

$$\begin{aligned} f(t) &= \sum_{k=-\infty}^{\infty} c_k\, e^{jk\omega_0 t} \\ &= c_0 + \sum_{k=-\infty}^{-1} c_k\, e^{jk\omega_0 t} + \sum_{k=1}^{\infty} c_k\, e^{jk\omega_0 t} \\ &= c_0 + \sum_{k=1}^{\infty} \left[c_k\, e^{jk\omega_0 t} + c_{-k}\, e^{-jk\omega_0 t} \right] \end{aligned}$$

Fourier series representation

DC component and harmonics

- $c_0 = \frac{1}{T} \int_{t_0}^{t_0+T} f(t)dt$ is the **DC or average component** of $f(t)$.
- $c_1 e^{j\omega_0 t} + c_{-1} e^{-j\omega_0 t}$ is the **fundamental component** of $f(t)$.
- $c_k e^{jk\omega_0 t} + c_{-k} e^{-jk\omega_0 t}$ is the **kth harmonic component**[39] of $f(t)$.

[39] $k > 1$

Fourier series representation: real-valued functions

Symmetry of line spectra

For a **real-valued** periodic signal $f(t)$, of period T, represented in the frequency domain by the Fourier coefficients $\{c_k = |c_k|e^{j\angle c_k}\}$ at harmonic frequencies $\{k\omega_0 = \frac{2k\pi}{T}\}$, we have that

$$c_k = \frac{1}{T}\int_{t_0}^{t_0+T} f(t)e^{-jk\omega_0 t}dt = c_{-k}^{\star}.$$

This yields

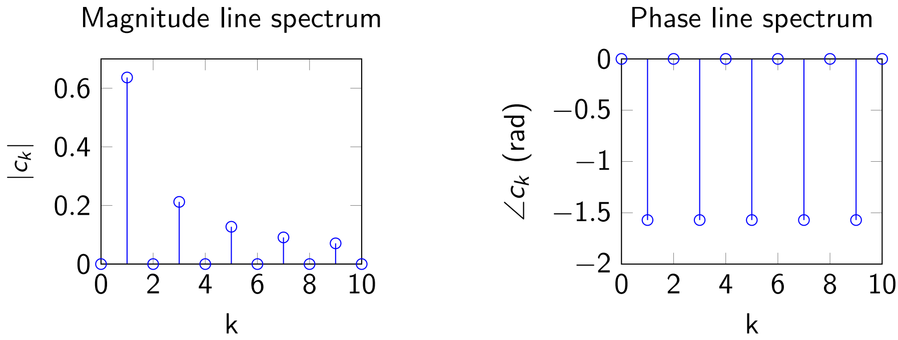
$$c_{-k} = |c_{-k}|e^{j\angle c_{-k}} = c_k^{\star} = |c_k|e^{-j\angle c_k}$$

and therefore

- $|c_k| = |c_{-k}|$, i.e. the magnitude line spectrum is even,
- $\angle c_k = -\angle c_{-k}$, i.e. the phase line spectrum is odd.

For real-valued signals we only need to display the spectra for $k \geq 0$.

Fourier series and real-valued functions: example



Fourier series representation: real-valued functions

The Fourier series of a **real-valued** function $f(t)$ of period T is

$$\begin{aligned} f(t) &= \sum_{k=-\infty}^{\infty} c_k e^{jk\omega_0 t} = \sum_{k=-\infty}^{\infty} |c_k| e^{j\angle c_k} e^{jk\omega_0 t} = \sum_{k=-\infty}^{\infty} |c_k| e^{j(k\omega_0 t + \angle c_k)} \\ &= c_0 + \sum_{k=1}^{\infty} |c_k| e^{j(k\omega_0 t + \angle c_k)} + |c_{-k}| e^{j(-k\omega_0 t + \angle c_{-k})} \\ &= c_0 + \sum_{k=1}^{\infty} |c_k| \left(e^{j(k\omega_0 t + \angle c_k)} + e^{-j(k\omega_0 t + \angle c_k)} \right) \\ &= c_0 + 2\sum_{k=1}^{\infty} |c_k| \cos(k\omega_0 t + \angle c_k) \end{aligned}$$

Using trigonometry[40], one obtains

$$f(t) = c_0 + 2\sum_{k=1}^{\infty} |c_k| (\cos(\angle c_k)\cos(k\omega_0 t) - \sin(\angle c_k)\sin(k\omega_0 t)).$$

[40] $\cos(\alpha+\beta) = \cos(\alpha)\cos(\beta) - \sin(\alpha)\sin(\beta)$

Fourier series representation: real-valued functions

DC component and harmonics

- $c_0 = \frac{1}{T} \int_{t_0}^{t_0+T} f(t)dt$ is the **DC or average component** of $f(t)$.
- $|c_1| \cos(\omega_0 t + \angle c_1)$ is the **fundamental component** of $f(t)$.
- $|c_k| \cos(k\omega_0 t + \angle c_k)$ is the **kth harmonic component**[41] of $f(t)$.

[41] $k > 1$

Fourier series representation: real-valued functions

The Fourier series of a **real-valued** function $f(t)$ of period T is

$$
\begin{aligned}
f(t) &= \sum_{k=-\infty}^{\infty} c_k\, e^{jk\omega_0 t} = c_0 + \sum_{k=-\infty}^{-1} c_k\, e^{jk\omega_0 t} + \sum_{k=1}^{\infty} c_k\, e^{jk\omega_0 t} \\
&= c_0 + \sum_{k=1}^{\infty} \left[c_k\, e^{jk\omega_0 t} + c_{-k}\, e^{-jk\omega_0 t} \right] \\
&= c_0 + \sum_{k=1}^{\infty} \left[c_k\, e^{jk\omega_0 t} + c_k^{\star}\, e^{-jk\omega_0 t} \right] \\
&= c_0 + \sum_{k=1}^{\infty} \left[(c_k + c_k^{\star}) \cos(k\omega_0 t) + j(c_k - c_k^{\star}) \sin(k\omega_0 t) \right] \\
&= c_0 + 2 \sum_{k=1}^{\infty} \left[\mathcal{R}_e[c_k] \cos(k\omega_0 t) - \mathcal{I}_m[c_k] \sin(k\omega_0 t) \right]
\end{aligned}
$$

Trigonometric Fourier series

Trigonometric Fourier series

The trigonometric Fourier series of a **real-valued** periodic signal $f(t)$ of period T is given by

$$f(t) = a_0 + \sum_{k=1}^{\infty} [a_k \cos(k\omega_0 t) + b_k \sin(k\omega_0 t)]$$

where

$$\begin{aligned} a_k &= \frac{1}{T} \int_0^T f(t) \cos(k\omega_0 t) dt, \;\; k = 0, 1, 2, \cdots \\ b_k &= \frac{1}{T} \int_0^T f(t) \sin(k\omega_0 t) dt, \;\; k = 1, 2, \cdots \end{aligned}$$

and a_0 is the DC or average component.

The basis functions that are used $\{\cos(k\omega_0 t), \sin(k\omega_0 t)\}$ are orthonormal.

Connection between the two forms of Fourier series

Connection between c_k and a_k, b_k

$$c_0 = a_0 = \frac{1}{T}\int_0^T f(t)dt$$

$$c_k = \frac{1}{2}(a_k - jb_k) \Rightarrow \begin{cases} |c_k| = \frac{1}{2}\sqrt{a_k^2 + b_k^2} \\ \angle c_k = -\arctan\left[\frac{b_k}{a_k}\right] \end{cases}$$

Proof 1: $2\,\mathcal{R}_e[c_k] = a_k$ and $-2\,\mathcal{I}_m[c_k] = b_k \;\Rightarrow\; a_k - jb_k = 2c_k$.

Proof 2: $a_k = 2|c_k|\cos(\angle c_k)$ and $b_k = -2|c_k|\sin(\angle c_k)$.

Fourier series representation

Properties of the Fourier series

Assume the Fourier series of a function $f(t)$ of period T is

$$f(t) = \sum_{k=-\infty}^{\infty} c_k\, e^{jk\omega_0 t}, \;\; \omega_0 = \frac{2\pi}{T}$$

where the Fourier coefficients are computed according to

$$c_k \; = \; \frac{1}{T} \int_{t_0}^{t_0+T} f(t) e^{-jk\omega_0 t} dt.$$

The Fourier series of the reflected function $f(-t)$ is

$$f(-t) = \sum_{k=-\infty}^{\infty} c_k\, e^{-jk\omega_0 t} = \sum_{k=-\infty}^{\infty} c_{-k}\, e^{jk\omega_0 t}.$$

Even real-valued functions

Properties of the Fourier series for real-value even functions

For a **real-valued even** function $f(t)$ of period T, one has

$$f(-t) = f(t) \Rightarrow c_k = c^{\star}_{-k} = c_{-k} \Rightarrow c_k = c^{\star}_k.$$

The Fourier coefficients $\{c_k\}$ are **real**. The Fourier series yields

$$\begin{aligned} f(t) &= c_0 + \sum_{k=1}^{\infty} \left[c_k\, e^{jk\omega_0 t} + c_{-k}\, e^{-jk\omega_0 t} \right], \\ &= c_0 + \sum_{k=1}^{\infty} c_k \left[e^{jk\omega_0 t} + e^{-jk\omega_0 t} \right], \\ &= c_0 + 2\sum_{k=1}^{\infty} c_k \cos(k\omega_0 t). \end{aligned}$$

The connection between both Fourier series is

$$\boxed{c_0 = a_0,\ b_k = 0 \text{ et } 2c_k = a_k.}$$

Odd real-valued functions

Properties of the Fourier series for real-value odd functions

For a **real-valued odd** function $f(t)$ of period T, one has

$$f(-t) = -f(t) \Rightarrow c_k = c^{\star}_{-k} = -c_{-k} \Rightarrow c_k = -c^{\star}_k.$$

The Fourier coefficients $\{c_k\}$ are **purely imaginary**. The Fourier series yields

$$\begin{aligned} f(t) &= \sum_{k=1}^{\infty} \left[c_k\, e^{jk\omega_0 t} + c_{-k}\, e^{-jk\omega_0 t}\right], \\ &= \sum_{k=1}^{\infty} c_k \left[\, e^{jk\omega_0 t} - \; e^{-jk\omega_0 t}\right], \\ &= 2j \sum_{k=1}^{\infty} c_k \sin(k\omega_0 t). \end{aligned}$$

The connection between both Fourier series is

$$\boxed{c_0 = 0,\; a_k = 0 \text{ et } 2jc_k = b_k.}$$

Fourier coefficients from Laplace

The computation of the c_k coefficients requires integration that for some signals can be rather complicated. The integration can be avoided whenever we know the Laplace transform of a period of the signal $f(t)$.

Fourier coefficients from Laplace transform

If the Laplace transform $F_T(s)$ of **one period** of the periodic signal $f(t)$ of period T is known, then the Fourier coefficients of $f(t)$ are given by

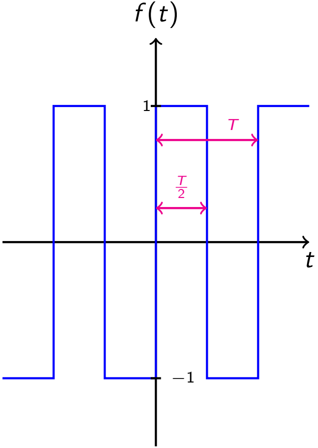
$$c_k = \frac{1}{T}\left(F_T(s)|_{s=jk\omega_0}\right),\ \omega_0 = \frac{2\pi}{T}.$$

Note that

$$F_T(s) = \mathcal{L}[f_T(t)]$$

with $f_T(t) = f(t)\left(u(t-t_0) - u(t-t_0-T)\right)$ and any t_0.

Fourier series representation: example



Note that

$$\begin{aligned} f_T(t) &= u(t) - 2u(t - \frac{T}{2}) + u(t - T) \text{ and} \\ F_T(s) &= \frac{1}{s}\left(1 - 2e^{-\frac{T}{2}s} + e^{-Ts}\right) \end{aligned}$$

The Fourier coefficients are

$$\begin{aligned} c_k &= \frac{1}{T}\frac{1}{s}\left(1 - 2e^{-\frac{T}{2}s} + e^{-Ts}\right)\Big|_{s=jk\omega_0=\frac{j2k\pi}{T}} \\ &= \frac{1}{j2k\pi}\left(1 - 2e^{-jk\pi} + e^{-j2k\pi}\right) \\ &= \frac{1}{jk\pi}\left(1 - e^{-jk\pi}\right) \end{aligned}$$

$$\boxed{c_k = 0 \text{ for } k \text{ even or zero and } c_k = \frac{2}{j\pi k} \text{ for } k \text{ odd.}}$$

Basic properties of Fourier series		
Properties	Time domain	Frequency domain
Signals of period T	$f(t),\ g(t)$	$c_k^f,\ c_k^g$
Linearity	$\alpha f(t) + \beta g(t)$	$\alpha c_k^f + \beta c_k^g$
Time shift	$f(t-\tau)$	$e^{-j\tau\omega_0} c_k^f$
Frequency shift	$e^{-jm\omega_0 t} f(t)$	c_{k+m}^f
Differentiation	$\dfrac{df(t)}{dt}$	$jk\omega_0\, c_k^f$

Basic properties of Fourier series		
Properties	Time domain	Frequency domain
Parseval's identity	$\frac{1}{T}\int_T \lvert f(t)\rvert^2 dt$	$\sum_k \lvert c_k^f\rvert^2$
Symmetry	real-valued $f(t)$	$\begin{cases} c_k^f = c_{-k}^{f\star} \\ \lvert c_k^f\rvert = \lvert c_{-k}^f\rvert, \\ \angle c_k^f = -\angle c_{-k} \end{cases}$
Simplification	real-valued even $f(t)$	$c_k^f = c_{-k}^{f\star} = c_{-k}^f$ $\{c_k\}$ real
Simplification	real-valued odd $f(t)$	$c_k^f = c_{-k}^{f\star} = -c_{-k}^f$ $\{c_k\}$ purely imaginary

Introduction

- The frequency representation of signals is a tool of great significance in signal processing, communications, and control theory.
- It would be nice to complete the Fourier representation of signals by extending it to **aperiodic signals**.
- **Idea**: to obtain the Fourier representation of aperiodic signals, we use the Fourier series representation in a limiting process, i.e. the aperiodic signal $f(t)$ is seen as a periodic signal of infinite period. Mathematically, this yields

$$f(t) = \lim_{T \to \infty} \tilde{f}(t)$$

where $\tilde{f}(t)$ is a function of period T.

Introduction

$$\tilde{f}(t) = \sum_{k=-\infty}^{\infty} c_k \, e^{jk\omega_0 t}, \;\; \omega_0 = \frac{2\pi}{T} \text{ avec } c_k \; = \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} \tilde{f}(t) e^{-jk\omega_0 t} dt$$

If $F(\omega_k) = T c_k$ with $\omega_k = k\omega_0 = \frac{2k\pi}{T}$ and $\Delta\omega = \omega_0 = \frac{2\pi}{T}$, one has

$$\begin{aligned} \tilde{f}(t) &= \sum_{k=-\infty}^{\infty} \frac{F(\omega_k)}{T} \, e^{j\omega_k t} = \sum_{k=-\infty}^{\infty} F(\omega_k) \, e^{j\omega_k t} \, \frac{\Delta\omega}{2\pi} \\ F(\omega_k) &= \int_{-\frac{T}{2}}^{\frac{T}{2}} \tilde{f}(t) e^{-j\omega_k t} dt \end{aligned}$$

When $T \to \infty$, $\omega_k \to \omega$, $\Delta\omega \to d\omega$, $\tilde{f}(t) \to f(t)$, the sum becomes an integral, the lines in the line spectrum get closer and, in the limit, a continuous spectrum is obtained.

Fourier transform

Fourier transform

The Fourier transform of a signal $f(t)$[42] is

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t}dt.$$

The inverse transform is

$$f(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} F(\omega)e^{j\omega t}d\omega.$$

Existence of the Fourier transform

The existence conditions of the Fourier transform are the Dirichlet conditions.

[42] that is not necessarily periodic

Fourier transform

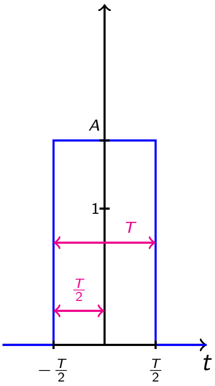
Interpretation

The Fourier transform of a signal $f(t)$ is the projection of $f(t)$ on a continuous complex exponential basis, i.e.

$$F(\omega) = \mathcal{F}[f(t)] = < f, e^{j\omega t} > \cdot$$

Fourier transform: example[43]

$f(t) = A\,\Pi(\frac{t}{T})$



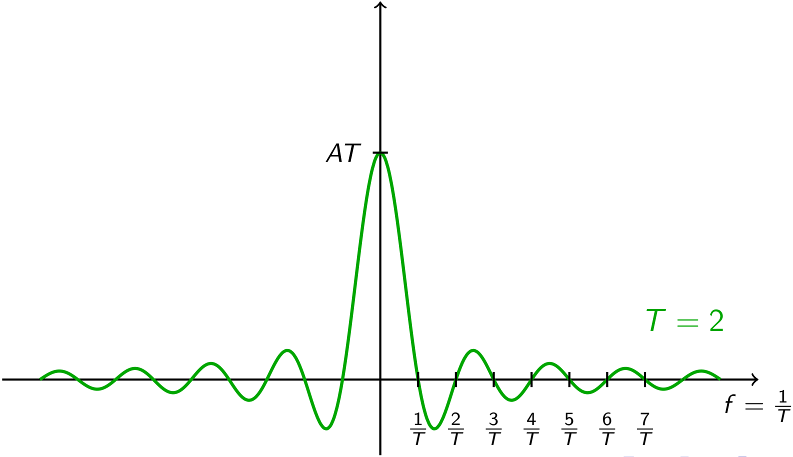
Given the definition of the Fourier transform, one obtains

$$\begin{aligned} F(\omega) &= \int_{-\infty}^{\infty} f(t)e^{-j\omega t}dt = \int_{-\frac{T}{2}}^{\frac{T}{2}} Ae^{-j\omega t}dt \\ &= \frac{A}{-j\omega}\left[e^{-j\omega t}\right]_{-\frac{T}{2}}^{\frac{T}{2}} \\ &= \frac{A}{-j\omega}\left(e^{-j\omega\frac{T}{2}} - e^{j\omega\frac{T}{2}}\right) = \frac{A}{-j\omega}\left(-2j\sin({}^{\omega T}/_{2})\right) \\ &= \frac{2A}{\omega}\sin({}^{\omega T}/_{2}) \\ &= AT\,\frac{\sin(\pi fT)}{\pi fT} = AT\,\mathrm{sinc}(fT) \end{aligned}$$

[43]The cardinal sine is defined as $\mathrm{sinc}(t) = \frac{\sin(\pi t)}{\pi t}$.

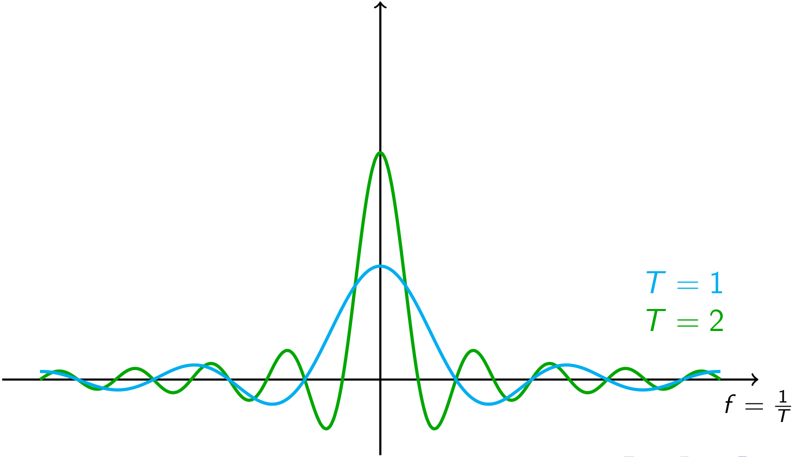
Cardinal sine

$$F(f) = AT\left(\frac{\sin(\pi f T)}{\pi f T}\right)$$



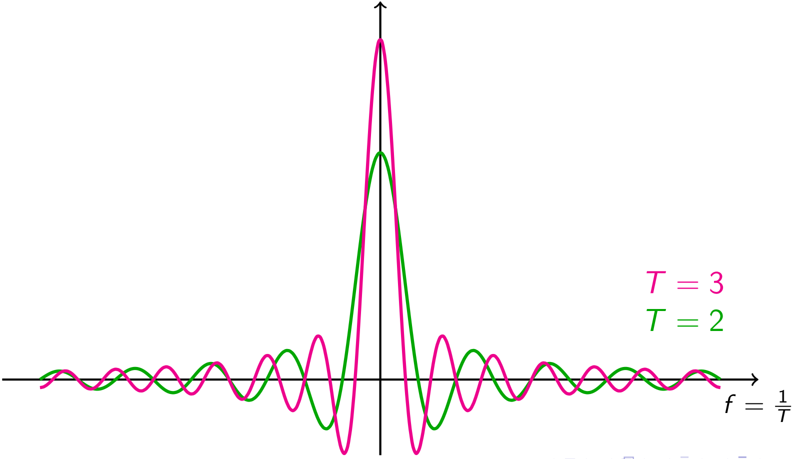
Cardinal sine

$$F(f) = AT\left(\frac{\sin(\pi f T)}{\pi f T}\right)$$



Cardinal sine

$$F(f) = AT \left(\frac{\sin(\pi f T)}{\pi f T} \right)$$



Fourier transforms from Laplace transform

Fourier transforms from Laplace transform

If the region of convergence (ROC) of $F(s) = \mathcal{L}[f(t)]$ contains the $j\omega$-axis, so that $F(s)$ can be defined for $s = j\omega$, then

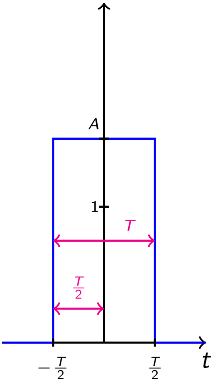
$$\begin{aligned}\mathcal{F}[f(t)] &= \mathcal{L}[f(t)]|_{s=j\omega} = \int_{-\infty}^{\infty} f(t)e^{-j\omega t}dt \\ &= F(s)|_{s=j\omega}\end{aligned}$$

Do **not** generalise this property when the ROC does not contain the imaginary axis !

Consider the **Laplace** transform if the interest is in **transients and steady state**, and the **Fourier** transform if **steady-state** behavior is of interest.

Fourier transforms from Laplace transform: example

$f(t) = A\,\Pi(\frac{t}{T})$



Note that

$$\begin{aligned} f(t) &= A\left(u(t+{}^{T}\!/_{2}) - u(t-{}^{T}\!/_{2})\right) \text{ et} \\ F(s) &= \frac{A}{s}\left(e^{\frac{T}{2}s} - e^{-\frac{T}{2}s}\right) \end{aligned}$$

$$\begin{aligned} \mathcal{F}[f(t)] &= F(s)\big|_{s=j\omega} \\ &= \frac{A}{j\omega}\left(e^{\frac{T}{2}j\omega} - e^{-\frac{T}{2}j\omega}\right) \\ &= \frac{2A}{\omega}\sin({}^{\omega T}\!/_{2}) \\ &= AT\,\frac{\sin(\pi f T)}{\pi f T} = AT\,\mathrm{sinc}(fT) \end{aligned}$$

Fourier transform: frequency shift

Frequency shift

If $F(\omega) = \mathcal{F}[f(t)]$ is the Fourier transform of $f(t)$, then we have the pair

$$f(t)e^{-j\omega_0 t} \Longleftrightarrow F(\omega + \omega_0)$$

$$\begin{aligned}
\mathcal{F}[f(t)e^{-j\omega_0 t}] &= \int_{-\infty}^{\infty} f(t)e^{-j\omega_0 t}e^{-j\omega t}dt \\
&= \int_{-\infty}^{\infty} f(t)e^{-j(\omega+\omega_0)t}dt \\
&= F(\omega + \omega_0)
\end{aligned}$$

Fourier transform: amplitude modulation

Amplitude modulation consists in multiplying an incoming message $f(t)$ by a sinusoid of frequency higher than the maximum frequency of the incoming signal. The modulated signal is $f(t)\cos(\omega_0 t)$.

Amplitude modulation

If $F(\omega) = \mathcal{F}[f(t)]$ is the Fourier transform of $f(t)$, then we have the pair

$$f(t)\cos(\omega_0 t) \Longleftrightarrow 0.5[F(\omega + \omega_0) + F(\omega - \omega_0)].$$

That is, the transform of the modulated signal is $F(\omega)$ shifted to frequencies ω_0 and $-\omega_0$, and multiplied by 0.5.

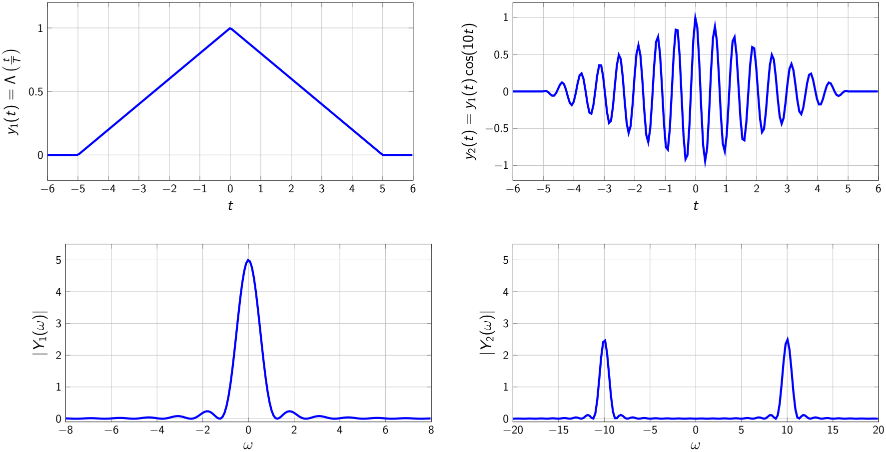
The Fourier pair is easily obtained using

$$f(t)\cos(\omega_0 t) = 0.5\, f(t)(e^{j\omega_0 t} + e^{-j\omega_0 t}).$$

Applications of amplitude modulation

- **Telecommunications**:
  - Acoustic signals are audible up to 20kHz.
  - When emitting signals with an antenna, the length of the antenna is related to the quarter wavelength.
  - The wavelength is give by $\lambda = \frac{c}{f} = \frac{3\times 10^8}{f}$ meters.
  - To emit signals with a frequency content up to 30kHz, the wavelength is 10km and the length of the antenna is 2.5km.
  - Amplitude modulation is an important application of the Fourier transform, as it allows us to change the original frequencies of a message to much higher frequencies, making it possible to transmit the signal over the airwaves.
- **Condition monitoring**: some mechanical faults (gearbox, bearings, etc.) introduce amplitude (or phase) modulation in vibration or current signals.

Amplitude modulation



Time-frequency duality

Time-frequency duality

To the Fourier transform pair

$$f(t) \Longleftrightarrow F(\omega) = \mathcal{F}[f(t)]$$

corresponds the following dual Fourier transform pair

$$F(t) \Longleftrightarrow 2\pi f(-\omega).$$

This duality property allows us to obtain the Fourier transform of signals for which we already have a Fourier pair and that would be difficult to obtain directly.

It is thus one more method to obtain the Fourier transform, besides the Laplace transform and the integral definition of the Fourier transform.

Time-frequency duality

Time-frequency duality

The inverse Fourier transform is obtained using

$$f(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} F(\omega)e^{j\omega t}d\omega = \frac{1}{2\pi}\int_{-\infty}^{\infty} F(\rho)e^{j\rho t}d\rho.$$

Replacing t by $-\omega$ and multiplying by 2π, one obtains

$$\begin{aligned} 2\pi f(-\omega) &= \int_{-\infty}^{\infty} F(\rho)e^{-j\rho\omega}d\rho, \\ &= \int_{-\infty}^{\infty} F(t)e^{-j\omega t}dt, \\ &= \mathcal{F}[F(t)]. \end{aligned}$$

Time-frequency duality: constant signal

Fourier transform of a constant signal

To the Fourier transform pair

$$A\,\delta(t) \Longleftrightarrow A$$

corresponds the dual Fourier transform pair

$$A \Longleftrightarrow 2\pi A\,\delta(-\omega) = 2\pi A\,\delta(\omega).$$

Time-frequency duality: cardinal sine

Fourier transform of a cardinal sine

To the Fourier transform pair

$$A\,\Pi(t/T) = A\left(u(t + T/2) - u(t - T/2)\right) \Longleftrightarrow \frac{2A}{\omega}\sin(\omega T/2)$$

corresponds the dual Fourier transform pair ($\frac{T}{2}$ replaced by ω_0)

$$\frac{2A}{t}\sin(\omega_0 t) \quad \Longleftrightarrow \quad 2\pi A\,\Pi(-\omega/2\omega_0) = 2\pi A\,\Pi(\omega/2\omega_0).$$

Therefore

$$\begin{aligned} \frac{\sin(\omega_0 t)}{\pi t} \quad &\Longleftrightarrow \quad u(-\omega + \omega_0) - u(-\omega - \omega_0), \\ &\Longleftrightarrow \quad u(\omega + \omega_0) - u(\omega - \omega_0). \end{aligned}$$

Time-frequency duality: Fourier transform of a cosine

To the Fourier transform pair

$$\delta(t-\tau)+\delta(t+\tau) \Longleftrightarrow e^{-j\omega\tau}+e^{j\omega\tau}=2\cos(\omega\tau)$$

corresponds the dual Fourier transform pair

$$2\cos(\tau t) \Longleftrightarrow 2\pi[\delta(-\omega-\tau)+\delta(-\omega+\tau)]=2\pi[\delta(\omega+\tau)+\delta(\omega-\tau)].$$

Remplacing τ by ω_0, the dual Fourier pair becomes

$$\cos(\omega_0 t) \Longleftrightarrow \pi\,[\delta(\omega+\omega_0)+\delta(\omega-\omega_0)].$$

The Fourier transform cannot be computed using

- the integral definition since this signal is not absolutely integrable,
- the Laplace transform: the $j\omega$-axis is not included in the ROC.

The Fourier transform is not defined $-\omega_0$ and ω_0.

Time-frequency duality: Fourier transform of a sine

To the Fourier transform pair

$$\delta(t+\tau) - \delta(t-\tau) \Longleftrightarrow e^{j\omega\tau} - e^{-j\omega\tau} = 2j\sin(\omega\tau)$$

corresponds the dual Fourier transform pair

$$2j\sin(\tau t) \Longleftrightarrow 2\pi[\delta(-\omega+\tau) - \delta(-\omega-\tau)] = 2\pi[-\delta(\omega+\tau) + \delta(\omega-\tau)].$$

Remplacing τ by ω_0, the dual Fourier pair becomes

$$\sin(\omega_0 t) \Longleftrightarrow j\pi\,[\delta(\omega+\omega_0) - \delta(\omega-\omega_0)].$$

The Fourier transform cannot be computed using

- the integral definition since this signal is not absolutely integrable,
- the Laplace transform: the $j\omega$-axis is not included in the ROC.

The Fourier transform is not defined $-\omega_0$ and ω_0.

Fourier transform of periodic signals

By applying the frequency-shifting property to compute the Fourier transform of periodic signals, we are able to unify the Fourier representation of aperiodic as well as periodic signals.

Fourier transform of periodic signals

For a periodic signal $f(t)$ of period T, we have the Fourier pair

$$f(t) = \sum_k c_k e^{jk\omega_0 t} \Longleftrightarrow F(\omega) = 2\pi \sum_k c_k \, \delta(\omega - k\omega_0)$$

obtained by representing $f(t)$ by its Fourier series.

Since a periodic signal $f(t)$ is not absolutely integrable, its Fourier transform cannot be computed using the integral formula. The Fourier series is used to circumvent this problem.

Parsevals energy conservation

Parsevals energy conservation

For a finite-energy signal $f(t)$ with Fourier transform $F(\omega)$, its energy is conserved when going from the time to the frequency domain, or

$$E = \int_{-\infty}^{\infty} |f(t)|^2 dt = \frac{1}{2\pi} \int_{-\infty}^{\infty} |F(\omega)|^2 d\omega.$$

Thus, $|F(\omega)|^2$ is an energy density indicating the amount of energy at each of the frequencies ω.

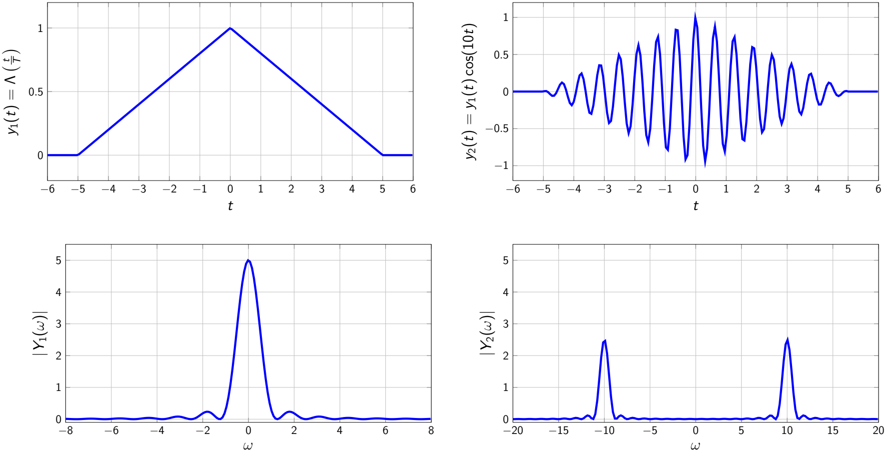
The plot $|F(\omega)|^2$ versus ω is called the energy spectrum of $f(t)$, and it displays how the energy of the signal is distributed over frequency.

