6.300 Signal Processing

Week 5, Lecture B: Systems

- System Abstraction
- Linearity and Time Invariance

Lecture slides are available on CATSOOP:

https://sigproc.mit.edu/fall25

- Quiz 1 grade on website; if need help with the class please reach out to us
- When submitting lab code or homework, please make sure you submit the correct version

5) Collaboration Policies

We encourage students to discuss 6.300 concepts and approaches with other students and with the teaching staff to better understand these materials. However, it is important that these conversation be held at a high level, and work that you submit under your name -- including derivations, programs, plots, and explanations -- must be your own. **When you submit an assignment under your name, you are certifying that the details are entirely your own work and that you played at least a substantial role in the conception stage.**

Students should **not** take credit for work done by other students. Students should **not** use solutions of other students (from this semester or from previous semesters) in preparing their own solutions. Students should **not** use artificial intelligence (AI) or large language models (LLMs) to complete exercises, problems, or labs. And students should **not** share their work with other students, including through public repositories such as GitHub.

Copying work and/or knowingly making work available for copying are serious offenses that may result in reduced grades, failing the course, and disciplinary action.

Weekly homework assignments provide an opportunity to develop intuition for new concepts by actively applying the new concepts to solve problems and answer questions. The process of actively struggling with the use of new ideas until you understand them is an effective and rewarding form of education. Reading someone's solution to a problem is not educationally equivalent to generating your own solution. If you skip the process of **personally struggling** with new concepts by getting the answers from someone else, **you will have lost a valuable learning experience.**

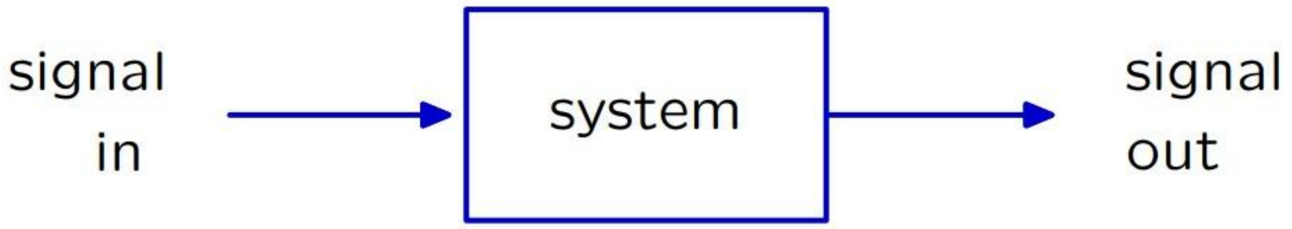
Good problems are a valuable resource. Don't squander that resource.

These policies are in place with the primary goal of *helping you learn more effectively*. If you have any questions about why the policies are structured as they are, or if a certain type of collaboration is allowed, just ask! You can do so by sending e-mail to the instructors at `sigproc-instructors@mit.edu`.

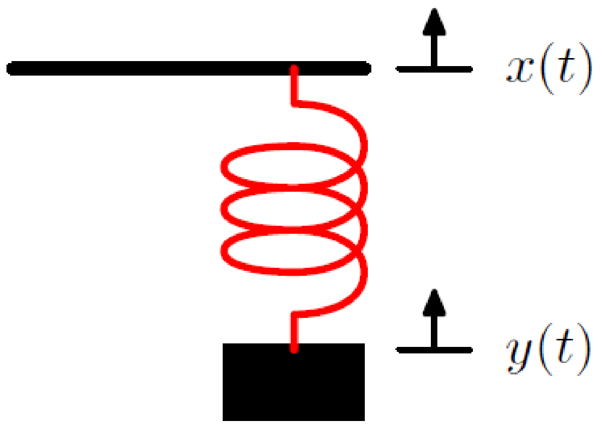
For more information, see the academic integrity handbook.

