Time-Domain Analysis of Continuous-Time Systems*

*Systems are LTI from now on unless otherwise stated

Recall course objectives

Main Course Objective:

Fundamentals of systems/signals interaction

(we'd like to understand how systems transform or affect signals)

Specific Course Topics:

-Basic test signals and their properties

-Systems and their properties

-Signals and systems interaction

Time Domain: convolution

Frequency Domain: frequency response

-Signals & systems applications:

audio effects, filtering, AM/FM radio

-Signal sampling and signal reconstruction

Signals & Systems interaction in the TD

Goals

I. Impulse Response (IR) and Convolution Formula

- Definition of IR and its use for system identification
- Convolution formula and its graphical interpretation

II. Properties of systems from IR and convolution

- Impulse response as a measure of system memory/stability
- Alternative measures of memory/stability: step response

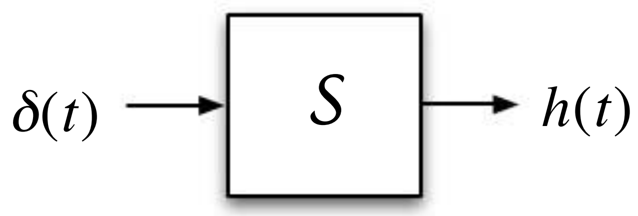
III. Applications of convolution

- Audio effects: reverberation
- Noise removal (i.e. signal filtering or smoothing)

The Impulse Response

Problem:

Find the response of a system to an impulsive input.



$h(t)$ is called the ***Impulse Response***.

Use what you know about LTI systems to compute $h(t)$.

Easy to "compute" in state space.

The Impulse Response

Problem:

Find the response of an LTI system in state space to an impulsive input.

Solution:

If the LTI system is causal it can be represented in state space. We are looking for a solution to

$$\dot{x}(t) = Ax(t) + B\delta(t), \qquad y(t) = Cx(t)$$

Since the system is causal, and assuming zero initial conditions, we have that $x(t) = 0$ for all $t < 0$. In particular $x(0^-) = 0$. Integrating from 0^- to 0^+

$$x(0^+) - x(0-) = \int_{0^-}^{0^+} \dot{x}(t)\, dt = A \int_{0^-}^{0^+} x(t)\, dt + B$$

The Impulse Response

Solution (continued):

Fact #1: $x(t)$ does not contain impulses! Therefore

$$\int_{0-}^{0^+} x(t)\,dt = 0 \quad \Longrightarrow \quad x(0^+) = B$$

From this point on the system has zero input and its response is the homogenous solution:

$$x(t) = e^{At}Ku(t), \qquad x(0^+) = B \;\Longrightarrow\; e^{A0^+}K = K = B$$

Putting it all together:

$$y(t) = Cx(t) = h(t), \qquad\qquad h(t) := Ce^{At}Bu(t)$$

where $h(t)$ is the impulse response.

The Convolution Formula

Question: Why should I care about the Impulse Response?

Answer #1: Because for LTI systems, knowledge of the impulse response lets you compute solutions to **ANY** input!

The convolution formula*:

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

The formula assumes *zero initial conditions*

The formula is easy to prove using system properties

*ATTENTION: *x(t)* is the input, not the state!

The Convolution Formula

Proof using system properties:

Let the impulsive input $g(t)$ produce the response $y_g(t)$:

$$g(t) = \delta(t) \quad \Longrightarrow \quad y_g(t) = h(t)$$

If the system is linear and time-invariant:

$$g(t-\tau) = x(\tau)\delta(t-\tau) \quad \Longrightarrow \quad y_g(t-\tau) = x(\tau)h(t-\tau)$$

Recall the sampling property of the impulse:

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

Using linearity, the integral (summation) of the input produces:

$$\int_{-\infty}^{\infty} g(t-\tau)\, d\tau = x(t) \quad \Longrightarrow \quad \begin{aligned} y(t) &= \int_{-\infty}^{\infty} y_g(t-\tau)\, d\tau \\ &= \int_{-\infty}^{\infty} x(\tau)h(t-\tau)\, d\tau \end{aligned}$$

The Convolution Formula

Question: Why should I care about the Impulse Response?

Answer #1: Because for LTI systems, knowledge of the impulse response lets you compute solutions to **ANY** input!

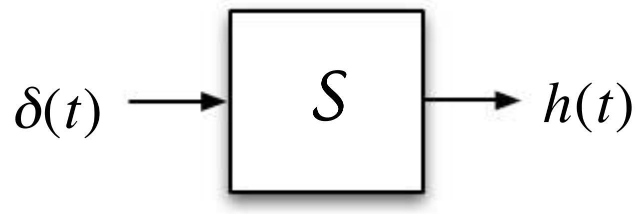
Answer #2: Because for LTI systems, knowledge of the impulse response **equals knowledge of the system**!

System identification:

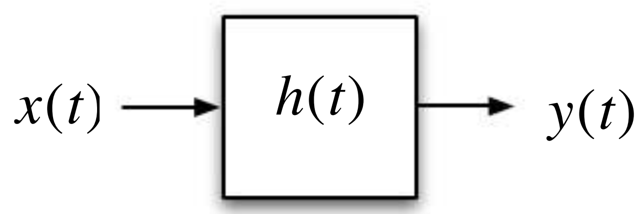
When no mathematical model is available to describe a system, then we can measure one signal (the impulse response) and use this as a model!

System Identification

Perform the experiment and record the impulse response:



If the system is LTI, **the Impulse Response is all we need to know** to obtain the response of the system to any input:



The Convolution Formula

Two equivalent formulas: $y(t) = \int_{-\infty}^{+\infty} h(\tau)x(t-\tau)d\tau$ or $y(t) = \int_{-\infty}^{+\infty} h(t-\tau)x(\tau)d\tau$

Observe that:

- t is treated as a constant in the integration
- τ is the integration variable
- The limits of integration $+\infty,-\infty$ may be simplified to finite values depending on the signals $h(t)$ $x(t)$
- Both integrals give the same values, so they are equivalent

Example: suppose $h(t) = ramp(t)$ $x(t) = 7u(t)$ then:

$$y(t) = \int_{-\infty}^{+\infty} h(\tau)x(t-\tau)d\tau = \int_{-\infty}^{+\infty} ramp(\tau)7u(t-\tau)d\tau$$

$$= \int_{0}^{+\infty} \tau 7u(t-\tau)d\tau = (\int_{0}^{t} \tau 7 d\tau)u(t)$$

$$= 7(\int_{0}^{t} \tau d\tau)u(t) = 7\left(\frac{\tau^2}{2}\right)_0^t u(t) = 7\left(\frac{t^2}{2} - 0\right)u(t) = \frac{7t^2}{2}u(t)$$

Graphical Interpretation of Convolution

Notation: From now on, we will use a * to denote the convolution-formula operation. That is:

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

The Impulse Response tells us through the convolution formula how different the output will be from the input.

Graphical Interpretation of Convolution

Notation: From now on, we will use a * to denote the convolution-formula operation. That is:

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

The Impulse Response tells us through the convolution formula how different the output will be from the input.

You can look at the integral as $h(t)$ being a **weighting function** and convolution as being a **weighted average** of the input over the integration interval.

Graphical Interpretation of Convolution

Notation: From now on, we will use a * to denote the convolution-formula operation. That is:

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

The Impulse Response tells us through the convolution formula how different the output will be from the input.

You can look at the integral as $h(t)$ being a **weighting function** and convolution as being a **weighted average** of the input over the integration interval.

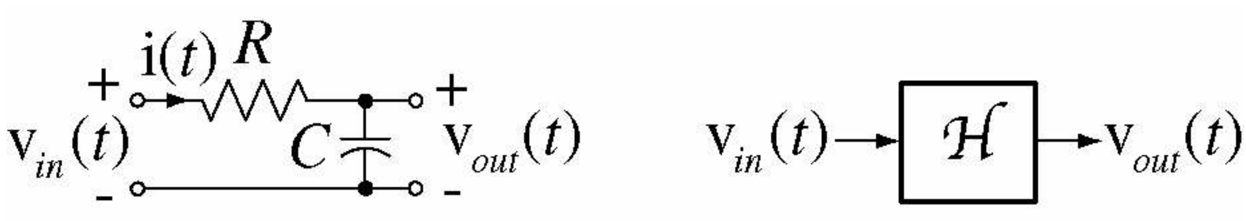
The output value $y(t)$ is then a **compromise of the memories of the input** $x(t)$ **from the past**. In other words, the $h(\tau)$ values $x(t-\tau)$ tell how well the system remembers .

