# System Identification

Control Engineering EN, 3rd year B.Sc.
Technical University of Cluj-Napoca
Romania

Lecturer: Lucian Buşoniu

# Part III

## Transient Analysis of Step and Impulse Responses

# Motivation

In general:

Sometimes a simple first or second-order model is sufficient; transient analysis offers an easy way to obtain it.

For students:

Closest relation to prior knowledge from system theory $\Rightarrow$ gentle transition towards other techniques.

# Classification

Recall **Taxonomy of mathematical models** from Part I:

By number of parameters:

1. Parametric models: have a fixed form (mathematical formula), with a known, often small number of parameters
2. Nonparametric models: cannot be described by a fixed, small number of parameters
   Often represented as graphs or tables

By amount of prior knowledge ("color"):

1. First-principles, white-box models: fully known in advance
2. Black-box models: entirely unknown
3. Gray-box models: partially known

# Classification (continued)

Step and impulse response models can be seen as nonparametric models, for those steps in which we study the graph of the response.

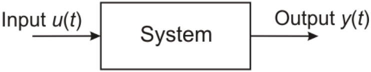
However, based on information from the graph, we will in the end find a transfer function – a parametric model.

These models are best classified as gray-box.

The study of these models is called *transient analysis*, since it relies in a large part on the transient regime of the response.

# Table of contents

# A definition of linear systems

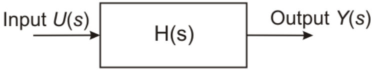



A system is *linear* if it satisfies:

Superposition: If for input $u_1(t)$ the system responds with output $y_1(t)$; and for $u_2(t)$ the system responds with $y_2(t)$; then for input $u_1(t) + u_2(t)$ the system will respond with $y_1(t) + y_2(t)$.

Homogeneity: If for input $u(t)$ the system responds with output $y(t)$; then for input $\alpha u(t)$ the system will respond with $\alpha y(t)$.

# Transfer function representation




The transfer function is:

$$H(s) = \frac{Y(s)}{U(s)} = \frac{b_m s^m + b_{m-1} s^{m-1} + \ldots + b_1 s + b_0}{a_n s^n + a_{n-1} s^{n-1} + \ldots + a_1 s + a_0}, \quad m \leq n$$

where $U(s)$ and $Y(s)$ are, respectively, the Laplace transforms of the input and output signals $u(t)$ and $y(t)$.
(Important: in zero initial conditions.)

The *Laplace transform* of a signal $f(t)$ is:

$$F(s) = \mathcal{L}[f(t)] = \int_0^\infty f(t) e^{-st} \mathrm{d}t$$

# Laplace transform interpretation

- $s$ is called *complex* argument (it is a complex number), and the Laplace transform can be seen as taking a function from the time domain $t$ to the complex domain $s$.
- The motivation is that many signal operations common in engineering (differentiation, integration, etc.) become much simpler in the $s$ domain.

# Table of contents

# First order system: Motivating example

First-order systems are common. Typical example: a thermal system. Consider an object at temperature $\theta_1$ (output variable) placed in an environment at temperature $\theta_2$ (input variable). Then:

$$C\dot{\theta}_1(t) = \frac{\theta_2(t) - \theta_1(t)}{R}$$

where $C$ is the thermal capacitance and $R$ is the thermal resistance.

Applying the Laplace transform on both sides:

$$Cs\Theta_1(s) = \frac{\Theta_2(s) - \Theta_1(s)}{R}$$

leading to the transfer function:

$$H(s) = \frac{\Theta_1(s)}{\Theta_2(s)} = \frac{1}{CRs + 1}$$

# First order system: General form

$$H(s) = \frac{K}{Ts + 1}$$

where:

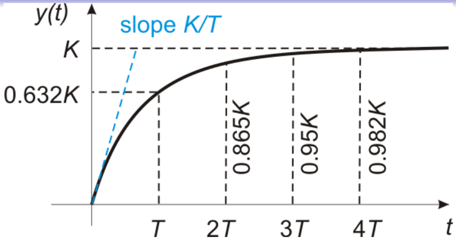
- $K$ is the gain ($= 1$ in the example)
- $T$ is the time constant ($= CR$ in the example)

# Table of contents

# Ideal step input


$$u_S(t) = \begin{cases} 0 & t \leq 0 \\ 1 & t > 0 \end{cases}$$

# Ideal 1st order response




Solving the differential equation for $y(t)$ (or easier: solve for $Y(s)$ and then apply inverse Laplace transform $\mathcal{L}^{-1}$), we get:

$$y(t) = K(1 - e^{-t/T})$$

from where:

$$\lim_{t\to\infty} y(t) = K(1-0) = K$$

$$\dot{y}(t) = \frac{K}{T}e^{-t/T}, \quad \dot{y}(0) = \frac{K}{T}e^{0} = \frac{K}{T}$$

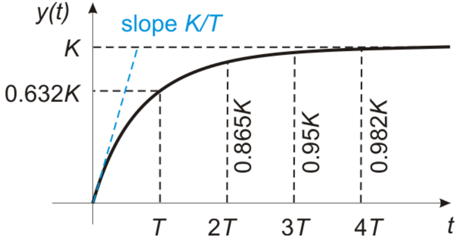
$$y(T) = K(1 - e^{-1}) \approx 0.632K$$

and similar for $t = 2T, 3T, 4T$ (see figure).

# Determining the parameters

So far, everything known from: Sys. Theory, Process Modeling.

Now, consider we are given a step response of an unknown system. We can use it to find an approximate transfer function of the system.



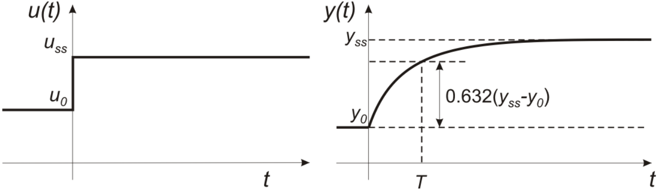

## Algorithm for **system identification**

1. Read the steady-state value. That is the gain $K$.
2. Determine the time value where the output reaches 0.632 of its steady state value. That is the time constant $T$.

# Nonzero initial conditions

In practice, we often cannot use ideal step signals: the system must be kept around a safe/profitable operating point. In particular, assume the system was in steady-state at $y_0$ with input held constant at $u_0$.

Real-life step inputs are often rectangular signals as below. The system response is therefore nonstandard.

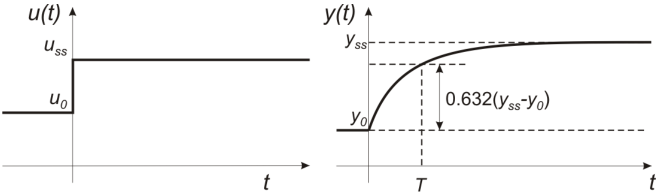



But recall linearity properties. Our new input is $u(t) = u_0 + (u_{ss} - u_0)u_S(t)$ with $u_S(t)$ the ideal step input. Then, denoting the ideal step response by $y_S(t)$, we have the new output:

$$y(t) = y_0 + (u_{ss} - u_0)y_S(t)$$

simply a shifted and scaled version.