Amplitude modulation

```
function X = Afourier(x,t0) % Approximate Fourier transform

syms t w
X = int(x*exp(-1i*w*t), t,-t0,t0);
```

```
% Signals and Systems using Matlab by Chaparro: example 5.7

clear all; close all;
syms t w
m = heaviside(t+5) - heaviside(t);
m1 = heaviside(t) - heaviside(t-5);
x2 = (t+5)*m + m1*(-t+5); x2 = x2/5;
x = x2*exp(-1i*10*t)/2;
y = x2*exp(+1i*10*t)/2;
Y = Afourier(y,5);
X = Afourier(x,5);
X2 = Afourier(x2,5);

figure(1)
subplot(221), ezplot(x2,[-6,6]), grid, axis([-6 6 -0.2 1.2]);
title(''), xlabel('t'), ylabel('x2(t)'),
subplot(222), ezplot(x+y,[-6,6]), grid, axis([-6 6 -1.2 1.2]);
title(''), xlabel('t'), ylabel('y2(t) = x2(t).cos(10t)'),
subplot(223), ezplot(abs(X2),[-8,8]), grid, axis([-8 8 -0.1 5.5])
xlabel('\omega'), ylabel('|X_2(\omega)|'),
subplot(224), ezplot(abs(X) + abs(Y),[-20,20]), grid; axis([-20 20 -0.1 5.5])
xlabel('\omega'), ylabel('|Y_2(\omega)|')
```

Amplitude modulation



Fourier transform pairs

$f(t)$	$\mathcal{F}[f(t)]$
$\delta(t)$	1
$\delta(t-\tau)$	$e^{-j\omega\tau}$
A	$2\pi A\,\delta(\omega)$
$u(t)$	$\frac{1}{j\omega}+\pi\delta(\omega)$
$e^{-at}u(t), a>0$	$\dfrac{1}{j\omega+a}$
$\sin(\omega_0 t)$	$j\pi[\delta(\omega+\omega_0)-\delta(\omega-\omega_0)]$
$\cos(\omega_0 t)$	$\pi[\delta(\omega+\omega_0)+\delta(\omega-\omega_0)]$

Fourier transform pairs	
$f(t)$	$\mathcal{F}[f(t)]$
$e^{-a\|t\|}, a > 0$	$\dfrac{2a}{(\omega^2 + a^2)}$
$t^n e^{-at} u(t),\ a > 0$	$\dfrac{n!}{(j\omega + a)^{n+1}}$
$A\,(u(t+T) - u(t-T))$	$2AT\,\dfrac{\sin(\omega T)}{\omega T}$
$\dfrac{\sin(\omega_0 t)}{\pi t}$	$u(\omega + \omega_0) - u(\omega - \omega_0)$
$f(t)\cos(\omega_0 t)$	$0.5[F(\omega + \omega_0) + F(\omega - \omega_0)]$

Basic properties of the Fourier transform		
Properties	$f(t)$	$F(\omega)$
Signals and constants	$\alpha f(t),\ \beta g(t)$	$\alpha F(\omega),\ \beta G(\omega)$
Linearity	$\alpha f(t) + \beta g(t)$	$\alpha F(\omega) + \beta G(\omega)$
Time shift	$f(t-\tau)$	$e^{-j\tau\omega} F(\omega)$
Frequency shift	$e^{-\omega_0 t} f(t)$	$F(\omega+\omega_0)$
Modulation	$f(t)\cos(\omega_0 t)$	$0.5[F(\omega+\omega_0) + F(\omega-\omega_0)]$

Basic properties of the Fourier transform

Properties	$f(t)$	$F(\omega)$
Time convolution	$f(t) * g(t)$	$F(\omega)G(\omega)$
Windowing / multiplication	$f(t)\,\omega(t)$	$\frac{1}{2\pi}[F * W](\omega)$
Time differentiation	$\dfrac{df(t)}{dt}$	$j\omega F(\omega)$
Integration	$\displaystyle\int_{-\infty}^{t} f(\bar{t})d\bar{t}$	$\frac{F(\omega)}{\omega} + \pi F(0)\delta(\omega)$
Expansion / contraction	$f(at), (a \neq 0)$	$\dfrac{1}{a} F\left(\dfrac{\omega}{a}\right)$

Basic properties of the Fourier transform

Properties	$f(t)$	$F(\omega)$
Parseval	$E = \int_{-\infty}^{\infty} \lvert f(t)\rvert^2 dt$	$E = \frac{1}{2\pi} \int_{-\infty}^{\infty} \lvert F(\omega)\rvert^2 d\omega$
Periodic signals	$\sum_k c_k e^{jk\omega_0 t}$	$2\pi \sum_k c_k \,\delta(\omega - k\omega_0)$
Symmetry	real-valued $f(t)$	$\begin{cases} \lvert F(\omega)\rvert = \lvert F(-\omega)\rvert, \\ \angle F(\omega) = -\angle F(-\omega) \end{cases}$
Cosine	real-valued even $f(t)$	$\int_{-\infty}^{\infty} f(t)\cos(\omega t)dt$, real
Sine	real-valued odd $f(t)$	$-j\int_{-\infty}^{\infty} f(t)\sin(\omega t)dt$, imaginary

6. Sampling theory

Ideal impulse sampling
Frequency folding
Analog-to-Digital Conversion
Digital-to-Analog Conversion

Introduction

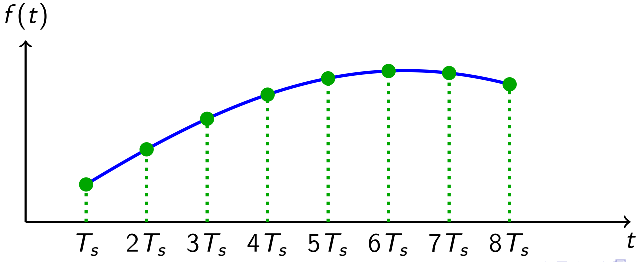
- Since many of the signals found in applications such as communications and control are analog, if we wish to process these signals with a computer it is necessary to sample, quantize, and code them to obtain digital signals.
- It is therefore necessary to understand the bridge between analog and discrete signals and systems.
- The device that samples, quantizes, and codes an analog signal is called an **Analog-to-Digital Converter (ADC)**, while the device that converts digital signals into analog signals is called a **Digital-to-Analog Converter (DAC)**.
- These devices are far from ideal and thus some practical aspects of sampling and reconstruction need to be considered: loss or information, **aliasing or frequency folding**, quantisation error, etc.

Uniform sampling

Uniform sampling

Sampling an analog continuous-time signal $f(t)$ yields samples of the system representing the amplitude of the signal at the sampling instants.

The samples $f(kT_s)$ are taken on a regular basis with a sampling period T_s.

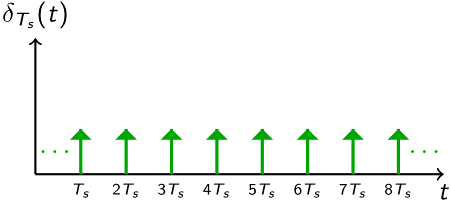


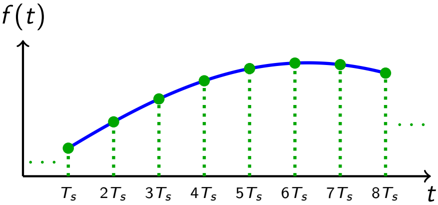
Ideal impulse sampling

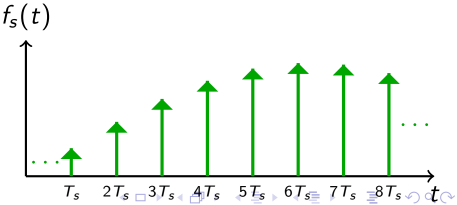
The sampling process is ideal if the samples are considered to be instantaneous values precisely at the sample instants.

Ideal sampling can be modeled using the sampling function.

$$\begin{aligned} \delta_{T_s}(t) &= \sum_{k=-\infty}^{\infty} \delta(t - kT_s) \\ f_s(t) &= f(t)\,\delta_{T_s}(t) \\ &= \sum_{k=-\infty}^{\infty} f(kT_s)\,\delta(t - kT_s) \end{aligned}$$







Fourier transform of a sampled signal

The sampling function is periodic with period T_s. It can therefore be written as a Fourier series of complex exponentials:

$$\begin{aligned}
\delta_{T_s}(t) &= \sum_{k=-\infty}^{\infty} c_k \, e^{jk\omega_s t}, \;\; \omega_s = \frac{2\pi}{T_s} \\
c_k &= \frac{1}{T_s}\int_{-\frac{T_s}{2}}^{\frac{T_s}{2}} \delta_{T_s}(t) e^{-jk\omega_s t} dt \\
&= \frac{1}{T_s}\int_{-\frac{T_s}{2}}^{\frac{T_s}{2}} \sum_{n=-\infty}^{\infty} \delta(t - nT_s) e^{-jk\omega_s t} dt = \frac{1}{T_s}
\end{aligned}$$

$$\boxed{\delta_{T_s}(t) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} e^{jk\omega_s t}, \;\; \omega_s = \frac{2\pi}{T_s}}$$

Fourier transform of a sampled signal

We have

$$f_s(t) = f(t)\,\delta_{T_s}(t).$$

Therefore

$$\begin{aligned} F_s(\omega) &= \mathcal{F}[f(t)\,\delta_{T_s}(t)] \\ &= \frac{1}{T_s}\sum_{k=-\infty}^{\infty}\mathcal{F}[f(t)\,e^{jk\omega_s t}] \\ &= \frac{1}{T_s}\sum_{k=-\infty}^{\infty}F(\omega - k\omega_s) \end{aligned}$$

The Fourier transform of the sampled system is a superposition of shifted analog spectra $F(\omega - k\omega_s)$ multiplied by $1/T_s$.

Uniform sampling and Fourier transform

Sampling a continuous-time $f(t)$ with a uniform sampling period T_s yields the signal

$$f_s(t) = \sum_{k=-\infty}^{\infty} f(kT_s)\,\delta(t - kT_s).$$

Sampling is equivalent to modulating the sampling signal

$$\delta_{T_s}(t) = \sum_{k=-\infty}^{\infty} \delta(t - kT_s).$$

If $F(\omega)$ is the Fourier transform of $f(t)$, the Fourier transform of the sampled signal $f_s(t)$ is

$$F_s(\omega) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} F(\omega - k\omega_s).$$

Low-pass spectrum of finite support

Low-pass spectrum of finite support

If the initial signal $f(t)$ has a low-pass spectrum of finite support, i.e. $F(\omega) = 0$ pour $\omega > \omega_{max}$ where ω_{max} is the maximum frequency present in the signal[44], it is possible to choose ω_s so that the spectrum of the sampled signal consists of shifted but not overlapping scaled versions of $F(\omega)$.

The Nyquist condition on ω_s is $\omega_s - \omega_{max} \geq \omega_{max}$, or

$$\omega_s \geq 2\,\omega_{max}.$$

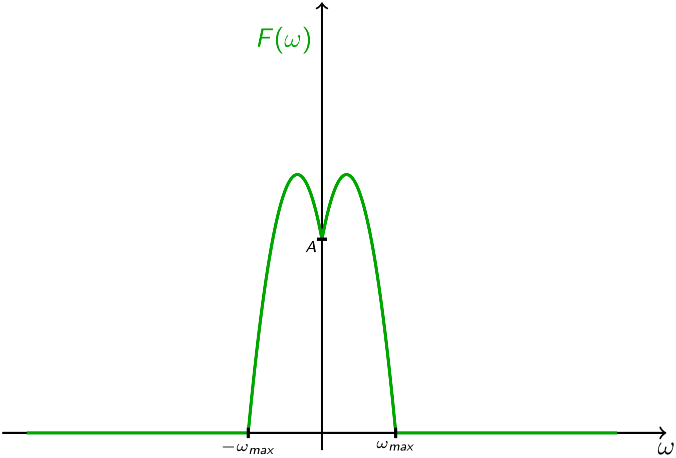
This is sometimes written under the form

$$\omega_N = \frac{\omega_s}{2} \geq \omega_{max}$$

where ω_N is called the Nyquist frequency.

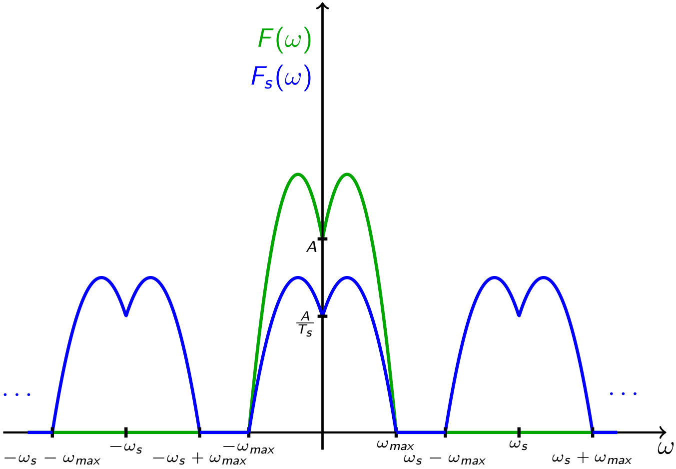
[44] Such a signal is called band limited.

Low-pass spectrum of finite support



Nyquist condition is respected !

Low-pass spectrum of finite support



Nyquist condition is respected !

Frequency folding or aliasing

When the Nyquist condition is not respected, it is impossible to distinguish between a frequency ω and its aliases $\omega + n\,\omega_s$.

Consequence : Loss of information !

When ?

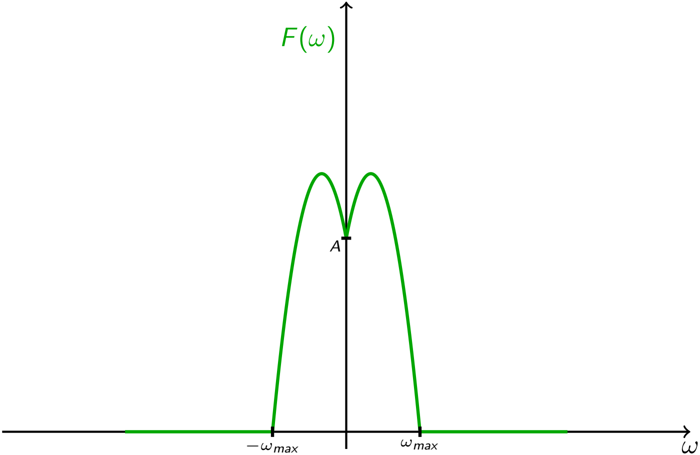
- The initial signal $f(t)$ has a spectrum of unlimited support

  Solution: Pass the signal through a low-pass anti-aliasing filter.

- The initial signal $f(t)$ is band limited but the Nyquist sampling condition is not respected.

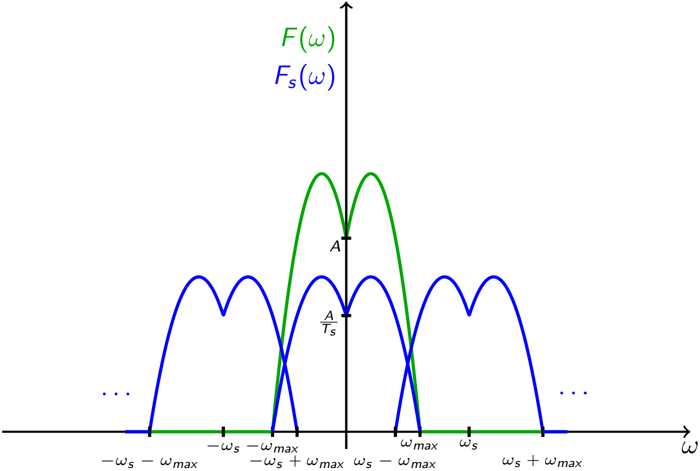
  Solution: Increase the sampling frequency.

Frequency folding or aliasing



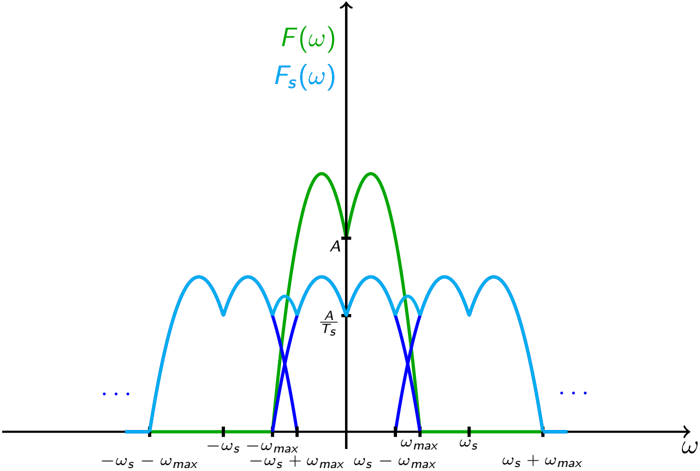
Nyquist condition is **not** respected !

Frequency folding or aliasing



Nyquist condition is **not** respected !

Frequency folding or aliasing

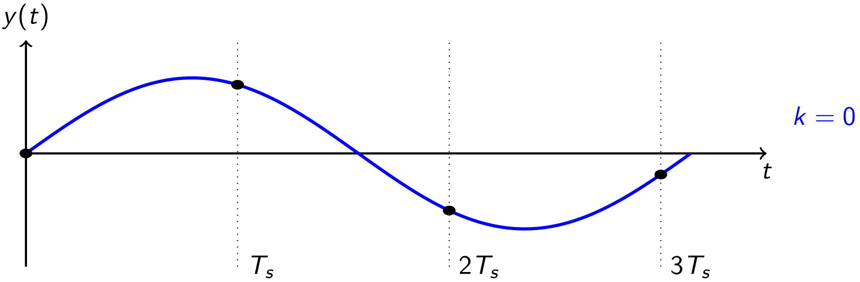


Nyquist condition is **not** respected !

Sampling issues

Several distinct signals can coincide on the same discrete samples !

Example: $y = \sin(\omega_f\, t), \omega_{max} = \omega_f = 1, \omega_s = \pi > 2\,\omega_{max} \rightarrow T_s = \frac{2\pi}{\omega_s} = 2$



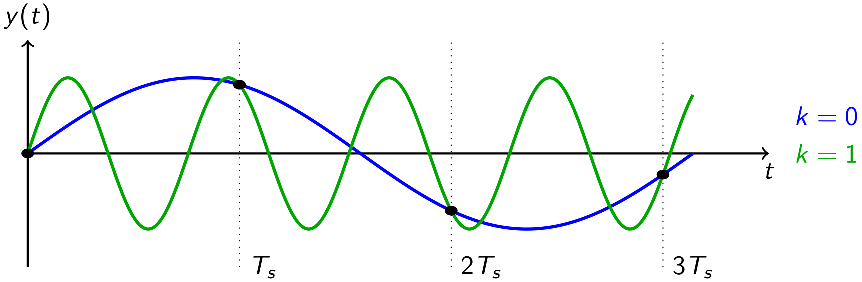
[45]The Nyquist condition is not verified anymore !

Sampling issues

Several distinct signals can coincide on the same discrete samples !

Example: $y = \sin(\omega_f\, t), \omega_{max} = \omega_f = 1, \omega_s = \pi > 2\,\omega_{max} \rightarrow T_s = \frac{2\pi}{\omega_s} = 2$

If the signals $y_k = \sin((\omega_f + k\omega_s)\, t)$ pollute the base signal $y(t)$, they are identical[45] to the base signal $y(t)$ at the sample instants.



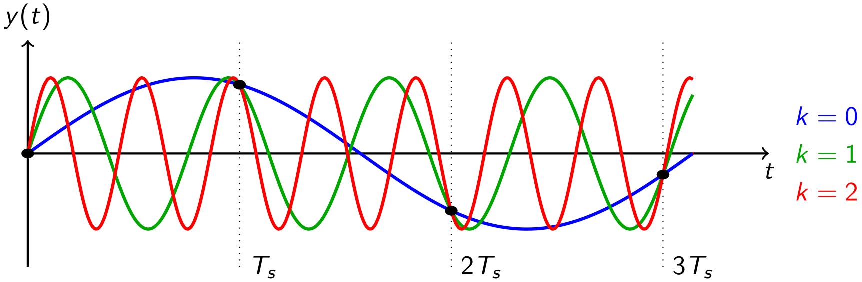
[45] The Nyquist condition is not verified anymore !

Sampling issues

Several distinct signals can coincide on the same discrete samples !

Example: $y = \sin(\omega_f\, t), \omega_{max} = \omega_f = 1, \omega_s = \pi > 2\,\omega_{max} \rightarrow T_s = \frac{2\pi}{\omega_s} = 2$

If the signals $y_k = \sin((\omega_f + k\omega_s)\, t)$ pollute the base signal $y(t)$, they are identical[45] to the base signal $y(t)$ at the sample instants.



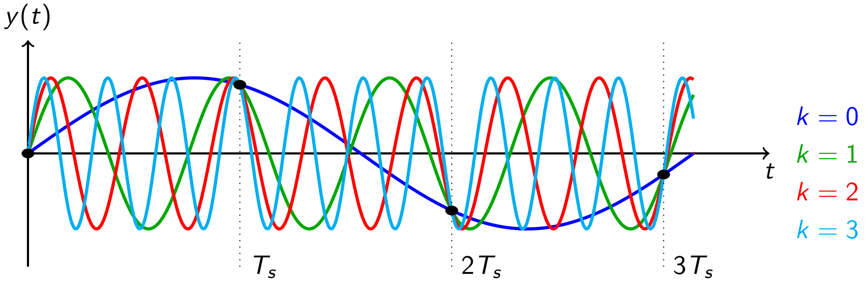
[45] The Nyquist condition is not verified anymore !

Sampling issues

Several distinct signals can coincide on the same discrete samples !

Example: $y = \sin(\omega_f\, t), \omega_{max} = \omega_f = 1, \omega_s = \pi > 2\,\omega_{max} \rightarrow T_s = \frac{2\pi}{\omega_s} = 2$

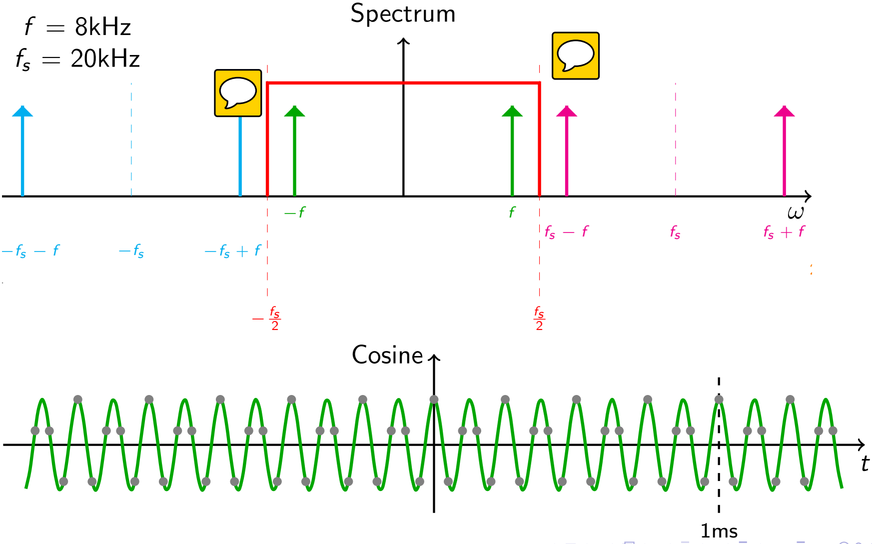
If the signals $y_k = \sin((\omega_f + k\omega_s)\, t)$ pollute the base signal $y(t)$, they are identical[45] to the base signal $y(t)$ at the sample instants.



[45] The Nyquist condition is not verified anymore !

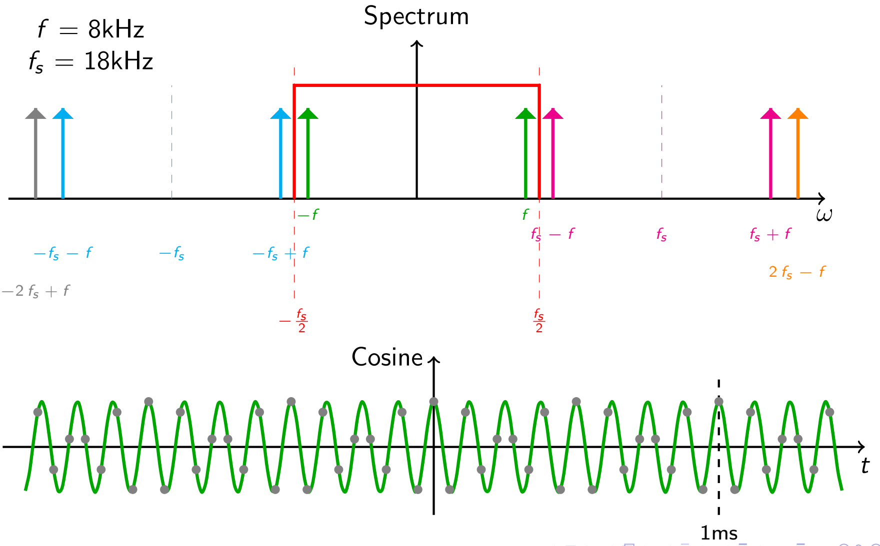
8kHz cosine sampled at 20kHz

$f = 8\text{kHz}$
$f_s = 20\text{kHz}$



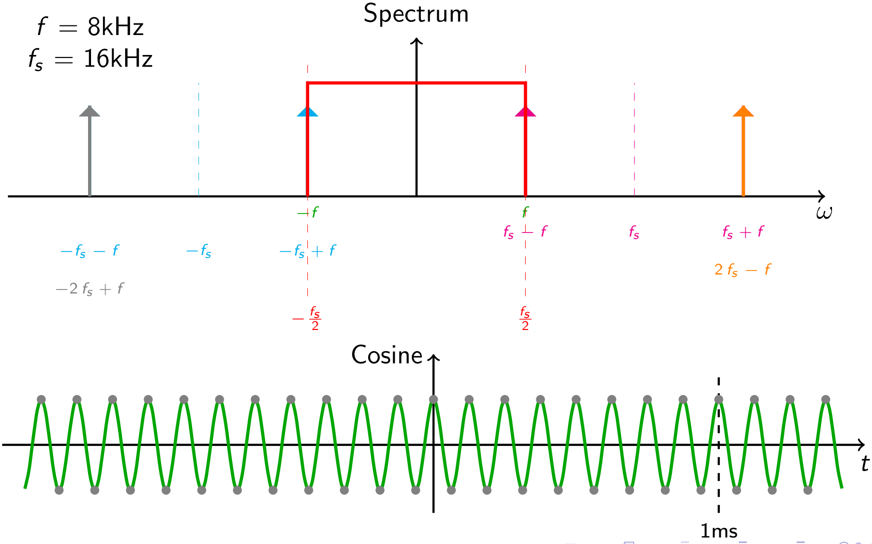
8kHz cosine sampled at 18kHz

$$f = 8\text{kHz}$$
$$f_s = 18\text{kHz}$$



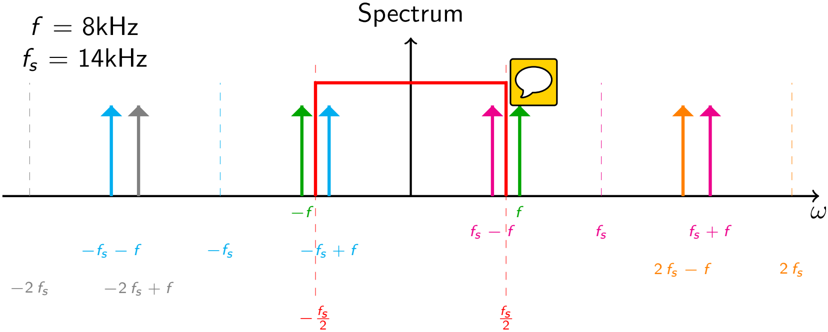
8kHz cosine sampled at 16kHz

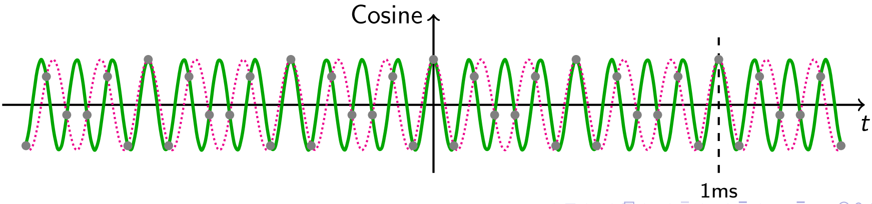
$$f = 8\text{kHz}$$
$$f_s = 16\text{kHz}$$



8kHz cosine sampled at 14kHz

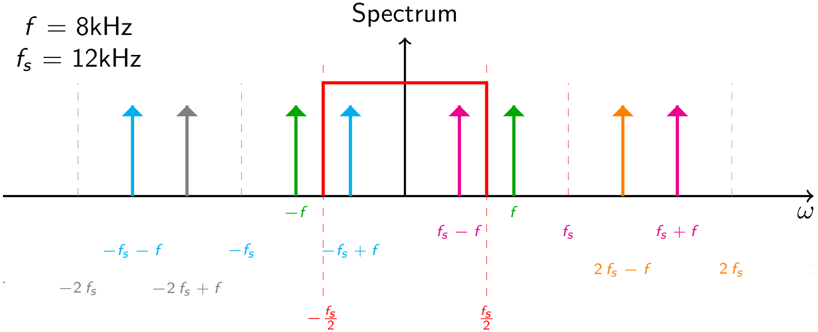
$f = 8\text{kHz}$
$f_s = 14\text{kHz}$

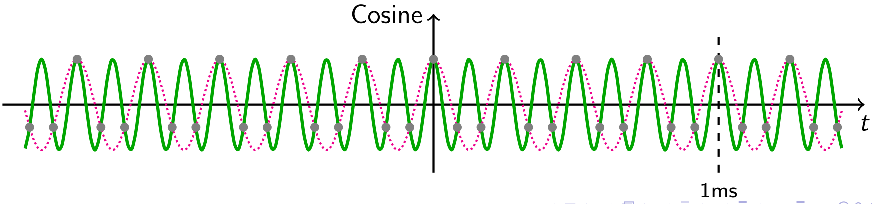




8kHz cosine sampled at 12kHz

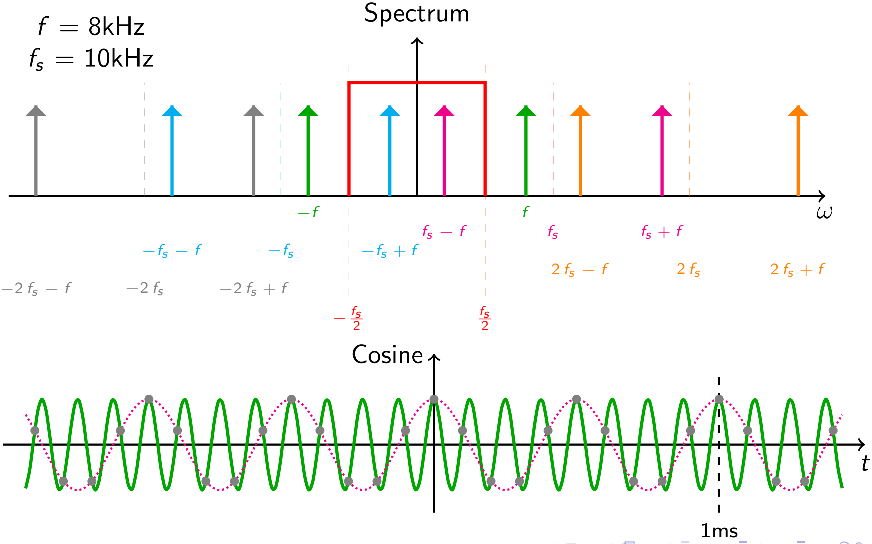
$$f = 8\text{kHz}$$
$$f_s = 12\text{kHz}$$





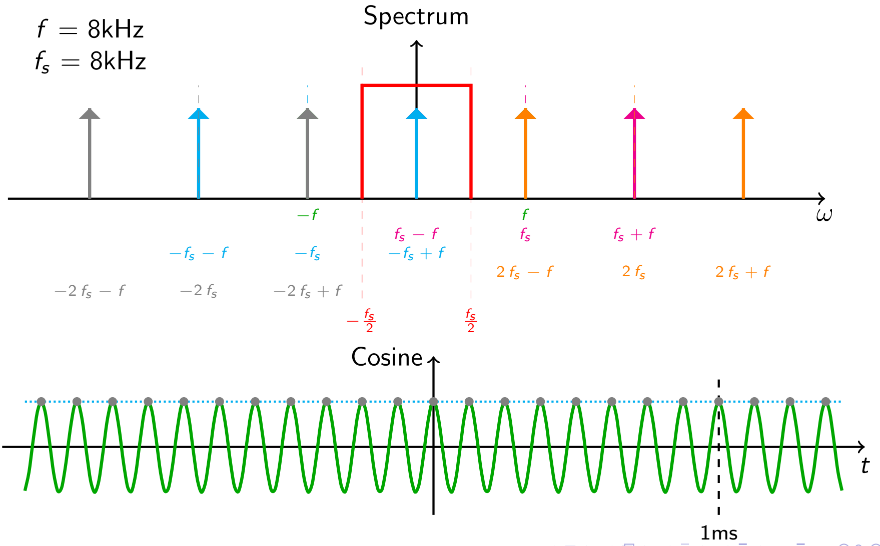
8kHz cosine sampled at 10kHz

$f = 8\text{kHz}$
$f_s = 10\text{kHz}$



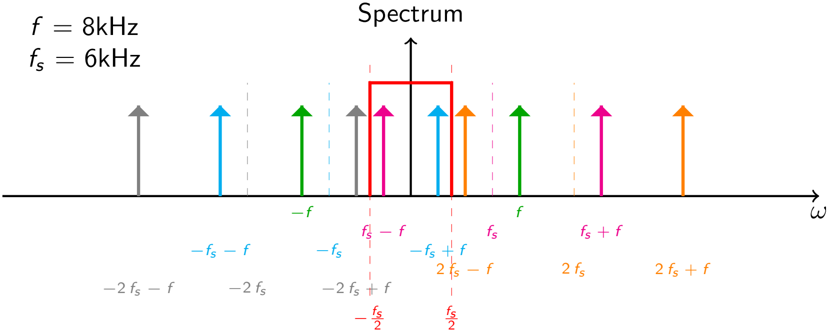
8kHz cosine sampled at 8kHz

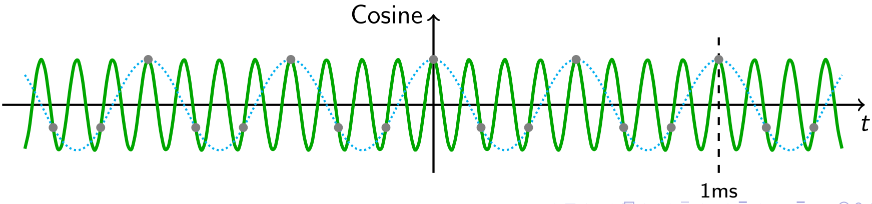
$f = 8\text{kHz}$
$f_s = 8\text{kHz}$



8kHz cosine sampled at 6kHz

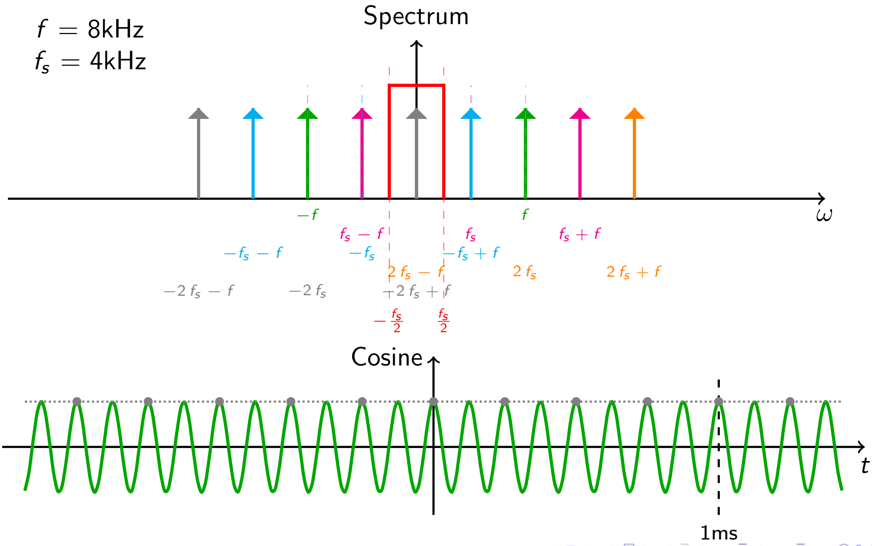
$f = 8\text{kHz}$
$f_s = 6\text{kHz}$





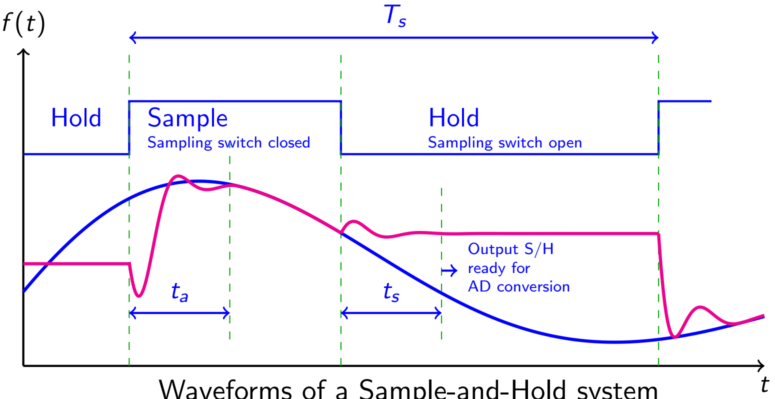
8kHz cosine sampled at 4kHz

$$f = 8\text{kHz}$$
$$f_s = 4\text{kHz}$$



Practical aspects of sampling

The Sample-and-Hold[46] (S/H) system first acquires the signal then takes the sample and holds it long enough for quantisation and coding to be done.



Waveforms of a Sample-and-Hold system

[46] Echantillonnage-blocage

Practical aspects of sampling

Definitions

- **Acquisition time** (t_a): time to acquire the analog signal
- **Settling time** (t_s): time to settle to the final held value within an accuracy tolerance. After that time, the sample is ready for quantification and coding.