Therefore, the IR is a **measure of the memory** of the system.

Graphical Interpretation of Convolution

Consider a RC low-pass filter

- Assume the capacitor is initially discharged (zero energy).
- Suppose we apply a pulse waveform at the voltage source.
- This leads to charging and discharging of the capacitor.



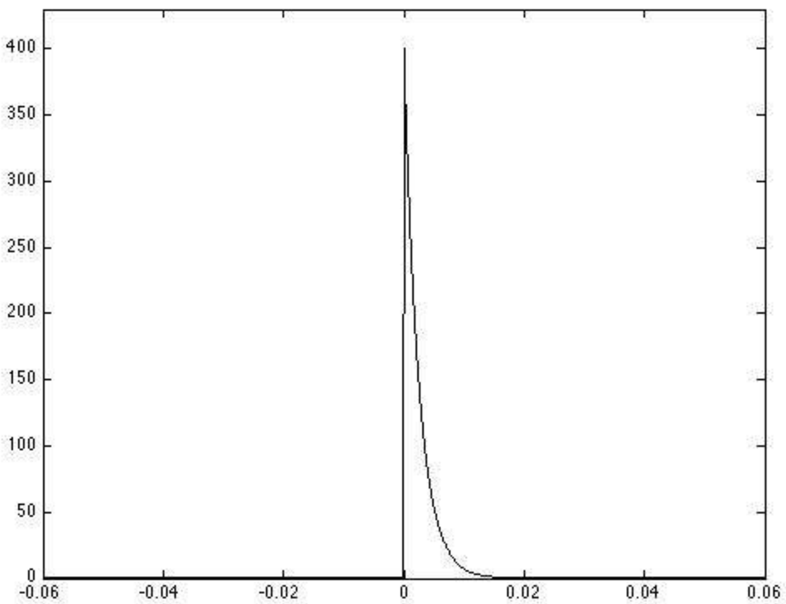
The impulse response of the RC low-pass filter is: $h(t) = \frac{1}{RC} e^{\frac{-1}{RC} t} u(t)$

The time constant of the exponential is RC (a small value) for example, a typical value is $RC = 2.5 \times 10^{-3} s \quad (RC)^{-1} = 400$

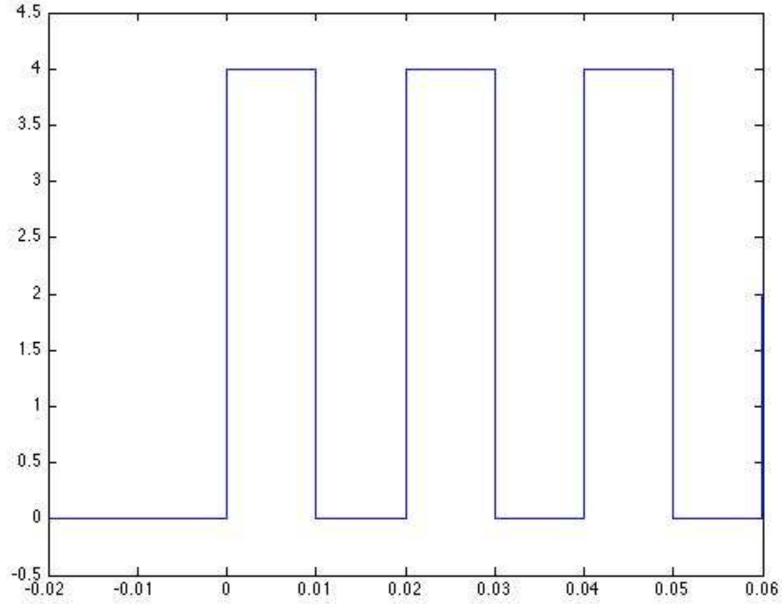
Graphical Interpretation of Convolution

Let us compare the signals $h(\tau)$ and $x(t-\tau)$, and the output value $y(t)$. The impulse response and input signals are:

$h(\tau)$



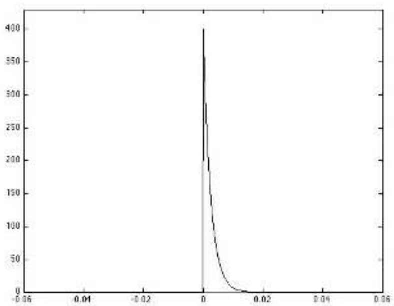
$x(\tau)$

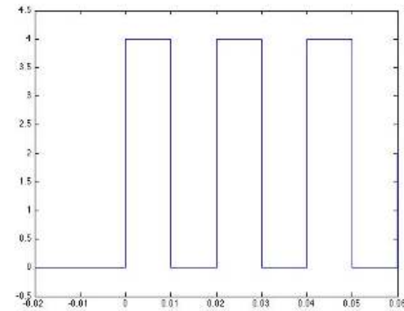


Graphical Interpretation of Convolution

The output signal $y(t)$ becomes:

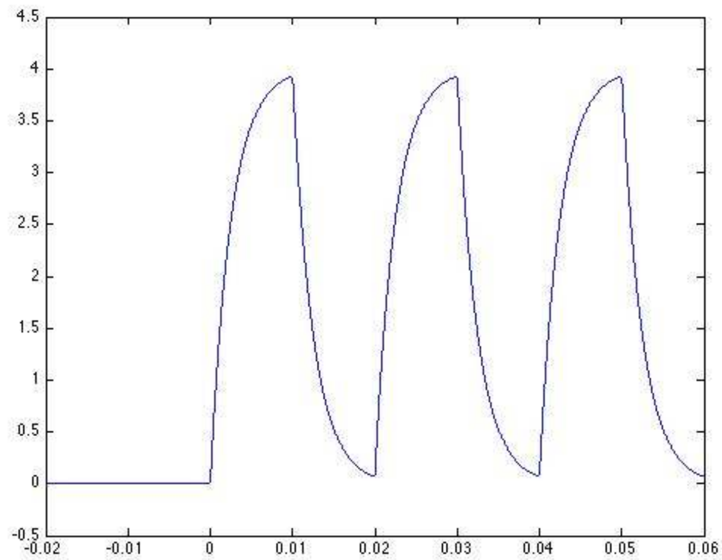
$h(\tau)$





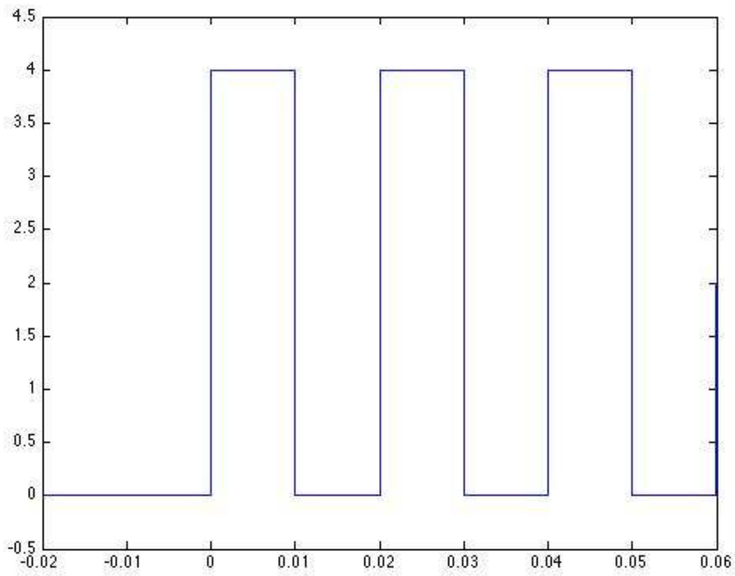
$x(\tau)$

$y(t)$

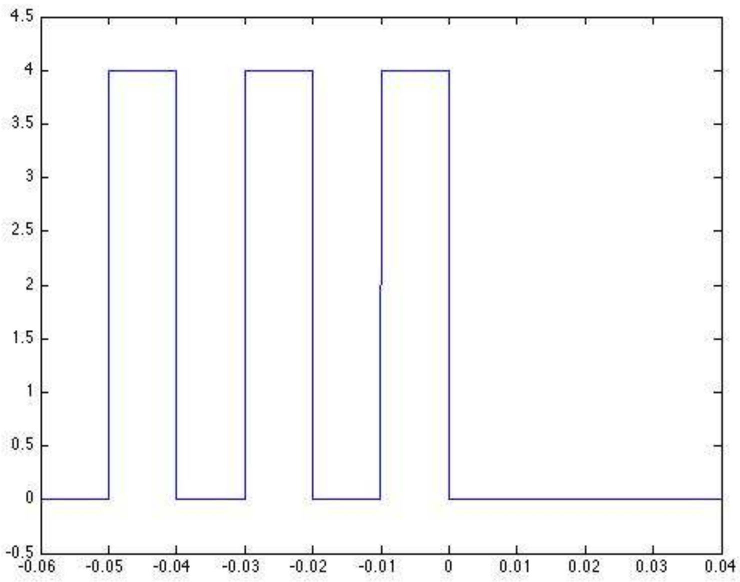


Graphical Interpretation of Convolution

What is $x(t-\tau)$ for different values of t ?