# Nonzero initial conditions (continued)

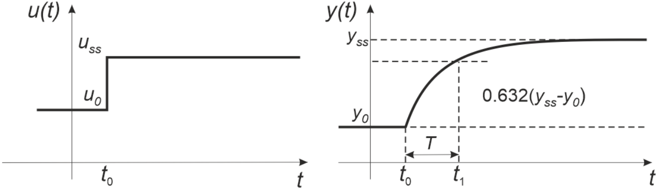



Then we obtain:

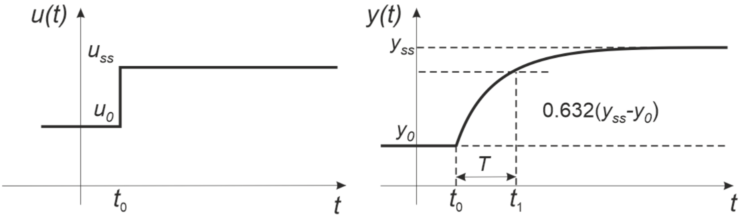
$$y_{ss} = y_0 + (u_{ss} - u_0)K$$
$$y(T) = y_0 + 0.632(y_{ss} - y_0)$$

# Nonzero step time

The start time of the step may also be different from 0, solved easily by shifting the time axis. Such a situation can occur for any of the step and impulse responses throughout the remainder of this part, and it is always handled the same way. We will provide details for the step responses, and the same idea can be applied for impulse responses.



# General algorithm

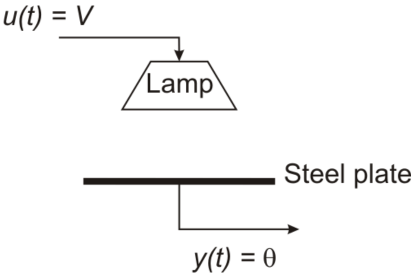



## General algorithm

1. Read $u_0$, $y_0$, $u_{ss}$, $y_{ss}$ the initial and steady-state values of the input and output signals. Compute $\boxed{K = \frac{y_{ss} - y_0}{u_{ss} - u_0}}$.
2. Read $t_0$ the start time of the step, $t_1$ the time where the output raises 0.632 of the difference. Compute $\boxed{T = t_1 - t_0}$.

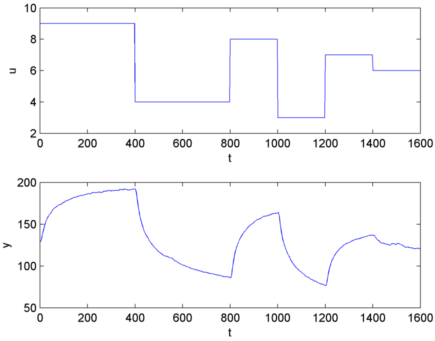
# Table of contents

# Example: Thermal system




Consider the thermal system in the figure (different from the example above). The input is the voltage $V$ applied to the lamp, the output is the temperature $\theta$ read by a thermocouple at the back of the steel plate.

# Thermal system: Experimental data




The data is obtained from the Daisy database. The signals are sampled in discrete time with sampling period $T_s = 2\,\text{s}$, but for transient analysis, we will treat them as continuous-time.

Note: the presence of noise in the data! This is virtually always true in identification experiments.

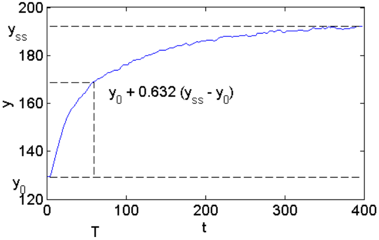
We use the first step for identification, and the others for validation.

# Sampling period

Definition: The sampling period $T_s$ is the continuous time-interval between two successive discrete-time sampling points (of the input, output, or other signals in the system).

Don't confuse sampling period $T_s$ with the time constant $T$ multiplied by complex argument $s$!

# Thermal system: Model and parameters




We have $y_{ss} \approx 192^\circ$ C, $y_0 \approx 129^\circ$ C. Also, the input $u_{ss} = 9$ V and (from the experiment) we know that $u_0 = 6$ V. Therefore:

$$K = \frac{y_{ss} - y_0}{u_{ss} - u_0} \approx \frac{192 - 129}{9 - 6} \approx 21$$

Further, $y(T) = y_0 + 0.632(y_{ss} - y_0) \approx 169$, and identifying this point on the graph we get $T \approx 60$.

# Thermal system: Transfer function model

$$\widehat{K} = 21$$

$$\widehat{T} = 60$$

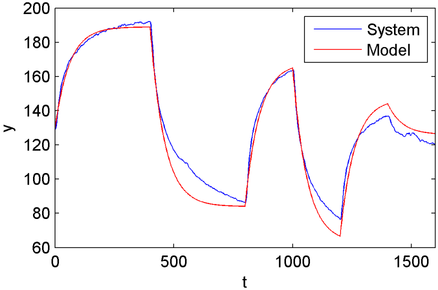
$$\widehat{H}(s) = \frac{\widehat{K}}{\widehat{T}s + 1} = \frac{21}{60s + 1}$$

The "hat" notation makes explicit the fact that the model is an approximation.

Matlab: `H = tf(num, den)`, with polynomials represented as vectors of coefficients in decreasing powers of $s$.

(Note: Actual calculations done in double representation with Matlab, so using the numbers given in the slides will lead to slightly different results. This remark applies to all the examples.)

# Thermal system: Validation of transfer function model

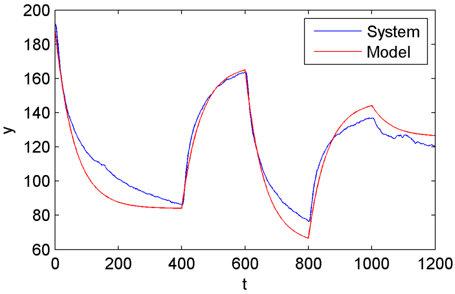



Note special steps needed to take into account the nonzero initial condition of the system; we will learn about them under impulse response analysis.

The fit is not great – the cooling dynamics are quite slower than the heating dynamics, for example, so in reality this is not a simple first-order system.

Nevertheless, the transfer function is sufficient for a rough initial model: this is the typical use of transient analysis.

# Thermal system: Validation (continued)




Mean squared error (MSE) on the validation data (second and further steps):

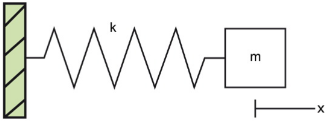
$$J = \frac{1}{N} \sum_{k=1}^{N} e^2(k) = \frac{1}{N} \sum_{k=1}^{N} (\widehat{y}(k) - y(k))^2 \approx 62.10$$

Recall that the data is actually sampled in discrete time, so a meaningful MSE can be computed.

# Table of contents

# Second order system: Motivating example

Second-order systems are also quite common.




Consider a mass $m$ tied to a spring, to which we apply a force $f$ (the input) away from the spring. We measure the position $x$ of the mass relative to the resting spring position (output). From Newton's second law:

$$m\ddot{x}(t) = f(t) - kx(t)$$

where $k$ is the spring constant.

Applying the Laplace transform on both sides:

$$ms^2X(s) = F(s) - kX(s)$$

leading to the transfer function:

$$H(s) = \frac{X(s)}{F(s)} = \frac{1}{ms^2 + k}$$

# Second order system: General form

$$H(s) = \frac{K\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

where:

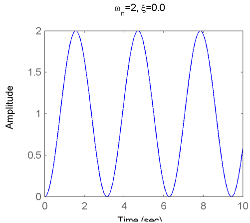
- $K$ is the gain ($= \frac{1}{k}$ in the example)
- $\xi$ is the damping ($= 0$ in the example)
- $\omega_n$ is the natural frequency ($= \sqrt{k/m}$ in the example)

# Table of contents

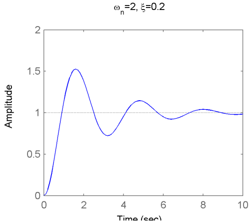
# 2nd order step response shapes

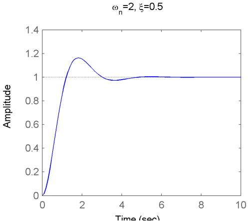
Damping factor $\xi$ is crucial in determining step response shape.