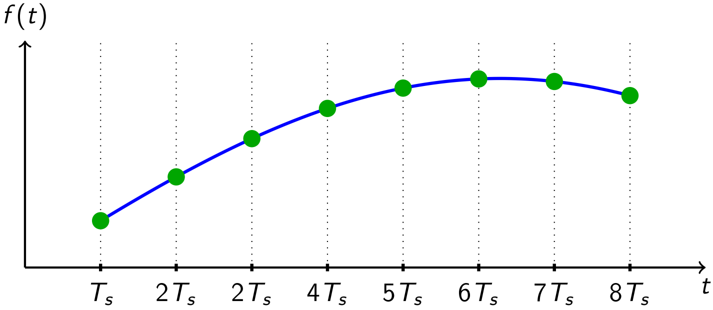
Other considerations

- **Aperture time**: time for the sampling switch to open.
- **Aperture jitter**[47]: variations on the aperture time due to clock variations and noise.

[47] Gigue d'échantillonnage

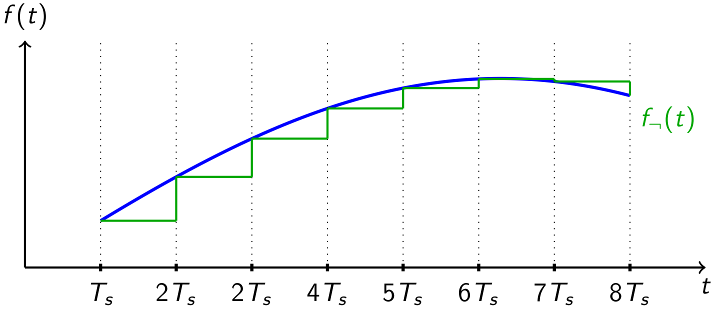
Practical aspects of sampling

Assuming acquisition is instantaneous, the signal resulting from the Sample-and-Hold has a staircase shape of $f_{\neg}(t)$.



Practical aspects of sampling

Assuming acquisition is instantaneous, the signal resulting from the Sample-and-Hold has a staircase shape of $f_{\neg}(t)$.



Quantisation

- Quantisation is the second step in the digitisation[48] of signals.
- It allows storage and signal processing once the signal is coded.
- The role of quantisation is to affect a finite resolution value to a sample which amplitude has in theory an infinite resolution.

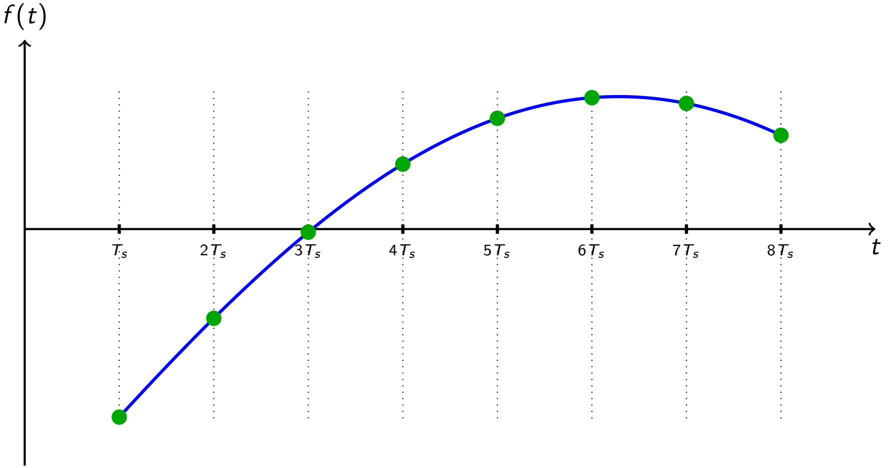
Quantisation

Quantising a sample is to take the one of the fixed amplitude levels that best represents the sample according to some approximation scheme. It is a nonlinear process.

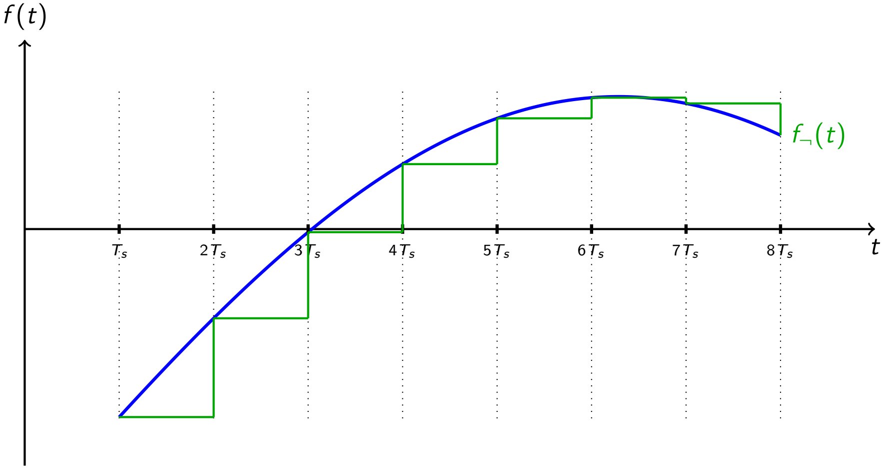
When the amplitude levels are equidistant, quantisation is uniform.

[48] Also known as digitalisation

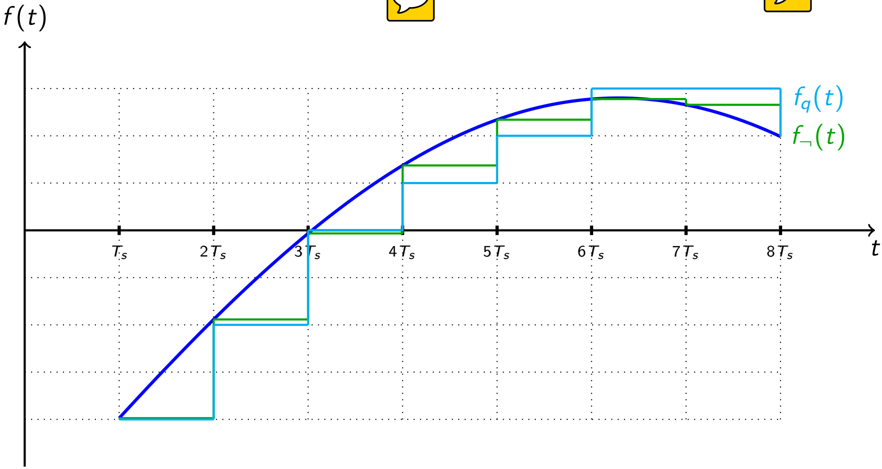
Quantisation



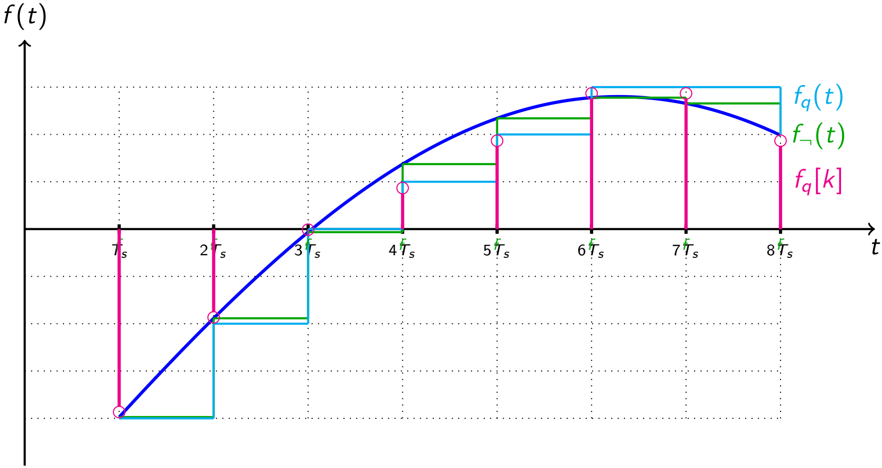
Quantisation



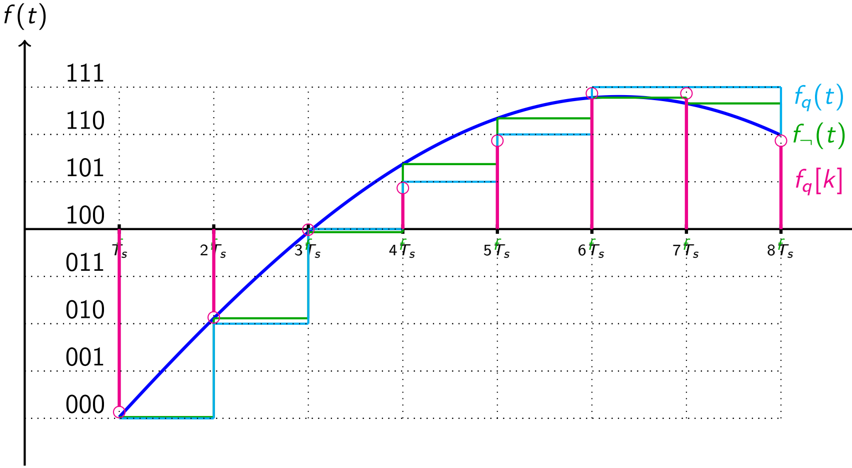
Quantisation



Quantisation



Quantisation and coding



Coding

Coding

By coding the quantised signal, we make it accessible to a processor

- The quantisation grid must cover the dynamic range of the signal.
- With N bits, it is possible to code 2^N levels of quantisation.
- Quantisation followed by coding results in loss of information, and the quantisation noise relates to the rounding or truncating error in the representation of each sample.

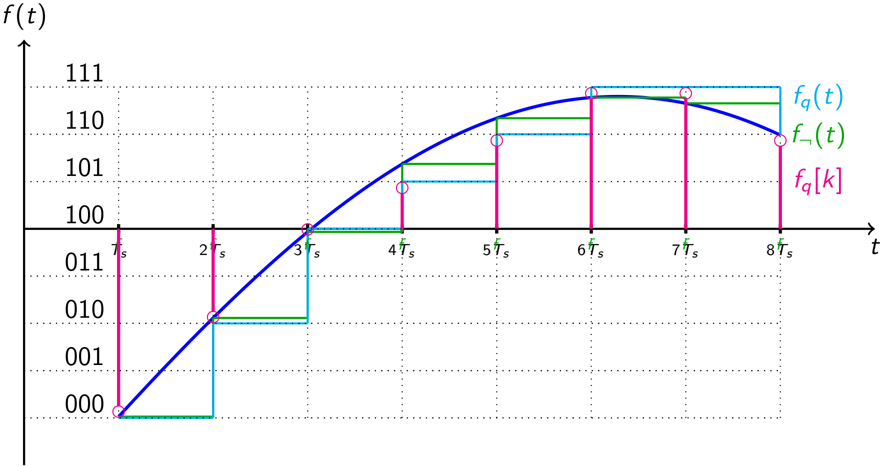
Coding: fundamentals

- When all representative bits representing the sample are 0, the associated value of the digital signal is f_{min}
- When all representative bits representing the sample are 1, the associated value of the digital signal is f_{max}
- The precision or resolution is related to the Least Significant Bit (LSB). It is called the quantum, the bit resolution or the quantisation step size. With N bits, there will be 2^N quantisation levels and the quantisation step size is

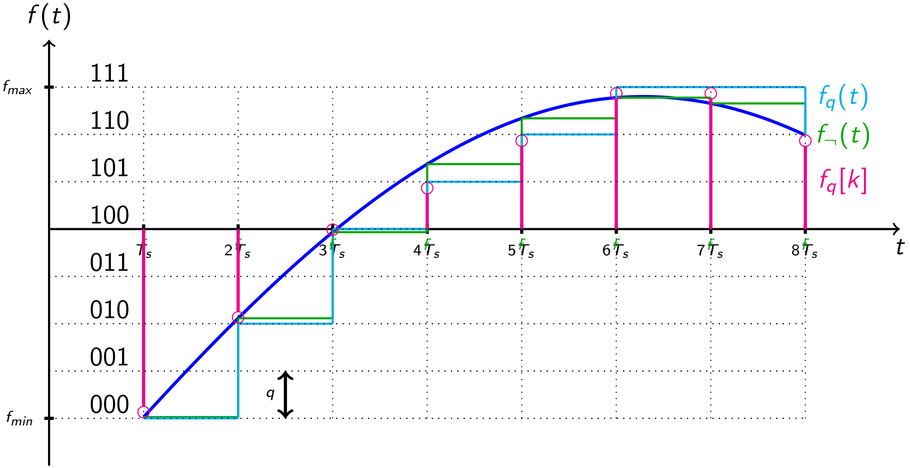
$$q = (f_{max} - f_{min})/(2^N - 1)$$

- The quantisation noise varies between $\pm\frac{\text{LSB}}{2}$.

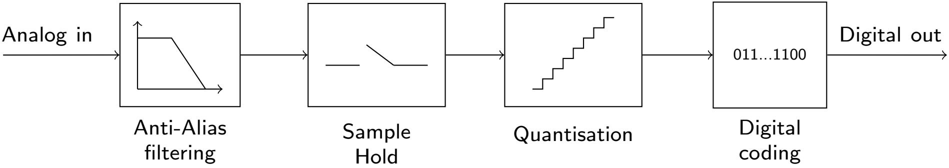
Quantisation and coding



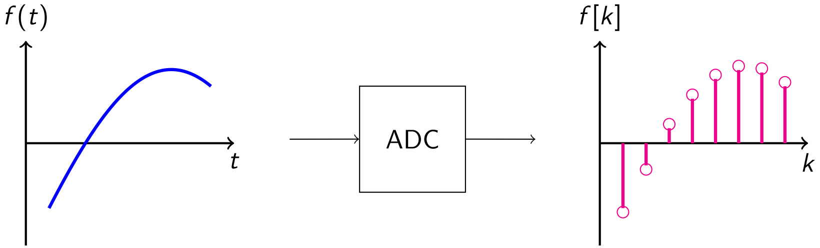
Quantisation and coding



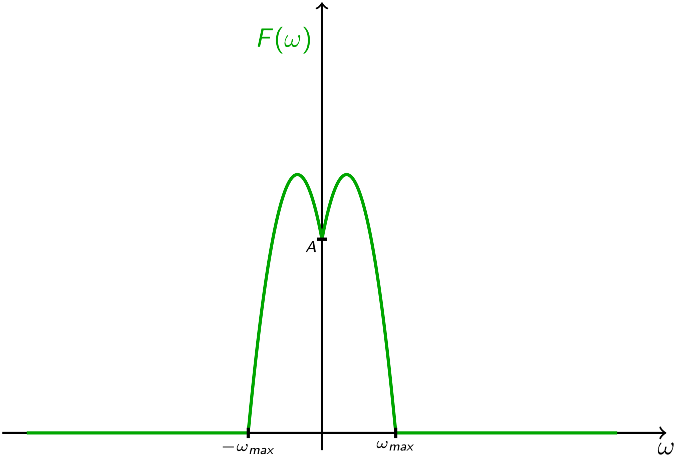
Analog-to-Digital Conversion (ADC)



With an infinite resolution and neglecting coding effects, we have

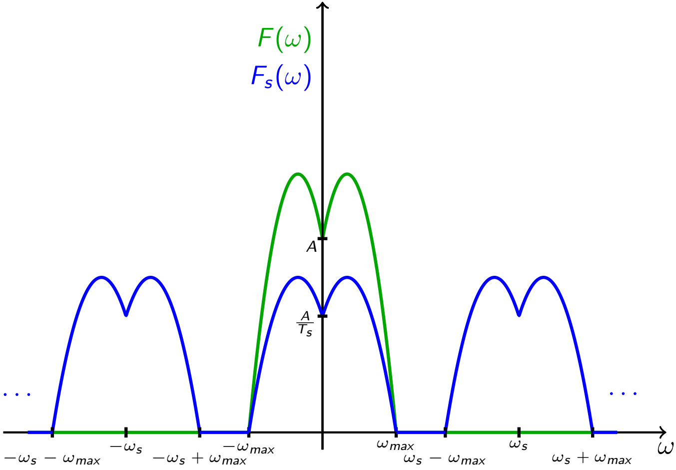


Low-pass spectrum of finite support: ideal reconstruction



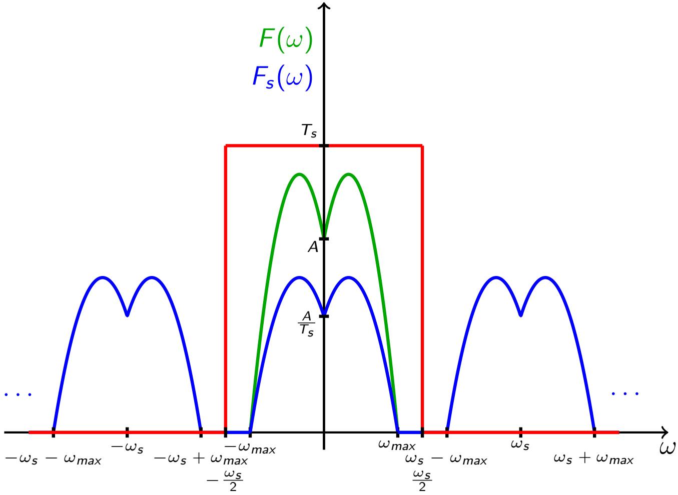
Nyquist condition is respected !

Low-pass spectrum of finite support: ideal reconstruction



Nyquist condition is respected !

Low-pass spectrum of finite support: ideal reconstruction



Nyquist condition is respected !

Low-pass spectrum of finite support: ideal reconstruction

If the initial signal $f(t)$ has a low-pass spectrum of finite support:, i.e. $F(\omega) = 0$ pour $\omega > \omega_{max}$, it is possible to reconstruct $f(t)$ from $f_s(t)$ using a scaled ideal low-pass filter

$$H_{lp}(\omega) = \begin{cases} T_s & -\frac{\omega_s}{2} < \omega < \frac{\omega_s}{2} \\ 0 & \text{elsewhere} \end{cases}$$

The Fourier transform of the filtered signal is $F_r(\omega) = H_{lp}(\omega) F_s(\omega)$, i.e.

$$F_r(\omega) = \begin{cases} F(\omega) & -\frac{\omega_s}{2} < \omega < \frac{\omega_s}{2} \\ 0 & \text{elsewhere} \end{cases}$$

This corresponds to the Fourier transform of the original filter.

Low-pass spectrum of finite support: reconstruction

The ideal low-pass filter $H_{lp}(\omega)$ has **non-causal** impulse response[49]

$$h_{lp}(t) = \frac{T_s}{2\pi} \int_{-\frac{\omega_s}{2}}^{\frac{\omega_s}{2}} e^{j\omega t} d\omega = \frac{\sin(\pi t/T_s)}{\pi t/T_s} = \mathrm{sinc}(t/T_s)$$

The reconstructed signal $f_r(t)$ is the convolution of $f_s(t)$ and $h_{lp}(t)$

$$\begin{aligned} f_r(t) &= [f_s * h_{lp}](t) = \int_{-\infty}^{\infty} f_s(\tau) h_{lp}(t-\tau)\, d\tau \\ &= \int_{-\infty}^{\infty} \left(\sum_{n=-\infty}^{\infty} f(nT_s)\, \delta(\tau - n\,T_s) \right) \frac{\sin(\pi(t-\tau)/T_s)}{\pi(t-\tau)/T_s}\, d\tau \\ &= \sum_{n=-\infty}^{\infty} f(nT_s) \frac{\sin(\pi(t-nT_s)/T_s)}{\pi(t-nT_s)/T_s} \end{aligned}$$

[49] The inverse Fourier transform is used here !

Low-pass spectrum of finite support: reconstruction

The reconstructed signal $f_r(t)$ is the convolution of $f_s(t)$ and $h_{lp}(t)$

$$f_r(t) = \sum_{n=-\infty}^{\infty} f(nT_s) \frac{\sin(\pi(t-nT_s)/T_s)}{\pi(t-nT_s)/T_s}.$$

The reconstructed signal is thus an interpolation of time-shifted cardinal sine signals with amplitudes the samples $\{f(nT_s)\}$. Taking $t = kT_s$ we can see that

$$f_r(kT_s) = \sum_{n=-\infty}^{\infty} f(nT_s) \frac{\sin(\pi(k-n))}{\pi(k-n)} = f(kT_s)$$

since

$$\frac{\sin(\pi(k-n))}{\pi(k-n)} = \begin{cases} 1 & k = n \\ 0 & k \neq n \end{cases}$$

The reconstruction is exact at the sampling instants $t = k\,T_s$. The ideal reconstruction filter is **not realisable in practice** as it is non-causal and it requires an infinite amount of data ! There is always some level of **approximation**.

Nyquist-Shannon sampling theorem

If a low-pass continuous-time signal $f(t)$ is band limited, i.e. $F(\omega) = 0$ pour $\omega > \omega_{max}$, the signal $f(t)$ is uniquely determined by its samples $f(kTs)$ provided that the sampling frequency satisfies the Nyquist condition

$$\omega_s > 2\,\omega_{max} \text{ or } f_s = \frac{1}{T_s} > \frac{\omega_{max}}{\pi}.$$

The original signal $f(t)$ can be reconstructed by passing the sampled signal through an ideal low-pass filter

$$H_{lp}(\omega) = \begin{cases} T_s & -\frac{\omega_s}{2} < \omega < \frac{\omega_s}{2} \\ 0 & \text{elsewhere} \end{cases}$$

The ideal reconstruction of signal is thus an interpolation of time-shifted cardinal sine signals from the samples $f(kT_s)$

$$f_r(t) = \sum_{k=-\infty}^{\infty} f(kT_s)\,\frac{\sin(\pi(t - kT_s)/T_s)}{\pi(t - kT_s)/T_s}.$$

Ideal reconstruction is impossible in practice

In practice, the exact recovery of the original signal is not possible for several reasons:

- The continuous-time signal is never really band limited due to the presence of noise.
- The sampling is not done exactly at uniform times. Slight random variations of the sampling times are always present.
- In reality, the values taken by the samples do not correspond to the amplitude at a precise sampling instant, but rather they are the average over a small interval around kT_s.
- Quantisation and coding effects have to be taken into account.
- The filter required for exact recovery is an ideal low-pass filter , which cannot be realized; only an approximation is possible.
- When the signal passes through a transmission channel, it can be subjected to small distortions.

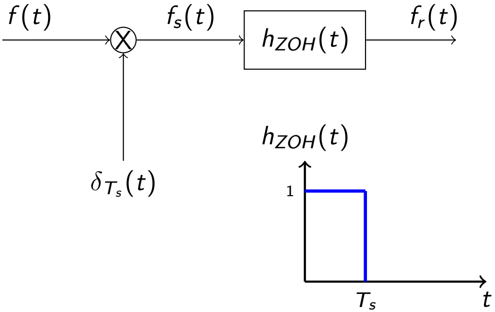
Digital-to-Analog Conversion (DAC)

- The ideal filter operation just described assumes availability of infinitely many samples. **This is not realistic !**
- Practical operation uses only a finite number of samples. Many techniques can be used to approximately reconstruct the signal.
- One such technique is to use a **Zero-Order Holder (ZOH)** reconstruction filter. We will only consider ZOH reconstruction.
- A **First-Order-Hold (FOH)** is sometimes used instead of ZOH. For the FOH, the signal is reconstructed as a piecewise linear approximation to the original signal that was sampled.

Zero Order Hold (ZOH) reconstruction

The Zero Order Hold (ZOH) reconstruction process can be represented by an LTI system having impulse response $h_{ZOH}(t)$ of width T_s.

Let us represent ideal sampling followed by ZOH reconstruction



Zero Order Hold (ZOH) reconstruction

The output of the ZOH process is

$$f_r(t) = [f_s * h_{ZOH}](t)$$

The associated Fourier transform is

$$\begin{aligned}
F_r(\omega) &= F_s(\omega)\, H_{ZOH}(\omega) \\
&= \left[\frac{1}{T_s}\sum_{k=-\infty}^{\infty} F(\omega - k\omega_s)\right] H_{ZOH}(\omega) \\
H_{ZOH}(\omega) &= \int_{-\infty}^{\infty} h_{ZOH}(t) e^{-j\omega t} dt = \int_{0}^{T_s} h_{ZOH}(t) e^{-j\omega t} dt = \frac{1}{j\omega}\left(1 - e^{-j\omega T_s}\right) \\
&= \frac{e^{-j\omega\frac{T_s}{2}}}{j\omega}\left(e^{j\omega\frac{T_s}{2}} - e^{-j\omega\frac{T_s}{2}}\right) = \frac{e^{-j\omega\frac{T_s}{2}}}{j\omega}\left(2j\sin(\omega\frac{T_s}{2})\right) \\
&= T_s\, e^{-j\omega\frac{T_s}{2}}\left[\frac{\sin(\omega\frac{T_s}{2})}{\omega\frac{T_s}{2}}\right] = T_s\, e^{-j\omega\frac{T_s}{2}}\ \mathrm{sinc}(fT_s)
\end{aligned}$$

Zero Order Hold (ZOH) reconstruction

Ideal reconstruction assuming the Nyquist condition is respected:

$$F_r(\omega) = F(\omega)$$

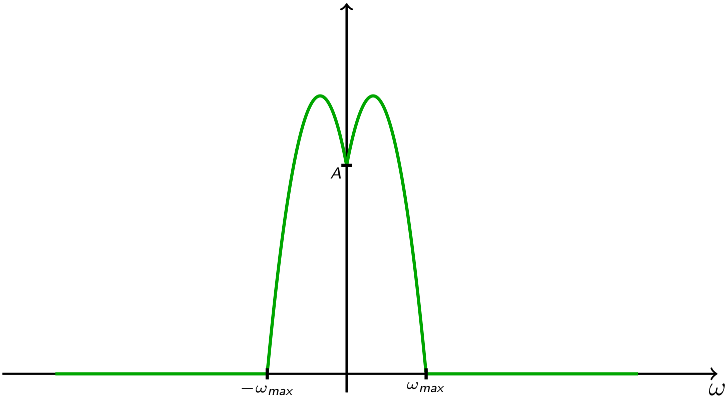
ZOH reconstruction assuming the Nyquist condition is respected:

$$F_r(\omega) = e^{-j\omega\frac{T_s}{2}} \frac{\sin(\omega\frac{T_s}{2})}{\omega\frac{T_s}{2}} \sum_{k=-\infty}^{\infty} F(\omega - k\omega_s)$$

The recovered signal $f_r(t)$ is distorted by the spectrum of $h_{ZOH}(t)$. The repetitions of the spectrum are still present although attenuated.

Low-pass spectrum of finite support: ZOH spectrum

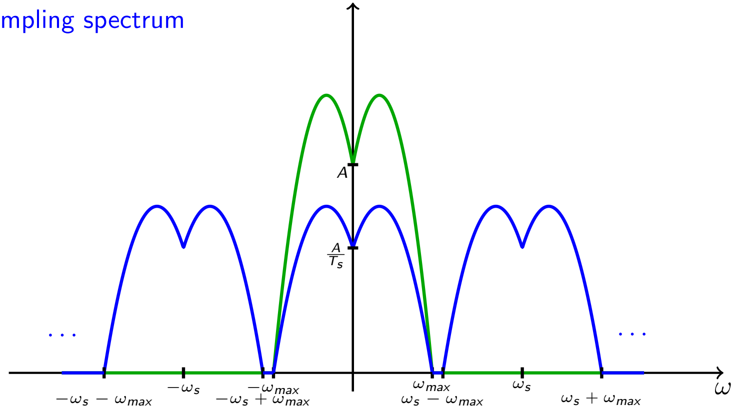
$F(\omega)$: Original spectrum



Low-pass spectrum of finite support: ZOH spectrum

$F(\omega)$: Original spectrum

$F_s(\omega)$: Ideal sampling spectrum

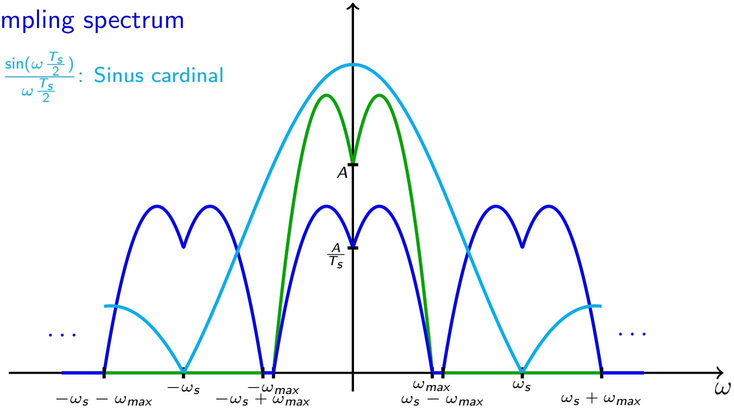


Low-pass spectrum of finite support: ZOH spectrum

$F(\omega)$: Original spectrum

$F_s(\omega)$: Ideal sampling spectrum

$|H_{ZOH}(j\omega)| = T_s \frac{\sin(\omega \frac{T_s}{2})}{\omega \frac{T_s}{2}}$: Sinus cardinal



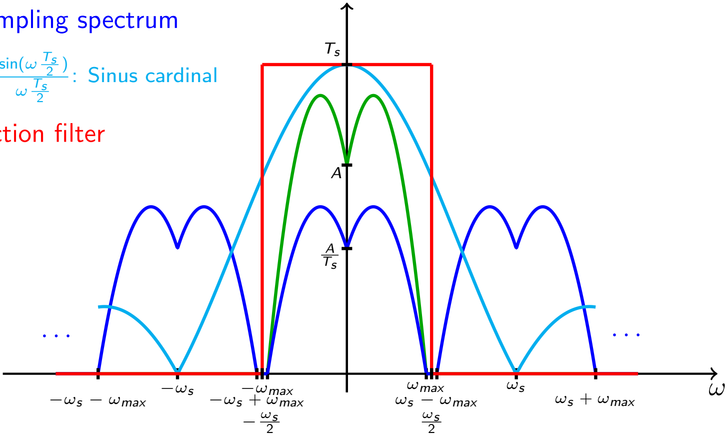
Low-pass spectrum of finite support: ZOH spectrum

$F(\omega)$: Original spectrum

$F_s(\omega)$: Ideal sampling spectrum

$|H_{ZOH}(j\omega)| = T_s \frac{\sin(\omega \frac{T_s}{2})}{\omega \frac{T_s}{2}}$: Sinus cardinal

Ideal reconstruction filter



Low-pass spectrum of finite support: ZOH spectrum

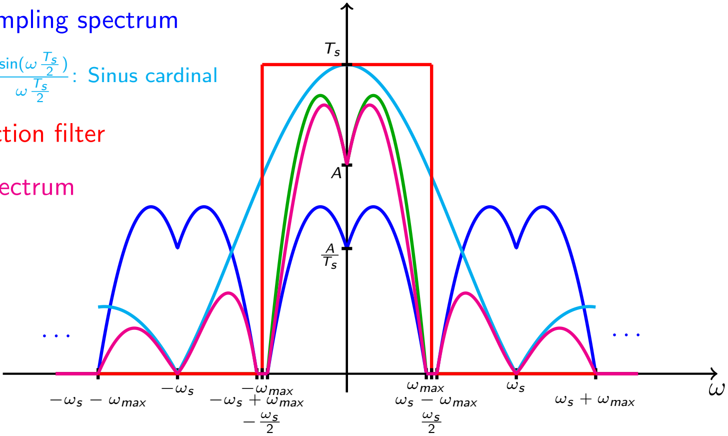
$F(\omega)$: Original spectrum

$F_s(\omega)$: Ideal sampling spectrum

$|H_{ZOH}(j\omega)| = T_s \, \frac{\sin(\omega \frac{T_s}{2})}{\omega \frac{T_s}{2}}$: Sinus cardinal

Ideal reconstruction filter

$F_r(\omega)$: ZOH spectrum



ZOH reconstruction: consequences

The recovered signal $f_r(t)$ is distorted by the spectrum of $h_{ZOH}(t)$. The repetitions of the spectrum are still present although attenuated.

As a consequence, the sampling frequency will have to satisfy the Nyquist condition with

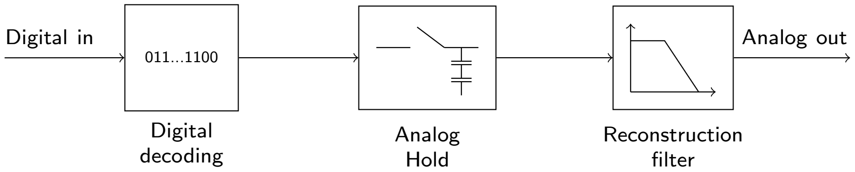
$$\omega_s >> 2\,\omega_{max} \text{ or } f_s = \frac{1}{T_s} >> \frac{\omega_{max}}{\pi}$$

to attenuate the distortion effects of the ZOH filter. This is known as **oversampling**.

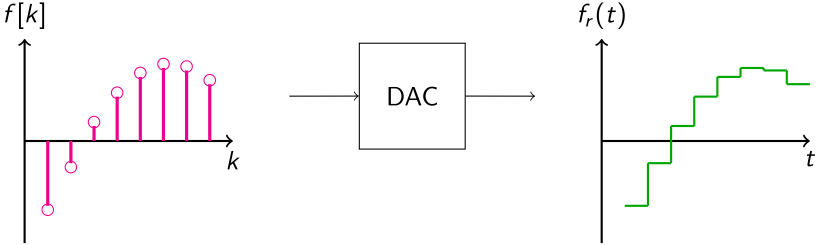
The ZOH reconstruction filter is often followed by a **reconstruction low-pass filter** to attenuate the repetitions in the spectrum of $f_r(t)$.

Sometimes an additional filter is included to attenuate (invert) the distortion effects of the ZOH filter within the frequency band of interest.

Digital-to-Analog Conversion (DAC)



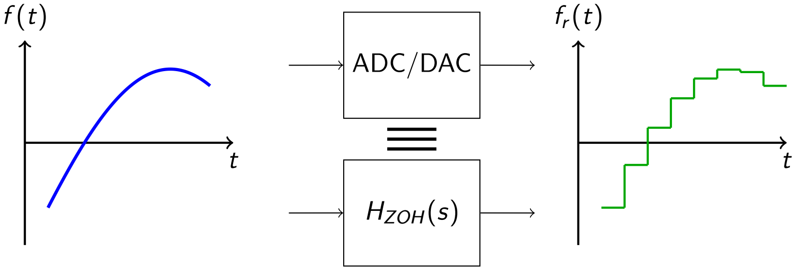
With an infinite resolution and neglecting decoding effects, we have[50]



[50] Often the Analog Hold is a Zero Order Hold (ZOH) !

ADC followed by DAC

With an infinite resolution and neglecting decoding effects, we have



where

$$H_{ZOH} = \frac{1}{s}\left(1 - e^{-T_s\, s}\right)$$

7. Discrete-time signals and systems

Discrete-time signals
Discrete-time systems

Discrete-time signals

Discrete-time signals

A discrete-time signal $x[k]$ can be tought of as a real- or complex-valued function of integer sample index k:

$$\begin{array}{rccc} x[.]: & \mathbb{Z} & \longrightarrow & \mathbb{R} \quad (\mathbb{C}) \\ & k & \longrightarrow & x[k] \end{array}$$

The signal is defined for all integers k; it is not defined for non-integer values of k.

Although in many situations, discrete-time signals are obtained from continuous-time signals by sampling, that is not always the case. Many signals are inherently discrete: final values attained daily by company shares.

Fibonacci sequence: $x[n+1] = x[n] + x[n-1]$, $n \geq 2$, $x[0] = 1$, $x[1] = 1$.

Periodic signals

A discrete-time signal $x[k]$ is periodic if

- it is defined for all possible integer values of k, $-\infty < k < \infty$ and
- there is a positive integer N, the period of $x[k]$, such that

$$x[k+N] = x[k], \;\; \forall k \in \mathbb{Z}.$$

A discrete sinusoid is **not necessarily** periodic.

A sequence $x[k] = A\cos(\Omega_0 k + \theta)$ is periodic **only if** there exist non-divisible positive integers N and m such that $\Omega_0 = \frac{2\pi m}{N}$, i.e. Ω_0 is a rational multiple of 2π. The period is N. Indeed,

$$\begin{aligned} x[k+N] = A\cos\left(\frac{2\pi m}{N}(k+N)+\theta\right) &= A\cos\left(\frac{2\pi m}{N}k + 2\pi m + \theta\right) \\ &= A\cos\left(\frac{2\pi m}{N}k+\theta\right) = x[k] \end{aligned}$$

Sampling of periodic signals

When sampling an analog sinusoid $x(t) = A\cos(\omega_0 t + \theta)$ of period T, the discrete signal

$$x[k] = A\cos(\omega_0 k T_s + \theta) = A\cos\left(\frac{2\pi T_s}{T} k + \theta\right)$$

is obtained. This signal is periodic if

$$\frac{T_s}{T} = \frac{m}{N} \iff mT = NT_s$$

for positive integers N and m, which are not divisible by each other.