$x(\tau)$

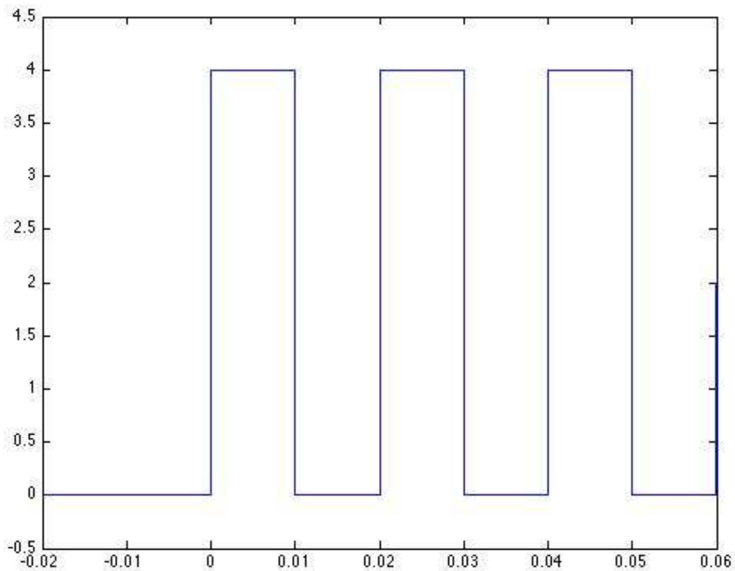


$x(-\tau)$

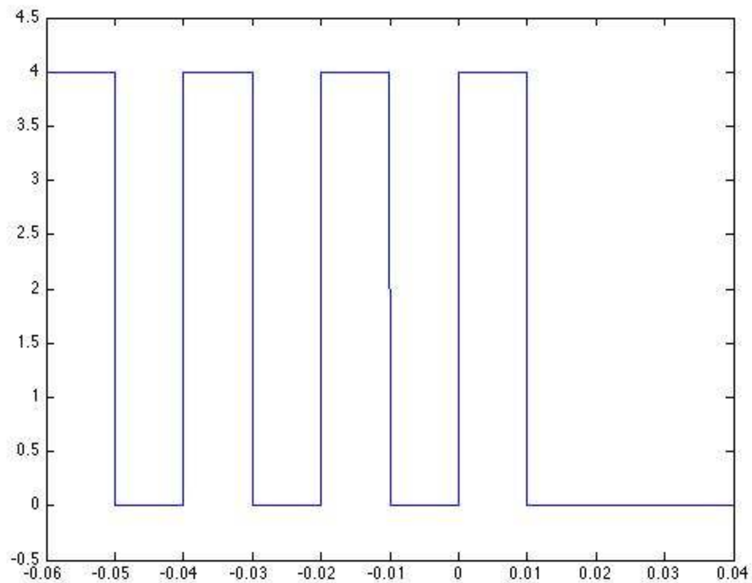
$t=0$

Graphical Interpretation of Convolution

What is $x(t-\tau)$ for different values of t?



$x(\tau)$

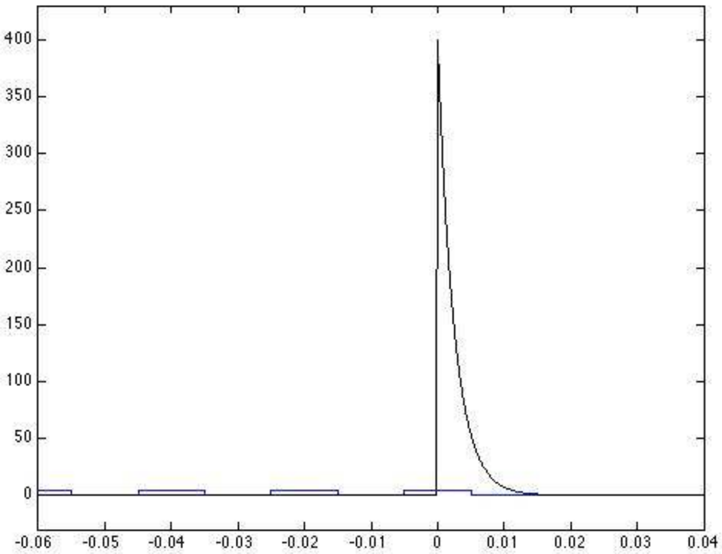


$x(0.01-\tau)$

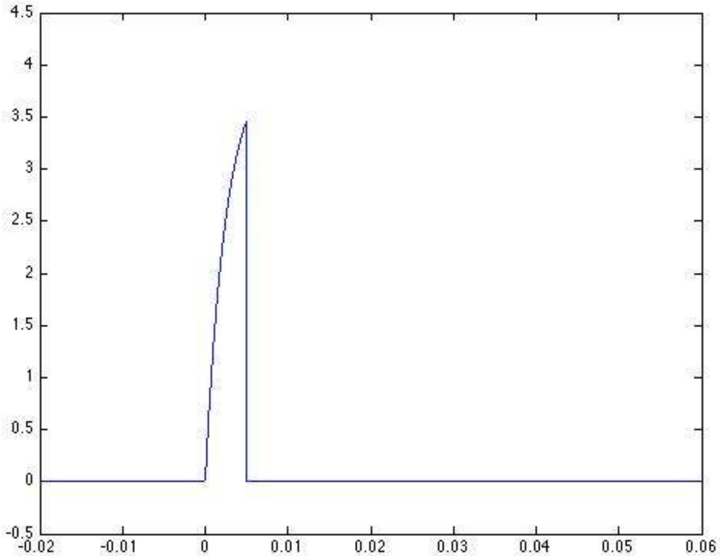
$t = 0.01$

Graphical Interpretation of Convolution

As time t increases from $t=0$ to $t=0.01$, $x(t-\tau)$and $h(\tau)$ start to overlap. The more overlap, the higher the value of the convolution integral and the more charge in the capacitor. At $t=0.01$ the voltage in the capacitor is at its maximum value.