$\xi = 0$, undamped



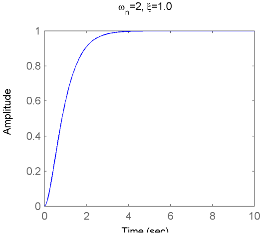

$\xi \in (0, 1)$, underdamped; smaller $\xi$ gives larger oscillations



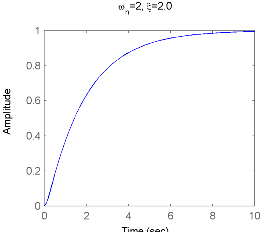



# 2nd order step response shapes (continued)

$\xi = 1$, critically damped

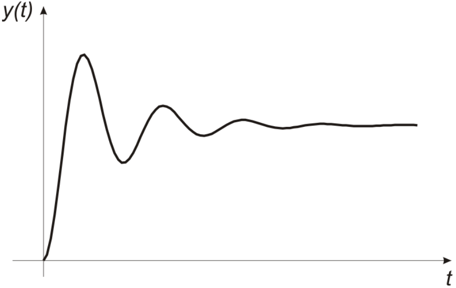



$\xi > 1$, overdamped



# Underdamped 2nd order step response

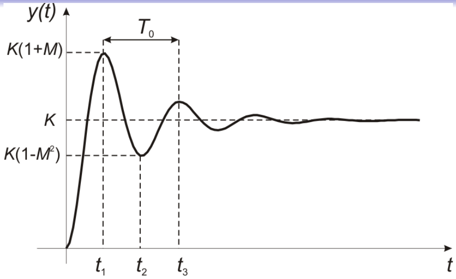
We are mostly interested in the underdamped case ($\xi \in (0, 1)$)




Solving for $y(t)$ we get:

$$y(t) = K \left[ 1 - \frac{1}{\sqrt{1-\xi^2}} e^{-\xi \omega_n t} \sin(\omega_n \sqrt{1-\xi^2} t + \arccos \xi) \right]$$

# Response characteristics




Steady-state value: $\lim_{t\to\infty} K\left[1 - \frac{1}{\sqrt{1-\xi^2}} e^{-\xi\omega_n t}\sin(\dots)\right] = K$

To get the peaks and valleys, we solve for zero derivative:

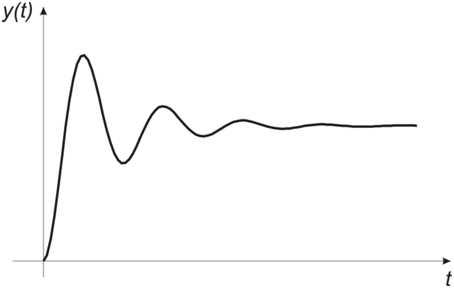
$$\dot{y}(t) = \frac{K\omega_n}{\sqrt{1-\xi^2}} e^{-\xi\omega_n t}\sin(\omega_n\sqrt{1-\xi^2}t) = 0$$

$$\Rightarrow t_m = \frac{m\pi}{\omega_n\sqrt{1-\xi^2}}, \quad m \geq 0$$

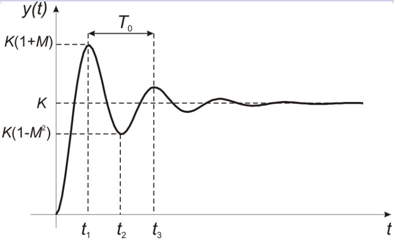
$$y(t_m) = K[1 + (-1)^{m+1}M^m], \text{ where } \textit{overshoot } M = e^{-\frac{\xi\pi}{\sqrt{1-\xi^2}}}$$

# Starting from the step response

Now, consider we are given a step response of an unknown system. Using the insight developed above, we can find an approximate transfer function of the system.



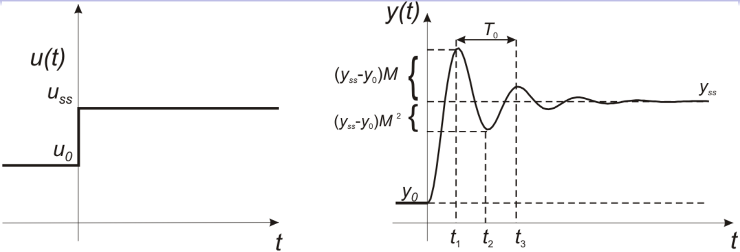
# Determining the parameters




## Algorithm

1. Determine steady-state output value $y_{ss}$. That is the gain $K$.
2. Determine overshoot $M$, (a) from the first peak: $M = \frac{y(t_1)-y_{ss}}{y_{ss}}$, or (b) from ratio of first valley and first peak: $M = \frac{y_{ss}-y(t_2)}{y(t_1)-y_{ss}}$.
3. Solve $M = e^{-\frac{\xi\pi}{\sqrt{1-\xi^2}}}$, leading to $\boxed{\xi = \frac{\log 1/M}{\sqrt{\pi^2+\log^2 1/M}}}$
4. Read oscillation period as the time between first two peaks $\boxed{T_0 = t_3 - t_1} = \frac{2\pi}{\omega_n\sqrt{1-\xi^2}}$; or $T_0 = 2(t_2 - t_1)$. Then, $\boxed{\omega_n = \frac{2\pi}{T_0\sqrt{1-\xi^2}}}$, or $\omega_n = \frac{2}{T_0}\sqrt{\pi^2 + \log^2 1/M}$.

# Nonzero initial conditions

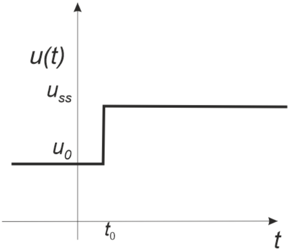


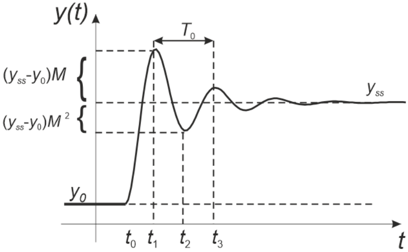

Similar to 1st order case: new input $u(t) = u_0 + (u_{ss} - u_0)u_S(t)$, so the new output is again just a shifted and scaled version of the ideal step response $y_S(t)$: $y(t) = y_0 + (u_{ss} - u_0)y_S(t)$. Modified algorithm:

1. Gain $\boxed{K = \frac{y_{ss}-y_0}{u_{ss}-u_0}}$.
2. Overshoot (a) $\boxed{M = \frac{y(t_1)-y_{ss}}{y_{ss}-y_0}}$ (we need to subtract $y_0$), or (b) $M = \frac{y_{ss}-y(t_2)}{y(t_1)-y_{ss}}$ (no change in this formula).

$\xi$, $T_0$: same as before.