To avoid frequency aliasing the sampling period should also satisfy

$$\omega_s \geq 2\omega_0 \iff T_s \leq \frac{\pi}{\omega_0} = \frac{T}{2}.$$

Periodic signals

The discrete signal

$$x[k] = A\cos(\omega_0 k T_s + \theta) = A\cos\left(\frac{2\pi T_s}{T} k + \theta\right)$$

is periodic if

$$\frac{T_s}{T} = \frac{m}{N} \iff mT = NT_s$$

for positive integers N and m, which are not divisible by each other.

This conditions says that a period $(m = 1)$ or several periods $(m > 1)$ should be divided into $N > 0$ segments of duration T_s seconds.

If the condition is **not** satisfied, then the discretized sinusoid is **not periodic**.

Energy

Energy

The energy of a discrete-time signal is defined as

$$E = \sum_{k=-\infty}^{\infty} |x[k]|^2$$

The signal $x[k]$ is said to have **finite energy** or to be **square summable** if $E < \infty$.

The signal $x[k]$ is called **absolutely summable** if

$$\sum_{k=-\infty}^{\infty} |x[k]| < \infty.$$

Power

Power

If a signal has infinite power, the average power is defined as

$$P = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{k=-N}^{N} |x[k]|^2$$

The signal $x[k]$ is said to have **finite power** if $P < \infty$.

Example: discrete unit step[51] (infinite energy)

$$P = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{0}^{N} 1 = \lim_{N\to\infty} \frac{N+1}{2N+1} = \frac{1}{2}$$

[51] The discrete unit step is introduced subsequently.

Time shifting and reflection

Time shifting and reflection are very similar to the continuous-time cases, the only difference being that the operations are done using integers.

Time shifting and reflection

Given a positive integer N and a discrete-time signal $x[k]$.

- The discrete-time signal $y[k]$ is $x[k]$ **delayed** by N samples if $y[k] = x[k-N]$, i.e. $x[k]$ shifted to the **right** N samples.
- The discrete-time signal $y[k]$ is $x[k]$ **advanced** by N samples if $y[k] = x[k+N]$, i.e. $x[k]$ shifted to the **left** N samples.
- The discrete-time signal $y[k]$ is $x[k]$ **reflected** if $y[k] = x[-k]$.

Even and odd signals

Even and odd signals

A discrete signal is

$$
\begin{aligned}
x[k] \text{ is even} &\iff x[k] = x[-k] \\
x[k] \text{ is odd} &\iff x[k] = -x[-k]
\end{aligned}
$$

A discrete signal $x[k]$ is representable as the sum of an even component and an odd component, i.e.

$$
\begin{aligned}
x[k] &= \underbrace{\frac{1}{2}(x[k] + x[-k])}_{x_e[k]} + \underbrace{\frac{1}{2}(x[k] - x[-k])}_{x_o[k]}, \\
&= x_e[k] + x_o[k].
\end{aligned}
$$

Discrete-time complex exponential

Discrete-time complex exponential

A discrete-time complex exponential is a signal of the form

$$\begin{aligned} x[k] &= A\,\alpha^k = |A||\alpha|^k e^{j(\Omega_0 k+\theta)} \\ &= |A||\alpha|^k(\cos(\Omega_0 k+\theta)+j\sin(\Omega_0 k+\theta)) \end{aligned}$$

with $A = |A|e^{j\theta}$ and $\alpha = |\alpha|e^{j\Omega_0}$.

The discrete-time complex exponential looks different from its continuous-time counterpart. Sample a continuous-time exponential

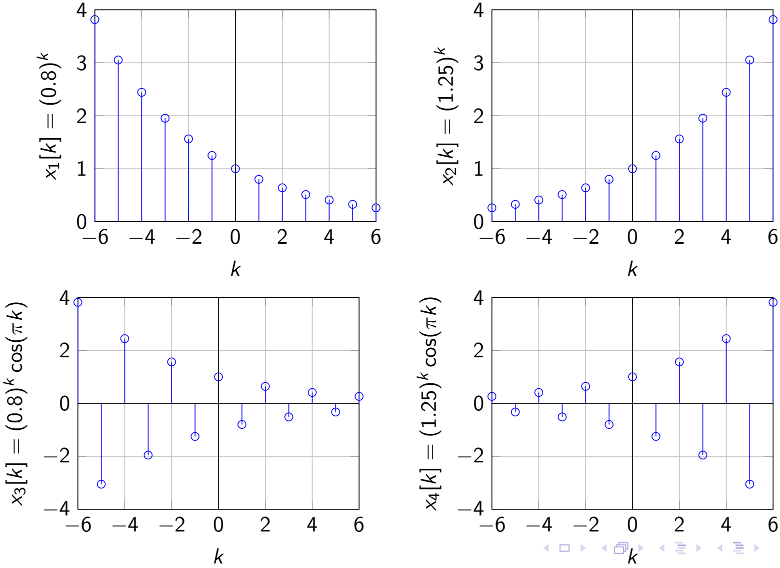
$$x(t) = A\,e^{(a+j\omega_0)t}, \quad A \text{ real}$$

using a sampling period T_s. The sampled signal is

$$x[k] = A\,e^{(a+j\omega_0)kT_s} = A\,(e^{aT_s})^k e^{j(\omega_0 T_s k)} = |A||\alpha|^k e^{j\Omega_0 k}$$

where $\alpha = e^{aT_s}$ and $\Omega_0 = \omega_0 T_s$.

Discrete-time complex exponential



Discrete-time unit-step and unit-sample

Discrete-time unit-step and unit-sample

The discrete-time unit-step $u[k]$ and unit-impulse or unit sample $\delta[k]$ are defined as

$$\begin{aligned} u[k] &= \begin{cases} 1 & k \geq 0 \\ 0 & k < 0 \end{cases} \\ \delta[k] &= \begin{cases} 1 & k = 0 \\ 0 & k \neq 0 \end{cases} \end{aligned}$$

These two signals are related as follows:

$$\begin{aligned} \delta[k] &= u[k] - u[k-1] \\ u[k] &= \sum_{n=0}^{\infty} \delta[k-n] = \sum_{m=-\infty}^{k} \delta[m] \end{aligned}$$

Generic representation of discrete-time signals

Generic representation of discrete-time signals

Any discrete-time signal $x[k]$ can be represented using shifted unit-impulse signals as

$$x[k] = \sum_{n=-\infty}^{\infty} x[n]\,\delta[k-n].$$

Properties of discrete-time systems

Linearity

A discrete-time process H is **linear**, if the superposition principle is applicable, i.e.

$$y[k] = H(\alpha_1 x_1[k] + \alpha_2 x_2[k]) = \alpha_1 H(x_1[k]) + \alpha_2 H(x_2[k]).$$

Time invariance

A discrete-time process H is **time-invariant**, if for an input $x[k]$ and corresponding output $y[k] = H(x[k])$, the output corresponding to a delayed or advanced version of $x[k]$, $x[k \pm N]$, is

$$y[k \pm N] = H(x[k \pm N]).$$

Properties of discrete-time systems

Causality

A discrete-time process H is **causal** if

- for an input $x[k] = 0$ up to sample k and zero initial conditions, this implies $y[k] = 0$.
- The output $y[k]$ does not depend on future inputs.

Convolution sum

Convolution sum

Suppose $h[k]$ is the impulse response of a Linear Time-Invariant (LTI) system, i.e. the response to a unit-impulse $\delta[k]$ with zero initial conditions. The response of the system to any input $x[k]$ is

$$y[k] = [x * h][k] = [h * x][k] = \sum_{n=-\infty}^{\infty} x[n]\, h[k-n] = \sum_{m=-\infty}^{\infty} h[m]\, x[k-m].$$

If $h[k]$ is the response to $\delta[k]$, by time invariance the response to $\delta[k-n]$ is $h[k-n]$. By superposition, the response to

$$x[k] = \sum_{n=-\infty}^{\infty} x[n]\, \delta[k-n]$$

is a sum of responses to $x[n]\, \delta[k-n]$, i.e. a sum of $x[n]\, h[k-n]$ terms.

Causal system

Causal system

Suppose $h[k]$ is the impulse response of a causal LTI systems, i.e.

$$h[k] = 0, \quad k < 0.$$

The response to a causal signal

$$x[k] = 0, \quad k < 0$$

is

$$y[k] = \sum_{n=0}^{k} x[n]\, h[k-n] = \sum_{m=0}^{k} h[m]\, x[k-m].$$

The lower limit of the first sum results from the causality of the input signal. The upper limit of the first sum results from the causality of the impulse response.

Graphical convolution

Suppose the input signal $x[k]$ and the impulse response $h[k]$ are both causal signals.

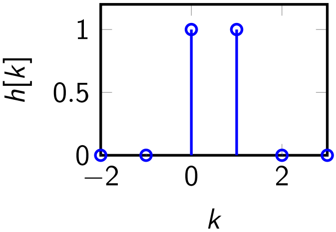
Graphical convolution

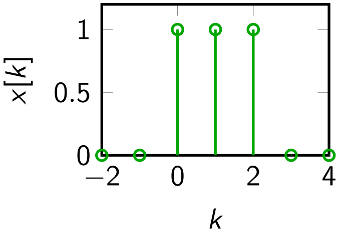
- Multiply $x[n]$ with a version of the impulse response $h[k-n]$ that is reflected and shifted by k samples to the right.
- Sum the resulting product from 0 to k.

- The convolution sum is computationally quite expensive. A more efficient method will be to use the z-transform.
- The convolution of signals of length (supports) M and N results in a signal of length $M+N-1$.

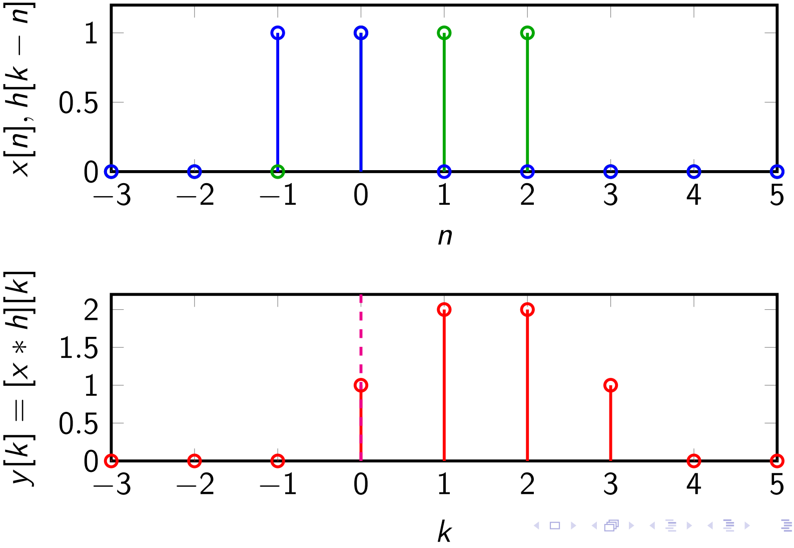
Example

- Impulse response: $h[k] = u[k] - u[k-2]$
- Input: $x[k] = u[k] - u[k-3]$

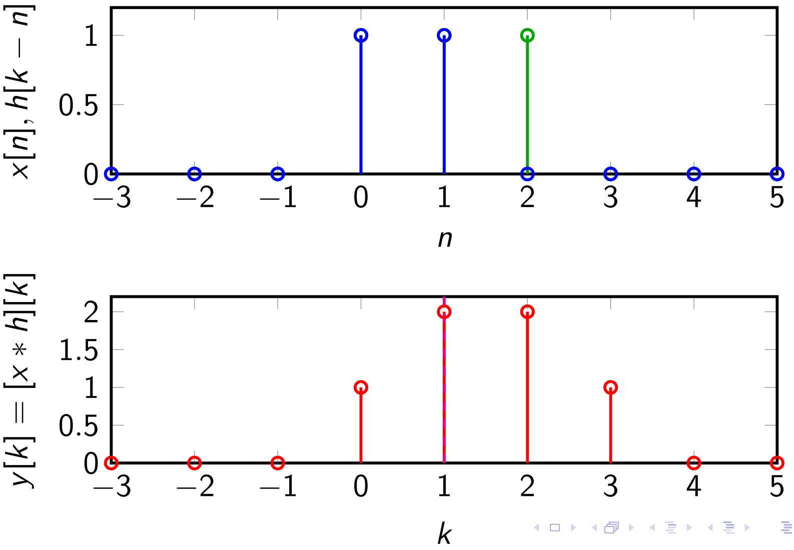




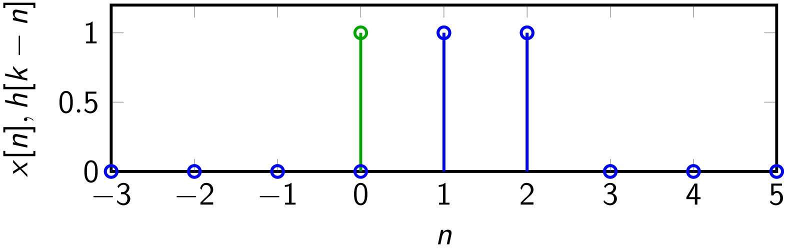
Example ($k = 0$)

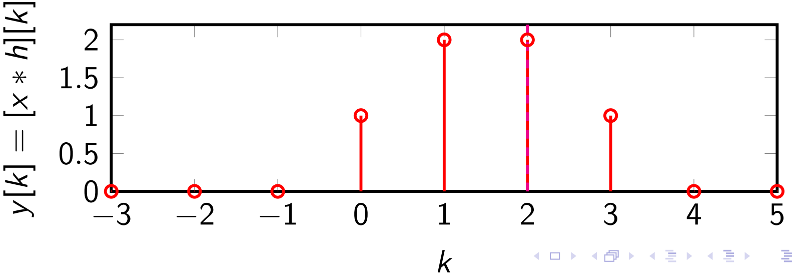


Example ($k = 1$)

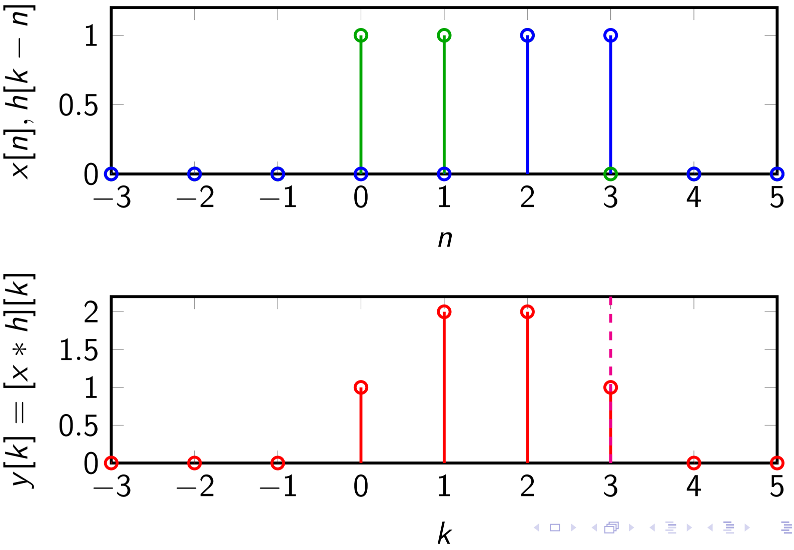


Example ($k = 2$)

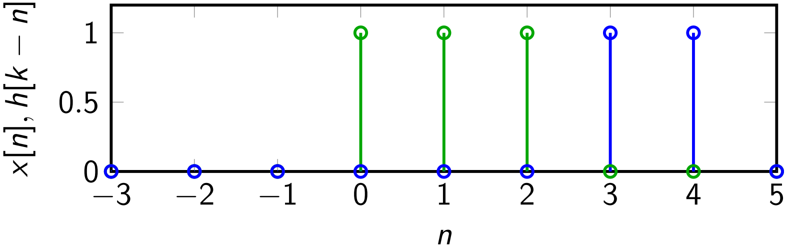


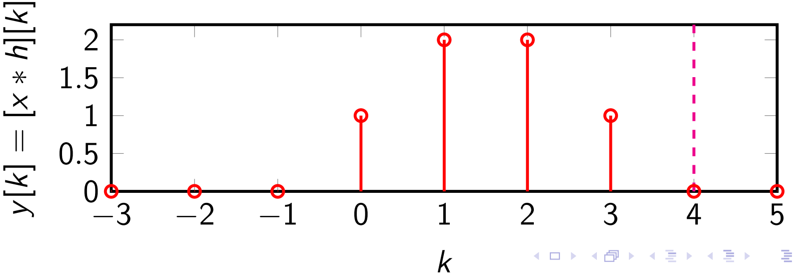


Example ($k = 3$)

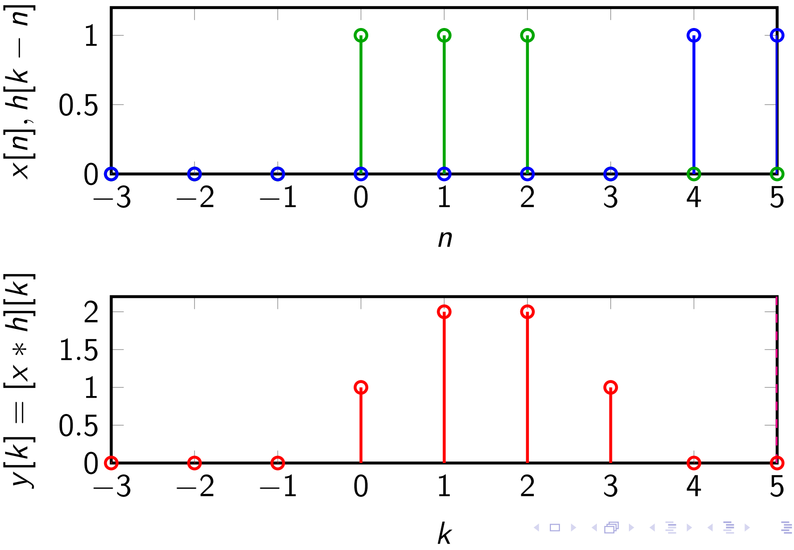


Example ($k = 4$)





Example ($k = 5$)



Difference equations

Difference equations in the discrete-time case are analogous to differential equations in the continuous-time case.

This type of equations with constant coefficients are used to describe discrete-time systems using the addition, multiplication and time-shifting operations.

Recursive system

In general, the relation between the input $x[k]$ and the output $y[k]$ can be written

$$y[k] = -\sum_{m=1}^{N} a_m \, y[k-m] + \sum_{m=0}^{M} b_m \, x[k-m], \quad k > 0$$

with initial conditions $y[-k], k = 1, \cdots, N$. This system is **recursive** and is also called an **Infinite Impulse Response** (IIR) system.

FIR system

The second type of discrete-time system is a finite impulse response system, also called a non-recursive system.

FIR system or non-recursive system

The relation between the $x[k]$ and the output $y[k]$ can be written as

$$y[k] = \sum_{m=0}^{M} b_m \, x[k-m]$$

There are no initial conditions as the system is **non-recursive**. This system is also called a **Finite Impulse Response** (FIR) system.

FIR system

FIR system or non-recursive system

The response of a non recursive filter is the convolution between a causal input signal $x[k]$ and the finite impulse response

$$h[k] = \begin{cases} b_k, & k = 0, \cdots, M \\ 0 & \text{ailleurs} \end{cases}$$

i.e.

$$y[k] = [h * x][k] = \sum_{m=-\infty}^{\infty} h[m]\, x[k-m] = \sum_{m=0}^{M} b_m\, x[k-m].$$

BIBO stability

BIBO stability

A LTI discrete-time system is called Bounded-Input-Bounded-Output (BIBO) stable if and only if for any bounded input $|x[k]| < M$, the output $y[k]$ is bounded.

A LTI discrete-time system is BIBO stable, if its impulse response $h[k]$ est absolutely summable,

$$\sum_k |h[k]| < \infty.$$

A much simpler way to test the stability of an IIR system will be based on the location of the poles of the z-transform. Note that a FIR system is always stable.

BIBO stability: sufficiency

Assume that the input $x[k]$ is bounded, i.e. that there is a bounded M for which $|x[k]| < M$ for all k. Then

$$\begin{aligned} |y[k]| = \left| \sum_{n=-\infty}^{\infty} x[n]\, h[k-n] \right| &= \left| \sum_{m=-\infty}^{\infty} h[m]\, x[k-m] \right| \\ &\leq \sum_{m=-\infty}^{\infty} |h[m]|\, |x[k-m]| \\ &\leq M \sum_{m=-\infty}^{\infty} |h[m]|. \end{aligned}$$

It follows that

$$\sum_{m=-\infty}^{\infty} |h[m]| < \infty$$

is a sufficient condition for BIBO stability[52].

[52] It is also possible to show that this condition is necessary.

8. Z-transform

Reminder

Geometrical series

Geometrical series with $k_0 \leq k_1, k_0, k_1 \in \mathbb{Z}$

$$S_{k_0,k_1} = \sum_{k=k_0}^{k_1} q^k = \frac{q^{k_0} - q^{k_1+1}}{1-q}$$

Proof by induction: The property is verified for $k_1 = k_0$.
Suppose that the property is verified for $k_1 = k$, then

$$\begin{aligned} S_{k_0,k+1} &= S_{k_0,k} + q^{k+1} = \frac{q^{k_0} - q^{k+1}}{1-q} + q^{k+1} \\ &= \frac{q^{k_0} - q^{k+1} + q^{k+1} - q^{k+2}}{1-q} = \frac{q^{k_0} - q^{k+2}}{1-q} \end{aligned}$$

i.e. the property is verified for $k_1 = k+1$ and therefore for all k_1.

Reminder

Geometrical series

Geometrical series with $k_1 > 0$

$$S_{0,k_1} = \sum_{k=0}^{k_1} q^k = \frac{1 - q^{k_1+1}}{1 - q}$$

When $|q| < 1$, the series converges to

$$S_{0,\infty} = \sum_{k=0}^{\infty} q^k = \frac{1}{1 - q}.$$

Z-transform: Laplace transform of a sampled system

Z-transform

The Laplace transform of a sampled system

$$x_s(t) = \sum_k x(kT_s)\,\delta(t - kT_s)$$

is given by

$$X_s(s) = \mathcal{L}[x_s(t)] = \sum_k x(kT_s)\,\mathcal{L}[\delta(t - kT_s)] = \sum_k x(kT_s)\,e^{-kT_s s}.$$

Defining $z = e^{T_s s}$, the equation can rewritten as

$$\begin{aligned} \mathcal{Z}[x(kT_s)] &= \mathcal{L}[x_s(t)]|_{z=e^{T_s s}} \\ &= \sum_k x(kT_s) z^{-k} \end{aligned}$$

which is the z-transform of the sampled signal $x(kT_s)$.

Mapping the Laplace plane into the z-plane

The relation $z = e^{T_s s}$ give a connection between the Laplace plane and the z-plane:

$$z = e^{T_s s} = e^{(\sigma + j\omega)T_s} = e^{\sigma T_s} e^{j\omega T_s}$$

Defining $r = e^{\sigma T_s}$ et $\Omega = \omega T_s$, one obtains

$$z = r\, e^{j\Omega}$$

which is a complex variable in polar coordinates, with a radius $0 \leq r < \infty$ and angles Ω in radians. The variable r corresponds to a damping factor and Ω is a discrete frequency expressed in radians.

The z-plane corresponds to circles of radius r and angles $-\pi \leq \Omega < \pi$.

Mapping the Laplace plane into the z-plane

Let us study the correspondence $z = e^{T_s s}$ between the z-domain and the Laplace domain.

Consider strips of width $\frac{2\pi}{T_s}$ in the left and right halve planes of the Laplace domain.

The width of the strip is of course related to the Nyquist condition establishing that the analog signals that we are considering have a maximum frequency $\omega_N = \frac{\omega_s}{2} = \frac{\pi}{T_s}$ where T_s is the sampling period.

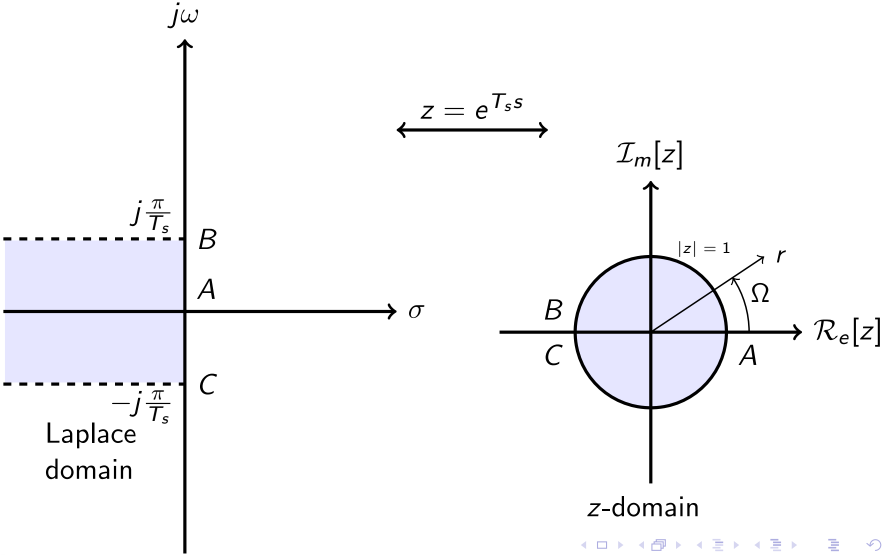
If $T_s \to 0$, one would consider the class of signals with maximum frequency approaching ∞, i.e. the entire left and right halve planes.

Mapping the Laplace plane into the z-plane

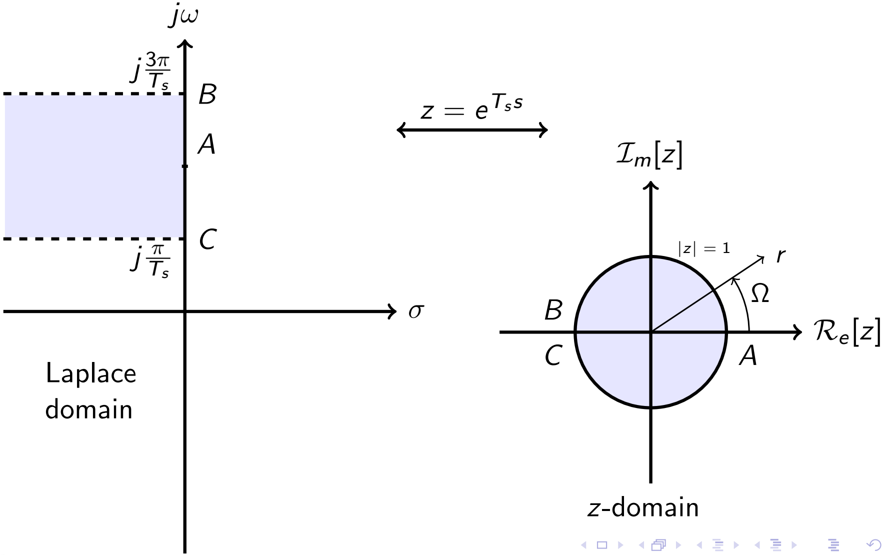
The relation $z = e^{T_s s}$ maps

- the real part $s = \sigma + j\omega$, $\mathcal{R}_e[s] = \sigma$ into the radius $r = e^{\sigma T_s}$ with
  - $0 \leq r \leq 1$ when $\sigma \leq 0$,
  - $r > 1$ when $\sigma > 0$,
- the analog frequencies
  - $-\dfrac{\pi}{T_s} \leq \omega \leq \dfrac{\pi}{T_s}$ on $-\pi \leq \Omega \leq \pi$,
  - $\dfrac{-\pi + 2k\pi}{T_s} \leq \omega \leq \dfrac{\pi + 2k\pi}{T_s}$ on $-\pi + 2k\pi \leq \Omega \leq \pi + 2k\pi$

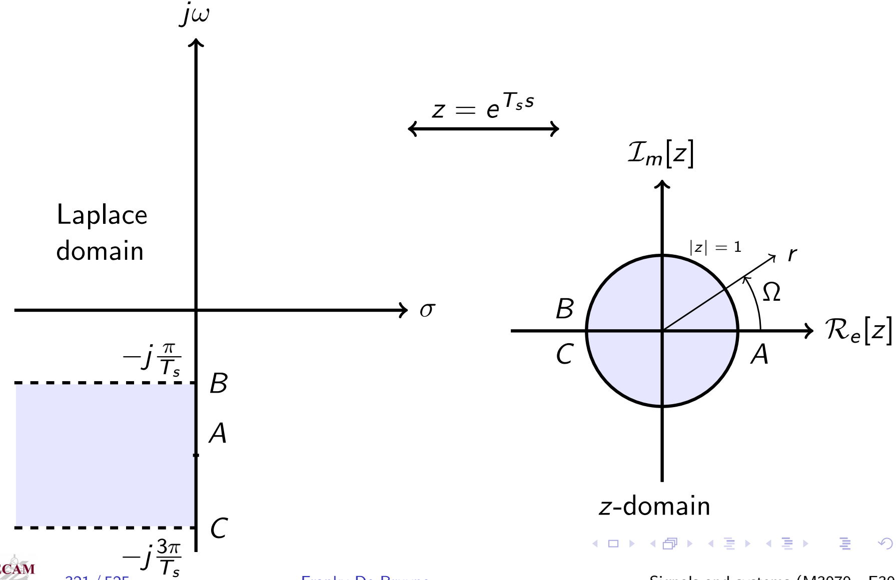
Mapping the Laplace plane into the z-plane



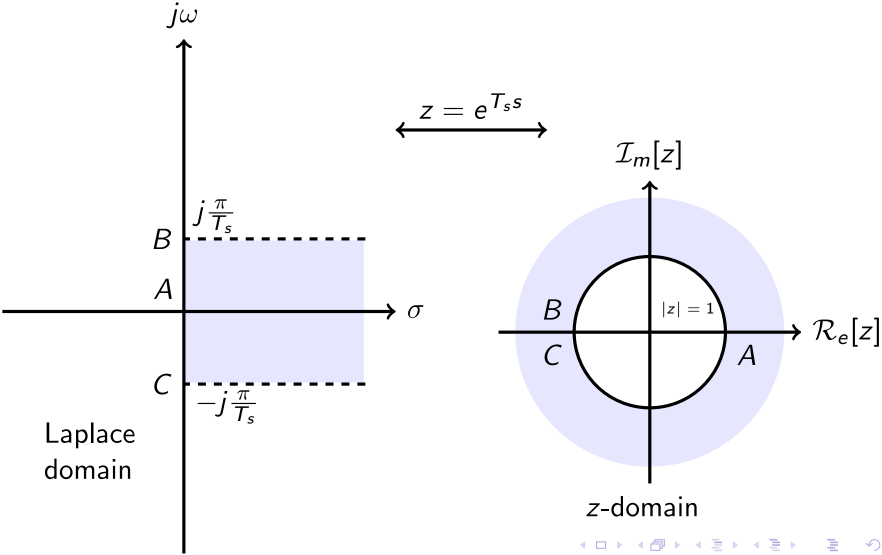
Mapping the Laplace plane into the z-plane



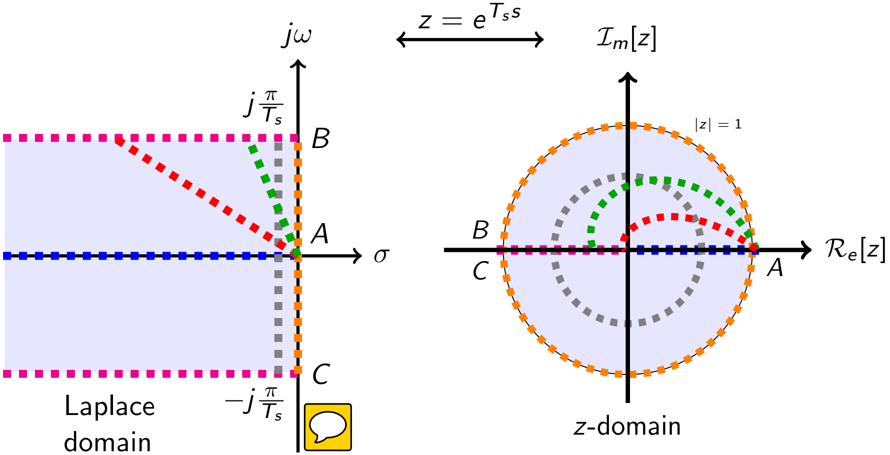
Mapping the Laplace plane into the z-plane



Mapping the Laplace plane into the z-plane



Mapping the Laplace plane into the z plane



Definition

Two-sided z-transform

Given a discrete-time signal $x[k]$, $-\infty < k < \infty$, the two-sided z-transform is

$$X(z) = \mathcal{Z}[x[k]] = \sum_{k=-\infty}^{\infty} x[k]z^{-k}.$$

This transform is defined in a Region Of Convergence (ROC) in the z-plane.

The two-sided z-transform is not useful in solving difference equations with initial conditions, just as the two-sided Laplace transform is not useful in solving differential equations with initial conditions. It is necessary to define a one-sided z-transform.

Region of convergence

For the two-sided z-transform to be defined for a discrete-time $x[k]$,

$$
\begin{aligned}
|X(z)| = \left| \sum_{k=-\infty}^{\infty} x[k] z^{-k} \right| &\leq \sum_{k=-\infty}^{\infty} |x[k] r^{-k} e^{-jk\Omega}| \\
&= \sum_{k=-\infty}^{\infty} |x[k]\, r^{-k}| < \infty,
\end{aligned}
$$

that is $x[k]\, r^{-k}$ must be absolutely summable[53]. This is often the case by an appropriate choice of r even if $x[k]$ itself is not absolutely summable. The values of r for which the integral converge define the ROC; the discrete frequency Ω does not affect the ROC.

$$
\mathrm{ROC} = \left\{ z = r\, e^{j\Omega} \text{ such that } \sum_{k=-\infty}^{\infty} |x[k]\, r^{-k}| < \infty \right\}
$$

[53] Here we have shown the sufficiency of the condition. It is possible to show it is also necessary.

Definition

One-sided z-transform

The one-sided z-transform of a given discrete-time signal $x[k]$, causal with $x[k] = 0$ for $k < 0$, or non-causal and made causal by multiplication with a unit step $u[k]$ is

$$X(z) = \mathcal{Z}[x[k]u[k]] = \sum_{k=0}^{\infty} x[k]z^{-k}$$

with region of convergence ROC_c.

Definition

Two-sided and one-sided z-transforms

The two-sided z-transform can be expressed using the one-sided z-transform

$$X(z) = \mathcal{Z}[x[k]u[k]] + \mathcal{Z}[x[-k]u[k]]|_z - x[0]$$

The region of convergence of $X(z)$ is $\text{ROC} = \text{ROC}_c \cap \text{ROC}_{ac}$.

Here ROC_c is the region of convergence of $\mathcal{Z}[x[k]u[k]]$ and ROC_{ac} the region of convergence of $\mathcal{Z}[x[-k]u[k]]|_z$.

Poles and zeros

Poles and zeros

The poles of the z transfer function $X(z)$, are the complex valued $\{p_k\}$ such that

$$X(p_k) \to \infty.$$

The zeros of the z transfer function $X(z)$, are the complex valued $\{z_k\}$ such that

$$X(z_k) = 0.$$

If the poles or zeros are complex, they come in complex conjugate pairs.

Region of convergence of a finite support signal

The region of convergence of a signal $x[k]$ of finite support $[k_0, k_1]$ where $-\infty < k_0 \leq k \leq k_1 < \infty$ is

$$X(z) = \mathcal{Z}[x[k]] = \sum_{k=k_0}^{k_1} x[k] z^{-k}.$$

Its region of convergence is the whole z-plane.

It is sometimes necessary to exclude $z = 0$ and $z = \pm\infty$ from the region of convergence depending on the values of k_0 and k_1.