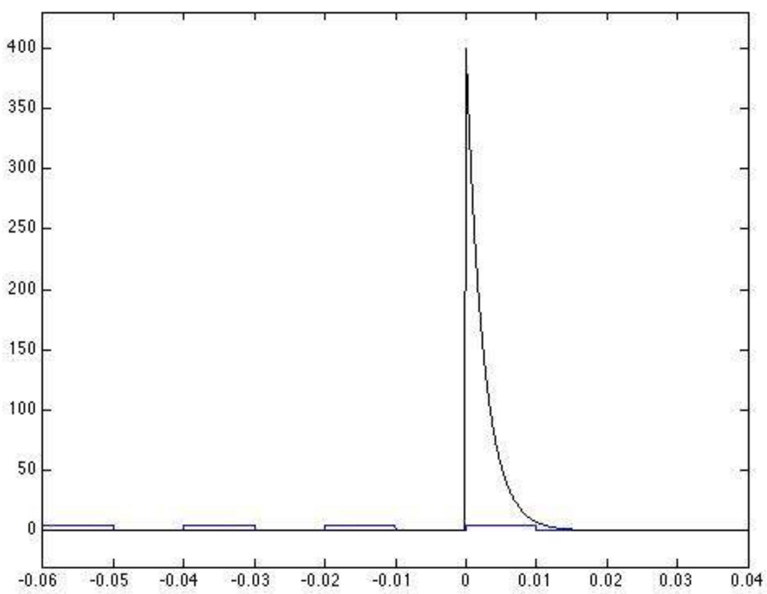
$$x(0.005-\tau), h(\tau)$$



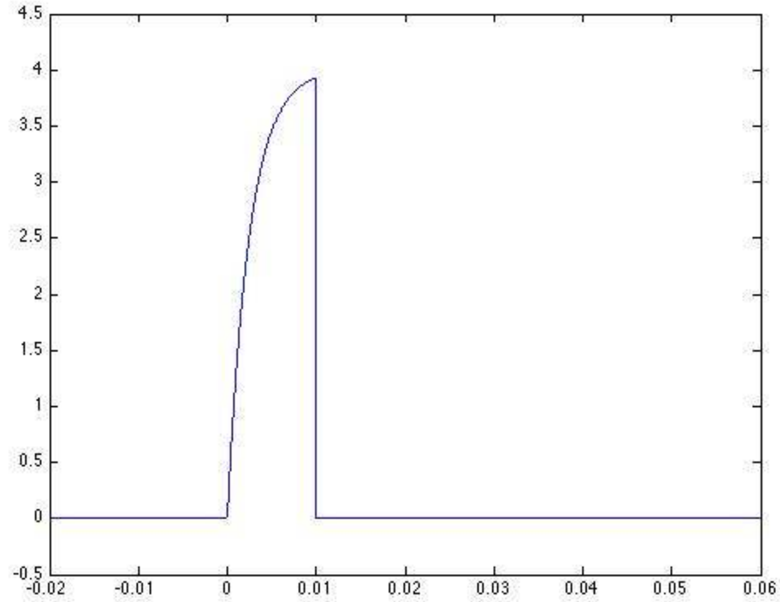
$$y(0.005) \approx 3.5$$

Graphical Interpretation of Convolution

As time t increases from $t=0$ to $t=0.01$, $x(t-\tau)$and $h(\tau)$ start to overlap. The more overlap, the higher the value of the convolution integral and the more charge in the capacitor. At $t=0.01$ the voltage in the capacitor is at its maximum value.



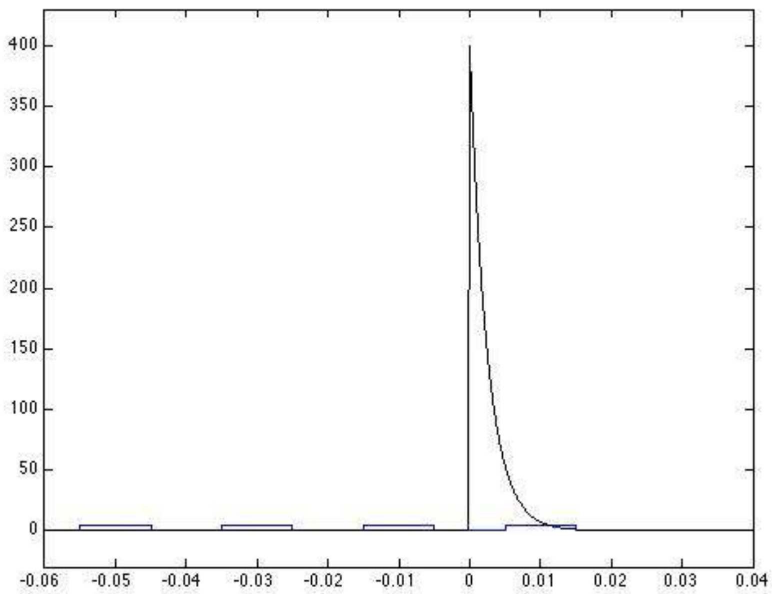
$x(0.01-\tau), h(\tau)$



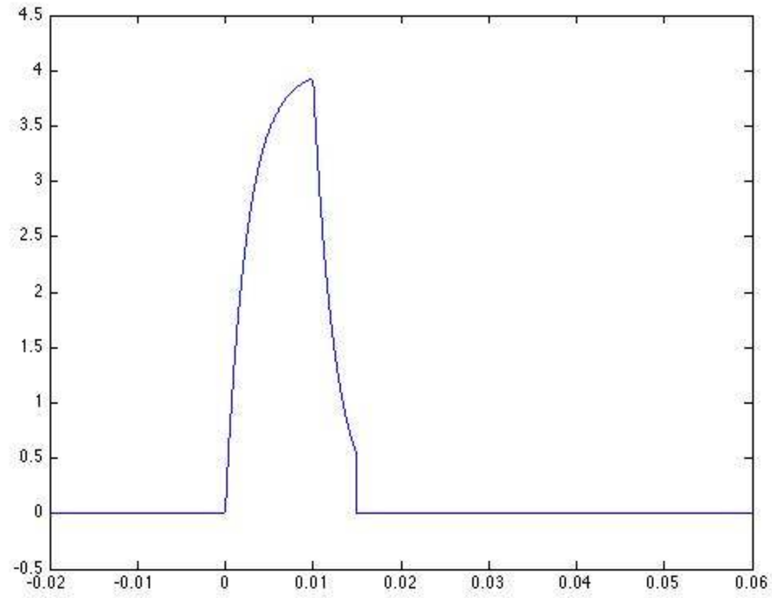
$y(0.01) \approx 4$

Graphical Interpretation of Convolution

As time t increases from $t = 0.01$ to $t = 0.02$, less of the non-zero part of $x(t - \tau)$ overlaps with the non-zero part of $h(\tau)$, and the capacitor starts discharging. At $t = 0.02$ the voltage of the capacitor reaches the minimum value.



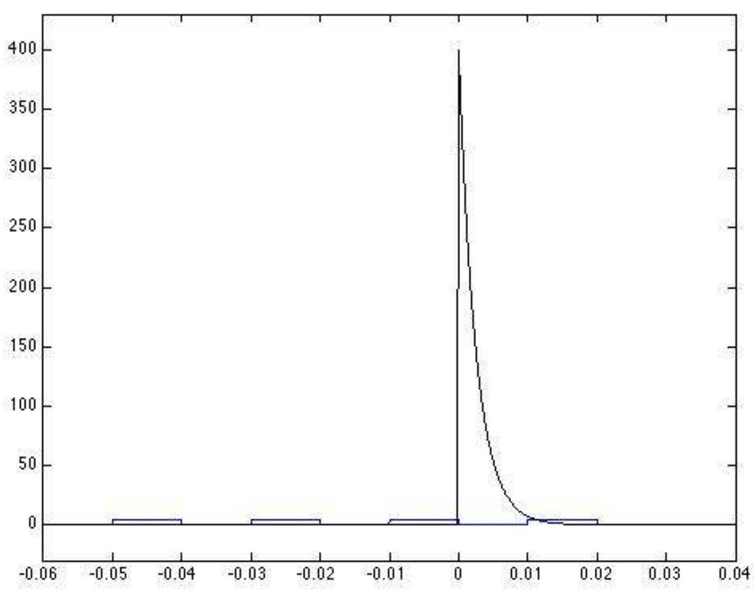
$x(0.015 - \tau), h(\tau)$



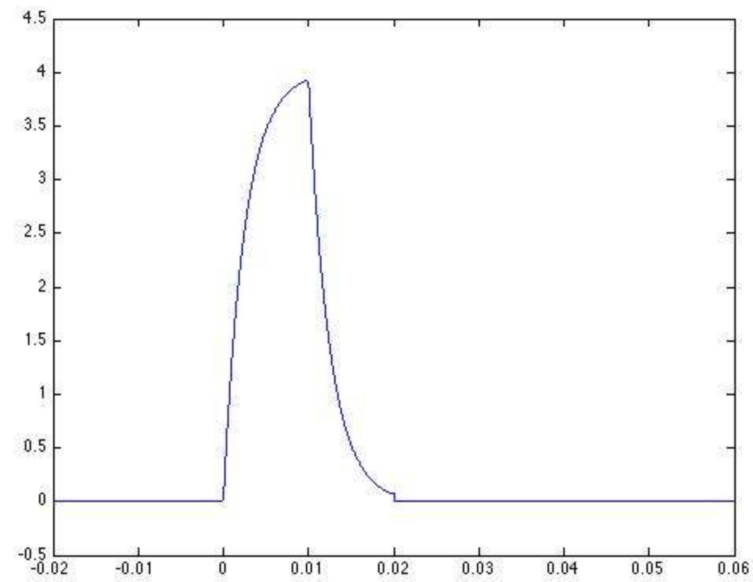
$y(0.01) \approx 0.5$

Graphical Interpretation of Convolution

As time t increases from $t = 0.01$ to $t = 0.02$, less of the non-zero part of $x(t-\tau)$ overlaps with the non-zero part of $h(\tau)$, and the capacitor starts discharging. At $t = 0.02$ the voltage of the capacitor reaches the minimum value.