# Nonzero initial time



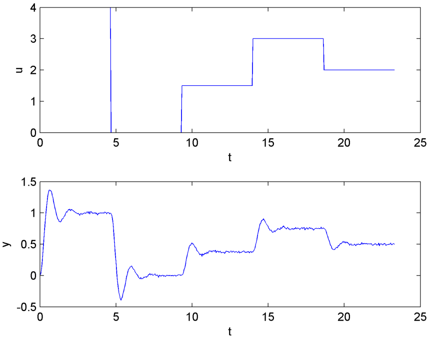




Like in the first-order case, we simply shift everything by the starting time $t_0$ of the step. This has no impact in the algorithm as we use relative times to compute the oscillation period, anyway.

# Table of contents

# 2nd order example

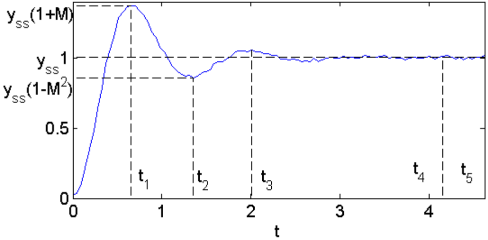
Simulation example, 500 samples with sampling time $\approx 0.047$.




Note again the measurement noise. Also, while the experiment has zero initial condition ($u_0 = y_0 = 0$), the steps still have nonstandard values (different from 1).

We will use step 1 for identification, steps 3-5 for validation (using the fact that step 2 returns the system to zero initial condition).

# Example: step response model

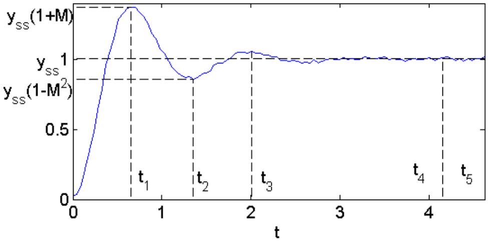



Since the output is noisy, we determine the steady state value by averaging a few last samples in steady-state, namely numbers 90 to 100, between $t_4$ and $t_5$:

$$y_{ss} \approx \frac{1}{11} \sum_{k=90}^{100} y(k) \approx 1.00$$

We read on the graph: $t_1 \approx 0.65$, $t_2 \approx 1.35$, $t_3 \approx 1.96$, $y(t_1) \approx 1.37$, $y(t_2) \approx 0.86$. Finally, $u_{ss} = 4$.

# Example: Determining the parameters




1. Gain $K = \frac{y_{ss}-y_0}{u_{ss}-u_0} = \frac{y_{ss}}{u_{ss}} \approx 0.25$.
2. Overshoot $M = \frac{y(t_1)-y_{ss}}{y_{ss}-y_0} = \frac{y(t_1)-y_{ss}}{y_{ss}} \approx 0.36$.
3. Damping $\xi = \frac{\log 1/M}{\sqrt{\pi^2+\log^2 1/M}} \approx 0.31$.
4. Period $T_0 = t_3 - t_1 \approx 1.31$, so natural frequency $\omega_n = \frac{2\pi}{T_0\sqrt{1-\xi^2}} \approx 5.05$.

# Example: Transfer function model

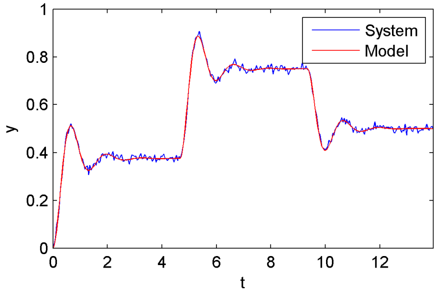
$$\widehat{K} = 0.25$$

$$\widehat{\xi} = 0.31$$

$$\widehat{\omega_n} = 5.05$$

$$\widehat{H}(s) = \frac{\widehat{K}\widehat{\omega}_n^2}{s^2 + 2\widehat{\xi}\widehat{\omega}_n s + \widehat{\omega}_n^2} = \frac{6.38}{s^2 + 3.09s + 25.51}$$

# Example: Validation of transfer function model



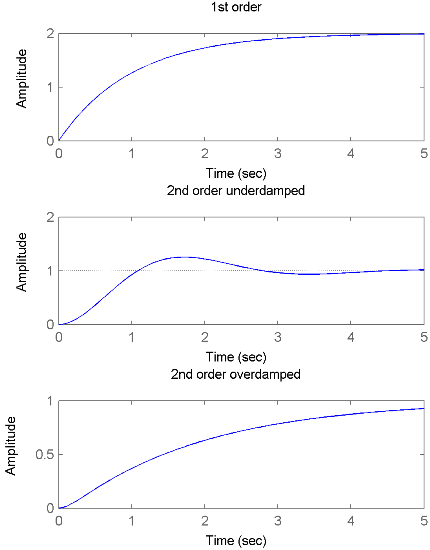

Very good fit (not surprising since this is synthetic data).

Mean squared error (MSE):

$$J = \frac{1}{N}\sum_{k=1}^{N} e^2(k) = \frac{1}{N}\sum_{k=1}^{N}(\widehat{y}(k) - y(k))^2 \approx 9.66 \cdot 10^{-5}$$

# Table of contents

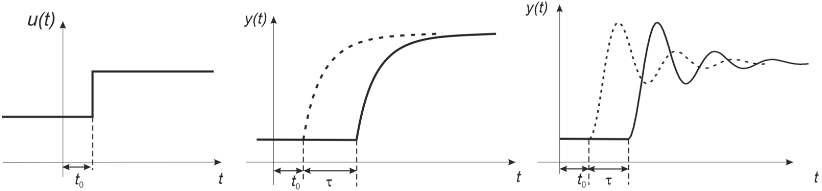
# Choosing the order




Even when it is overdamped or critically-damped, at $t = 0$ a 2nd order system response will have a derivative of 0: it will be tangent to the time axis. In contrast, the tangent slope is $K/T$ for 1st order systems.

# Time delay

The response of a 1st or 2nd order system with a time delay of $\tau$ has the same shape as before, but after the input changes, there is a delay of $\tau$ before the output responds. Don't mix it up with the (possibly nonzero) step time $t_0$! The delay is the interval *after* $t_0$ taken by the output to start reacting.




The delay is represented in the transfer function as follows, for first and second-order systems:

$$H(s) = \frac{K}{Ts+1} e^{-s\tau}, \quad H(s) = \frac{K\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} e^{-s\tau}$$

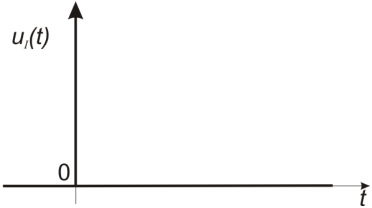
The value of $\tau$ can be simply read on the graph.

# Summary step response

- Gain $K$: difference between output levels / difference between input levels.
- 1st order: time constant $T$ found on *time* axis when the *output* axis reaches 63.2% of the difference.
- 2nd order: period $T_0$ read on the graph, overshoot $M$ computed from the peaks and valleys. $\xi, \omega_n$ follow.
- Graph inspection + mean squared error used to validate models.
- Averaging to reject noise in initial/steady-state values.
- Nonzero initial time and time delays handled by shifting the time values appropriately; time delay goes in transfer function.

# Table of contents

# Ideal impulse input




The ideal impulse is the Dirac delta. An informal definition:

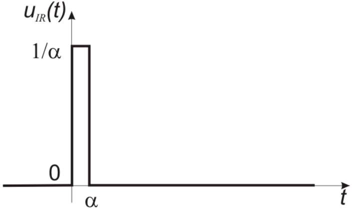
$$u_{\mathrm{I}}(t) = \begin{cases} \infty & t = 0 \\ 0 & t \neq 0 \end{cases}$$

with the additional condition $\int_{-\infty}^{\infty} u_{\mathrm{I}}(t) dt = 1$.