ROC for causal signals of infinite support

The z-transform of a causal $x_c[k]$ with infinite support is

$$X_c(z) = \sum_{k=0}^{\infty} x_c[k] z^{-k} = \sum_{k=0}^{\infty} x_c[k] r^{-k} e^{-jk\Omega}.$$

For the series to converge, we need to determine appropriate values of r, the damping factor. The discrete frequency Ω has no influence on the ROC.

If R_1 is the radius of the farthest poles of $X_c(z)$, then there is an exponential $R_1^k\, u[k]$ such that $x_c[k] < M\, R_1^k$ for $k \geq 0$ and some value $M > 0$. Then for $X_c(z)$ to converge, we need

$$|X_c(z)| \leq \sum_{k=0}^{\infty} |x_c[k]||r^{-k}| < M \sum_{k=0}^{\infty} \left| \frac{R_1}{r} \right|^k < \infty$$

We need $|z| = r > R_1$. This also indicates that the ROC does not include any of the poles of $X_c(z)$.

ROC for signals of infinite support

For a signal $x[k]$ of infinite support which is

- **causal**, the z-transform has a ROC $|z| > R_1$ where R_1 is the largest radius of the poles of $X_c(z)$, i.e. the region of convergence is outside a circle of radius R_1.
- **anti-causal**, the z-transform has a ROC $|z| < R_2$ where R_2 is smallest radius of the poles of $X(z)$, i.e. the region of convergence is inside a circle of radius R_2.
- **non-causal**, the z-transform has a ROC $R_1 < |z| < R_2$ where R_1 and R_2 are, respectively, the largest and smallest radius of the poles of $X_c(z)$ and $X_{ac}(z)$, the z-transform of the causal and anti-causal parts of $x[k]$.

Discrete unit-impulse

The z-transform of a discrete unit-impulse $\delta[k]$ is

$$\mathcal{Z}[\delta[k]] = \sum_{k=-\infty}^{\infty} \delta[k] z^{-k} = \sum_{k=0}^{\infty} \delta[k] z^{0} = 1.$$

There are no conditions on z for the sum to converge. The ROC is the whole z-domain.

Discrete unit-step

The z-transform of a discrete unit-step $u[k]$ is

$$\begin{aligned}\mathcal{Z}[u[k]] &= \sum_{k=-\infty}^{\infty} u[k]z^{-k} = \sum_{k=0}^{\infty} z^{-k}, \\ &= \frac{1}{1-z^{-1}}.\end{aligned}$$

The sum converges if $|z| > 1$. The region of convergence is outside the unit circle $|z| = 1$.

Discrete complex exponentials

For discrete complex exponentials of the form $x[k] = \alpha^k u[k]$

$$\begin{aligned} \mathcal{Z}[\alpha^k u[k]] &= \sum_{k=-\infty}^{\infty} \alpha^k u[k] z^{-k} = \sum_{k=0}^{\infty} (\alpha z^{-1})^k, \\ &= \frac{1}{1-\alpha z^{-1}}. \end{aligned}$$

The sum converges if $|z| > |\alpha|$.The region of convergence is outside the circle of radius $|\alpha|$.

When α is **real** and

- **positive**: the exponential signal is less and less damped for α increasing in the interval $[0, 1[$. For $\alpha > 1$, the series diverges. The value $\alpha = 1$ corresponds a unit step.
- **negative**: the exponential signal changes sign each sample. The modulated exponential signal is less and less damped for α decreasing in the interval $[0, -1[$. For $\alpha < -1$, the series diverges.

Discrete complex exponentials

To compute the z-transform of $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$, the Euler identity is used to obtain

$$x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k] = \frac{1}{2}\left[r^k\, e^{j(\Omega_0 k+\theta)} + r^k\, e^{-j(\Omega_0 k+\theta)}\right] u[k]$$

Using the linearity property and the z-transform with $\alpha = re^{j\Omega_0}$ and its complex conjugate $\alpha^* = re^{-j\Omega_0}$, one obtains

$$\begin{aligned} X(z) &= \frac{1}{2}\left[\frac{e^{j\theta}}{1 - r\, e^{j\Omega_0}\, z^{-1}} + \frac{e^{-j\theta}}{1 - r\, e^{-j\Omega_0}\, z^{-1}}\right] \\ &= \frac{1}{2}\left[\frac{2\cos(\theta) - 2\, r\, \cos(\Omega_0 - \theta)\, z^{-1}}{1 - 2\, r\, \cos(\Omega_0)\, z^{-1} + r^2\, z^{-2}}\right] \\ &= \frac{\cos(\theta) - r\, \cos(\Omega_0 - \theta)\, z^{-1}}{1 - 2\, r\, \cos(\Omega_0)\, z^{-1} + r^2\, z^{-2}} \end{aligned}$$

with ROC the outside of the unit circle, i.e. $|z| > r$.

Discrete complex exponentials

The z-transform of $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$:

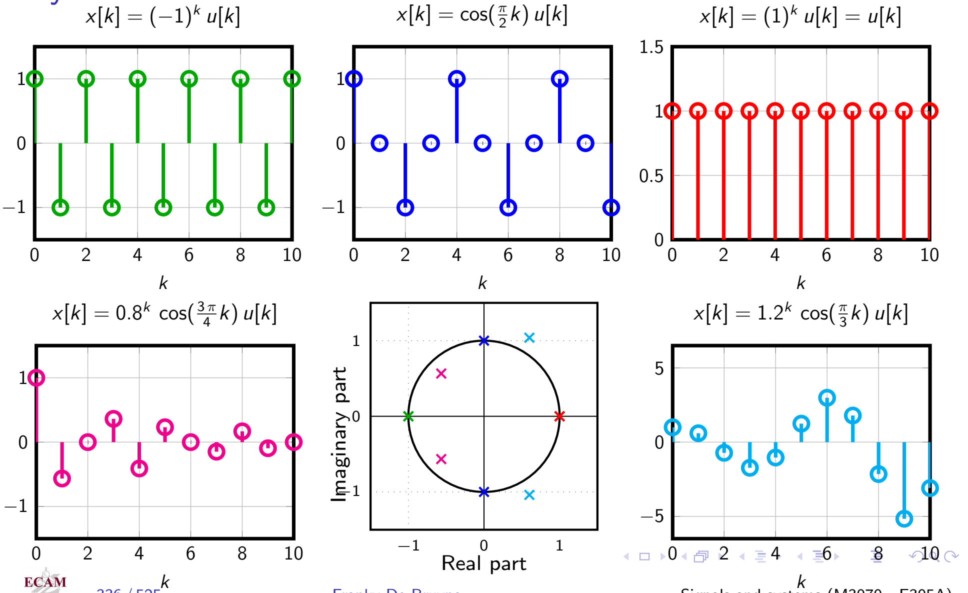
$$\mathcal{Z}[r^k \cos(\Omega_0 k + \theta)\, u[k]] = \frac{\cos(\theta) - r \cos(\Omega_0 - \theta)\, z^{-1}}{1 - 2\, r \cos(\Omega_0)\, z^{-1} + r^2\, z^{-2}}$$

In particular,

- $\mathcal{Z}[\cos(\Omega_0 k) u[k]] = \dfrac{z(z - \cos(\Omega_0))}{(z - e^{j\Omega_0})(z - e^{-j\Omega_0})} \quad (\theta = 0,\ r = 1)$
- $\mathcal{Z}[\sin(\Omega_0 k) u[k]] = \dfrac{z \sin(\Omega_0)}{(z - e^{j\Omega_0})(z - e^{-j\Omega_0})} \quad (\theta = -\frac{\pi}{2},\ r = 1)$

with ROC the outside of the unit circle, i.e. $|z| > 1$.

Dynamic behaviour

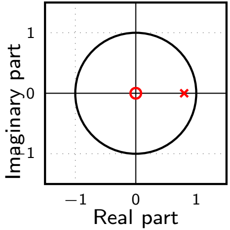


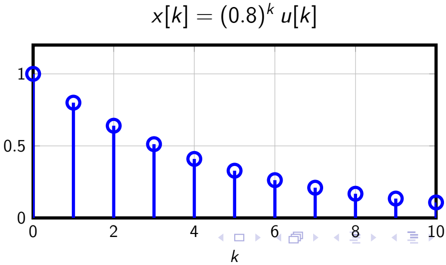
Dynamic behaviour: real exponentials

Discrete complex exponentials $x[k] = \alpha^k u[k]$ with α real

$$\begin{aligned}\mathcal{Z}[\alpha^k u[k]] &= \sum_{k=-\infty}^{\infty} \alpha^k u[k] z^{-k} = \sum_{k=0}^{\infty} (\alpha z^{-1})^k \\ &= \frac{1}{1-\alpha z^{-1}} = \frac{z}{z-\alpha}\end{aligned}$$

which has a single real pole at $z = \alpha$, plus a zero at $z = 0$. We have the following correspondence between pole location and pulse response.



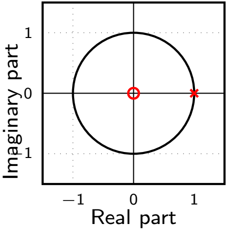


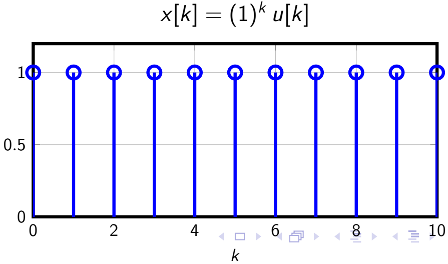
Dynamic behaviour: real exponentials

Discrete complex exponentials $x[k] = \alpha^k u[k]$ with α real

$$\begin{aligned}\mathcal{Z}[\alpha^k u[k]] &= \sum_{k=-\infty}^{\infty} \alpha^k u[k] z^{-k} = \sum_{k=0}^{\infty} (\alpha z^{-1})^k \\ &= \frac{1}{1-\alpha z^{-1}} = \frac{z}{z-\alpha}\end{aligned}$$

which has a single real pole at $z = \alpha$, plus a zero at $z = 0$. We have the following correspondence between pole location and pulse response.



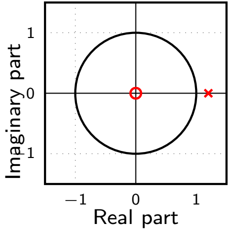


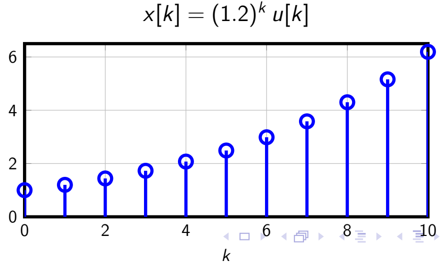
Dynamic behaviour: real exponentials

Discrete complex exponentials $x[k] = \alpha^k u[k]$ with α real

$$\begin{aligned} \mathcal{Z}[\alpha^k u[k]] &= \sum_{k=-\infty}^{\infty} \alpha^k u[k] z^{-k} = \sum_{k=0}^{\infty} (\alpha z^{-1})^k \\ &= \frac{1}{1-\alpha z^{-1}} = \frac{z}{z-\alpha} \end{aligned}$$

which has a single real pole at $z = \alpha$, plus a zero at $z = 0$. We have the following correspondence between pole location and pulse response.



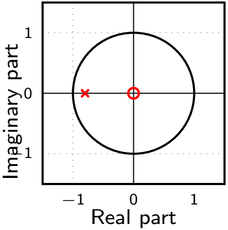


Dynamic behaviour: real exponentials

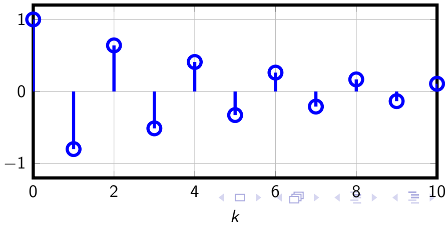
Discrete complex exponentials $x[k] = \alpha^k u[k]$ with α real

$$\mathcal{Z}[\alpha^k u[k]] = \sum_{k=-\infty}^{\infty} \alpha^k u[k] z^{-k} = \sum_{k=0}^{\infty} (\alpha z^{-1})^k$$
$$= \frac{1}{1-\alpha z^{-1}} = \frac{z}{z-\alpha}$$

which has a single real pole at $z = \alpha$, plus a zero at $z = 0$. We have the following correspondence between pole location and pulse response.



$x[k] = (-0.8)^k u[k]$

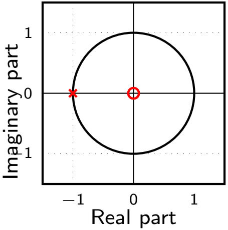


Dynamic behaviour: real exponentials

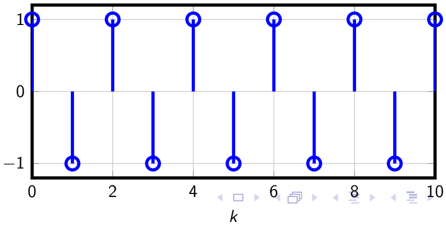
Discrete complex exponentials $x[k] = \alpha^k u[k]$ with α real

$$\begin{aligned}\mathcal{Z}[\alpha^k u[k]] &= \sum_{k=-\infty}^{\infty} \alpha^k u[k] z^{-k} = \sum_{k=0}^{\infty} (\alpha z^{-1})^k \\ &= \frac{1}{1-\alpha z^{-1}} = \frac{z}{z-\alpha}\end{aligned}$$

which has a single real pole at $z = \alpha$, plus a zero at $z = 0$. We have the following correspondence between pole location and pulse response.



$x[k] = (-1)^k\, u[k]$

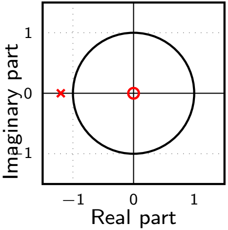


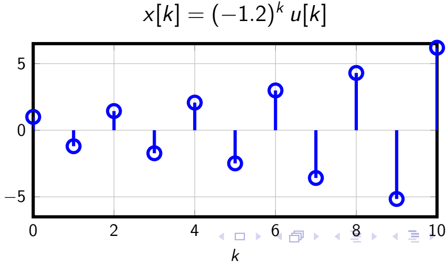
Dynamic behaviour: real exponentials

Discrete complex exponentials $x[k] = \alpha^k u[k]$ with α real

$$\begin{aligned}\mathcal{Z}[\alpha^k u[k]] &= \sum_{k=-\infty}^{\infty} \alpha^k u[k] z^{-k} = \sum_{k=0}^{\infty} (\alpha z^{-1})^k \\ &= \frac{1}{1-\alpha z^{-1}} = \frac{z}{z-\alpha}\end{aligned}$$

which has a single real pole at $z = \alpha$, plus a zero at $z = 0$. We have the following correspondence between pole location and pulse response.





Dynamic behaviour: complex exponentials

To compute the z-transform of $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$, the Euler identity is used to obtain

$$x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k] = \frac{1}{2}\left[r^k\, e^{j(\Omega_0 k + \theta)} + r^k\, e^{-j(\Omega_0 k + \theta)}\right] u[k]$$

Using the linearity property and the z-transform with $\alpha = re^{j\Omega_0}$ and its complex conjugate $\alpha^* = re^{-j\Omega_0}$, one obtains

$$\begin{aligned} X(z) &= \frac{1}{2}\left[\frac{z\, e^{j\theta}}{z - r\, e^{j\Omega_0}} + \frac{z\, e^{-j\theta}}{z - r\, e^{-j\Omega_0}}\right] \\ &= \frac{z\,(z\, \cos(\theta) - r\, \cos(\Omega_0 - \theta))}{(z - r\, e^{j\Omega_0})(z - r\, e^{-j\Omega_0})} \end{aligned}$$

The poles are complex conjugate $p_{12} = r\, e^{j\pm\Omega_0}$. The zeros are real $z_1 = 0$ and $z_2 = \frac{r\, \cos(\Omega_0 - \theta)}{\cos(\theta)}$.

Dynamic behaviour: complex exponentials

The z-transform $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$ is

$$X(z) = \frac{z\,(z\, \cos(\theta) - r\, \cos(\Omega_0 - \theta))}{(z - r\, e^{j\Omega_0})(z - r\, e^{-j\Omega_0})}$$

The poles are complex conjugate $p_{12} = r\, e^{j\pm\Omega_0}$. The zeros are real $z_1 = 0$ and $z_2 = \frac{r\, \cos(\Omega_0 - \theta)}{\cos(\theta)}$. Note that $\theta = -\frac{\pi}{2}$ corresponds to

$$x[k] = r^k \sin(\Omega_0 k)\, u[k]$$

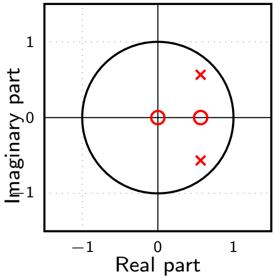
and

$$X(z) = \frac{z\, r\, \sin(\Omega_0)}{(z - r\, e^{j\Omega_0})(z - r\, e^{-j\Omega_0})}.$$

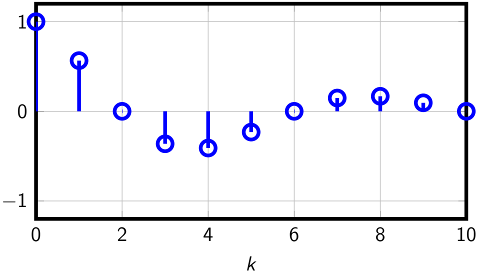
The zeros are real $z_1 = 0$ and $z_2 = \infty$. The poles are unaffected by θ phase changes.

Dynamic behaviour: complex exponentials

Discrete complex exponential $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$



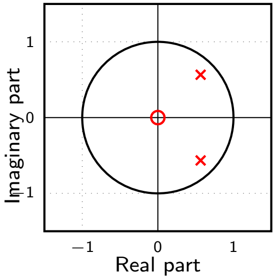
$x[k] = (0.8)^k \cos(\frac{\pi}{4} k)\, u[k]$

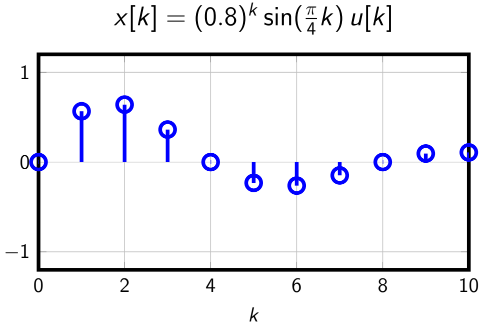


with $\Omega_0 = \frac{\pi}{4}, \theta = 0, r = 0.8.$

Dynamic behaviour: complex exponentials

Discrete complex exponential $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$

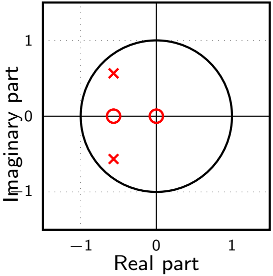




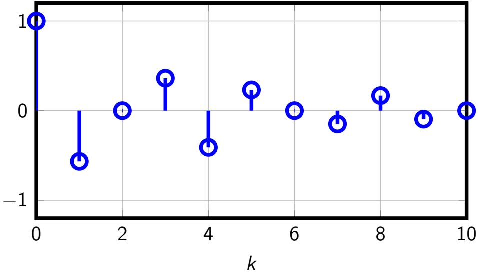
with $\Omega_0 = \frac{\pi}{4}, \theta = -\frac{\pi}{2}, r = 0.8.$

Dynamic behaviour: complex exponentials

Discrete complex exponential $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$



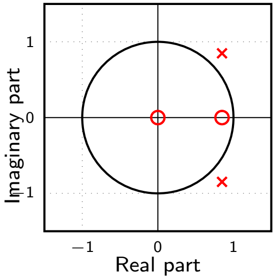
$x[k] = (0.8)^k \cos(\frac{3\pi}{4} k)\, u[k]$

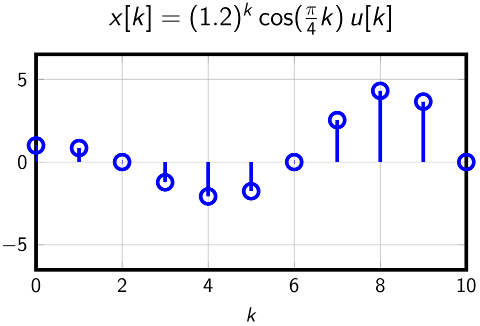


with $\Omega_0 = \frac{3\pi}{4}, \theta = 0, r = 0.8.$

Dynamic behaviour: complex exponentials

Discrete complex exponential $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$

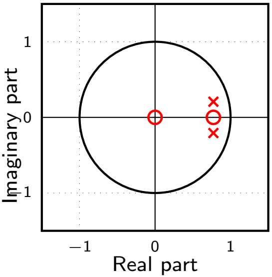




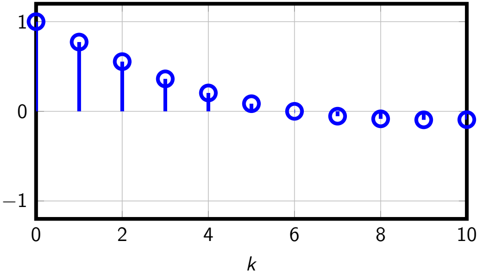
with $\Omega_0 = \frac{\pi}{4}, \theta = 0, r = 1.2.$

Dynamic behaviour: complex exponentials

Discrete complex exponential $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$



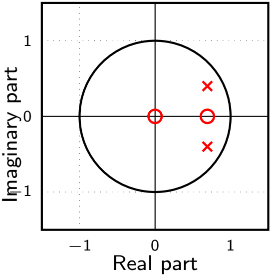
$x[k] = (0.8)^k \cos(\frac{\pi}{12} k)\, u[k]$



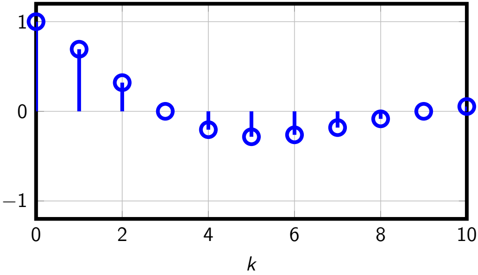
with $\Omega_0 = \frac{\pi}{12}, \theta = 0, r = 0.8.$

Dynamic behaviour: complex exponentials

Discrete complex exponential $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$



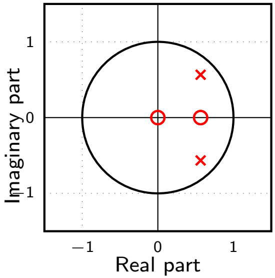
$x[k] = (0.8)^k \cos(\frac{\pi}{6} k)\, u[k]$

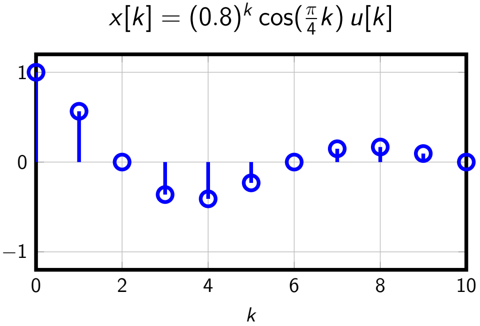


with $\Omega_0 = \frac{\pi}{6}, \theta = 0, r = 0.8.$

Dynamic behaviour: complex exponentials

Discrete complex exponential $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$

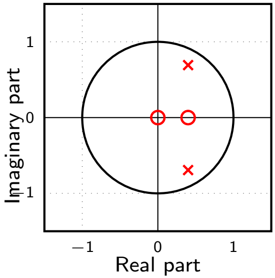




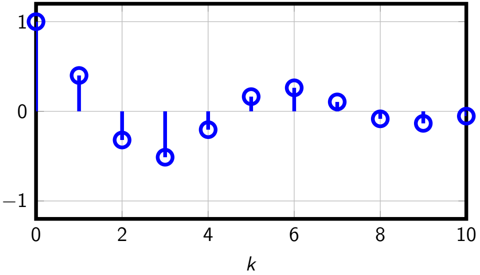
with $\Omega_0 = \frac{\pi}{4}, \theta = 0, r = 0.8.$

Dynamic behaviour: complex exponentials

Discrete complex exponential $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$



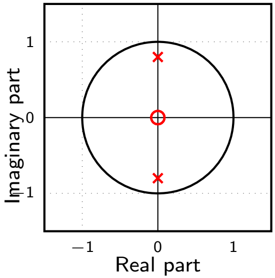
$x[k] = (0.8)^k \cos(\frac{\pi}{3} k)\, u[k]$

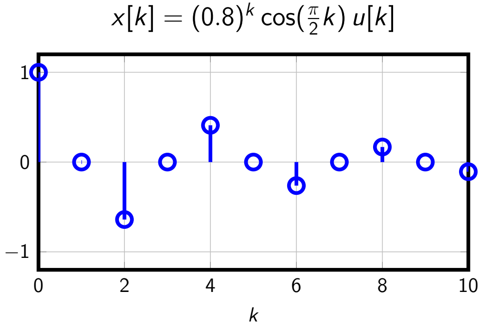


with $\Omega_0 = \frac{\pi}{3}, \theta = 0, r = 0.8.$

Dynamic behaviour: complex exponentials

Discrete complex exponential $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$



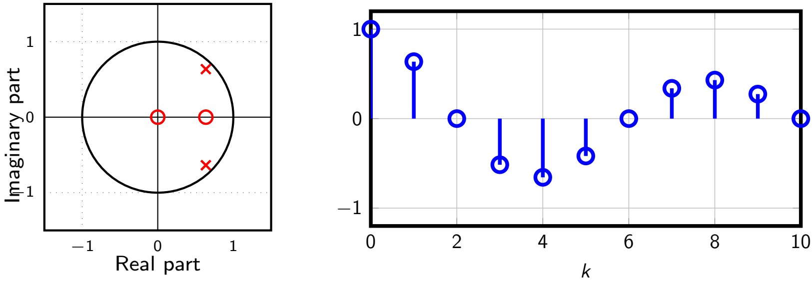


with $\Omega_0 = \frac{\pi}{2}, \theta = 0, r = 0.8.$

Dynamic behaviour: complex exponentials

Discrete complex exponential $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$

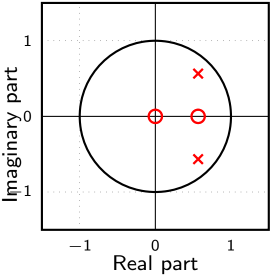
$x[k] = (0.9)^k \cos(\frac{\pi}{4} k)\, u[k]$



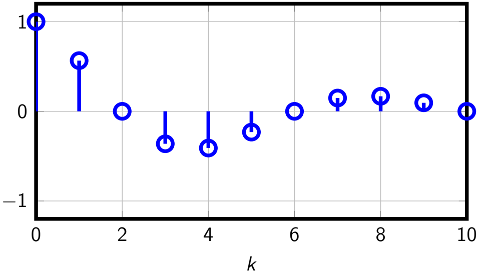
with $\Omega_0 = \frac{\pi}{4}, \theta = 0, r = 0.9.$

Dynamic behaviour: complex exponentials

Discrete complex exponential $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$



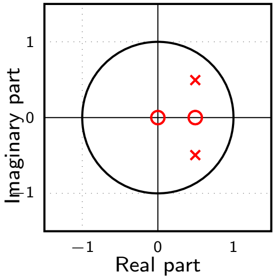
$x[k] = (0.8)^k \cos(\frac{\pi}{4}k)\, u[k]$



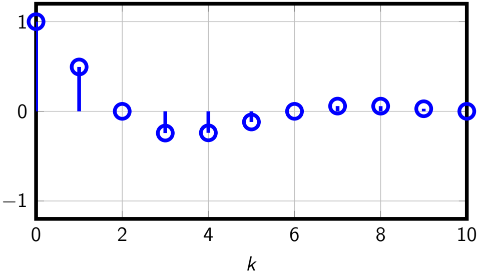
with $\Omega_0 = \frac{\pi}{4}, \theta = 0, r = 0.8.$

Dynamic behaviour: complex exponentials

Discrete complex exponential $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$



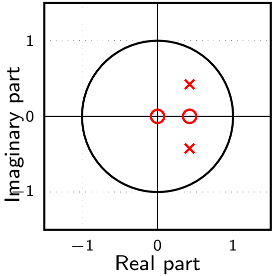
$x[k] = (0.7)^k \cos(\frac{\pi}{4} k)\, u[k]$



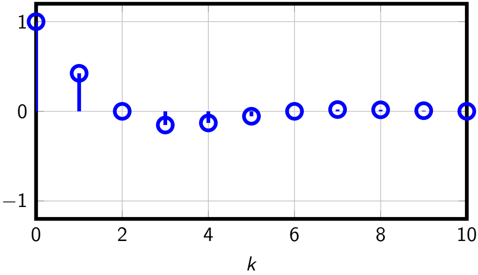
with $\Omega_0 = \frac{\pi}{4}, \theta = 0, r = 0.7$.

Dynamic behaviour: complex exponentials

Discrete complex exponential $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$



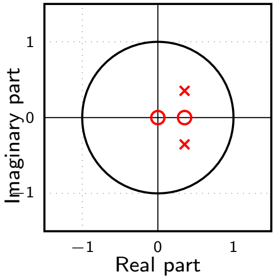
$x[k] = (0.6)^k \cos(\frac{\pi}{4}k)\, u[k]$



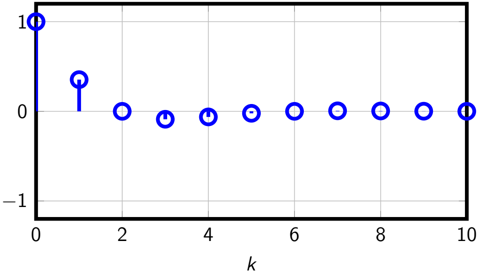
with $\Omega_0 = \frac{\pi}{4}, \theta = 0, r = 0.6.$

Dynamic behaviour: complex exponentials

Discrete complex exponential $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$



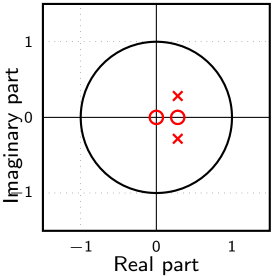
$x[k] = (0.5)^k \cos(\frac{\pi}{4} k)\, u[k]$



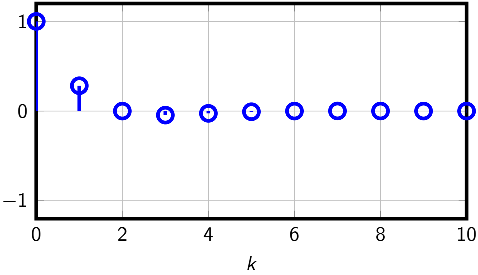
with $\Omega_0 = \frac{\pi}{4}, \theta = 0, r = 0.5.$

Dynamic behaviour: complex exponentials

Discrete complex exponential $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$



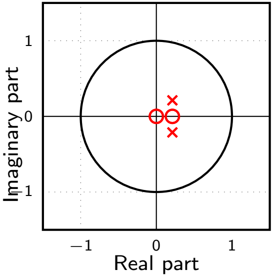
$x[k] = (0.4)^k \cos(\frac{\pi}{4} k)\, u[k]$



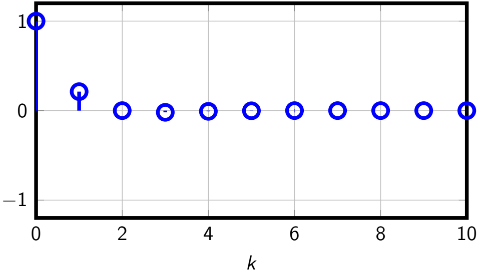
with $\Omega_0 = \frac{\pi}{4}, \theta = 0, r = 0.4.$

Dynamic behaviour: complex exponentials

Discrete complex exponential $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$



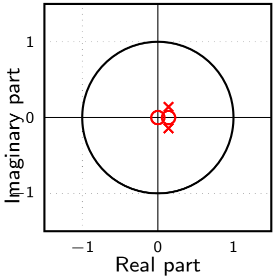
$x[k] = (0.3)^k \cos(\frac{\pi}{4} k)\, u[k]$

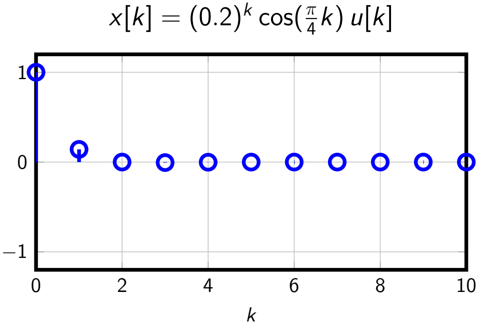


with $\Omega_0 = \frac{\pi}{4}, \theta = 0, r = 0.3$.

Dynamic behaviour: complex exponentials

Discrete complex exponential $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$

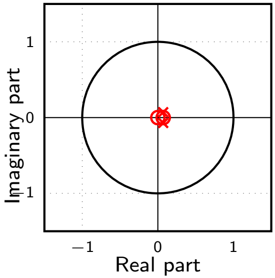




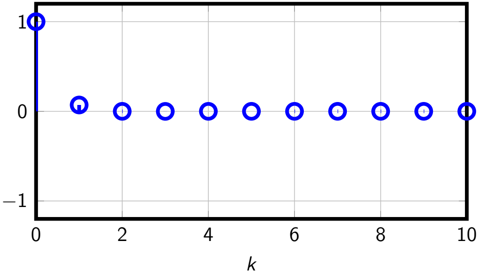
with $\Omega_0 = \frac{\pi}{4}, \theta = 0, r = 0.2.$

Dynamic behaviour: complex exponentials

Discrete complex exponential $x[k] = r^k \cos(\Omega_0 k + \theta)\, u[k]$



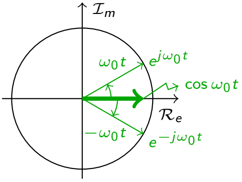
$x[k] = (0.1)^k \cos(\frac{\pi}{4} k)\, u[k]$



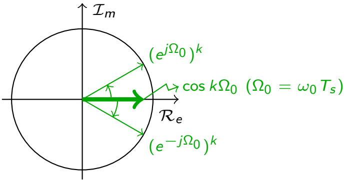
with $\Omega_0 = \frac{\pi}{4}, \theta = 0, r = 0.1$.