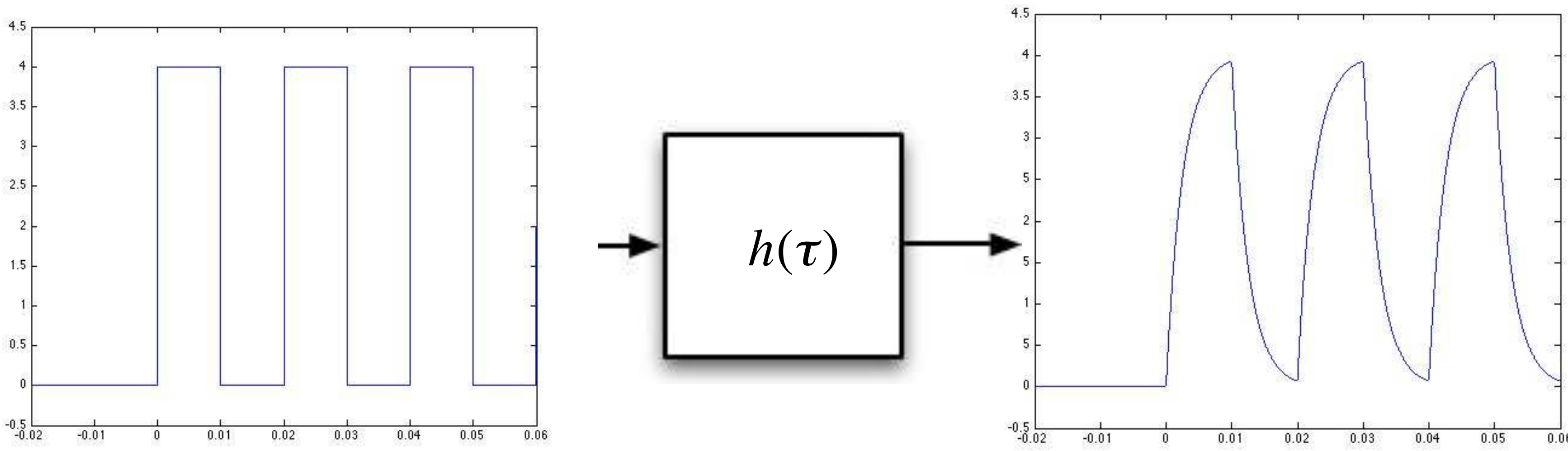


$$x(0.02-\tau), h(\tau)$$



$$y(0.02) \approx 0$$

Graphical Interpretation of Convolution



$x(\tau)$ $y(t)$

$h(\tau)$ tells us how different $y(t)$ will be from $x(\tau)$

In this case the output of the system is

a "rounded" version of the input

Signals & Systems Interaction in the TD

Goals

I. Impulse Response (IR) and Convolution Formula

-Definition of IR and its use for system identification

-Convolution formula and its graphical interpretation

II. Properties of systems from IR and convolution

-Impulse response as a measure of system memory/stability

-Alternative measures of memory/stability: step response

III. Applications of convolution

-Audio effects: reverberation

-Noise removal (i.e. signal filtering or smoothing)

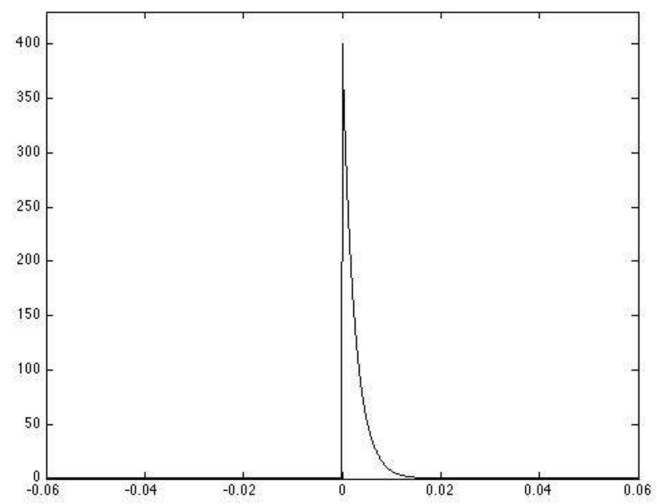
Impulse Response and System Memory

The memory of an LTI system will clearly define the shape of the IR (how fast it decays to zero or not.)

However, from the previous discussion on convolution, we also observe that the the shape of $h(t)$ is what determines how much the system recalls previous input values:

The larger the range of non-negative values of the $h(t)$ for positive t , the "more memory" the system has

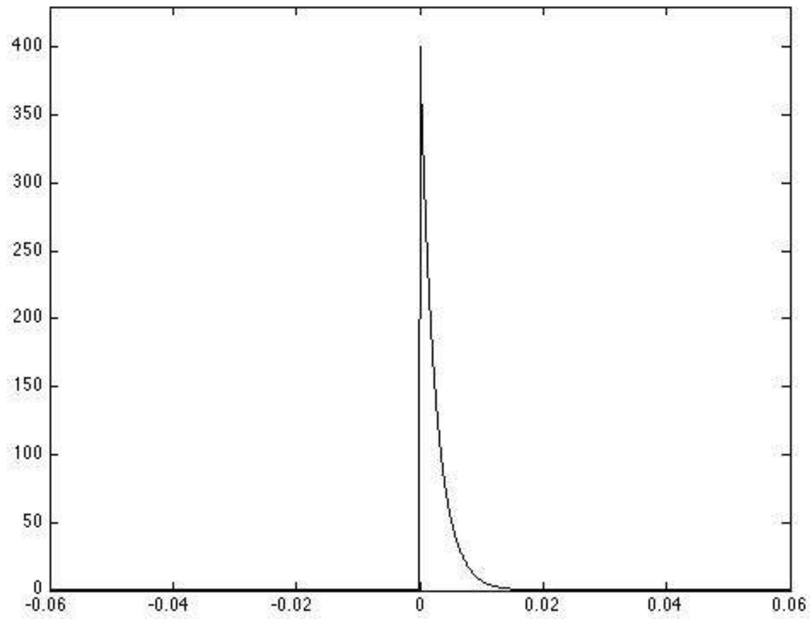
The memory of the RC low-pass filter is small and related to the IR **settling time**



Impulse Response and System Memory

Definition:

The **settling time** of a signal is the time it takes the signal to reach its steady-state value.



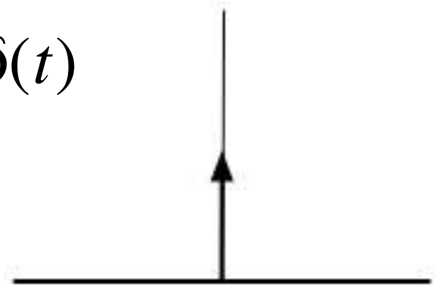
In this case we just compute t_s to be the time when $h(t)$ reaches the 0.01 value:

$$h(t_s) = \frac{e^{-400 t_s}}{400} \approx 0.01 \qquad \boxed{t_s \approx 0.0265}$$

Impulse Response and System Memory

If the IR of a system is a unit impulse signal, then the system has **no memory** of the past and leaves inputs unchanged!

$$h(t) = \delta(t)$$



From the convolution formula, we obtain that

$$y(t_0) = \int_{-\infty}^{+\infty} x(t)h(t - t_0)dt = \int_{-\infty}^{+\infty} x(t)\delta(t - t_0)dt = x(t_0)$$

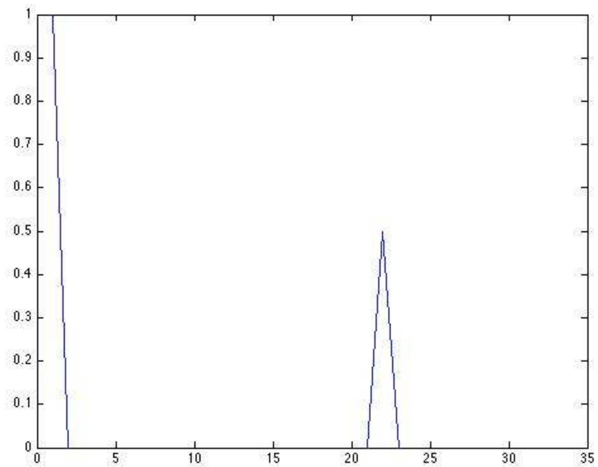
As an IR, the unit impulse just "samples" the input signal $x(t)$ at the value t_0 . This is why we call this property of the unit impulse function the "sampling property" (the associated system has no memory and leaves inputs unchanged.)