(In fact, the ideal impulse is not a function and requires the notion of distributions to be formally defined.)

# Practical impulse realization

In practice, we cannot create signals of infinite amplitude.
So, an approximate impulse is realized with the rectangular signal:




$$u_{\mathrm{IR}}(t) = \begin{cases} \frac{1}{\alpha} & t \in [0, \alpha) \\ 0 & \text{otherwise} \end{cases}$$

where $\alpha \ll$ (much smaller than the time constants in the system).

Note the signal still obeys $\int_{-\infty}^{\infty} u_{\mathrm{IR}}(t)dt = 1$ (rectangle has area 1).

This approximate impulse will introduce differences (error) from the true impulse response, but for small $\alpha$ the error is not large. We develop the analysis in the ideal case, while the examples use the practical realization.

# Useful property of impulse response

In the Laplace domain:

$$\text{step input } U_S(s) = \frac{1}{s}, \qquad \text{impulse input } U_{\mathrm{I}}(s) = 1$$

Recall that the time-domain response of a system can be expressed as: $y(t) = \mathcal{L}^{-1}\{Y(s)\}$, and $Y(s) = H(s)U(s)$. So:

$$Y_S(s) = \frac{1}{s} Y_{\mathrm{I}}(s), \qquad Y_{\mathrm{I}}(s) = sY_S(s)$$

$$y_S(t) = \int_0^t y_{\mathrm{I}}(\tau)d\tau, \qquad y_{\mathrm{I}}(t) = \dot{y}_S(t)$$

The impulse response is the *derivative of the step response*.

# Table of contents

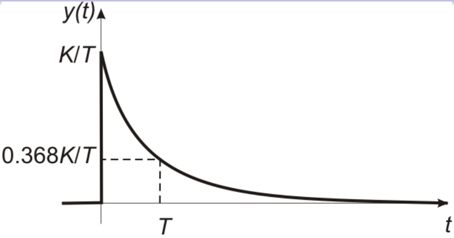
# Recall: First order system

$$H(s) = \frac{K}{Ts + 1}$$

where:

- $K$ is the gain
- $T$ is the time constant

# Ideal 1st order impulse response




Using the relation to the step response, and the derivative of the step response we already computed, we have the impulse response:

$$y_{\mathrm{I}}(t) = \frac{K}{T} e^{-t/T}, \qquad t \geq 0$$

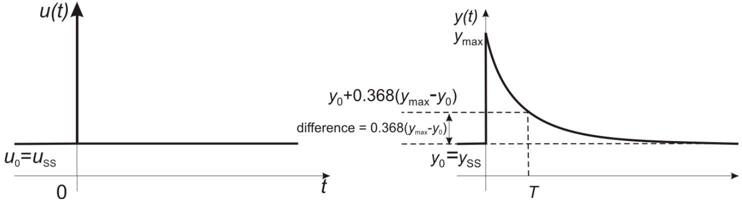
from where:

$$\begin{cases} y_{\mathrm{I}}(0) = \dfrac{K}{T} = y_{\max} \\ y_{\mathrm{I}}(T) = \dfrac{K}{T} e^{-1} = y_{\max} e^{-1} \approx 0.368 y_{\max} \end{cases}$$

Note: $y_{\mathrm{I}}(4T) = 0.0183 y_{\max}$, so like for the step response, the output is roughly in steady state after $4T$.

# Nonzero initial conditions

When the initial conditions are nonzero, the impulse is shifted along the vertical axis. Assume the system was in steady-state at $y_0$ with input held constant at $u_0$.



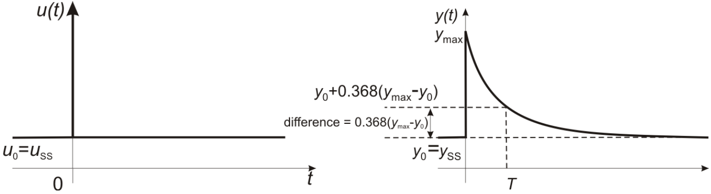

From linearity, given the shifted $u(t) = u_0 + u_{\mathrm{I}}(t)$, we have a shifted $y(t) = y_0 + y_I(t)$. Note the input is not scaled as the result would no longer be an approximate impulse (area different from 1).

So the behavior is: $$\begin{cases} y_{\max} = y_0 + \dfrac{K}{T} \\ y(T) = y_0 + 0.368(y_{\max} - y_0) \end{cases}$$

Note $u_0 = u_{ss}, y_0 = y_{ss}$.

# Determining the parameters

Consider now that we are given the impulse response of an unknown system. As in the step case, we can use this response to find an approximate transfer function (a parametric model) of the system.

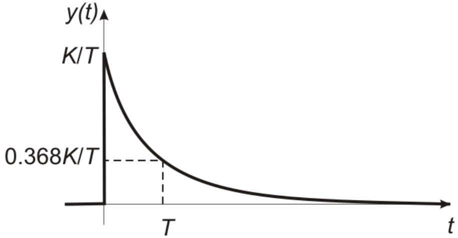



We consider first nonzero initial conditions because that is actually a favorable situation: we have a reliable way to find the gain $K$.

## Algorithm

1. Read the steady-state (or initial) output $y_{ss} = y_0$ and input $u_{ss} = u_0$. Then, $\boxed{K = y_{ss}/u_{ss}}$.
2. Read $y_{\max}$ and read the time constant $T$ at the moment where the output decreases to $\boxed{0.368 \text{ of the difference } y_{\max} - y_0}$.

# Determining the parameters in zero initial conditions




We can estimate the gain by using $y_{\max} = \frac{K}{T}$, but in practice this will not be as accurate (e.g. because of noise and the non-ideal impulse signal).

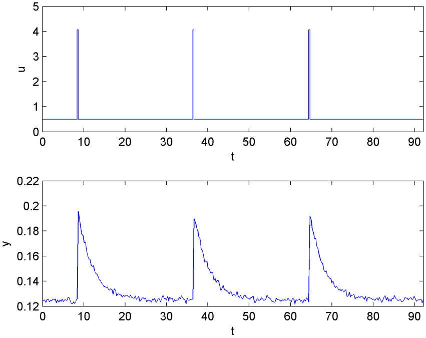
## Algorithm

1. Read $y_{\max}$ and determine the time where the output decreases to 0.368 of $y_{\max}$. That is the time constant $T$.
2. Find $K = y_{\max} T$.

# Table of contents

# 1st order example

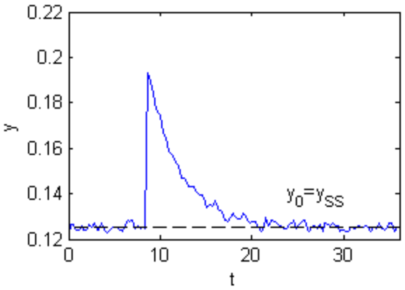
Simulation example, 330 samples with sampling time $T_s = 0.28$ (30 samples are the initial steady-state regime, then 100 for each impulse response). The practical impulses are realized with $\alpha = T_s = 0.28$, amplitude $1/\alpha \approx 3.57$.




Note the measurement noise and the nonzero initial condition.

We will use impulse 1 for identification, impulses 2-3 for validation.