Sampled cosine: Nyquist OK $(\omega_0 T_s < \pi \iff \omega_0 < \omega_N)$

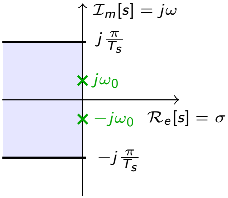
Continuous-time



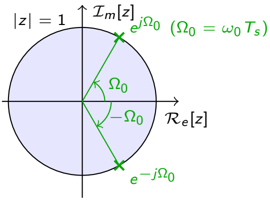
Discrete time $(t = kT_s)$



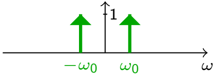
Laplace domain $(s = \sigma + j\omega)$



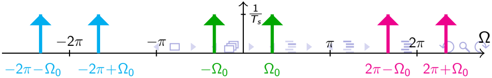
Z-domain $(z = r\, e^{j\Omega})$



Fourier transform

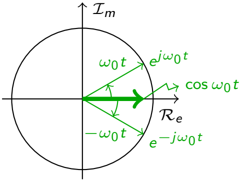


Fourier transform

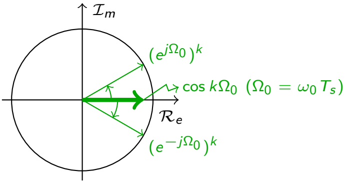


Sampled cosine: Nyquist OK $(\omega_0 T_s < \pi \iff \omega_0 < \omega_N)$

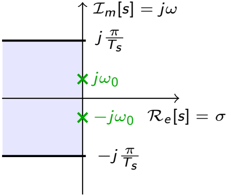
Continuous-time



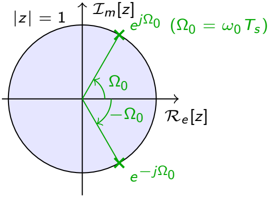
Discrete time $(t = kT_s)$



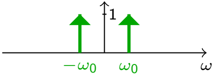
Laplace domain $(s = \sigma + j\omega)$



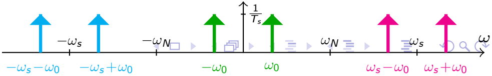
Z-domain $(z = r\, e^{j\Omega})$



Fourier transform

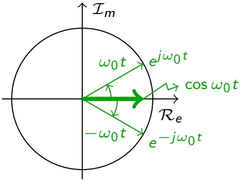


Fourier transform

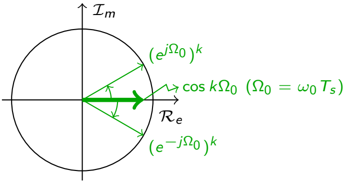


Sampled cosine: Nyquist NOK $(\omega_0 T_s > \pi \iff \omega_0 > \omega_N)$

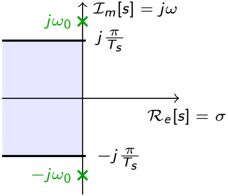
Continuous-time



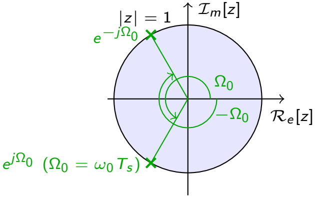
Discrete time $(t = kT_s)$



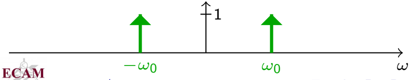
Laplace domain $(s = \sigma + j\omega)$



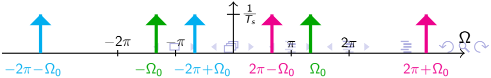
Z-domain $(z = r\,e^{j\Omega})$



Fourier transform

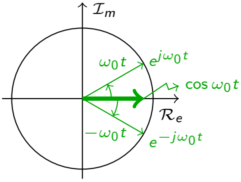


Fourier transform

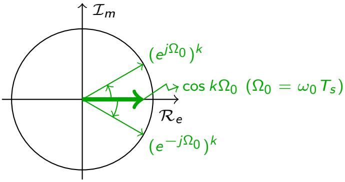


Sampled cosine: Nyquist NOK ($\omega_0 T_s > \pi \iff \omega_0 > \omega_N$)

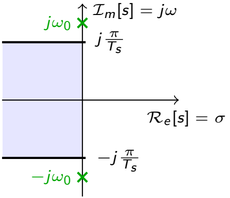
Continuous-time



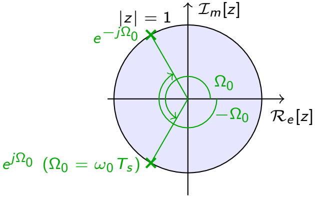
Discrete time ($t = kT_s$)



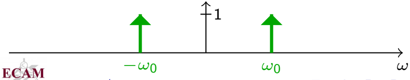
Laplace domain ($s = \sigma + j\omega$)



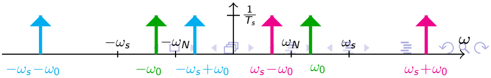
Z-domain ($z = r\,e^{j\Omega}$)



Fourier transform



Fourier transform



Linearity

Linearity

For the signals $x[k]$ and $y[k]$ with z-transform

- $\mathcal{Z}[x[k]] = X(z)$,
- $\mathcal{Z}[y[k]] = Y(z)$ and
- constants α and $\beta \in \mathbb{C}$

one has

$$\mathcal{Z}[\alpha x[k] + \beta y[k]] = \alpha\, \mathcal{Z}[x[k]] + \beta\, \mathcal{Z}[y[k]] = \alpha X(z) + \beta Y(z)$$

with ROC that is the intersection of the regions of convergence of $\mathcal{Z}[x[k]]$ and $\mathcal{Z}[y[k]]$.

Time shifting

Time shifting

For a signal $x[k]$ with z-transform $\mathcal{Z}[x[k]] = X(z)$ and $k_0 > 0$, one has

$$\begin{aligned}
\mathcal{Z}[x[k-k_0]] &= \sum_{k=0}^{\infty} x[k-k_0]z^{-k} = \sum_{n=-k_0}^{\infty} x[n]z^{-(n+k_0)} \\
&= z^{-k_0}\sum_{n=0}^{\infty} x[n]z^{-n} + \sum_{n=-k_0}^{-1} x[n]z^{-(n+k_0)} \\
&= z^{-k_0}X(z) + x[-1]z^{-k_0+1} \\
&\qquad + x[-2]z^{-k_0+2} + \cdots + x[-k_0]
\end{aligned}$$

with the same ROC as $\mathcal{Z}[x[k]]$ and, possibly, additional restrictions in $z = \pm\infty$.

With zero initial conditions, one has $\mathcal{Z}[x[k-k_0]] = z^{-k_0}X(z)$.

Time shifting

Time shifting

For a signal $x[k]$ with z-transform $\mathcal{Z}[x[k]] = X(z)$ and $k_0 > 0$, one has a:

$$\begin{aligned}
\mathcal{Z}[x[k+k_0]] &= \sum_{k=0}^{\infty} x[k+k_0] z^{-k} = \sum_{n=k_0}^{\infty} x[n] z^{-n} z^{k_0} \\
&= z^{k_0} \sum_{n=0}^{\infty} x[n] z^{-n} - \sum_{n=0}^{k_0-1} x[n] z^{-n} z^{k_0} \\
&= z^{k_0} X(z) - x[0] z^{k_0} \\
&\qquad -x[1] z^{k_0-1} - \cdots - x[k_0-1] z
\end{aligned}$$

with the same ROC as $\mathcal{Z}[x[k]]$ and, possibly, additional restrictions $z = 0$. With zero initial conditions, one has $\mathcal{Z}[x[k+k_0]] = z^{k_0} X(z)$.

Unit delay operator

From the time shifting property, we know that, with zero initial conditions, one has

$$\mathcal{Z}[x[k-k_0]] = z^{-k_0}X(z).$$

Suppose the system with transfer function $H(z) = z^{-1}$ and an input signal $x[k]$ with associated z-transform $X(z)$. The response is easily obtained in the z-domain, i.e.

$$Y(z) = z^{-1}X(z).$$

with associated time response $y[k] = x[k-1]$. The operator z^{-1} is therefore the **unit delay operator**.

Finite difference

Finite difference

For a signal $x[k]$ with z-transform $X(z) = \mathcal{Z}[x[k]]$, the z-transform of the finite difference

$$\mathcal{Z}[x[k] - x[k-1]] = (1 - z^{-1})X(z) - x[-1]$$

where $x[-1]$ is the initial condition.

Derivative of $X(z)$

Derivative of $X(z)$

For a signal $x[k]$ with z-transform $X(z) = \mathcal{Z}[x[k]]$, the derivative of $X(z)$ with respect to z is

$$\frac{dX(z)}{dz} = \sum_{k=-\infty}^{\infty} x[k]\frac{dz^{-k}}{dz} = -z^{-1}\sum_{k=-\infty}^{\infty} kx[k]z^{-k}.$$

One obtains

$$\mathcal{Z}[kx[k]] = -z\,\frac{dX(z)}{dz}.$$

From the property of the derivative of $X(z)$, one can calculate the following transform

$$\mathcal{Z}[k\alpha^k u[k]] = -z\frac{d}{dz}\left(\frac{1}{1-\alpha z^{-1}}\right) = -z\frac{d}{dz}\left(\frac{z}{z-\alpha}\right) = \frac{\alpha z}{(z-\alpha)^2}.$$

Z-scaling / damping

Z-scaling / damping

For a signal $x[k]$ with z-transform $X(z) = \mathcal{Z}[x[k]]$, one has

$$\begin{aligned}
\mathcal{Z}[\alpha^k x[k]] &= \sum_{k=-\infty}^{\infty} \alpha^k x[k] z^{-k}, \\
&= \sum_{k=-\infty}^{\infty} x[k] \left(\frac{z}{\alpha}\right)^{-k} = \sum_{k=-\infty}^{\infty} x[k] \left(\alpha z^{-1}\right)^{k}, \\
&= X\left(\frac{z}{\alpha}\right) = X(\alpha z^{-1}).
\end{aligned}$$

The radius defining the ROC is multiplied by $|\alpha|$.

Convolution

Convolution

The z-transform of the convolution of the signal $x[k]$ with $\mathcal{Z}[x[k]] = X(z)$ and the signal $w[k]$ with $\mathcal{Z}[w[k]] = W(z)$ is

$$\mathcal{Z}[x * w][k] = X(z)W(z).$$

Corollary

The response of a LTI system with impulse response $h[k]$ to a causal signal $x[k]$ is given by

$$y[k] = \mathcal{Z}^{-1}[Y(z)] = \mathcal{Z}^{-1}[H(z)X(z)] = \mathcal{Z}^{-1}[\mathcal{Z}[h * x][k]].$$

where $\mathcal{Z}[h[k]] = H(z)$ and $\mathcal{Z}[x[k]] = X(z)$.

In the z-domain, one has $Y(z) = H(z)X(z)$.

Transfer function

Transfer function

The transfer function $H(z) = \mathcal{Z}[h[k]]$, the z-transform of the impulse response $h[k]$ of a LTI system, can be expressed as the ratio

$$H(z) = \frac{\mathcal{Z}[y[k]]}{\mathcal{Z}[x[k]]} = \frac{\mathcal{Z}[\text{output signal}]}{\mathcal{Z}[\text{input signal}]}$$

The transfer function characterizes the system by its poles and zeros.

It is an important tool in the analysis and synthesis of systems.

Initial and final value theorem

In some control applications and to check a partial fraction expansion, it is useful to find the initial and final value of a discrete-time signal $x[k]$ from its z-transform $X(z)$.

Initial and final value theorem

From $X(z)$, it is possible to determine the initial and final value of the signal $x[k]$, in the vicinity of $k = 0$ and $k \to \infty$.

- **Initial value**: providing the limit exists,

$$x[0] = \lim_{z \to \infty} X(z).$$

- **Final value**: if all the poles of $(z-1)\,X(z)$ are inside the unit circle then

$$\lim_{k \to \infty} x[k] = \lim_{z \to 1} (z-1)\,X(z).$$

Stability in the z-domain

A LTI system is stable in a BIBO sense if

$$\sum_k |h[k]| < \infty$$

i.e. if its impulse response $h[k]$ is absolutely summable. Let $H(z)$ be the z-transform of the impulse response $h[k]$. The associated

$$\text{ROC} = \left\{ z = r\, e^{j\Omega} \text{ such that } \sum_{k=-\infty}^{\infty} |h[k]\, r^{-k}| < \infty \right\}$$

Therefore, the z-transform $H(z)$ of $h[k]$ must converge on the unit circle.

Stability in the z-domain

Stability in the z-domain

The discrete-time LTI system is stable if it can be represented by $H(z)$ with ROC including the unit circle.

The condition is applicable for causal, anti-causal and non-causal systems.

For a causal system, the condition is equivalent to requesting that the poles $H(z)$ are located strictly inside the unit circle.

Inverse z-transform: long division method

Long division

A rational transfer function $H(z) = \frac{B(z)}{A(z)}$ with ROC $|z| > R$ (i.e. $h[k]$ is causal) can be written

$$H(z) = h[0] + h[1]\, z^{-1} + h[2]\, z^{-2} + \cdots$$

This description can be obtained by the long division method. The inverse z-transform is given by

$$h[k] = h[0]\, \delta[k] + h[1]\, \delta[k-1] + h[2]\, \delta[k-2] + \cdots$$

Inverse z-transform: partial fraction expansion

Fractions simples

The basics of partial fraction expansion remain the same for the z-transform as for the Laplace transform:

- The transfer function must be proper, i.e. the degree of its numerator must be smaller than the degree of its denominator. If needed, perform a Euclidean division until the condition is satisfied.
- Expand in partial fractions after computing the poles of the transfer function. The expansion can be performed in z or z^{-1} **but the expansion in powers of z requires more care**.
- The resulting expansion is composed of terms of which the inverse z-transform can easily be obtained from z-transform tables.

Partial fraction expansion in z^{-1} of $Y(z^{-1})$: example

Find the inverse z-transform of

$$Y(z) = \frac{1+z^{-1}}{(1+0.5z^{-1})(1-0.5z^{-1})} = \frac{z(z+1)}{(z+0.5)(z-0.5)} \quad |z| > 0.5$$

Expanding in partial fractions using negative powers z^{-1}, one obtains

$$Y(z) = \frac{A}{(1+0.5z^{-1})} + \frac{B}{(1-0.5z^{-1})}$$

$$A = \left.\frac{1+z^{-1}}{1-0.5z^{-1}}\right|_{z^{-1}=-2} = -\frac{1}{2}, \quad B = \left.\frac{1+z^{-1}}{1+0.5z^{-1}}\right|_{z^{-1}=2} = \frac{3}{2}$$

$$y[k] = [1.5\,(0.5)^k - 0.5\,(-0.5)^k]\,u[k].$$

Partial fraction expansion in z of $Y(z)/z$: example

Find the inverse z-transform of

$$Y(z) = \frac{1 + z^{-1}}{(1 + 0.5z^{-1})(1 - 0.5z^{-1})} = \frac{z(z+1)}{(z+0.5)(z-0.5)} \quad |z| > 0.5$$

Expanding in partial fractions using positive powers z of $Y(z)/z$, one obtains

$$\frac{Y(z)}{z} = \frac{A}{(z+0.5)} + \frac{B}{(z-0.5)} \Rightarrow Y(z) = \frac{A\,z}{(z+0.5)} + \frac{B\,z}{(z-0.5)}$$

$$A = \left.\frac{z+1}{z-0.5}\right|_{z=-0.5} = -\frac{1}{2}, \quad B = \left.\frac{z+1}{z+0.5}\right|_{z=0.5} = \frac{3}{2}$$

$$y[k] = [1.5\,(0.5)^k - 0.5\,(-0.5)^k]\,u[k].$$

Partial fraction expansion in z of $Y(z)$: example

Find the inverse z-transform of

$$Y(z) = \frac{1+z^{-1}}{(1+0.5z^{-1})(1-0.5z^{-1})} = \frac{z(z+1)}{(z+0.5)(z-0.5)} \quad |z| > 0.5$$

Expanding in partial fractions using positive powers z, after Euclidean division, one obtains

$$\begin{aligned} Y(z) &= 1 + \frac{z+0.25}{(z+0.5)(z-0.5)}, \\ &= 1 + \frac{0.75}{z-0.5} + \frac{0.25}{z+0.5}, \\ &= 1 + z^{-1}\,\frac{0.75\,z}{z-0.5} + z^{-1}\,\frac{0.25\,z}{z+0.5}. \end{aligned}$$

$$y[k] = \delta[k] + \left[0.75\,(0.5)^{(k-1)} + 0.25\,(-0.5)^{(k-1)}\right] u[k-1].$$

Partial fraction expansion in z: example

$$
\begin{aligned}
y[k] &= \delta[k] + \left[0.75\,(0.5)^{(k-1)} + 0.25\,(-0.5)^{(k-1)}\right] u[k-1], \\
&= \delta[k] + \left[\frac{0.75}{0.5}\,(0.5)^{k} + \frac{0.25}{-0.5}\,(-0.5)^{k}\right] u[k-1], \\
&= \delta[k] + \left[1.5\,(0.5)^{k} - 0.5\,(-0.5)^{k}\right] u[k-1], \\
&= \delta[k] + \left[1.5\,(0.5)^{k} - 0.5\,(-0.5)^{k}\right] u[k] - \delta[k], \\
&= [1.5\,(0.5)^{k} - 0.5\,(-0.5)^{k}]\, u[k].
\end{aligned}
$$

LTI system analysis

The response $y[k]$ of a system described by a difference equation of order N with real constant coefficients

$$y[k] = -\sum_{m=1}^{N} a_m \, y[k-m] + \sum_{m=0}^{M} b_m \, x[k-m]$$

with $N \geq M$ and initial conditions $y[-k], k = 1, \cdots, N$ is obtained by inverting the z-transform

$$Y(z) = \frac{B(z)}{A(z)} X(z) + \frac{1}{A(z)} I(z)$$

where $Y(z) = \mathcal{Z}[y[k]]$, $X(z) = \mathcal{Z}[x[k]]$, $I(z)$ depends on the initial conditions and

$$A(z) = 1 + \sum_{m=1}^{N} a_m \, z^{-m}, \quad B(z) = \sum_{m=0}^{M} b_m \, z^{-m}.$$

Example 1

Consider, the discrete-time IIR system represented by the first order recursive equation

$$y[k] = a\,y[k-1] + x[k], \quad k > 0.$$

Using the z-transform, one obtains

$$\begin{aligned} \mathcal{Z}[y[k]] &= \mathcal{Z}[a\,y[k-1]] + \mathcal{Z}[x[k]] \\ Y(z) &= a\,(z^{-1}Y(z) + y[-1]) + X(z) \end{aligned}$$

and

$$Y(z) = \frac{X(z)}{1 - a\,z^{-1}} + \frac{\overbrace{a\,y[-1]}^{I(z)}}{1 - a\,z^{-1}}.$$

Example 1

Also

$$
\begin{aligned}
Y(z) &= \frac{X(z)}{1 - a\,z^{-1}} + \frac{\overbrace{a\,y[-1]}^{I(z)}}{1 - a\,z^{-1}} \\
&= \sum_{n=0}^{\infty} X(z)\,a^n z^{-n} + a\,y[-1] \sum_{n=0}^{\infty} a^n z^{-n} \\
&= X(z) + a\,X(z)\,z^{-1} + \cdots + a\,y[-1]\,(1 + a\,z^{-1} + \cdots) \\
y[k] &= x[k] + ax[k-1] + \cdots + a\,y[-1]\,(\delta[k] + a\,\delta[k-1] + \cdots) \\
&= \sum_{n=0}^{\infty} a^n\,x[k-n] + a\,y[-1] \sum_{n=0}^{\infty} a^n\,\delta[k-n]
\end{aligned}
$$

Example 2

Consider the first order recursive equation

$$y[n+1] - 2y[n] = (n+1)\,u[n] \quad \text{with} \quad y[0] = 2.$$

Going to the z-domain, one obtains

$$\begin{aligned}
(zY(z) - zy[0]) - 2Y(z) &= \frac{z}{(z-1)^2} + \frac{z}{z-1} \\
(z-2)Y(z) &= 2z + \frac{z^2}{(z-1)^2} \\
Y(z) &= \underbrace{\frac{2z}{z-2}}_{Y_1(z)} + \underbrace{\frac{z^2}{(z-2)(z-1)^2}}_{Y_2(z)}
\end{aligned}$$

$$\begin{aligned}
\frac{Y_2(z)}{z} &= \frac{z}{(z-2)(z-1)^2} = \frac{-2}{(z-1)} + \frac{-1}{(z-1)^2} + \frac{2}{(z-2)} \\
Y(z) &= \frac{2z}{z-2} + \frac{-2z}{(z-1)} + \frac{-z}{(z-1)^2} + \frac{2z}{(z-2)}
\end{aligned}$$

Example 2

We obtain

$$Y(z) = \frac{4z}{z-2} - \frac{2z}{(z-1)} - \frac{z}{(z-1)^2}.$$

The inverse z-transform yields

$$y[n] = [4\,2^n - (n+2)]\,u[n].$$

Notice that $y[0] = 2$!

Example 2: alternative method

Consider the first order recursive equation

$$y[n+1] - 2y[n] = (n+1)\,u[n] \quad \text{with} \quad y[0] = 2.$$

It is possible to rewrite this equation as

$$y[n] - 2y[n-1] = n\,u[n-1]$$

where one needs the initial condition $y[-1]$! The first equation applied in $n = -1$ yields

$$y[0] - 2y[-1] = 0 \Longrightarrow y[-1] = \frac{y[0]}{2} = 1.$$

Let us now solve

$$y[n] - 2y[n-1] = n\,u[n-1] \quad \text{with} \quad y[-1] = 1.$$

Example 2: alternative method

Let us now solve

$$y[n] - 2y[n-1] = n\,u[n-1] \ \text{ with } \ y[-1] = 1.$$

Going to the z-domain, one obtains

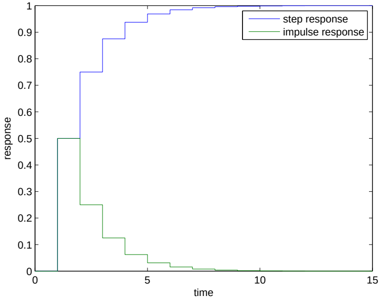
$$\begin{aligned}
Y(z) - 2(z^{-1}Y(z) + y[-1]) &= \mathcal{Z}[(n-1)\,u[n-1] + u[n-1]] \\
(1 - 2z^{-1})Y(z) &= 2 + \frac{z^{-2}}{(1-z^{-1})^2} + \frac{z^{-1}}{(1-z^{-1})} \\
(1 - 2z^{-1})Y(z) &= 2 + \frac{z^{-1}}{(1-z^{-1})^2} \\
Y(z) &= \frac{2z}{z-2} + \frac{z^2}{(z-2)(z-1)^2}
\end{aligned}$$

The inverse z-transform yields

$$y[n] = [4\,2^n - (n+2)]\,u[n].$$

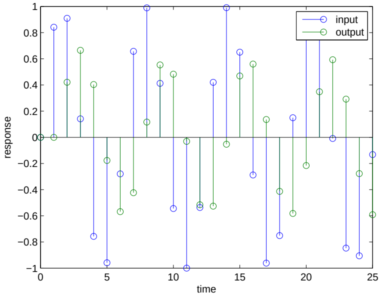
Step and impulse responses

```
% pkg load control % uncomment if running octave with control toolbox

numd = [0 0.5];
dend = [1 -0.5]; % Transfer function H(z) = 0.5/(z — 0.5)
Ts = 1;
sys = tf(numd,dend,Ts);
t = 0:Ts:15;
y1 = step(sys,t);
y2 = impulse(sys,t);
stairs(t,[y1 y2])
legend('step response','impulse response')
xlabel('time'); ylabel('response')
```



Simulated responses

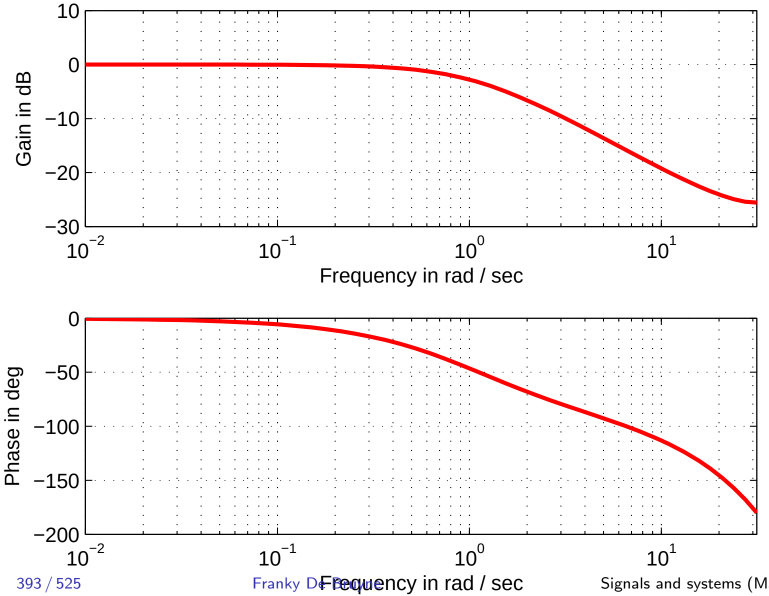
```
% pkg load control % uncomment if running octave with control toolbox

numd = [0 0.5];
dend = [1 -0.5]; % Transfer function H(z) = 0.5/(z — 0.5)
Ts = 1;
sys = tf(numd,dend,Ts);
t = 0:Ts:25;
u = sin(t);
y = lsim(sys,u,t);
stem(t,[u' y])
legend('input','output')
xlabel('time'); ylabel('response')
```



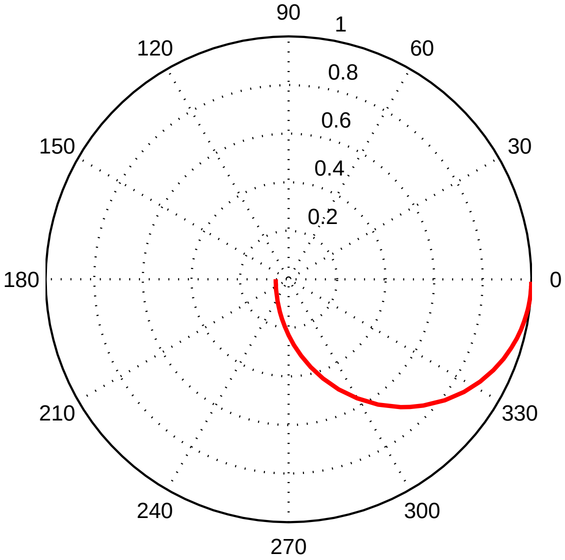
Bode and Nyquist plots

```
% pkg load control % uncomment if running octave with control toolbox

numd = [0 0.1]; dend = [1 -0.9];  % Transfer function H(z) = 0.1/(z-0.9)
Ts = 0.1;
sysd = tf(numd,dend,Ts);
[amplitude,phase,w] = bode(sysd);
amplitude = reshape(amplitude,[length(amplitude) 1]);
phase = reshape(phase,[length(phase) 1]);

% Bode diagram
figure(1)
subplot(211)
semilogx(w,20*log10(amplitude),'r','Linewidth',2);
grid on; axis([0.01 pi/Ts -30 10]);
ylabel('Gain in dB'); xlabel('Frequency in rad / sec');
subplot(212)
semilogx(w,phase ,'r','Linewidth',2);
grid on; axis([0.01 pi/Ts -200 0]);
ylabel('Phase in deg'); xlabel('Frequency in rad / sec');

% Nyquist diagram
figure(2)
h1 = polar(phase*(pi/180), amplitude );
set(h1 ,'color','r','linewidth',2);
grid
```

Bode plots

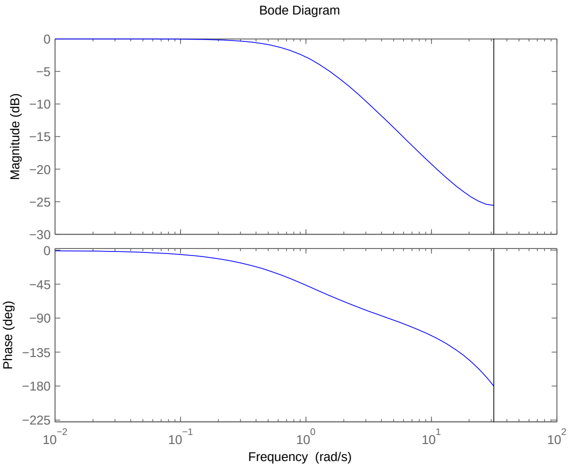

10
0
−10
−20
−30
Gain in dB
10^{-2}
10^{-1}
10^{0}
10^{1}
Frequency in rad / sec
0
−50
−100
−150
−200
Phase in deg
10^{-2}
10^{-1}
10^{0}
10^{1}
Frequency in rad / sec

Nyquist plots



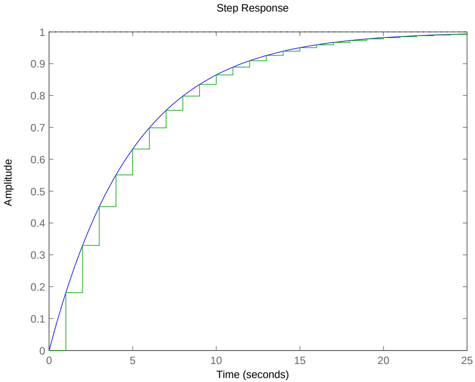
Bode plots

```
numd = [0 0.1]; dend = [1 -0.9];  % Transfer function H(z) = 0.1/(z—0.9)
dbode(numd,dend,Ts)
```



Discretisation

```
clear all
num = [0 1]; % H(s) = 1/(5s + 1)
den = [5 1];
sys = tf(num,den);
Ts = 1;
sysd = c2d(sys,Ts); % Discretisation of continuous system
step(sys,sysd,25)
```



Z-transforms

```
% pkg load symbolic % uncomment for use with octave with symbolic toolbox
syms a n z
f1 = a^n;
f2 = n*a^n;
ztrans(f1, z)
ztrans(f2, z)

syms n z
F = z/(z-3)^2 + z/(z-2);
f = iztrans(F, n)
simplify(f)
```

```
ans =

-z/(a - z)

ans =

(a*z)/(a - z)^2

f =

2^n + 3^n/3 + (3^n*(n - 1))/3

ans =

(3^n*n)/3 + 2^n
```

One-sided z-transforms		
$x[k]$	$\mathcal{Z}[x[k]u[k]]$	ROC
$\delta[k]$	1	z-domain
1	$\dfrac{1}{1-z^{-1}}$	$\lvert z\rvert > 1$
k	$\dfrac{z^{-1}}{(1-z^{-1})^2}$	$\lvert z\rvert > 1$
α^k	$\dfrac{1}{1-\alpha z^{-1}}$	$\lvert z\rvert > \lvert\alpha\rvert$
$k\alpha^k$	$\dfrac{\alpha z^{-1}}{(1-\alpha z^{-1})^2}$	$\lvert z\rvert > \lvert\alpha\rvert$

One-sided z-transforms		
$x[k]$	$\mathcal{Z}[x[k]u[k]]$	ROC
$\cos(\Omega_0 k)$	$\frac{1-\cos(\Omega_0)z^{-1}}{1-2\cos(\Omega_0)z^{-1}+z^{-2}}$	$\lvert z\rvert > 1$
$\sin(\Omega_0 k)$	$\frac{\sin(\Omega_0)z^{-1}}{1-2\cos(\Omega_0)z^{-1}+z^{-2}}$	$\lvert z\rvert > 1$
$r^k\cos(\Omega_0 k)$	$\frac{1-r\cos(\Omega_0)z^{-1}}{1-2r\cos(\Omega_0)z^{-1}+r^2z^{-2}}$	$\lvert z\rvert > \lvert r\rvert$
$r^k\sin(\Omega_0 k)$	$\frac{\sin(\Omega_0)rz^{-1}}{1-2r\cos(\Omega_0)z^{-1}+r^2z^{-2}}$	$\lvert z\rvert > \lvert r\rvert$

Basic properties of one-sided z-transforms		
Properties	$x[k]$	$X(z)$
Causal signals and constants	$\alpha x[k]$, $\beta y[k]$	$\alpha X(z)$, $\beta Y(z)$
Linearity	$\alpha x[k] + \beta y[k]$	$\alpha X(z) + \beta Y(z)$
Convolution	$[x * y][k]$ $=\sum_n x[k]y[k-n]$	$X(z)Y(z)$
Time shifting (zero I.C.)	$x[k-k_0]$	$z^{-k_0}X(z)$
Time shifting	$x[k-k_0]$	$z^{-k_0}X(z) + x[-1]z^{-k_0+1}$ $+x[-2]z^{-k_0+2} + \cdots + x[-k_0]$

Basic properties of one-sided z-transforms		
Properties	$x[k]$	$X(z)$
Time shifting (zero I.C.)	$x[k+k_0]$	$z^{k_0}X(z)$
Time shifting	$x[k+k_0]$	$z^{k_0}X(z) - x[0]z^{k_0} - x[1]z^{k_0-1} - \cdots - x[k_0-1]z$

Basic properties of one-sided z-transforms		
Properties	$x[k]$	$X(z)$
Multiplication by k	$k\,x[k]$	$-z\,\dfrac{dX(z)}{dz}$
Finite difference	$x[k]-x[k-1]$	$(1-z^{-1})\,X(z)-x[-1]$
Accumulation	$\sum_{n=0}^{k} x[k]$	$\frac{X(z)}{1-z^{-1}}$
Initial value	$x[0]$	$\lim_{z\to\infty} X(z)$
Final value	$\lim\limits_{k\to\infty} x[k]$	$\lim_{z\to 1}(z-1)\,X(z)$

9. Discrete-time Fourier frequency analysis

Discrete-time Fourier transform

Discrete-time Fourier transform

The Discrete-Time Fourier Transform (DTFT) of a discrete-time signal $x[k]$,

$$X(\Omega) = \sum_{k=-\infty}^{\infty} x[k]e^{-j\Omega k}, \quad -\pi \leq \Omega < \pi$$

converts $x[k]$ into $X(\Omega)$ of the discrete frequency[54] Ω (rad).

The inverse transform gives back $x[k]$ from $X(\Omega)$ according to

$$x[k] = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\Omega)e^{j\Omega k} d\Omega.$$

[54] The discrete frequency Ω is a continuous variable over $[-\pi, \pi)$.

Link with the Fourier transform

When sampling an analog signal $x(t)$, the sampled signal $x_s(t)$ can be written as

$$x_s(t) = \sum_k x(kT_s)\,\delta(t - kT_s).$$

Its Fourier transform is then

$$\mathcal{F}[x_s(t)] = \sum_k x(kT_s)\,\mathcal{F}[\delta(t - kT_s)] = \sum_k x(kT_s) e^{-j\omega k T_s} = X_s(e^{j\omega T_s}).$$

Letting $\Omega = \omega T_s$, the discrete frequency, the above equation can be written as

$$\begin{aligned} X_s(e^{j\Omega}) &= X_s(\Omega) = \sum_k x(kT_s)e^{-j\Omega k} = \sum_k x[k]e^{-j\Omega k} \\ &= \frac{1}{T_s}\sum_k X(\omega - k\omega_s) = \frac{1}{T_s}\sum_k X\left(\frac{\Omega - 2k\pi}{T_s}\right). \end{aligned}$$

Thus, sampling converts a continuous-time signal into a discrete-time signal with a periodic spectrum varying continuously in frequency.

Link with the z-transform

If in the above we ignore T_s and consider $x(kT_s)$ a function of k, $x[k]$, we can see that

$$X_s(e^{j\Omega}) = X(z)|_{z=e^{j\Omega}}.$$

That is, it is the z-transform computed on the unit circle. For the above to happen, $X(z)$ must have a region of convergence (ROC) that includes the unit circle.

However, any discrete-time signal $x[k]$, of finite support in time, has a z-transform $X(z)$ with a region of convergence the whole z-plane, excluding either the origin or infinity, and as such its DTFT $X(\Omega)$ is computed from $X(z)$ by letting $z = e^{j\Omega}$.

Discrete frequency Ω

- The DTFT spectrum is continuous periodic of period 2π, only the frequencies $\Omega \in [-\pi, \pi)$ need to be considered.
- When plotting or displaying the spectrum of a **real-valued** discrete-time signal it is important to know that it is only necessary to show the magnitude and the phase spectra for frequencies $\Omega \in [0 \quad \pi]$ since
  - the **magnitude spectrum** is an **even** function of Ω and
  - the **phase spectrum** is an **odd** function of Ω.
- Interpretation of Ω:

$$\Omega = \omega T_s = \omega T_s \frac{2\pi}{2\pi} = \omega \frac{2\pi}{\frac{2\pi}{T_s}} = 2\pi \frac{\omega}{\omega_s} = \pi \frac{\omega}{\frac{\omega_s}{2}} = \pi \frac{\omega}{\omega_N}$$

- The spectrum of a real-valued discrete signal is sometimes represented over $F \in [0 \quad 1]$ where F is the normalised frequency, i.e.

$$F = \frac{\Omega}{\pi} = \frac{\omega}{\omega_N} = \frac{f}{f_N}.$$

DTFT: example

Consider the DTFT of $\Pi[k] = u[k] - u[k-N]$. Since $\Pi[k]$ has finite support, its z-transform converges everywhere except at the origin, i.e.