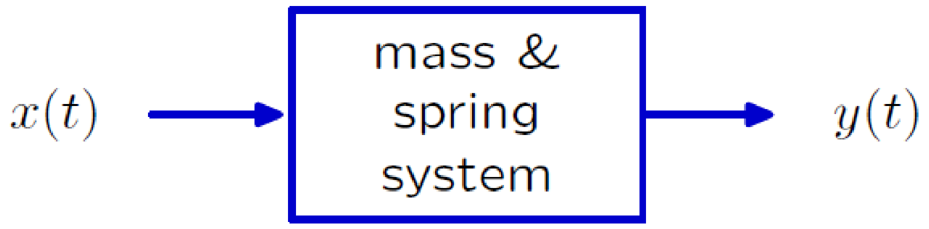
From Signal to Systems: The System Abstraction

Represent a system (physical, mathematical, or computational) by the way it transforms an input signal into an output signal.

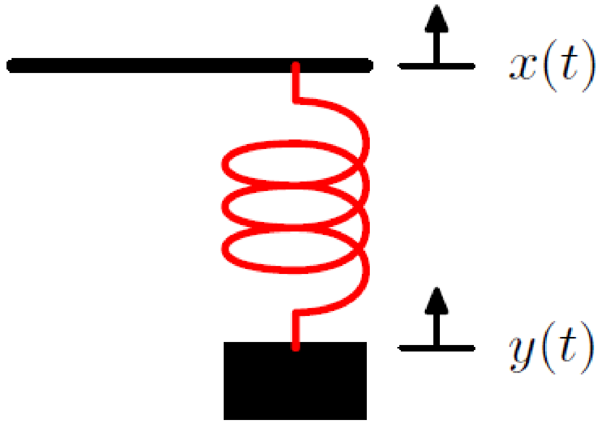


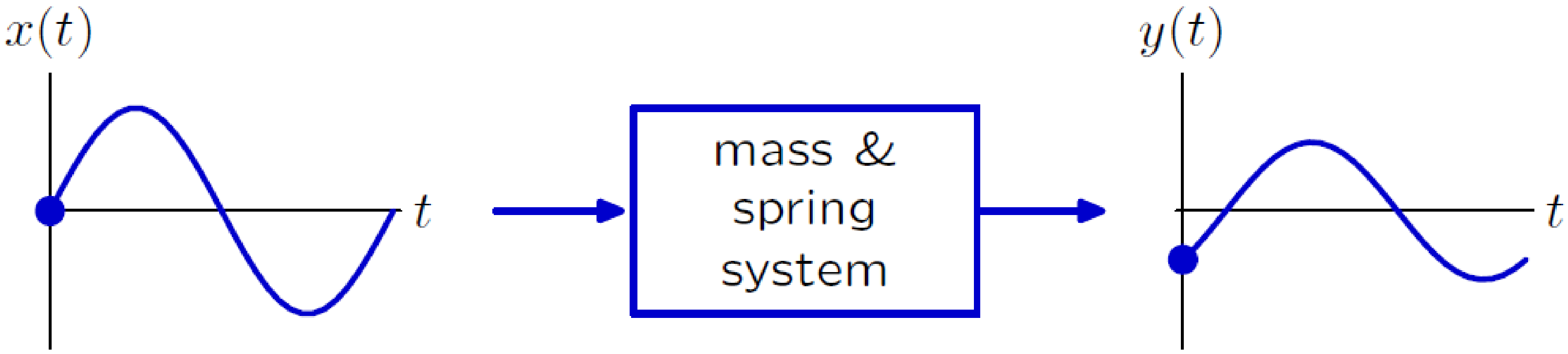
Example: Mass and Spring