Impulse Response and System Memory

The following impulse response has an ‘echo’ effect on signals

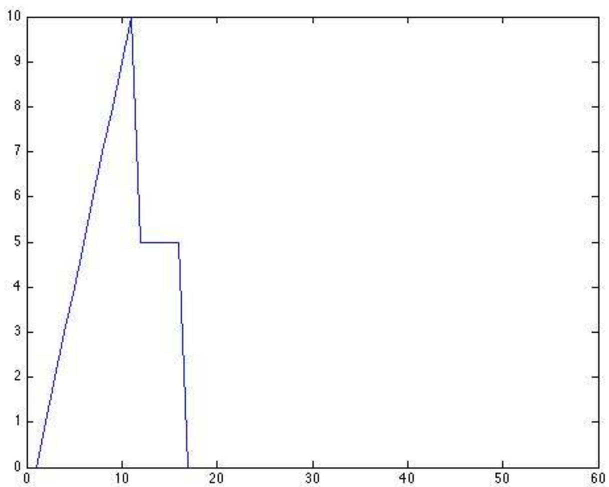
You can interpret the peaks of the IR as approximations of impulse signals of different strength

$h(t)$

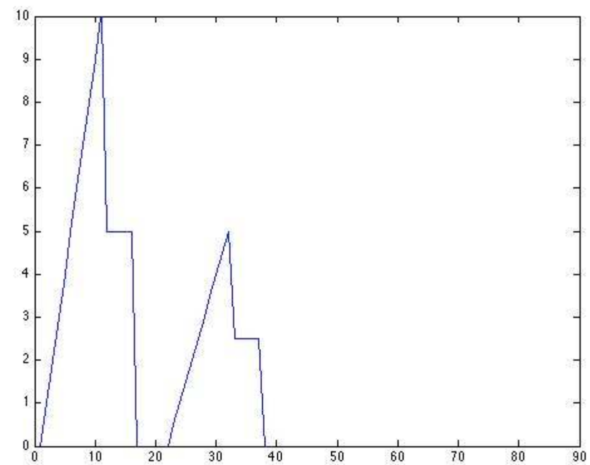


$$t_s \approx 23.5s$$

$x(t)$



$y(t)$



Impulse Response and System Stability

BIBO stability can also be inferred from the shape of the IR

We have that:

If $\int_{-\infty}^{+\infty} |h(\tau)|\, d\tau < \infty$ then BIBO stability holds

Observe that IR that do not satisfy the above formula will induce large system memories and, because of the convolution formula, it will possibly make some outputs unbounded.

NOTE: When $h(t)$ is defined as a sum of complex exponentials with negative real parts, then we can guarantee that $\int_{-\infty}^{+\infty} |h(\tau)|\, d\tau < \infty$

(Unit) step response

Suppose that x(t) produces the response y(t) in an LTI system.

Then the excitation $\frac{d}{dt}\left(\mathrm{x}\left(t\right)\right)$ will produce the response $\frac{d}{dt}\left(\mathrm{y}\left(t\right)\right)$

It turns out that $\delta(t) = u'(t)$, i.e. the generalized derivate of the unit step signal is the unit impulse signal

Then, if $s(t)$ is the system unit-step response, we have

$$\boxed{\int_{-\infty}^{t} h(\tau)d\tau = s(t)} \qquad \boxed{s'(t) = h(t)}$$

This means that knowing the (unit) step response is as informative as knowing the unit impulse response.

(Unit) step response

The IR and a step response are used in practice to capture the transient responses of the system (it tells you how the system reacts to disturbances and, qualitatively, about the system stability).

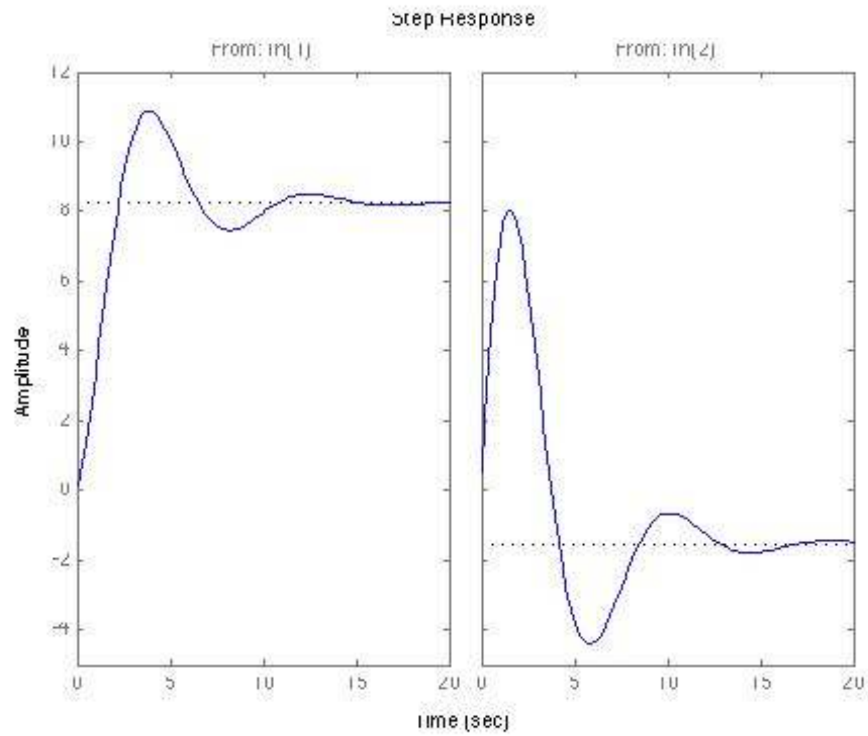
Constants of interest:

Rise time: time it takes the signal to reach the vicinity of new set point

Settling time

Overshoot: maximum amount the system overshoots its final value divided by it

Peak time: time to reach the overshoot value

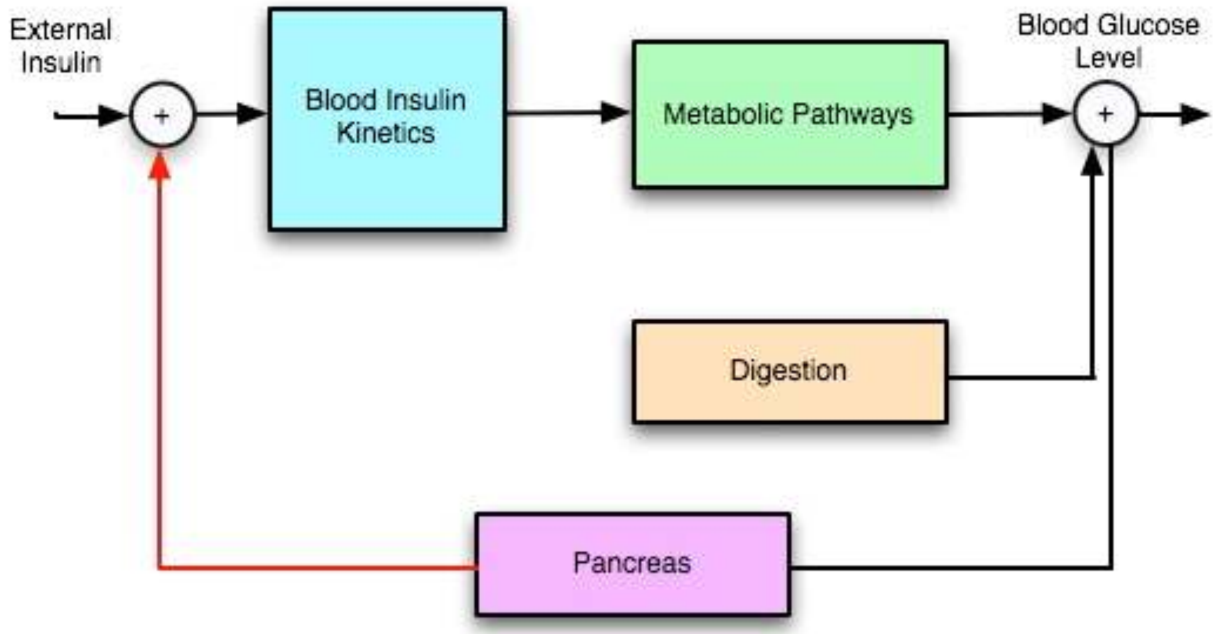


$t_r \approx 7.7s \quad t_s \approx 15s \quad t_p \approx 3.5s$

$Ov \approx 11.5/8.1$

Example: system behavior from step response

Glucose System diagram



A patient with no capability of secreting insulin requires external insulin shots. Suppose we want to **design a programmable pump** to provide regular shots to a patient.

Then, we would need to answer the following questions…

Example: system behavior from step response

1. Is the blood glucose level normal?

   Apply tests with insulin pump and blood analysis system

2. What steady state value of insulin should we deliver to the patient in order to bring his/her glucose level to normal?

   Measure the insulin step response

3. We would like to apply the patient small insulin shots to maintain his/her glucose level. How much does the effect of a shot last?

   Check whether the system is LTI through experiment,
   the system memory gives us an estimate of a shot duration
   we can simulate shot sequences using convolution

Example: system behavior from step response

4. Suppose the patient eats a meal. How should the insulin dose change?

We would like to find the input that cancels out the effect of the meal, in other words…

If $x_{meal}(t)$ produces the output glucose level $y_{meal}(t)$ and $x_{shot,fastng}(t)$ is the input we need to produce $y_{normal}(t)$ (under fasting conditions)

Then we need to find $x(t)$ that produces $h(t)*x(t) = -y_{meal}(t)$. Then:

$$h(t)*(x_{shot,fastn}(t) + x_{meal}(t) + x(t)) =$$
$$y_{normal}(t) + y_{meal}(t) - y_{meal}(t) = y_{normal}(t)$$

To solve for $x(t)$ in $h(t)*x(t) = -y_{meal}(t)$ we need to apply a "deconvolution" (easier in the Frequency Domain…)

Signals & Systems Interaction in the TD

Goals

I. Impulse Response (IR) and Convolution Formula

- Definition of IR and its use for system identification
- Convolution formula and its graphical interpretation

II. Properties of systems from IR and convolution

- Impulse response as a measure of system memory/stability
- Alternative measures of memory/stability: step response

III. <u>Applications of convolution</u>

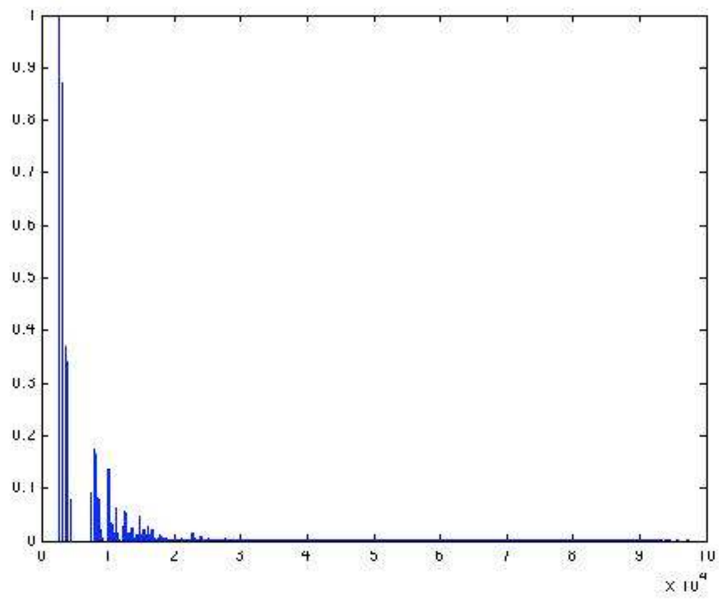
- Audio effects: reverberation
- Noise removal (i.e. signal filtering or smoothing)

Reverberation effects

Reverberation (or reverb) effects are probably one of the most heavily-used effects in music recording.

Reverberation is the result of many reflections of sound in a room. Reflected waves reach to the listener later than waves that reach him/her directly. This produces an “echo” effect.

One way of implementing reverb effects is to **convolve audio signals with impulse responses** like the following:



Reverberation effects

The process of obtaining a room Impulse Response is quite straightforward. The following options are available:

1) Record a short impulse (hand clap, drum hit) in the room
2) Room IR can be simulated in software also (e.g. MATLAB)
3) There is also commercially available software (e.g. Altiverb) that implements the reverb effects for different rooms

The IR records the room characteristics as follows:

Reverberation Effects

There are m files in MATLAB to approximate impulse responses in rooms. For example the function rir.m freely available in the Internet ("room impulse response", you can take it from WebCT.)

```
[h]=rir(fs, mic, n, r, rm, src);
%RIR   Room Impulse Response.
%   [h] = RIR(FS, MIC, N, R, RM, SRC) performs a room impulse
%         response calculation by means of the mirror image method.
%
%      FS  = sample rate.
%      MIC = row vector giving the x,y,z coordinates of
%            the microphone.
%      N   = The program will account for (2*N+1)^3 virtual sources
%      R   = reflection coefficient for the walls, in general -1<R<1.
%      RM  = row vector giving the dimensions of the room.
%      SRC = row vector giving the x,y,z coordinates of
%            the sound source.
```

Reverberation Effects

Once the IR has been generated, you need to use an approximation of convolution as follows:

$$y[n] = \sum_{m=-\infty}^{+\infty} x[m]h[n-m]$$

Here, $x[n]$ and $h[n]$ discrete-time versions of the input audio signals and the unit impulse signal sampled at the right rate (more on this when we go over the sampling of analog signals.)

Suppose that $x(t)$, $h(t)$ are stored into .wav files. Then, you need to use a function in MATLAB like ‘wavread’:

```
[x,Fs]=wavread(‘acoustic.wav’)
% x = vector corresponding to x[n]
% Fs = sampling rate of the signal acoustic.wav
```

Reverberation Effects

There are fast convolution functions in MATLAB, such as fconv, which do the previous convolution sum in a fast way:

```
function [y]=fconv(x, h)
%FCONV Fast Convolution
%   [y] = FCONV(x, h) convolves x and h.  The output of this
%         function is scaled.
%
%      x = input vector
%      h = input vector
%
%      See also CONV
```

In fact, the convolution makes use of Fast Fourier Transform methods (…) (more info about this in the book)

Reverberation Effects

An example of how we call this function given an input signal and an Impulse Response is the following (in WebCT):

```
% reverb_convolution_eg.m
% Script to call implement Convolution Reverb

close all;
clear all;

% read the sample waveform
filename='.acoustic.wav';
[x,Fs,bits] = wavread(filename);

% read the impulse response waveform or take it from rir.mand
    substitute imp by the output of rir.m

filename='impulse_room.wav';

[imp,Fsimp,bitsimp] = wavread(filename);

% Do convolution with FFT
y = fconv(x,imp);

% write output
wavwrite(y,Fs,bits,'out_IRreverb.wav');
```

Reverberation Effects

Finally, to play a wav file in MATLAB just use

```
[x,Fs]=wavread('acoustic.wav');
sound(x,Fs)
```

In WebCT there are examples of room impulse responses and audio files. Try them with your favorite wav files!