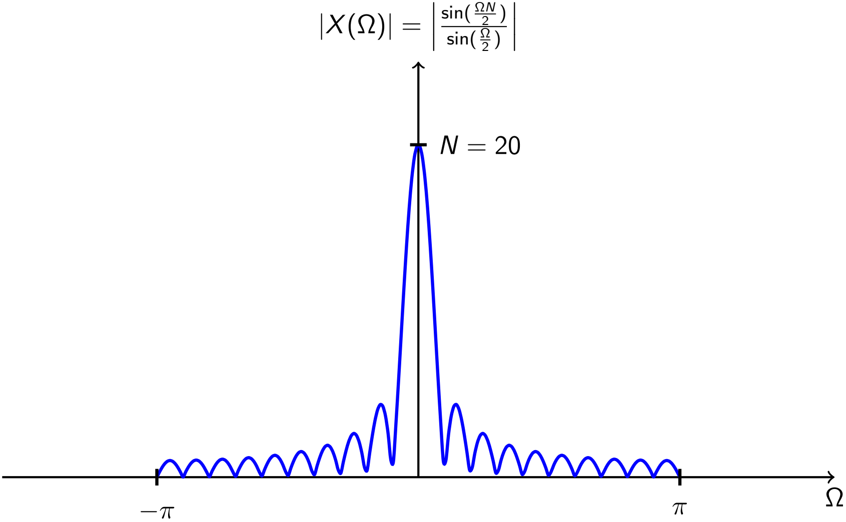
$$\Pi(z) = \sum_{k=0}^{N-1} z^{-k} = \frac{1-z^{-N}}{1-z^{-1}}.$$

The DTFT is given by

$$\begin{aligned}\Pi(e^{j\Omega}) &= \frac{1-e^{-j\Omega N}}{1-e^{-j\Omega}} = \frac{e^{-j\frac{\Omega N}{2}}}{e^{-j\frac{\Omega}{2}}} \frac{e^{j\frac{\Omega N}{2}} - e^{-j\frac{\Omega N}{2}}}{e^{j\frac{\Omega}{2}} - e^{-j\frac{\Omega}{2}}} \\ &= e^{-j\frac{\Omega(N-1)}{2}} \frac{\sin(\frac{\Omega N}{2})}{\sin(\frac{\Omega}{2})}\end{aligned}$$

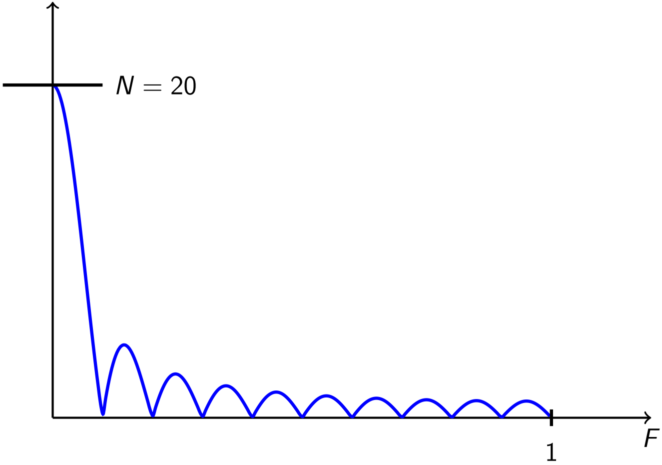
The function $\sin(\frac{\Omega N}{2})/\sin(\frac{\Omega}{2})$ plays the same role as the sinc function for the continuous-time Fourier transform.

DTFT: example



DTFT: example

$$|X(F)| = \left| \frac{\sin(\frac{\pi F N}{2})}{\sin(\frac{\pi F}{2})} \right|$$



Discrete Fourier basis

Discrete Fourier basis

The discrete Fourier basis is composed of functions

$$\phi[k,n] = e^{j\frac{2\pi kn}{N}}, k, n = 0, 1, \cdots, N.$$

The following properties hold:

- The basis is **periodic** with respect to k and n with period N.

$$\phi[k+mN,n] = e^{j\frac{2\pi(k+mN)n}{N}} = e^{j\frac{2\pi kn}{N}} e^{j2\pi mn} = \phi[k,n]$$

- The basis is **orthonormal** over a period T using the inner product

$$< \phi[k,n], \phi[l,n] > = \frac{1}{N} \sum_{n=n_0}^{n_0+N-1} \phi[k,n]\phi[l,n]^{\star} = \begin{cases} 1, & k = l \\ 0, & k \neq l \end{cases}$$

Discrete Fourier series

Discrete Fourier Series (DFS)

The complex DFS of a periodic signal of period N uses the discrete Fourier basis, i.e.

$$x[k] = \sum_{n=n_0}^{n_0+N-1} X[n]\,\phi[k,n] = \sum_{n=n_0}^{n_0+N-1} X[n]\,e^{j\frac{2\pi kn}{N}}$$

where the discrete Fourier coefficients are obtained as

$$X[n] = \; < x[k], \phi[k,n] > \; = \frac{1}{N}\sum_{k=k_0}^{k_0+N-1} x[k]e^{-j\frac{2\pi kn}{N}}.$$

The frequency $\Omega_0 = \frac{2\pi}{N}$ is the fundamental frequency and k_0, n_0 arbitrary coefficients. The discrete Fourier coefficients $X[n]$ are also periodic of period N.

Discrete Fourier transform

Discrete Fourier Transform (DFT) of a periodic signal

The DFT of a **periodic** signal of period N is

$$X[n] = \sum_{k=0}^{N-1} x[k] e^{-j\frac{2\pi kn}{N}}, \;\; 0 \leq n \leq N-1.$$

The inverse DFT is

$$x[k] = \frac{1}{N} \sum_{n=0}^{N-1} X[n]\, e^{j\frac{2\pi kn}{N}}, \;\; 0 \leq k \leq N-1.$$

Both $X[n]$ and $x[k]$ are discrete and periodic of the period N.

The link with the DFS is immediate:

- $X_{\mathrm{DFT}}[n] = N\, X_{\mathrm{DFS}}[n]$,
- the arbitrary constants n_0 and k_0 have been fixed to 0.

Generalisation to **aperiodic** signals ?

DFT of an aperiodic signal

- The DFT of a signal $x[k]$, $k = 0, \cdots, N-1$ is obtained by sampling its DTFT $X(\Omega)$ in frequency. Suppose

$$\Omega_n = \frac{2\pi n}{L}, n = 0, \cdots, L-1.$$

- Sampling a continuous signal over time yields a periodic spectrum over frequency.
- By time-frequency duality, sampling a spectrum over frequency yields a periodic time sequence, i.e.

$$\tilde{x}[k] = \sum_{m=-\infty}^{\infty} x[k+mL].$$

- If $L < N$, the first period of $\tilde{x}[k]$ does not coincide with $x[k]$ because of superposition of shifted versions of $x[k]$.

  This corresponds to **time aliasing**, the dual of frequency aliasing, which occurs in time sampling.

DFT of an aperiodic signal

- We therefore only consider the case $L \geq N$. Then the periodic expansion $\tilde{x}[k]$ clearly displays a first period equal to the given signal $x[k]$ with some zeros attached at the end when $L > N$.
- Therefore

$$\tilde{x}[k] = \sum_{m=-\infty}^{\infty} x[k + mL] \iff X[n] = \underbrace{\sum_{k=0}^{N-1} x[k] e^{-j\frac{2\pi kn}{L}}}_{\text{DFT of } x[k]}, \; 0 \leq n \leq L - 1.$$

- The inverse DFT is the sequence $\tilde{x}[k]$ limited to its first period, i.e.

$$x[k] = \frac{1}{L} \sum_{n=0}^{L-1} X[n] \, e^{j\frac{2\pi kn}{L}}, \; 0 \leq k \leq L - 1.$$

DFT of an aperiodic signal

DFT of an aperiodic signal

The DFT of an **aperiodic** signal $x[k]$ of length N is obtained by choosing $L \geq N$ with

$$\tilde{x}[k] = x[k] \text{ for } 0 \leq k < N, \;\; \tilde{x}[k] = 0 \text{ for } N \leq k < L,$$

$$X[n] = \sum_{k=0}^{L-1} \tilde{x}[k] e^{-j\frac{2\pi kn}{L}}, \;\; 0 \leq n \leq L-1.$$

The inverse DFT is

$$\tilde{x}[k] = \frac{1}{L} \sum_{n=0}^{L-1} X[n]\, e^{j\frac{2\pi kn}{L}}, \;\; 0 \leq k \leq L-1.$$

Note that both $X[n]$ and $\tilde{x}[k]$ are periodic of period L.

Also $X[n] = X(\Omega)|_{\Omega=\Omega_n=\frac{2\pi n}{L}}$.

Spectral resolution: why $L > N$?

Hypothesis: DFT of an **aperiodic** signal $x[k]$ of length N

- If N is small and $L = N$, the DFT produces the DTFT spectrum sampled over N points, i.e. the spectral resolution is **poor**.

  To **increase** the spectral resolution, $L > N$ is chosen.

  In practice, a number of zeros are added at the end of the sequence. This is called **zero padding**.

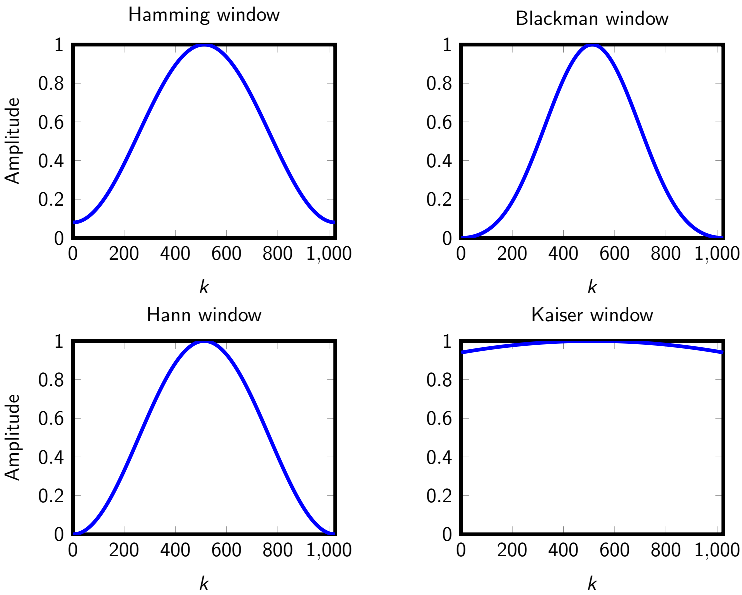
- When the FFT is used (efficient implementation of the DFT), N needs to be a power 2. If this not case a number of zeros are added at the end of the sequence so that the length of the resulting sequence becomes a power of 2.

Hypothesis: DFT of a **periodic** signal $x[k]$ of period N

- When N is small, several periods of the signal need to be considered. Zero padding should **not** be used in this case.

Spectral leakage

- Practice: a signal is measured over a limited time span.
- The DFT "presupposes" that the measured signal is repeating itself.
- Most measured signals are not equal at the beginning and at the end of the sequence. Since the DFT interprets the data as a repeating sequence, it sees discontinuities.
- The discontinuities between the ends and the beginnings manifest themselves as frequency components that do not really exist. The components are called **spurious frequencies** or **spectral leakage**.
- **Solutions:**
  - **Windowing** functions effectively taper the ends of the segments to 0 or small values so that they connect from beginning to end. A compromise between reducing spectral leakage and smearing the spectrum needs to be thought.
  - Select N (large, appropriate choice) to attenuate the effects of spectral leakage.


Hamming window
Blackman window
Hann window
Kaiser window
Amplitude
Amplitude
1
0.8
0.6
0.4
0.2
0
0
200
400
600
800
1,000
k

DFT: Practical aspects ($L = N$)

- Real-valued input sequence $x[k]$ with $k = 0, \cdots, N-1$.
- DFT produces the complex sequence $X[n]$ with $n = 0, \cdots, N-1$.
- Index n samples $\Omega_n = \frac{2\pi n}{N}$ over a period $0, \cdots, 2\pi\frac{N-1}{N}$ of the DFT.
- The amplitude and phase spectra are, respectively, even and odd. The $X[n]$ with indices $n = 0, \cdots, \frac{N}{2}$ contain all the relevant information.
- Is this consistent ? yes !
  - Input: N real samples
  - Output: $\frac{N}{2} + 1$ complex samples $\Longrightarrow N + 2$ reals !
  - 2 samples do not contain any information, i.e.

$$\mathcal{I}_m[\,X[0]\,] = \mathcal{I}_m[\,X[\tfrac{N}{2}]\,] = 0.$$

DFT: Practical aspects ($L = N$)

- Index n samples $\Omega_n = \frac{2\pi n}{N}$ over a period $0, \cdots, 2\pi \frac{N-1}{N}$ of the DFT.
- The normalised frequency is

$$F = \frac{\Omega}{\pi} = \frac{\omega}{\omega_N} = \frac{f}{f_N}.$$

- The indices $n = 0, \cdots, \frac{N}{2}$ of the DFT correspond to a frequency scale in the interval $[0 \ \ f_N]$, i.e.

$$\boxed{[0 : 1 : \tfrac{N}{2}] \quad \Longleftrightarrow \quad [0 : \tfrac{f_s}{N} : f_N = \tfrac{f_s}{2}].}$$

Fourier series and transforms: summary

- **Fourier Series (FS)**: applicable to continuous and periodic signals. The resulting spectrum is discrete.
- **Fourier Transform (FT)**: applicable to continuous signals. The resulting spectrum is continuous[55] and aperiodic.
- **Discrete-Time Fourier Transform (DTFT)**: applicable to discrete signals. The resulting spectrum is continuous and periodic.
- **Discrete Fourier Series (DFS)** : applicable to discrete and periodic signals. The resulting spectrum is discrete.
- **Discrete Fourier Transform (DFT)**: applicable to discrete signals. The resulting spectrum is discrete and periodic.

[55] For periodic signals, the resulting FT spectrum is discrete.

Introduction

- Cooley and Tukey are the 2 engineers credited for the invention of the modern Fast Fourier Transform (FFT), in an article published in 1965.
- Although the development of fast algorithms for DFT can be traced back to Gauss's work in 1805, the authors realised that Gauss's methods could be implemented efficiently with computers.
- The FFT is **not** a a new transform but rather the efficient implementation of the DFT.
- The FFT can be used to compute the inverse DFT with minor modifications, i.e. a change of sign.

Introduction

- As opposed to the DTFT, the DFT can easily be implemented on a processor.
- An implementation of the DFT based on its definition is not efficient because it does not use all symmetry properties of the DFT.
- The FFT is an efficient implementation of the DFT that uses these symmetries:
  - Evaluating the DFT's sums involves N^2 complex multiplications.
  - The well-known radix-2 CooleyTukey FFT algorithm requires $N \log_2 N$ complex multiplications[56].
  - The gain in complexity is approximately $\frac{N}{\log_2 N}$. For $N = 1024$, the gain in efficiency is around 100.
  - The practical implementation of the FFT requires N to be a power of 2.

[56] We have ignored the number of complex additions for simplicity.

Applications of the FFT

- Efficient discrete approximation of the continuous Fourier transform for **spectral analysis**.
- Computation of the **convolution of time sequences** of length M and K using the property $\mathcal{F}[x[k] * y[k]] = X(\Omega)Y(\Omega)$.
  - First the DFT is used. The resulting spectra are multiplied. Finally, the inverse DFT is used.
  - The gain is substantial for long sequences.
  - **Zero padding** is used to make sure that the 2 sequences have a minimum length $N \geq M + K - 1$. N is chosen as the first power of 2 that verifies the condition.
- **Denoising**: noise can often be detected and removed more easily in the frequency domain.

Discrete Fourier transform

Discrete Fourier transform

The Discrete Fourier Transform (DFT) of a periodic signal of period N is

$$X[n] = \sum_{k=0}^{N-1} x[k]\, W_N^{kn}, \quad 0 \leq n \leq N-1.$$

The inverse DFT is

$$x[k] = \frac{1}{N} \sum_{n=0}^{N-1} X[n]\, W_N^{-kn}, \quad 0 \leq k \leq N-1$$

with

$$W_N = e^{-j\frac{2\pi}{N}}.$$

Symmetry and periodicity of W_N

Properties:

$$W_N^{k+N} = e^{-j\frac{2\pi(k+N)}{N}} = e^{-j\frac{2\pi k}{N}} e^{-j\frac{2\pi N}{N}} = e^{-j\frac{2\pi k}{N}} = W_N^k$$

$$W_N^{k+\frac{N}{2}} = e^{-j\frac{2\pi(k+\frac{N}{2})}{N}} = e^{-j\frac{2\pi k}{N}} e^{-j\frac{\pi N}{N}} = -e^{-j\frac{2\pi k}{N}} = -W_N^k$$

$$W_N^{Nn} = e^{-j\frac{2\pi(Nn)}{N}} = e^{-j2\pi n} = 1$$

$$W_N^{2kn} = e^{-j\frac{2\pi}{N}2kn} = e^{-j\frac{2\pi}{\frac{N}{2}}kn} = W_{\frac{N}{2}}^{kn} \quad \text{(N even)}$$

The elementary DFT of length $N = 1$ is

$$X[0] = \sum_{k=0}^{N-1} x[k]\, W_N^{kn} = x[0].$$

Towards an efficient implementation of the DFT

- Choose a value of $N = r^l$.
- Often $r = 2$ is selected $\Longrightarrow$ **radix-2** algorithm[57].
- The radix-2 decimation-in-time algorithm rearranges the DFT equation into two parts: a sum over the even-numbered discrete-time indices and a sum over the odd-numbered indices.
- Efficient use of memory known as "in-place computation": FFT computations are normally performed in place in a one-dimensional array, with new values overwriting old values.

[57]The meaning of radix is "base", a Latin word meaning "root".

Radix-2 FFT algorithm

Suppose $N = 2^l$. Decompose the DFT sum into 2 parts corresponding to, respectively, the even and odd indices of the sequence $x[k]$

$$
\begin{aligned}
\underbrace{X[n]}_{\substack{\text{DFT over } N\\ \text{samples}}} &= \sum_{k=0}^{N-1} x[k] W_N^{kn} \\
&= \sum_{k=0}^{\frac{N-1}{2}} x[2k] W_N^{2kn} + \sum_{k=0}^{\frac{N-1}{2}} x[2k+1] W_N^{(2k+1)n} \\
&= \sum_{k=0}^{\frac{N-1}{2}} x[2k] W_N^{2kn} + W_N^n \sum_{k=0}^{\frac{N-1}{2}} x[2k+1] W_N^{2kn} \\
&= \underbrace{\sum_{k=0}^{\frac{N-1}{2}} x[2k] W_{\frac{N}{2}}^{kn}}_{\substack{\text{DFT over } \frac{N}{2}\\ \text{samples}\\ \text{even indices}}} + W_N^n \underbrace{\sum_{k=0}^{\frac{N-1}{2}} x[2k+1] W_{\frac{N}{2}}^{kn}}_{\substack{\text{DFT over } \frac{N}{2}\\ \text{samples}\\ \text{off indices}}}
\end{aligned}
$$

Radix-2 FFT algorithm: recursion

Idea: recursive use of the relation $\Longrightarrow$ decompose the DFT of length $N = 2^l$ in l steps in order to obtain 2^l DFTs of length 1.

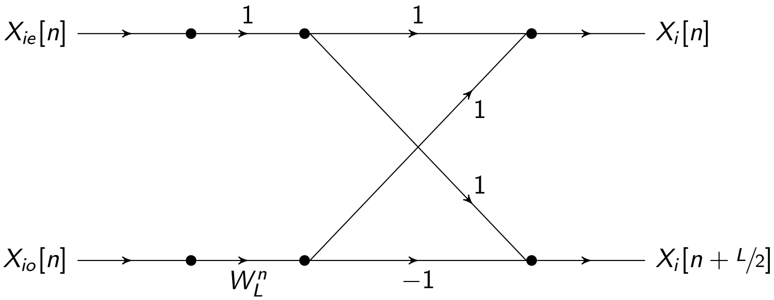
In step i, $X_i[n]$ is a DFT of length L and therefore $X_{ie}[n]$ and $X_{io}[n]$ are DFTs of length $^L/_2$

$$\begin{aligned} X_i[n] &= X_{ie}[n] + W_L^n X_{io}[n] \\ X_i[n + {}^L/_2] &= X_{ie}[n + {}^L/_2] + W_L^{(n+{}^L/_2)} X_{io}[n + {}^L/_2] \\ &= X_{ie}[n] + W_L^{(n+{}^L/_2)} X_{io}[n] \\ &= X_{ie}[n] - W_L^n X_{io}[n] \end{aligned}$$

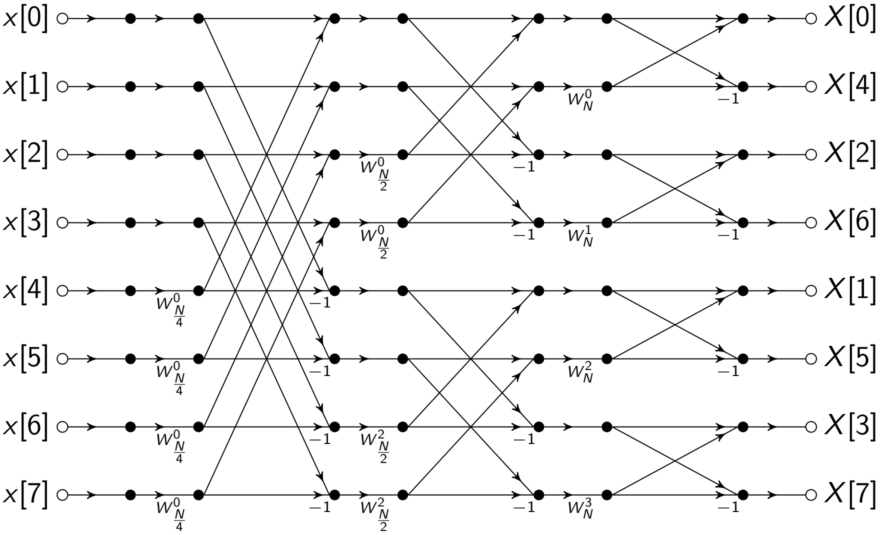
We obtain the elementary **butterfly**:

$$\boxed{\begin{aligned} X_i[n] &= X_{ie}[n] + W_L^n X_{io}[n] \\ X_i[n + {}^L/_2] &= X_{ie}[n] - W_L^n X_{io}[n] \end{aligned}}$$

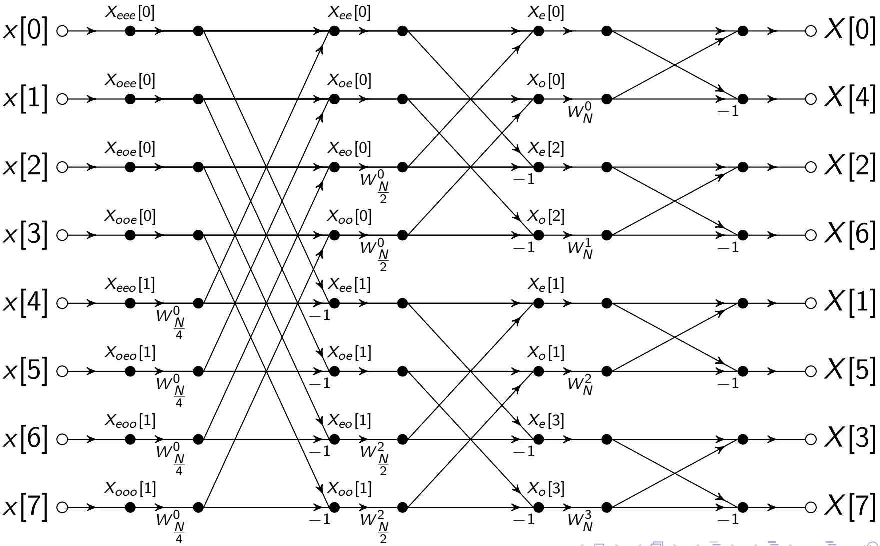
Radix-2 FFT algorithm: butterfly



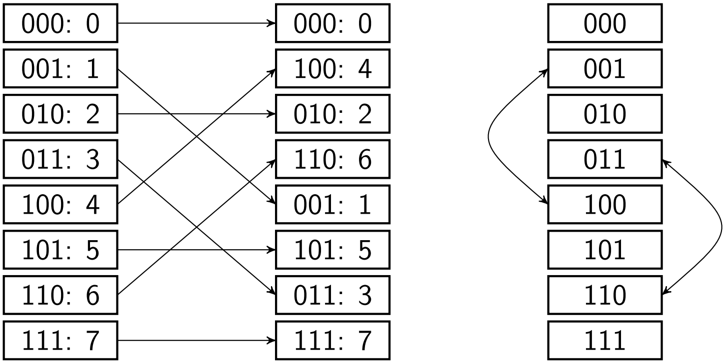
Radix-2 FFT algorithm: $N = 8$



Radix-2 FFT algorithm: $N = 8$



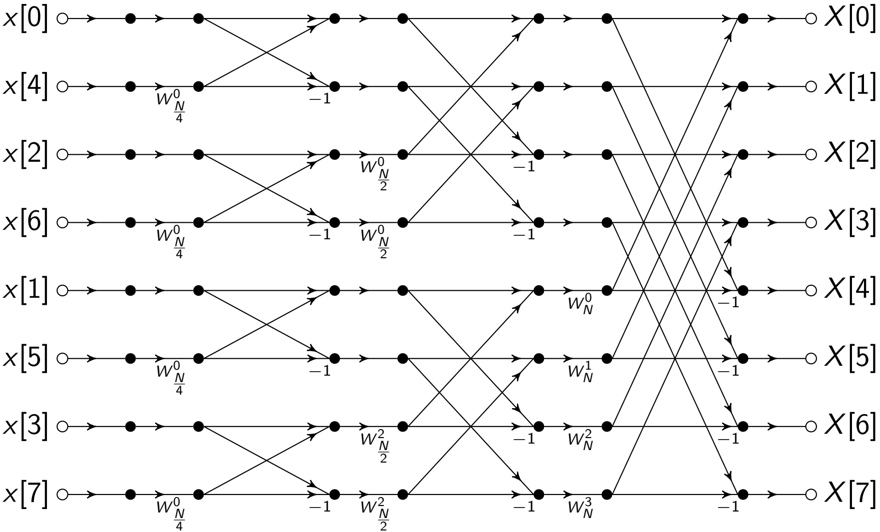
Radix-2 FFT algorithm: reverse-binary representation



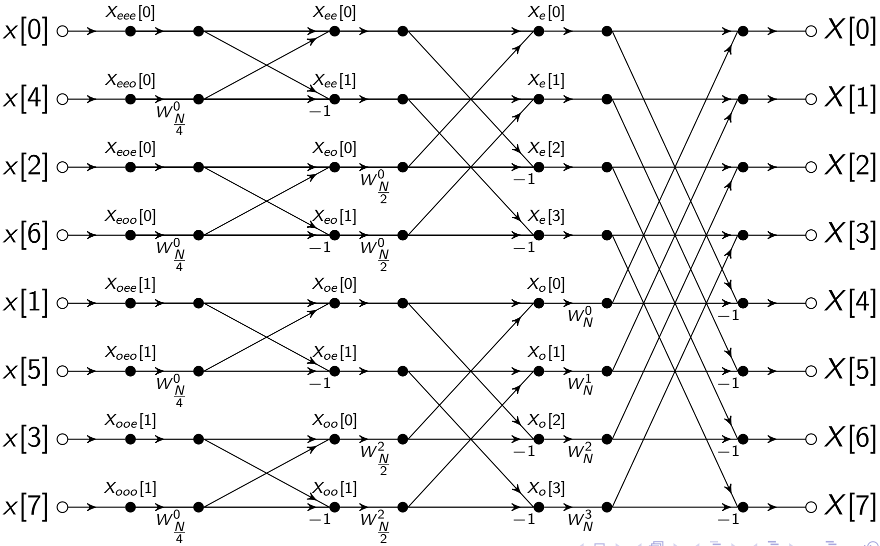
Binary representation

Reverse-binary representation

Radix-2 FFT algorithm: $N = 8$ with reorganisation



Radix-2 FFT algorithm: $N = 8$ with reorganisation



Radix-2 FFT algorithm: summary

- Rearrange the order of the N time domain samples by counting in binary with the bits flipped left-for-right.
- Find the frequency spectra of the 1 point time domain signals. This step is trivial as the frequency spectrum of a 1 point signal is equal to itself.
- Combine the N frequency spectra in the exact reverse order that the time domain decomposition took place. The basic butterfly pattern is repeated over and over.

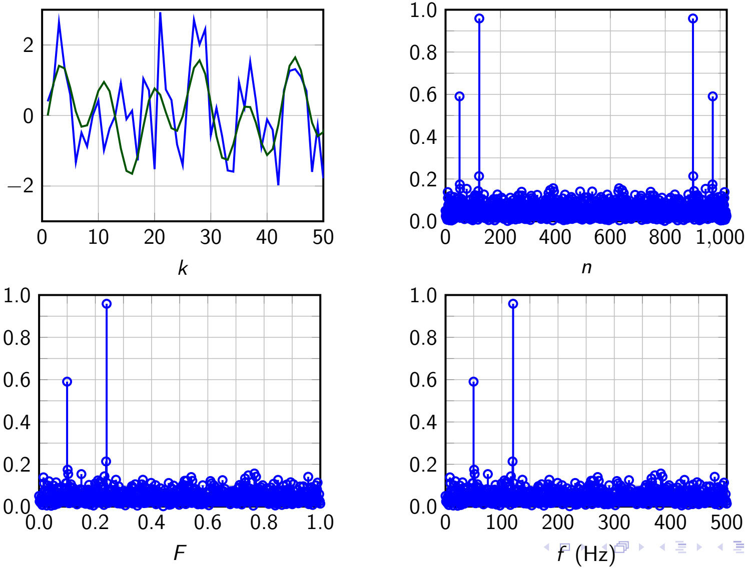
Radix-2 FFT algorithm

```
function [fft] = fft_comp(data,L)
%
% Franky De Bruyne
%
% Data is a column (!!!) vector with complex data
%
% N = 2^L
% if data has less than N points, it is padded with zeroes
% if data has more than N points, it is truncated to N points

% Make length a power of 2
% -------------------------

N = pow2(L);
N1 = length(data);
if (N1 < N)
    temp = [data; zeros(N-N1,1)];
elseif (N1 > N)
    temp = data(1:N);
else
    temp = data;
end
```

Radix-2 FFT algorithm

```
% Bit reversal routine
% ---------------------

for j=0:N-1
    nj = bin2dec(fliplr(dec2bin(j,L)));
    if ((j ~= nj) && (j < nj))
        tmp = temp(j+1);
        temp(j+1) = temp(nj+1); temp(nj+1) = tmp;
    end
end

% FFT computation
% ---------------

for j = 1:L     % Loop for each stage, i.e. DFTs with increasing data size
    theta = -2*pi/pow2(j);     % DFT of size 2^j
    Wj = exp(theta*1i); W = 1+0*1i;
    for k = 1:pow2(j-1)        % Loop for each sub DFT
        for l = k:pow2(j):N    % Implementation of butterfly
            tmp = temp(l+pow2(j-1))*W;
            temp(l+pow2(j-1)) = temp(l) - tmp; temp(l) = temp(l) + tmp;
        end
        W = W*Wj;
    end
end

fft = temp;
```

Noisy sinusoid



Noisy sinusoid

```
Fs = 1000;                  % Sampling frequency
T = 1/Fs;                   % Sample time
t = (0:999)*T;t = t';       % Time vector
t = t';
N = 2^10;

yr = randn(size(t));        % Random noise
k50 = 0.7; k120 = 1;        % Sum of a 50 Hz sinusoid and a 120 Hz sinusoid
ys = k50*sin(2*pi*50*t) + k120*sin(2*pi*120*t);
yrs = yr + ys;

figure(1)
plot([yrs ys]);
axis([0 50 -3 3]), grid on, xlabel('k')

figure(2)
X = fft_comp(yrs',10);
X = fft(yrs,N);             % built-in fft command
n = 0:1:N-1;
plot((n/(N/2))*Fs/2,(2/N)*abs(X),'o')
hold on
plot((n/(N/2))*Fs/2,(2/N)*abs(X))
hold off, axis([0 Fs/2 0 1]), grid on, xlabel('f (Hz)')
```

Convolution using the FFT

Give $x[k] = \{1, -3, 1, 5\}$ and $h[k] = \{4, 3, 2, 1\}$, compute their convolution using the DFT.

The resulting convolution sequence will have a support of $4 + 4 - 1 = 7$. It is therefore necessary to add 3 zeros at the end of each sequence. The DFT of each sequence yields

$$X[n] = \{4.0, -5.60 - j0.80, 3.88 + j7.27, 3.21 - j2.79, 3.21 + j2.79, 3.88 - j7.27, -5.60 + j0.80\},$$

$$H[n] = \{10, 4.52 - j4.73; 2.15 - j1.28; 2.32 - j0.72, 2.32 + j0.72, 2.15 + j1.28, 4.52 + j4.73\}.$$

Point to point multiplications yields

$$Y[n] = \{40, -29.11 + j22.86, 17.63 + j10.72, 5.47 - j8.77, 5.47 + j8.77, 17.63 - j10.70, -29.11 - j22.86\}.$$

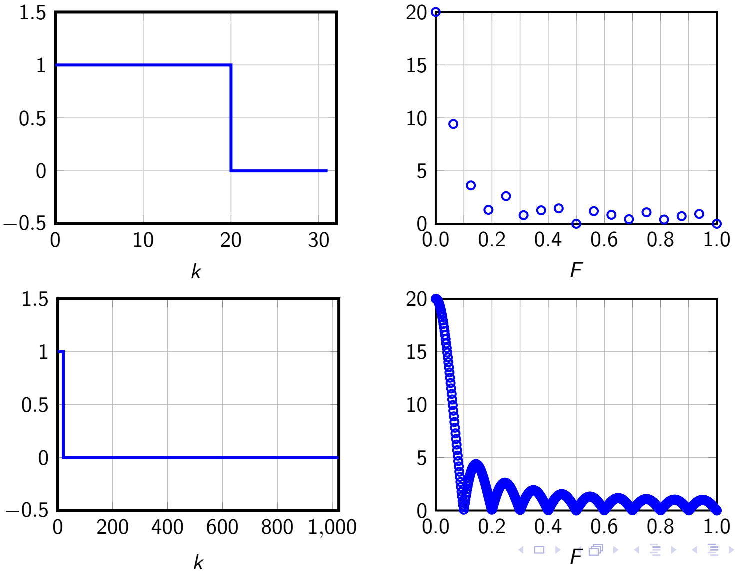
Finally, the inverse DFT gives

$$y[k] = \{4, -9; -3; 18; 14; 11; 5\}.$$

Convolution using the FFT

```
x = [1 -3 1 5];              % Length of convolution product
y = [4 3 2 1];               % 4 + 4 — 1 = 7
x_padded = [x zeros(1,3)];   % => Zero padding (3 samples)
y_padded = [y zeros(1,3)];
X = fft(x_padded,8);         % FFT
Y = fft(y_padded,8);         % FFT
Z = X.*Y;                    % Point to point multiplication
z = ifft(Z,8);               % Inverse FFT
```

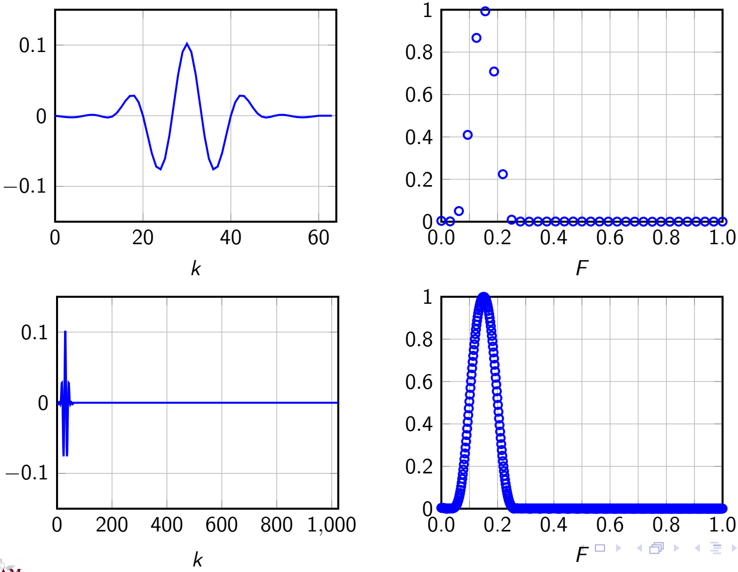
Zero padding



Zero padding

```
x1 = [ones(20,1); zeros(12,1)];
fft1 = fft(x1,32);
x2 = [x1;zeros(992,1)];
fft2 = fft(x2,1024);

subplot(2,2,1)
k1 = 0:1:31;
plot(k1',x1)
axis([0 32 -0.5 1.5]), xlabel('k')

subplot(2,2,2)
plot(k1/16,abs(fft1),'o')
hold on
plot(k1/16,abs(fft1))
axis([0 1 0 20]), xlabel('F')

subplot(2,2,3)
k2 = 0:1:1023;
plot(k2',x2)
axis([0 1024 -0.5 1.5]), xlabel('k')

subplot(2,2,4)
plot(k2/512,abs(fft2),'o')
hold on
plot(k2/512,abs(fft2))
hold off
axis([0 1 0 20]), xlabel('F')
```

Spectral analysis



Spectral analysis

```
h_band_pass = fir1(60,[0.1 0.2]);   % FIR bandpass filter: pkg load signal
imp = zeros(64,1); imp(1) = 1;      % if using Octave with signal toolbox
x1 = filter(h_band_pass,1,imp);     % Impulse response
fft1 = fft(x1,64);                  % FFT N=64
x2 = [x1;zeros(960,1)];
fft2 = fft(x2,1024);                % FFT N=1024

subplot(2,2,1)
k1 = 0:1:63;
plot(k1',x1)
axis([0 64 -0.15 0.15]), xlabel('k')

subplot(2,2,2)
plot(k1/32,abs(fft1),'o'); hold on
plot(k1/32,abs(fft1)); hold off
axis([0 1 0 1]), xlabel('F')

subplot(2,2,3)
k2 = 0:1:1023;
plot(k2',x2)
axis([0 1024 -0.15 0.15]), xlabel('k')

subplot(2,2,4)
plot(k2/512,abs(fft2),'o'); hold on
plot(k2/512,abs(fft2)); hold off
axis([0 1 0 1]), xlabel('F')
```

10. Getting started with ...

Getting started with ...