# Example: Model and parameters

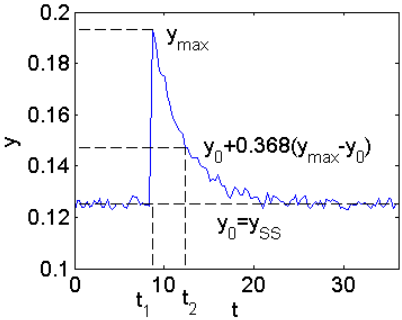



We use the graph to estimate a transfer function. We have $u_0 = u_{ss} = 0.5$.

We find the steady state output (equal to the initial output) by averaging a few steady-state samples:

$$y_{ss} = y_0 \approx \frac{1}{11} \sum_{k=120}^{130} y(k) \approx 0.13$$

# Example: Model and parameters (continued)




The maximum output value is $y_{\max} \approx 0.19$, reached at $t_1 \approx 8.86$.
Value $y_0 + 0.368(y_{\max} - y_0) \approx 0.15$ is reached at $t_2 \approx 12.60$.
Therefore:

1. $K = y_{ss}/u_{ss} \approx 0.25$.
2. $T = t_2 - t_1 \approx 3.92$.

Note we take into account the nonzero time $t_1$ when the impulse is applied!

# Example: Transfer function model

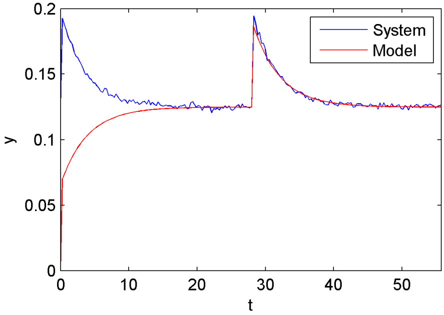
$$\widehat{K} = 0.25$$

$$\widehat{T} = 3.92$$

$$\widehat{H}(s) = \frac{\widehat{K}}{\widehat{T}s + 1} = \frac{0.25}{3.92s + 1}$$

# Example: Validation of transfer function model

Comparison of system data and model response for the validation data (second and third impulse responses):




The simulation does not take into account the nonzero initial condition of the system, hence the first part has large differences.

We will present a method to take into account initial conditions, which works not only for impulse signals, but for *any* input (step, etc.).

# State space model of an *n*th order system

A (continuous-time) state space model of a linear system is a representation of the system in the following form:

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t) + Du(t)$$

where:

- $x$ state vector, $x \in \mathbb{R}^n$ with $n$ the order of the system
- $u$ and $y$ are the usual input and output. They can be vectors if the system has several inputs or outputs, but for us here, a scalar input and output are enough.
- $A$ state matrix, $B$ input matrix, $C$ output matrix, and $D$ feedthrough matrix. They have appropriate dimensions: $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times 1}$ (a vector, due to scalar input), $C \in \mathbb{R}^{1 \times n}$ (a vector, due to scalar output), $D \in \mathbb{R}$ (a scalar, usually 0).

# State space model of a general 1st order system

Starting from transfer function model:

$$H(s) = \frac{K}{Ts + 1} = \frac{Y(s)}{U(s)}$$

and moving back to the time domain we get:

$$\dot{y}(t) = \frac{-1}{T} y(t) + \frac{K}{T} u(t)$$

By simply taking $x = y$ (recall that the system has order 1 so a single state suffices), we can write:

$$\dot{x}(t) = -\frac{1}{T} x(t) + \frac{K}{T} u(t)$$
$$y(t) = x(t)$$

so our state space model has $A = -\frac{1}{T}$, $B = \frac{K}{T}$, $C = 1$, $D = 0$.

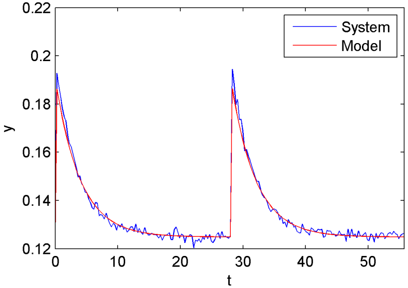
# Back to example: (Approximate) state space model

$$\dot{x}(t) = -\frac{1}{\widehat{T}}x(t) + \frac{\widehat{K}}{\widehat{T}}u(t) = -0.26x(t) + 0.06u(t)$$
$$y(t) = x(t)$$

Matlab: `Hss = ss(A, B, C, D)`

# Example: Validation with correct initial condition

To take the initial condition into account, we simply set $x(0) = y_0$ when starting the simulation.




Mean squared error (MSE) on the validation data:

$$J = \frac{1}{N} \sum_{k=1}^{N} e^2(k) \approx 3.74 \cdot 10^{-6}$$

# Table of contents

# Table of contents

# Recall: Second order system

$$H(s) = \frac{K\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

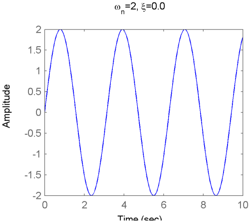
where:

- $K$ is the gain
- $\xi$ is the damping factor
- $\omega_n$ is the natural frequency

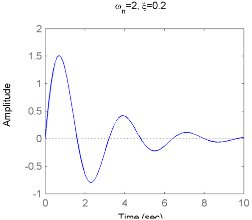
# 2nd order impulse response shapes

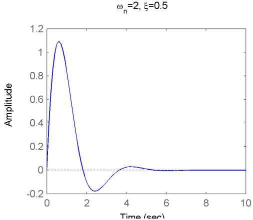
As for the step response, the damping factor $\xi$ determines the shape.

$\xi = 0$, undamped



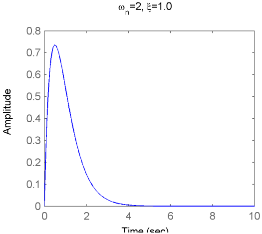

$\xi \in (0, 1)$, underdamped – we are interested in this case



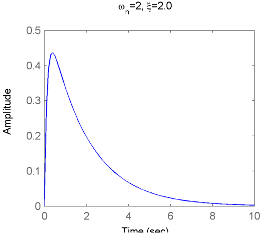



# 2nd order impulse response shapes (continued)

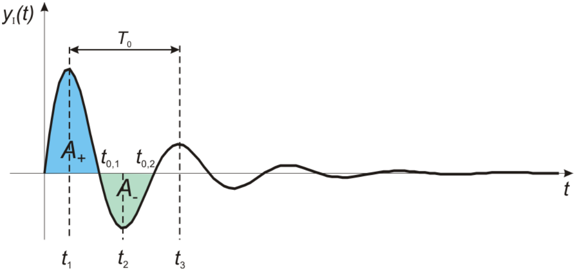
$\xi = 1$, critically damped




$\xi > 1$, overdamped



# Underdamped impulse response

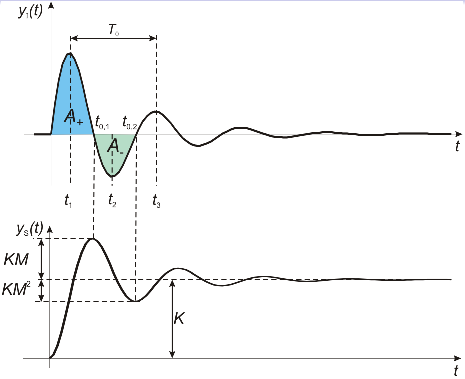



Using the derivative of the step response we already computed, we have the impulse response:

$$y_{\mathrm{I}}(t) = \frac{K\omega_n}{\sqrt{1-\xi^2}} e^{-\xi\omega_n t} \sin(\omega_n \sqrt{1-\xi^2}\, t)$$

Already note that the oscillation period is unchanged, so $T_0 = t_3 - t_1 = 2(t_2 - t_1)$.