Just do the following:

-Run ‘reverb_convolution_eg.m’ in MATLAB with the input.wav file and IR.wav file you like

-play out_IRreverb.wav using ‘wavread’ and ‘sound’ as above

More on sound processing and convolution

Connecting systems in series/parallel is useful for system design and removing unwanted effects from previously applied systems to signals ("echo control", more of this in MAE 143C)

Sound processing:

-Real-time room acoustic de-reverberation

-Introduction of reverberation effects in music files

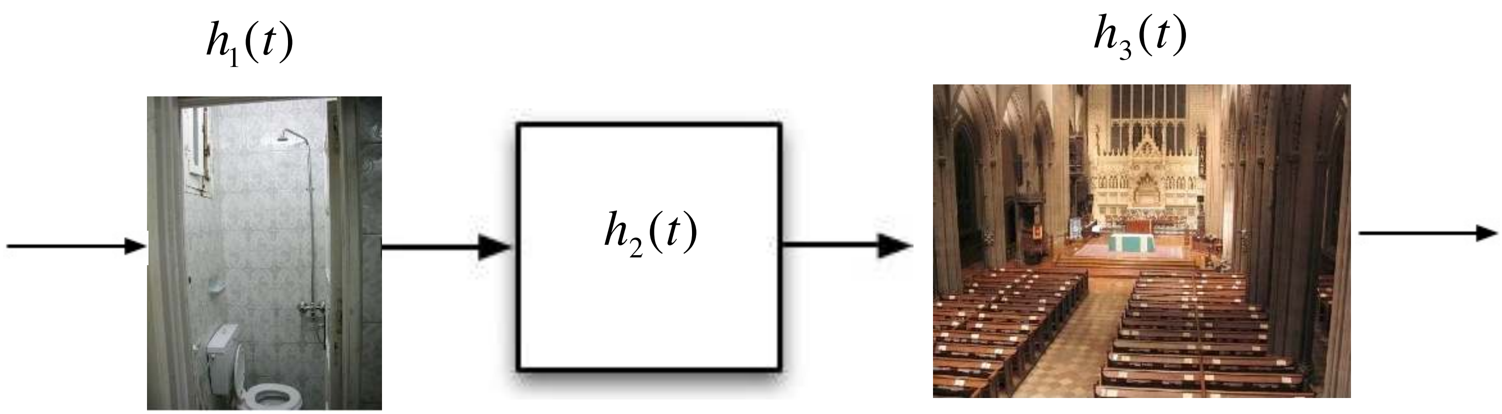
-Superposition of sounds

More on sound processing and convolution

Measure the room impulse response $h_1(t)$

Find another impulse response function such that $h_1(t) * h_2(t) = \delta(t)$ (This is possible for Invertible Systems. The new impulse response has to be found by "deconvolution". This is done in the Frequency Domain)

Consider your favorite impulse response $h_3(t)$ to filter the speech



Similar ideas used in image processing; for example this is done in camera auto-focusing sub-routines

However, for image/sound processing, we need to deal with discrete signals and discrete systems counterparts (more on this later.)

Noise removal and signal smoothing

Convolution is commonly used to implement linear operations on audio signals and images such as filtering (noise removal)

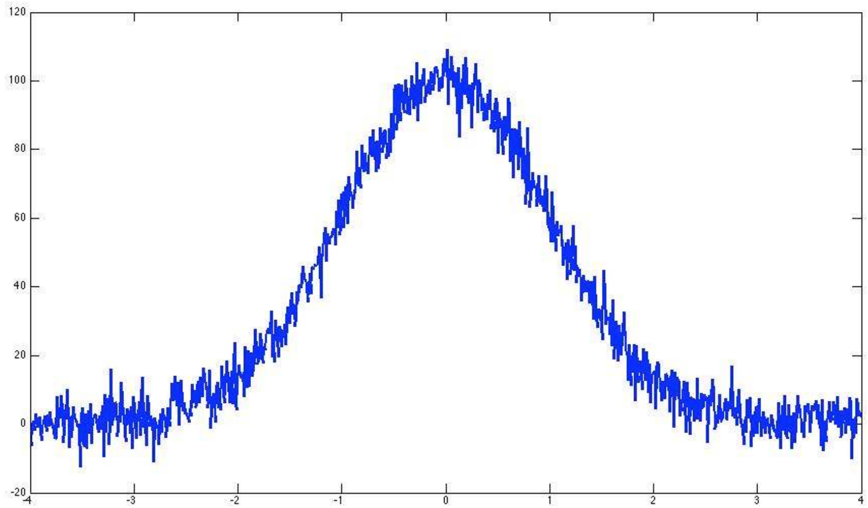
<u>Moving average filter</u>

Take $f(t)$ a noisy signal

Here $f(t) = s(t) + w(t)$

where $s(t) = e^{-t^2/2}$

and $w(t)$ is a white noise



$f(t)$

Gaussian white noise: for every t, $w(t)$ is Gaussian distributed (that is, the mean value of $w(t)$ is zero and $w(t)$ takes a real value close to zero with a standard deviation of σ)

Noise removal and signal smoothing

Take $h(t) = rect(t)$. The moving average filter is defined as the convolution $g(t) = h(t) * f(t)$. The output $g(t)$ is a new signal with the same shape as $s(t)$ (in general we can only say $s(t) \approx g(t)$)

Why does this work?

Recall that

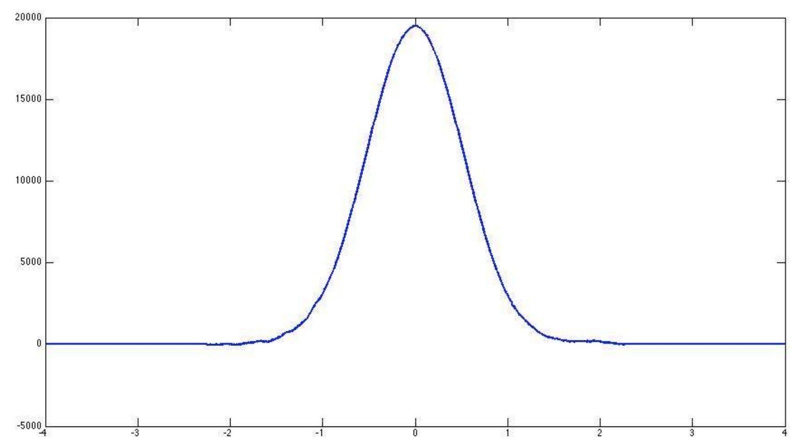
convolution = weighted average

Then have that

$$g(t) \approx \sum h(\tau)(s(t-\tau) + w(t-\tau))$$
$$\approx \frac{1}{m}\sum_{i=1}^{m}(s(t-\tau_i) + w(t-\tau_i))$$
$$\approx \frac{1}{m}\sum s(t-\tau_i) \approx s(t)$$

because the mean value of $w(t)$ is zero

$g(t)$



Noise removal and signal smoothing

Other convolutions that also remove noise are:

$$g_1(t) = h(t) * f(t) \qquad h(t) = tri(t) \qquad \text{(unit triangle signal)}$$

$$g_2(t) = h(t) * f(t) \qquad h(t) = \operatorname{sinc}(t) = \frac{\sin(\pi t)}{\pi t} \qquad \text{(sinc function)}$$

The unit triangle function makes outputs even more smooth because it turns out that $tri(t) = rect(t) * rect(t)$

The reason why the sinc function works will be explained when we study signals and systems interaction in the Frequency Domain (more on this later…this corresponds to the ideal low-pass filter)

The same ideas can be applied for image signals. For example, special convolution operations are also used for edge sharpening in images.

Summary

Important points to remember:

1. The **impulse response (IR)** of a system is the particular output that the system produces when excited with the unit impulse signal. In this way, the IR of a system can be obtained experimentally.

2. The IR of an LTI system **can be used to obtain the response of the system to an arbitrary excitation** via an operation called **convolution**. This turns out to be very useful if we don't know an ODE model of the LTI system.

3. The IR of an LTI system can be seen as a **measure of memory** of the system. It can also tell us whether the system is BIBO stable or not.

4. **Convolution** can be understood as a **weighted sum** of input values.

5. The **step response** of a system **is the (generalized) derivative** of the Impulse Response. Thus, it is as informative as the IR.

6. **Convolution has applications in** the prediction of general (nonlinear) systems behavior (system identification), and in the treatment of audio/image signals (e.g. reverberation effects and noise removal.)