Matlab

www.mathworks.com

registered trademarks of The MathWorks, Inc.

Octave

www.octave.org

distributed under the terms of the GNU General Public License

Scilab

www.scilab.org

open source software distributed under CeCILL license

Simulink

Xcos

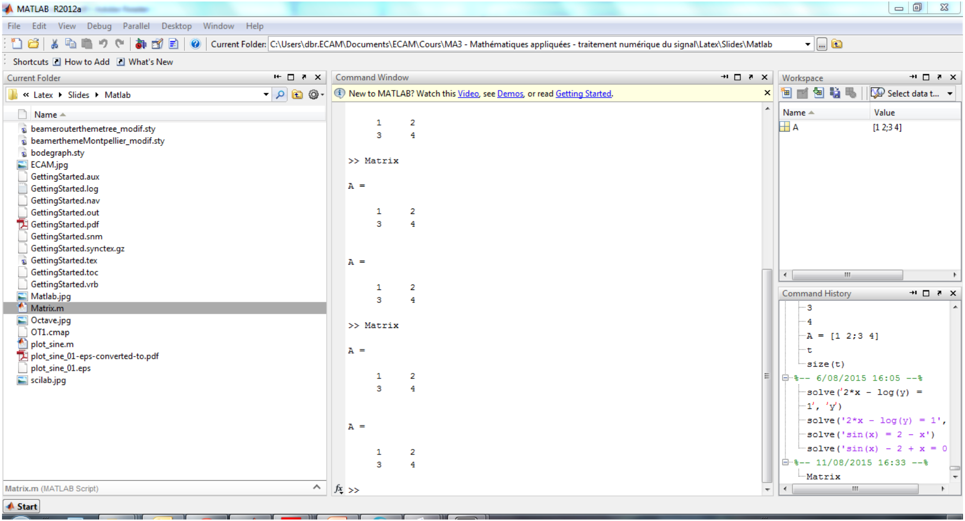
Getting started with ...

Matlab, Octave and Scilab are high-level languages and mathematical programming environments for:

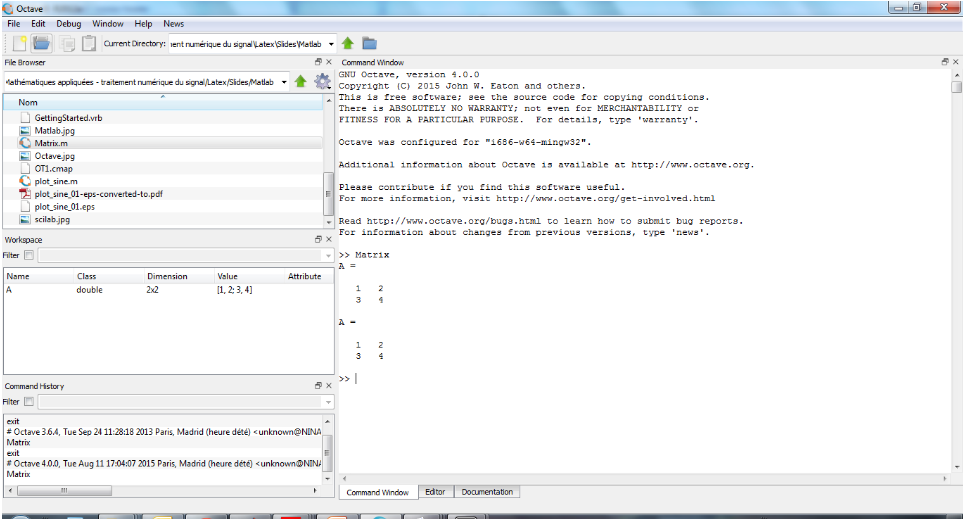
- Visualization
- Programming, algorithm development, prototyping
- Scientific computing: linear algebra, optimization, control, statistics, signal and image processing, etc.

This tutorial applies to Matlab, Octave and Scilab unless stated otherwise !

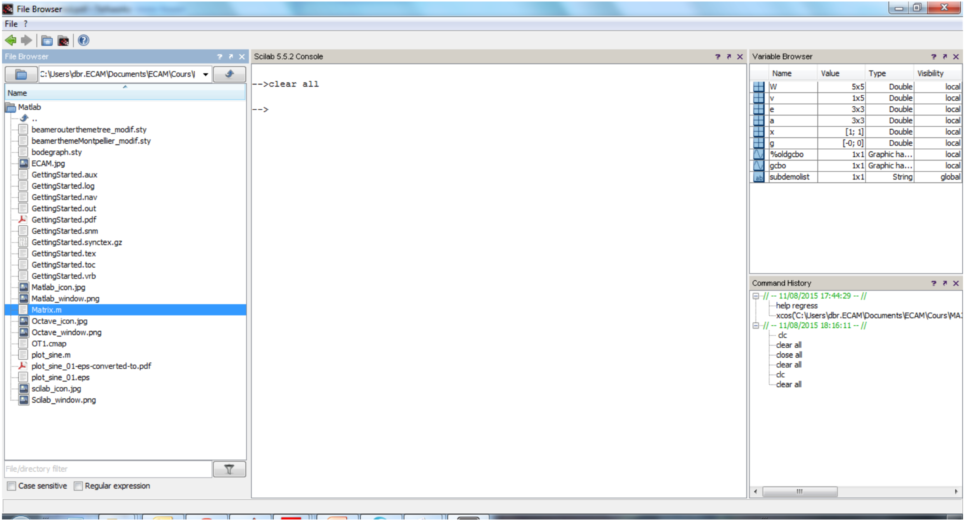
Matlab window (Matlab 2012a)



Octave window (version 4.0.0)



Scilab window (version 5.5.2)



Working environment

- Command window
- Command history
- Workspace / Variable browser
- Current Directory / File browser

Getting help

To get help on a specific command (= built-in function), use the `help` command.

```
help plot
```

```
PLOT   Linear plot.
   PLOT(X,Y) plots vector Y versus vector X. If X or Y is a matrix,
   then the vector is plotted versus the rows or columns of the matrix,
   whichever line up.  If X is a scalar and Y is a vector, disconnected
   line objects are created and plotted as discrete points vertically at
   X.

   PLOT(Y) plots the columns of Y versus their index.
   If Y is complex, PLOT(Y) is equivalent to PLOT(real(Y),imag(Y)).
   In all other uses of PLOT, the imaginary part is ignored.

   Various line types, plot symbols and colors may be obtained with
   PLOT(X,Y,S) where S is a character string made from one element
   from any or all the following 3 columns:

          b     blue          .     point              -     solid
```

Input and output

Input commands are entered in the command window. The output is returned in two ways:

- typically, **text or numerical output** is returned in the same **command window**,
- but **graphical output** appears in a separate **figure window**.

Arithmetics

Matlab, Octave and Scilab can be used to do arithmetic as one would with a calculator. You can add with +, subtract with −, multiply with ∗, divide with /, and exponentiate with ˆ.

For example:

```
3^2 - (5 + 4)/2 + 6*3
```

```
ans =

   22.5000
```

The answer is assigned to a variable called `ans`. If one wants to perform further calculations with the answer, one can use the variable `ans` rather than retype the answer.

Arithmetics

A number of **elementary functions** are available. Look for the associated help functionalities or google

"elementary functions program"

with program replaced by Matlab, Octave or Scilab.

For example, the Matlab help yields:

```
help elfun
```

```
  Elementary math functions.

  Trigonometric.
    sin         - Sine.
    sind        - Sine of argument in degrees.
    sinh        - Hyperbolic sine.
    asin        - Inverse sine.
```

Use of the semicolon (;)

If one wants to suppress and hide the output for an expression, add a semicolon (;) after the expression.

For example:

```
x = 3;
y = x + 5

y =

     8
```

Adding comments

The percent symbol (%) is used to indicate a comment line in .

Two consecutives slashes (//) are used in .

For example:

```
x = 3;       % Assign the value 3 to x
y = x + 5    % Add 3 to x and assign the result to y
```

```
y =

     8
```

Retrieving previous commands

- To select commands in the command history, press the up-arrow and down-arrow keys, $\uparrow$ and $\downarrow$, in the command window.
- To retrieve a command using a partial match, type any part of the command at the prompt, and then press the $\uparrow$ key.
- A new command can sometimes be entered more efficiently by modifying a previous command.

Saving the workspace

The `save` command is used for saving all the variables in the workspace in the current directory. The extension is program dependent. For example:

```
save myfile
```

You can reload the file anytime later using the `load` command.

```
load myfile
```

Variables

In the considered environments, every variable is an array or matrix.

You can assign variables in a simple way. For example:

```
x = 3
```

```
x =

     3
```

It creates a 1-by-1 matrix named `x` and stores the value 3 in its element.

Variables

Another example:

```
x = sqrt(16)
```

```
x =

     4
```

Please note that:

- Once a variable is entered into the system, one can refer to it later.
- Variables must have values before they are used.
- When an expression returns a result that is not assigned to any variable, the system assigns it to a variable named `ans`, which can be used later.

Variables: useful commands

The `who` command displays all the variable names one has used. The `whos` command displays more informations about the variables:

```
whos
```

```
Name      Size          Bytes  Class     Attributes

A         2x2              32  double
ans       1x1               8  double
x         2x1             112  sym
y         2x1             112  sym
```

Variables: useful commands

The `clear` command deletes all (or the specified) variable(s) from the memory. For example:

```
x = 3; y = 4;
who
clear x
who
clear
who
```

```
Your variables are:

x   y

Your variables are:

y
```

Variables: useful commands

By default, numbers are displayed with four decimal place values. This is known as the short format.

```
format short
x = 7 + 10/3 + 5^1.2
```

```
x =

  17.2320
```

However, if one wants more precision, one needs to use the `format` command. The format long command displays 16 digits after decimal.

```
format long
x = 7 + 10/3 + 5^1.2
```

```
x =

  17.231981640639408
```

Creating vectors

Row vectors are created by enclosing the set of elements in square brackets, using space or comma to delimit the elements.

For example:

```
x = [1 2 3]
y = [1,2,3]
```

```
x =

     1     2     3

y =

     1     2     3
```

Creating vectors

Column vectors are created by enclosing the set of elements in square brackets, using semicolon(;) to delimit the elements.

For example:

```
x = [1; 2; 3]
```

```
x =

     1
     2
     3
```

Creating matrices

A matrix is a two-dimensional array of numbers.

A **matrix** is created by entering each row as a sequence of space or comma separated elements; and end of a row is demarcated by a semicolon.

For example:

```
x = [1 2 3;4 5 6;7 8 9]
```

```
x =

     1     2     3
     4     5     6
     7     8     9
```

Accessing parts of a matrix

Accessing element (i, j), i.e. element in row i and column j.

For example:

```
x = [1 2 3;4 5 6;7 8 9];
y = x(2,3)
```

```
y =

     6
```

Accessing parts of a matrix

Accessing a row.

For example:

```
x = [1 2 3;4 5 6;7 8 9];
y = x(2,:)
```

```
y =

     4     5     6
```

Accessing parts of a matrix

Accessing a column.

For example:

```
x = [1 2 3;4 5 6;7 8 9];
y = x(:,3)
```

```
y =

     3
     6
     9
```

Accessing parts of a matrix

Accessing a given range.

For example:

```
x = [1 2 3;4 5 6;7 8 9];
y = x(1:2,2:3)
```

```
y =

     2     3
     5     6
```

Combinations of matrices

Combining matrices.

For example:

```
x = [1 2 3;4 5 6;7 8 9];
y = [3; 4; 9];
z = [0 9 8 7];
A = [x y; z]
```

```
A =

     1     2     3     3
     4     5     6     4
     7     8     9     9
     0     9     8     7
```

Matrix operations

Matlab, Octave and Scilab can be used to do **matrix operations**.

Assuming compatibily, one can add with +, subtract with −, multiply with ∗. The operator `.*` is used to do multiplication element by element.

For example:

```
A = [1 2;3 4]; B = [1; 2]; C = [1 2];
X = A^2
Y = A*B
Z = C*A
```

```
X =

     7    10
    15    22

Y =

     5
    11

Z =

     7    10
```

Matrix transpose

The operator ' is used for **matrix transposition**.

For example:

```
B = [1 2;3 4;5 6]
C = B'
```

```
B =

     1     2
     3     4
     5     6

C =

     1     3     5
     2     4     6
```

Matrices functions

Matrix functions.

For example:

```
x = [1 2 3;4 0 6;7 8 9];
d = det(x)
ix = inv(x)
```

```
d =

   60.0000

ix =

   -0.8000    0.1000    0.2000
    0.1000   -0.2000    0.1000
    0.5333    0.1000   -0.1333
```

Matrices functions

A number of **matrix functions** are available. Look for the associated help functionalities or google

"matrix functions program"

with program replaced by Matlab, Octave or Scilab.

For example, the Matlab help yields:

```
help matfun
```

```
  Matrix functions - numerical linear algebra.

  Matrix analysis.
    rank        - Matrix rank.
    det         - Determinant.
    trace       - Sum of diagonal elements.

  Linear equations.
    \ and /     - Linear equation solution; use "help slash".
    inv         - Matrix inverse.
```

Special matrices

Matrices of ones, matrices of zeros and identity matrices can easily be constructed using the commands `ones`, `zeros` and `eye`.

For example:

```
A = eye(3)
B = eye(2,3)
```

```
A =

     1     0     0
     0     1     0
     0     0     1


B =

     1     0     0
     0     1     0
```

Diagonal matrices

Diagonal matrices can be constructed using the command `diag`.

For example:

```
diag([1 2 3])
```

```
ans =

     1     0     0
     0     2     0
     0     0     3
```

Vector and matrix size

The size of a matrix array can be obtained using the command `size`. The length of a vector can be ontained using the command `length`.

For example:

```
a = [1 2];
length(a)
B = eye(2,3);
size(B)
```

```
ans =

     2


ans =

     2     3
```

Complex computations

Complex numbers can be constructed using the function `complex`. A number of functions like real, `imag`, `abs` and `conj` can be used.

For example:

```
a = complex(1,1);
real(a)
abs(a)
conj(a)
```

```
ans =

     1

ans =

    1.4142

ans =

   1.0000 - 1.0000i
```

Complex computations

Complex numbers can also be entered directly. The function `angle` can also be used.

For example:

```
a = 1 + 1i;
real(a)
abs(a)
angle(a)
```

```
ans =

     1

ans =

    1.4142

ans =

    0.7854
```

Script files

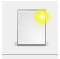
- For simple problems, entering your requests at the prompt is fast and efficient.
- However, as the number of commands increases typing the commands over and over at the prompt becomes tedious.
- Script files are the **main tool** for writing code.

Script files

In order to create and run a script file, one needs to:

- Open a new script file

Matlab



Octave

Scilab

- Give it a name. Be sure the name is not an existing function !
- Write your instructions inside the file. Write comments in your program !
- Save it in the current directory.
- "Call it", i.e. type the file name on the command window.

Strings

- Strings in MATLAB are written between **single quotes**.
- A quotation within the string is indicated by two quotation marks.
- **Concatenation** of strings is done by using **square brackets**.
- The function `num2str` is used to convert a numerical result into a string.
- **Reading from the keyboard** can be accomplished by using the `input` function.

Strings

```
clear all
str = 'Example''s result';
a = input('Enter a real value to be multiplied by 2: ');
a = 2*a;
str = [str ': ' num2str(a)];
disp(str)
whos
```

```
Enter a real value to be multiplied by 2: 3.3
Example's result: 6.6

  Name      Size            Bytes  Class     Attributes

  a         1x1                 8  double
  str       1x21               42  char
```

Plotting

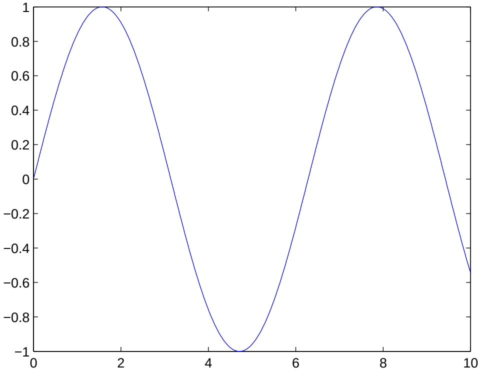
To plot the graph of a function, one needs the following steps:

- Define `x`, by specifying the range of values for the variable `x`, for which the function is to be plotted.
- Define the function, `y = f(x)`.
- Call the plot command as `plot(x,y)`.

Plotting

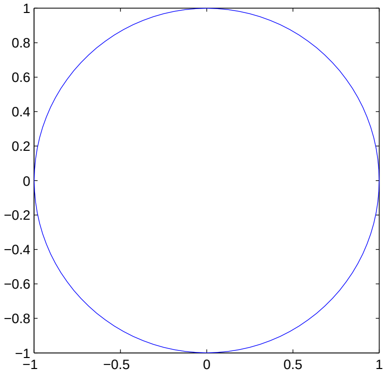
Create a script file and type the following code and execute it !

```
x = 0:0.01:10;
y = sin(x);
plot(x,y);
```



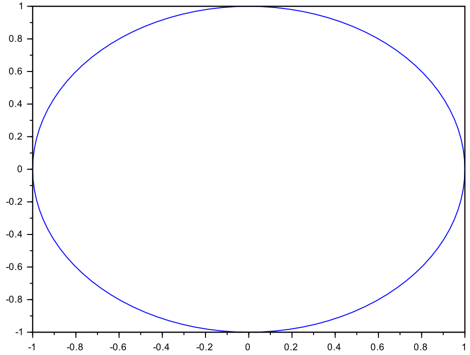
Parametric plot

```
t = 0:0.01:1;
plot(cos(2*pi*t),sin(2*pi*t))
axis square
```



Parametric plot

```
t = 0:0.01:1;
plot(cos(2*%pi*t),sin(2*%pi*t))
```



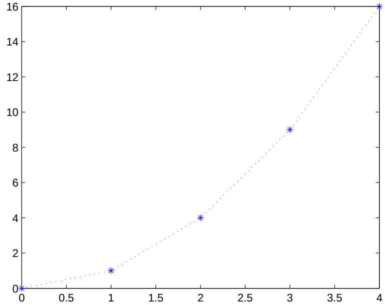
help plot

```
help plot
```

```
PLOT   Linear plot.
    PLOT(X,Y) plots vector Y versus vector X. If X or Y is a matrix,
    then the vector is plotted versus the rows or columns of the matrix,
    whichever line up.  If X is a scalar and Y is a vector, disconnected
    line objects are created and plotted as discrete points vertically at X.

    Various line types, plot symbols and colors may be obtained with
    PLOT(X,Y,S) where S is a character string made from one element
    from any or all the following 3 columns:

           b     blue          .     point              -     solid
           g     green         o     circle             :     dotted
           r     red           x     x-mark             -.    dashdot
           c     cyan          +     plus               --    dashed
           m     magenta       *     star             (none)  no line
           y     yellow        s     square
           k     black         d     diamond
           w     white         v     triangle (down)
                               ^     triangle (up)
```

Plotting points

```
x = [0 1 2 3 4];
y = [0 1 4 9 16];
plot(x,y,'b:*')
```

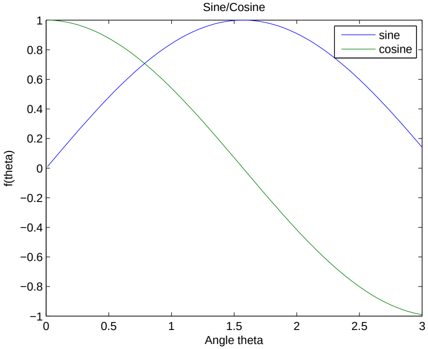


Plotting: summary of useful commands

- `figure` - opens window
- `hold on`, `hold off` - between these commands everything is plotted in the same window
- `grid on`, `grid off` - add or remove grid lines
- `subplot` - create several different plots in one figure
- `xlabel`, `ylabel` - adds text beside the X-axis/Y-axis on the current axis
- `title` - adds text at the top of the current axis
- `legend` - display legend
- `axis` - control axis scaling and appearance

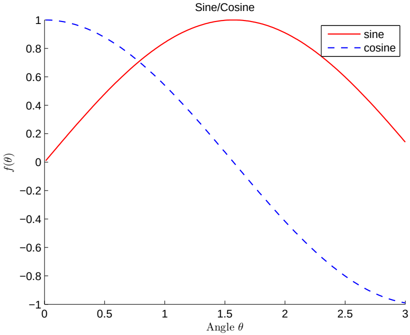
Several plots

```
figure
x = 0.01:0.01:3;
y1 = sin(x); y2 = cos(x);
plot(x,[y1; y2]);
legend('sine','cosine');
xlabel('Angle theta');
ylabel('f(theta)');
title('Sine/Cosine');
```



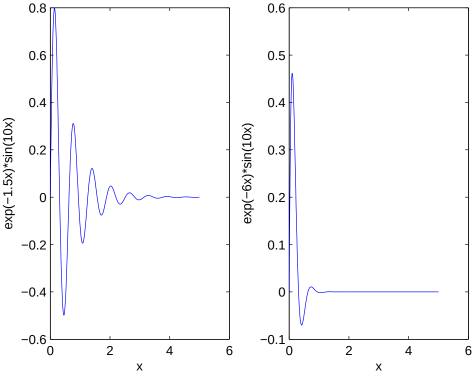
Several plots with hold function

```
figure, hold on
x = 0.01:0.01:3;
y1 = sin(x); y2 = cos(x);
plot(x,y1,'-r','LineWidth',1);
plot(x,y2,'--b','LineWidth',1);
legend('sine','cosine');
xlabel('Angle $\theta$','interpreter','latex');
ylabel('$f(\theta)$','interpreter','latex');
title('Sine/Cosine');
hold off
```



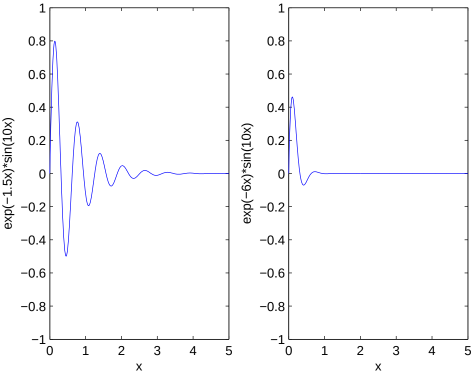
Subplots

```
x = 0:0.01:5;
y = exp(-1.5*x).*sin(10*x);
subplot(1,2,1)
plot(x,y), xlabel('x'), ylabel('exp(-1.5x)*sin(10x)')
y = exp(-6*x).*sin(10*x);
subplot(1,2,2)
plot(x,y),xlabel('x'), ylabel('exp(-6x)*sin(10x)')
```



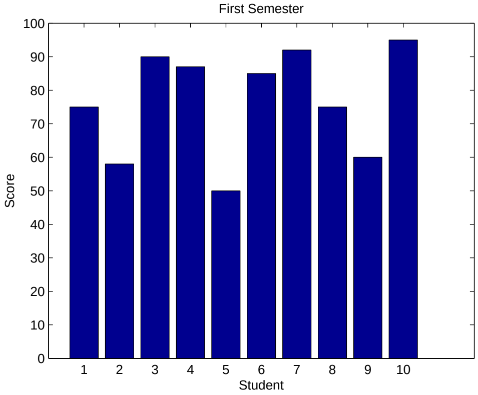
Subplots with axis function

```
x = 0:0.01:5;
y = exp(-1.5*x).*sin(10*x);
subplot(1,2,1)
plot(x,y), xlabel('x'), ylabel('exp(-1.5x)*sin(10x)'), axis([0 5 -1 1])
y = exp(-6*x).*sin(10*x);
subplot(1,2,2)
plot(x,y),xlabel('x'), ylabel('exp(-6x)*sin(10x)'), axis([0 5 -1 1])
```



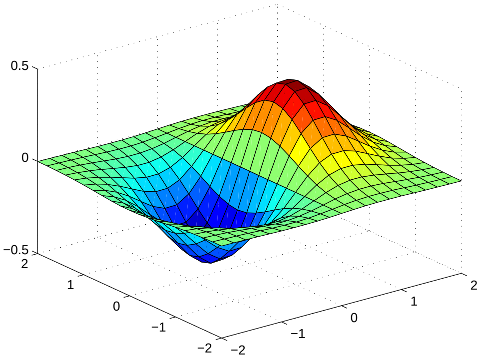
Bar charts

```
x = 1:10;
y = [75,58,90,87,50,85,92,75,60,95];
bar(x,y), xlabel('Student'), ylabel('Score'),
title('First Semester')
```



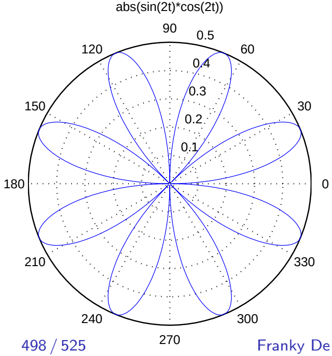
3D surface map

```
[x, y] = meshgrid(-2:.2:2);
g = x .* exp(-x.^2- y.^2);
surf(x, y, g)
```



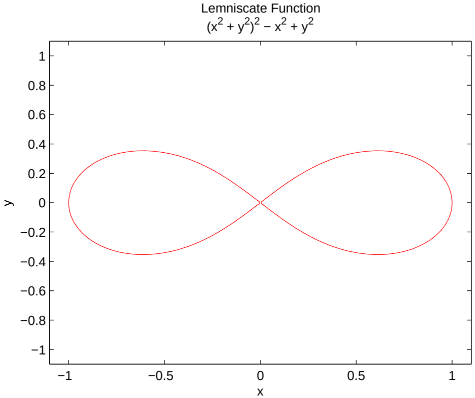
Polar plots

```
% Create data for the function
t = 0:0.01:2*pi;
r = abs(sin(2*t).*cos(2*t));

% Create a polar plot using polar
figure;
polar(t, r);

% Add a title
title('abs(sin(2t)*cos(2t))');
```



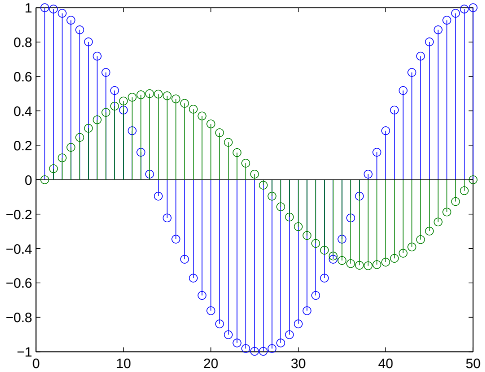
Plotting implicitly defined functions

```
figure;
h = ezplot('(x^2 + y^2)^2 - x^2 + y^2',[-1.1, 1.1], [-1.1, 1.1]);
set(h,'color','red')
grid on

title({'Lemniscate Function';'(x^2 + y^2)^2 - x^2 + y^2'});
```



Discrete time plots

```
figure
X = linspace(0,2*pi,50)';
Y = [cos(X), 0.5*sin(X)];
stem(Y)
```



Function files

- Function files play the role of **user defined commands** that often have input and output.
- One can create own commands for specific problems this way, which will have the same status as other commands.

Function files

```
function [a] = log3(x)
% [a] = log3(x) - Calculates the base 3 logarithm of x.
a = log(abs(x))./log(3);
% End of function
```

```
help log3
log3(2)
```

```
[a] = log3(x) - Calculates the base 3 logarithm of x.

ans =

    0.6309
```

Looping with `for`

- Structure:

```
for var = expression,
    body;
end
```

- Nesting of `for` loops is allowed.

Looping with `for`

```
A = [1 5 -3;2 4 0;-1 6 9];
for i = 1:3
    for j = 1:3
        B(i,j) = A(i,j)^2;
    end
end
B
```

```
B =

     1    25     9
     4    16     0
     1    36    81
```

Looping with `while`

- Structure:

```
while condition,
    body;
end
```

- Nesting of `while` loops is allowed.

Looping with `while`

```
i = 1;
while (i < 3)
  disp(i);
  i = i + 1;
end
```

1

2

Branching with `if`

- Structure:

```
if condition
    then-body;
elseif condition
    elseif-body;
else
    else-body;
end
```

- The `else` and `elseif` clauses are optional.
- Any number of `elseif` clauses may exist.
- Nesting of `if` branches is allowed.

Branching with `if`

```
a = 100;
if a == 10
    disp('Value of a is 10');
elseif( a == 20 )
    disp('Value of a is 20');
elseif a == 30
    disp('Value of a is 30');
else
    disp('None of the values are matching');
end
```

```
None of the values are matching
```

Branching with switch

- Structure:

```
switch expression

case label
    command-list;
case label
    command-list;
...
otherwise
    command-list;
end
```

- Any number of case labels are allowed.

Branching with switch

```
grade = 'B';
switch(grade)

case 'A'
        disp('Excellent!');
case 'B'
        disp('Above average');
case 'C'
        disp('Average');
case 'D'
        disp('Acceptable');
case 'F'
        disp('Fail');
otherwise
        disp('Invalid grade');
end
```

```
Above average
```

Relational operators

- `x < y` true if x is less than y
- `x <= y` true if x is less than or equal to y
- `x == y` true if x is equal to y
- `x >= y` true if x is greater than or equal to y
- `x > y` true if x is greater than y
- `x ~= y` true if x is not equal to y

Boolean expressions

- `B1 & B2` Element-wise logical and
- `B1 | B2` Element-wise logical or
- `~B` Element-wise logical not

```
A = [0 1 1 0 1];
B = [1 1 0 0 1];
A & B
```

```
ans =

     0     1     0     0     1
```

Polynomials

- The polynomial
$$p(x) = x^4 + 7x^2 - x$$
is represented by `p = [1, 0, 7, -1, 0]`.
- An n-th order polynomial is represented by vector of length $n + 1$.
- if the polynomial is missing any coefficients, zeros must be entered in the appropriate place(s) in the vector, as done above.
- The value of a polynomial at a given x can be found using the `polyval` command.
- Finding the roots of a polynomial is done using the `root` command.
- The product of two polynomials is found by taking the convolution of their coefficients using the command `conv`.

Polynomials

```
p = [1 0 7 -1 0];
polyval(p,-1)
roots(p)
```

```
ans =

     9

ans =

        0
  -0.0712 + 2.6486i
  -0.0712 - 2.6486i
   0.1424
```

Polynomials

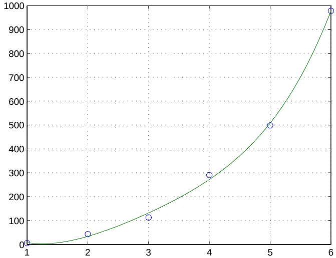
```
p = [1 0 -1];
q = [1 3];
r = conv(p,q)
roots(r)
```

```
r =

     1     3    -1    -3

ans =

   -3.0000
    1.0000
   -1.0000
```

Polynomial curve fitting

The `polyfit` function finds the coefficients of a polynomial that fits a set of data in a least-squares sense.

```
x =[1  2  3  4  5  6];
y = [5.5  43.1  112.8  290.7  498.4  978.7];

p = polyfit(x,y,4);

x2 = 1:.1:6;
y2 = polyval(p,x2);
plot(x,y,'o',x2,y2)
grid on
```



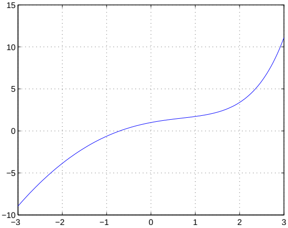
Numerical methods

- Suppose we are interested in finding the **roots** of a general non-linear function. This can be done in through the command `fzero`, which is used to approximate the root of a function of one variable, given an initial guess.
- Another useful command is `fminsearch` which finds the **minimum of a function**.
- The function `quad` can be used for **numerical integration**.

Numerical methods

```
function [val] = func(x)
val = exp(x) - x.^2;
```

```
x=-3:.01:3;
plot(x,func(x));
grid
fzero('func',-0.5)
```




```
ans =

   -0.7035
```

Numerical methods: minimisation

```
fun = inline('100*(x(2)-x(1)^2)^2+(1-x(1))^2');
x0 = [5 5];
[x,fval,exitflag,output] = fminsearch(fun,x0)
```

```
x =

    1.0000    1.0000

fval =

   5.6197e-10

exitflag =

     1

output =

    iterations: 106
     funcCount: 201
     algorithm: 'Nelder-Mead simplex direct search'
       message: [1x194 char]
```

Numerical methods: integration

```
function [val] = func(x)
val = exp(x) - x.^2;
```

```
quad('func',0,1)
quad('sin',0,pi)
```

```
ans =

    1.3849

ans =

    2.0000
```

Symbolic computations

- Using Matlabs and Octaves symbolic toolbox, one can carry out algebraic or symbolic calculations such as factoring and expanding polynomials.
- Type `help symbolic` to **make sure** that the symbolic toolbox is **installed** in Matlab.
- Type `pkg load symbolic` to **make sure** that the symbolic toolbox is **installed** in Octave. The installation procedure is quite elaborate.
- Use `syms` or `sym` to declare to declare the variables that one plans to be symbolic variables.
- Use the commands `expand`, `factor`, `simplify`, `subs` to perform symbolic computations.
- The commands `diff` and `int` can be used to differentiate and integrate symbolic expressions.
- The commands `limit` and `taylor` to compute limits and Taylor series.

Symbolic computations

```
syms x y
factor(x^3- y^3)
expand((x+2)*(x-3)*(x-5)*(x+7))
simplify((x^4-16)/(x^2-4))
z = x^3- y^3;
subs(z, x, 1)
```

```
ans =

(x - y)*(x^2 + x*y + y^2)

ans =

x^4 + x^3 - 43*x^2 + 23*x + 210

ans =

x^2 + 4

ans =

1 - y^3
```

Symbolic computations

```
syms x
diff(sin(x^2))
int(x^3)

limit(abs(x)/x, x, 0,'left')
taylor(sin(x),x,pi/2,'Order',6)
```

```
ans =

2*x*cos(x^2)

ans =

x^4/4

ans =

-1

ans =

(pi/2 - x)^4/24 - (pi/2 - x)^2/2 + 1
```

Symbolic computations

```
pkg load symbolic
x = sym('x'); y = sym('y');
factor(x^3- y^3)
expand((x+2)*(x-3)*(x-5)*(x+7))
simplify((x^4-16)/(x^2-4))
z = x^3- y^3;
subs(z, x, 1)
```

```
ans = (sym)

          / 2          2\
  (x - y)*\x  + x*y + y /

ans = (sym)

   4    3       2
  x  + x  - 43*x  + 23*x + 210

ans = (sym)

   2
  x  + 4

ans = (sym)

     3
  - y  + 1
```

Symbolic computations

```
pkg load symbolic
x = sym ('x');
diff(sin(x^2))
int(x^3)
limit(abs(x)/x, x, 0,'left')
taylor(sin(x),x,pi/2,'Order',6)
```

```
ans = (sym)

         / 2\
  2*x*cos\x /

ans = (sym)

   4
  x
  --
  4

ans = (sym) -1

ans = (sym)

          4            2
  /    pi\    /    pi\
  |x - --|    |x - --|
  \    2 /    \    2 /
  --------- - --------- + 1
      24           2
```