# Underdamped impulse response: areas

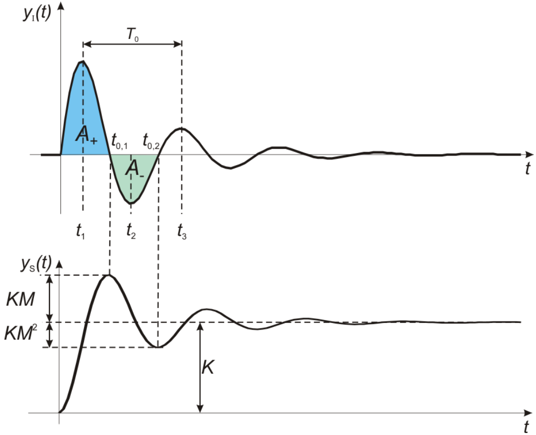



Since $y_S(t) = \int_0^t y_\mathrm{I}(\tau)d\tau$, and remembering the sizes of the first peak and valley in the step response as a function of the overshoot $M$:

$$A_+ = \int_0^{t_{0,1}} y_\mathrm{I}(\tau)d\tau = y_\mathrm{S}(t_{0,1}) = K + KM, \qquad A_- = -\int_{t_{0,1}}^{t_{0,2}} y_\mathrm{I}(\tau)d\tau =$$

$$= -[y_\mathrm{S}(t_{0,2}) - y_\mathrm{S}(t_{0,1})] = -[K - KM^2 - (K + KM)] = KM^2 + KM$$

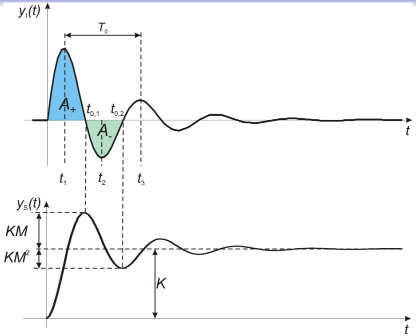
# Underdamped impulse response: overshoot




Therefore:

$$\frac{A_-}{A_+} = \frac{KM^2 + KM}{K + KM} = M$$

# Underdamped impulse response: area estimation



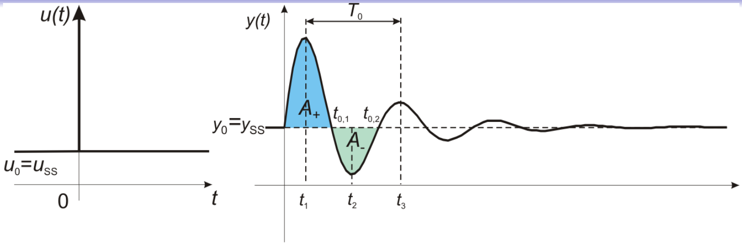

The areas are estimated numerically:

$$A_+ = \int_0^{t_{0,1}} (y(t) - y_0)dt \approx T_s \sum_{k=0}^{k_{0,1}} (y(k) - y_0)$$

$$A_- = \int_{t_{0,1}}^{t_{0,2}} (y_0 - y(t))dt \approx T_s \sum_{k=k_{0,1}}^{k_{0,2}} (y_0 - y(k))$$

with $k_{0,1}$, $k_{0,2}$ discrete-time (sample) indices corresponding to $t_{0,1}$, $t_{0,2}$.

# Nonzero initial conditions: estimating $K$




In nonzero initial conditions, the impulse is shifted, $u(t) = u_0 + u_{\mathrm{I}}(t)$, leading to a shifted $y(t) = y_0 + y_I(t)$. Note $u_0 = u_{ss}, y_0 = y_{ss}$.

From the steady-state values we can estimate the gain: $K = \frac{y_{ss}}{u_{ss}}$. There is no change in $T_0$, but the areas must now be found relative to the steady-state value:

$$A_+ = \int_0^{t_{0,1}} (y(\tau) - y_0) d\tau \qquad = K + KM$$

$$A_- = -\int_{t_{0,1}}^{t_{0,2}} (y(\tau) - y_0) d\tau = \int_{t_{0,1}}^{t_{0,2}} (y_0 - y(\tau)) d\tau \qquad = KM^2 + KM$$

# Determining the parameters

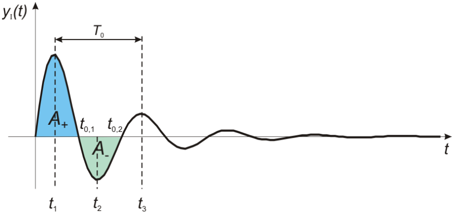
Given the impulse response of an unknown system, the transfer function is found as follows:

## Algorithm

1. Read steady-state output $y_{ss}$, and $u_{ss}$. The gain is $\boxed{K = \frac{y_{ss}}{u_{ss}}}$.
2. Read time values where $y(t)$ crosses $y_{ss}$: $t_{0,1}$, $t_{0,2}$,. Compute areas $A_+ = \int_0^{t_{0,1}} (y(\tau) - y_0) d\tau$, $A_- = \int_{t_{0,1}}^{t_{0,2}} (y_0 - y(\tau)) d\tau$. Find overshoot $\boxed{M = \frac{A_-}{A_+}}$.
3. The damping factor is $\xi = \frac{\log 1/M}{\sqrt{\pi^2 + \log^2 1/M}}$.
4. Read time values at peaks, $t_1, t_3$ (or peak and valley $t_1, t_2$). Find the oscillation period $\boxed{T_0 = t_3 - t_1}$, or $T_0 = 2(t_2 - t_1)$.
5. Natural frequency $\boxed{\omega_n = \frac{2\pi}{T_0\sqrt{1-\xi^2}}}$, or $\omega_n = \frac{2}{T_0}\sqrt{\pi^2 + \log^2 1/M}$.

Note the relationships between $M$, $T_0$, $\xi$, and $\omega_n$ are true regardless of the response type, so algorithm steps 3 and 5 use the same formulas as in the step-response case.

# Determining the gain in zero initial conditions




We solve $\dot{y}(t) = 0$ to get $t_1$ for the first peak, and replace it in $y(t)$ to get the value at the peak. After some calculation we obtain:

$$y(t_1) = K\omega_n e^{-\frac{\xi \arccos \xi}{\sqrt{1-\xi^2}}}$$

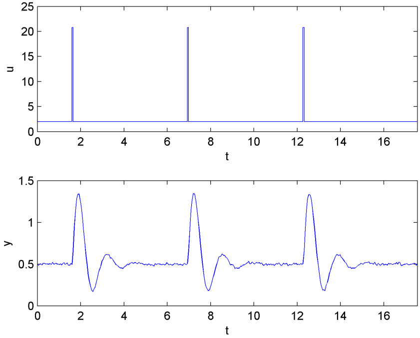
which can be used to estimate the gain as $K = \frac{y(t_1)}{\omega_n e^{-\frac{\xi \arccos \xi}{\sqrt{1-\xi^2}}}}$. This requires $\xi$ and $\omega_n$ to be computed by the methods above, which can be done regardless of the initial condition.

For the same reasons as in the first-order case, this method is less accurate than determining the gain from nonzero steady-state values.

# Table of contents

# 2nd order example

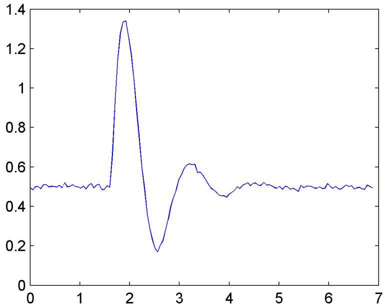
Simulation example, 330 samples with sampling time $\approx 0.053$.




We again have a nonzero initial condition (and as usual measurement noise).

We will use impulse 1 for identification, impulses 2-3 for validation.

# Example: Steady-state values and gain

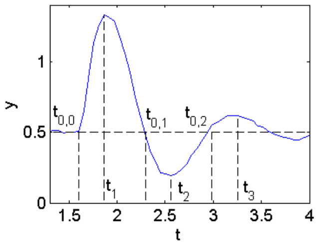



We use the graph to estimate a transfer function. We have $u_0 = u_{ss} = 2$.

We determine the steady state output (equal to the initial output) by averaging the last 11 samples:

$$y_{ss} = y_0 \approx \frac{1}{11} \sum_{k=120}^{130} y(k) \approx 0.5$$

# Example: Damping factor




We read $t_{0,0} \approx 1.6$, $t_{0,1} \approx 2.3$, $t_{0,3} \approx 2.99$. Note the impulse is applied at time $t_{0,0} \neq 0$, so we need to take this into account.

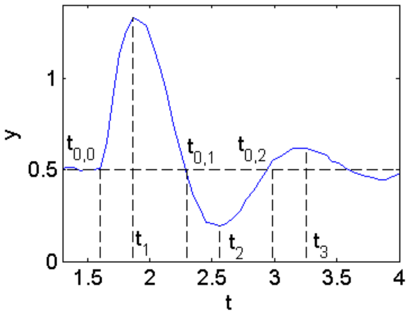
Recall area estimation:

$$A_+ = \int_{t_{0,0}}^{t_{0,1}} (y(t) - y_0) dt \approx T_s \sum_{k=k_{0,0}}^{k_{0,1}} (y(k) - y_0) \approx 0.34$$

$$A_- = \int_{t_{0,1}}^{t_{0,2}} (y_0 - y(t)) dt \approx T_s \sum_{k=k_{0,1}}^{k_{0,2}} (y_0 - y(k)) \approx 0.12$$

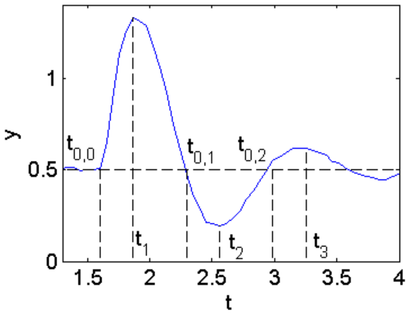
with $k_{0,0}$, $k_{0,1}$, $k_{0,2}$ sample indices corresponding to $t_{0,0}$, $t_{0,1}$, $t_{0,2}$.

# Example: Damping factor (continued)




From these areas, $M = \frac{A_-}{A_+} \approx 0.36$, and $\xi = \frac{\log 1/M}{\sqrt{\pi^2 + \log^2 1/M}} \approx 0.31$.

# Example: Oscillation period




We read $t_1 \approx 1.92$ and $t_3 \approx 3.2$, leading to $T_0 = 1.28$. From this, $\omega_n = \frac{2\pi}{T_0\sqrt{1-\xi^2}} \approx 5.16$.

# Example: Transfer function model

$$\widehat{K} = 0.25$$

$$\widehat{\xi} = 0.31$$

$$\widehat{\omega_n} = 5.16$$

$$\widehat{H}(s) = \frac{\widehat{K}\widehat{\omega}_n^2}{s^2 + 2\widehat{\xi}\widehat{\omega}_n s + \widehat{\omega}_n^2} = \frac{6.64}{s^2 + 3.21s + 26.68}$$

# General state space model of a 2nd order system

Recall that to simulate starting from nonzero initial conditions, we need a state space model $\dot{x}(t) = Ax(t) + Bu(t)$, $y(t) = Cx(t) + Du(t)$. Starting from $H(s)$ and moving to the time domain, we get:

$$\ddot{y}(t) = -2\xi\omega_n\dot{y}(t) - \omega_n^2 y(t) + K\omega_n^2 u(t)$$

By taking $x_1 = y, x_2 = \dot{y}$ (since system has order 2), we can write:

$$\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\end{bmatrix} = \begin{bmatrix} x_2(t) \\ -2\xi\omega_n x_2(t) - \omega_n^2 x_1(t) + K\omega_n^2 u(t)\end{bmatrix}$$
$$= \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\xi\omega_n\end{bmatrix}\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix} + \begin{bmatrix}0\\ K\omega_n^2\end{bmatrix} u(t)$$
$$y(t) = x_1(t) = \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix} + 0u(t)$$

from where the matrices *A*, *B*, *C*, *D* of the state-space model are obtained.

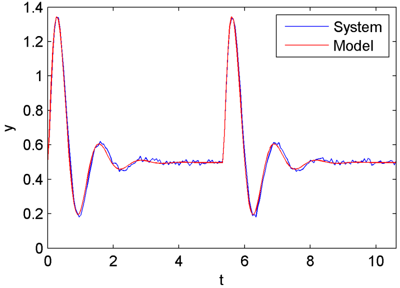
# Back to example: (Approximate) state space model

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ -26.68 & -3.22 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 6.64 \end{bmatrix} u(t)$$
$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} x(t) + 0u(t)$$

where $x$ is now the whole state vector, $x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$.

# Example: Validation

To take the initial condition into account, we set $x_1(0) = y_0, x_2(0) = 0$ when starting the simulation (we start from steady state, so $x_2(0) = \dot{y}(0) = 0$).



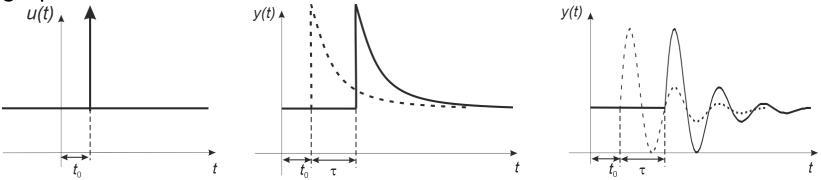

Mean squared error (MSE) on the validation data:

$$J = \frac{1}{N} \sum_{k=1}^{N} e^2(k) \approx 8 \cdot 10^{-4}$$

# Table of contents

# Time delay

Like the step response, the impulse response of a 1st or 2nd order system with a time delay of $\tau$ has the typical shape, but after the input changes (which may happen at nonzero time $t_0$), there is a delay of $\tau$ before the output responds. The value of $\tau$ can be simply read on the graph.




Transfer functions:

$$H(s) = \frac{K}{Ts+1} e^{-s\tau}, \quad H(s) = \frac{K\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} e^{-s\tau}$$

# Summary impulse response

- Impulse response = derivative of the step response.
- Gain $K$: output/input in nonzero initial conditions, otherwise from maximum value.
- 1st order: time constant $T$ found on *time* axis when the *output* axis reaches 36.8% of the difference.
- 2nd order: period $T_0$ read on the graph, overshoot $M$ computed via numerical integration. $\xi, \omega_n$ follow.
- State-space model to handle nonzero initial conditions.