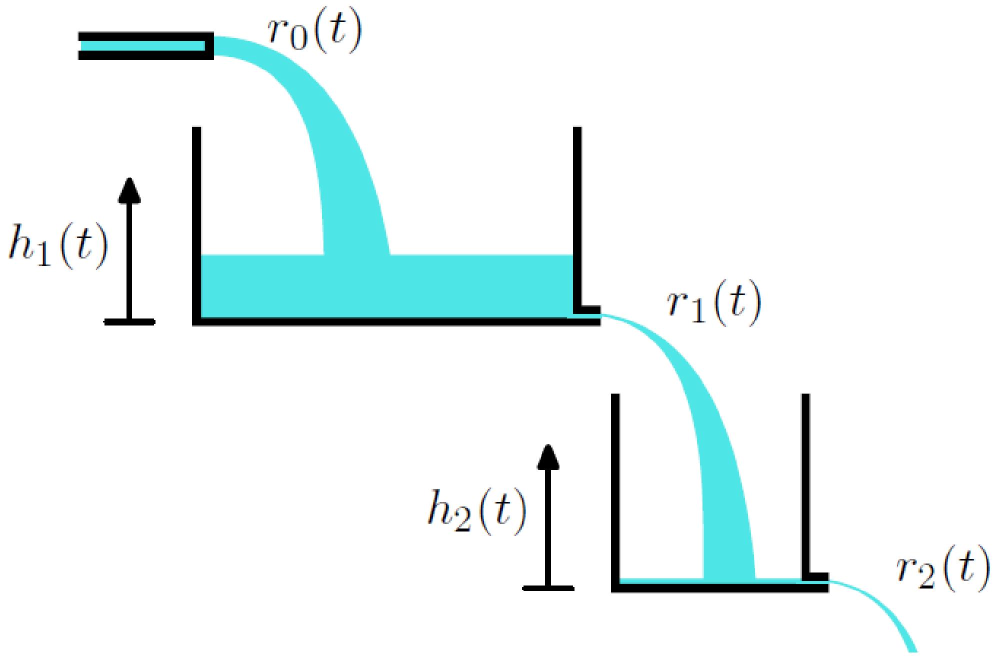


Example: Mass and Spring

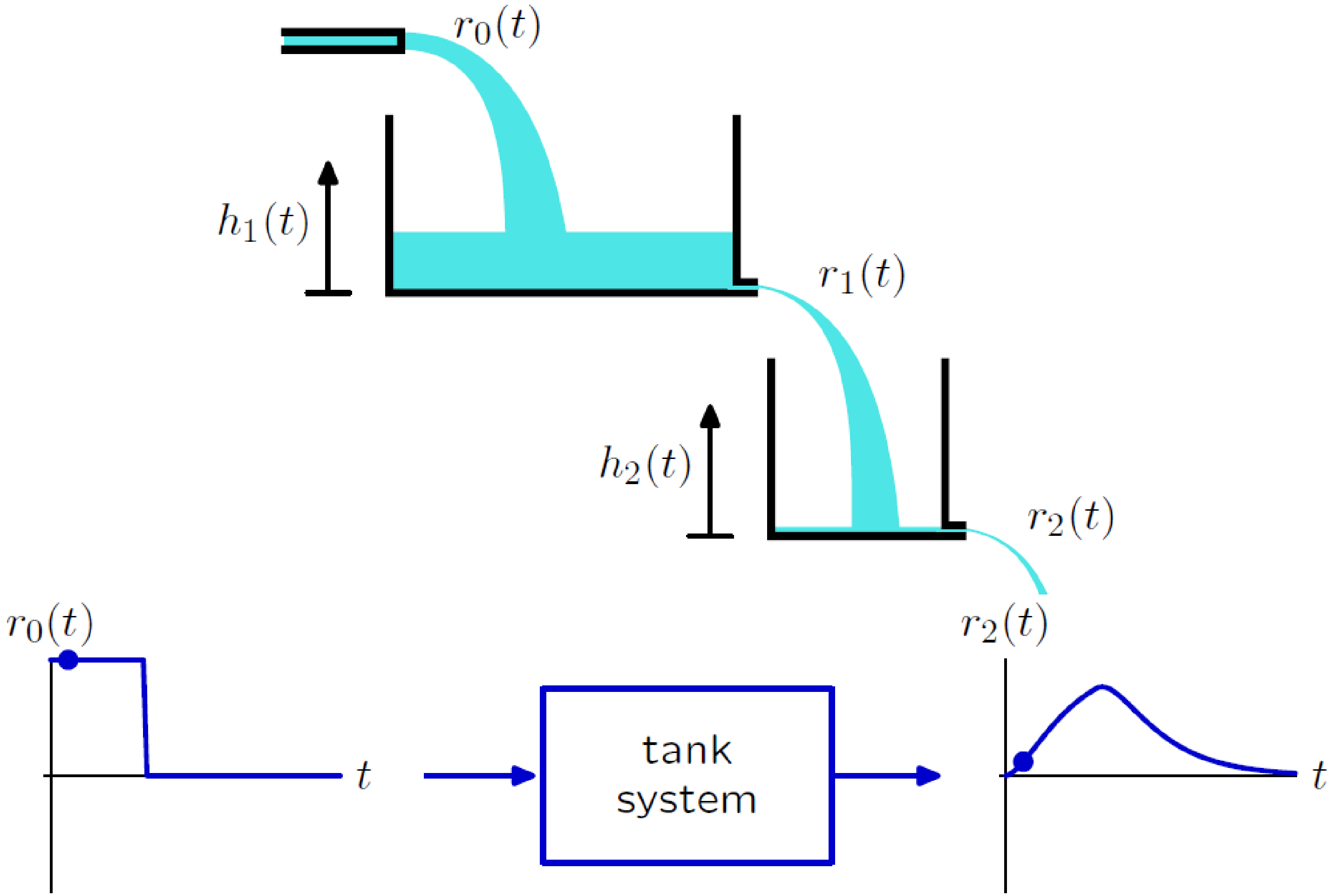




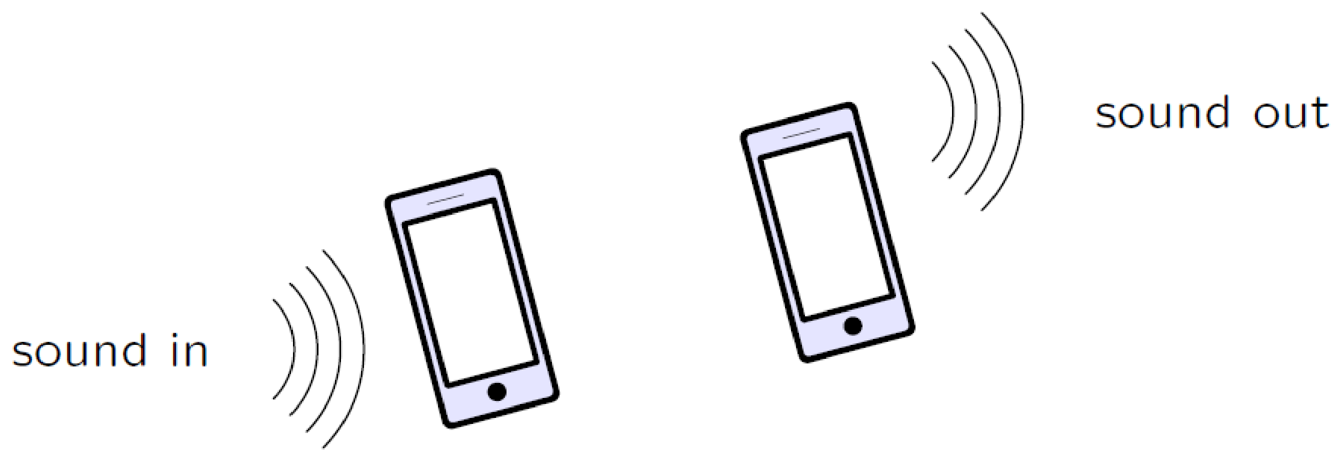
Example: Tanks

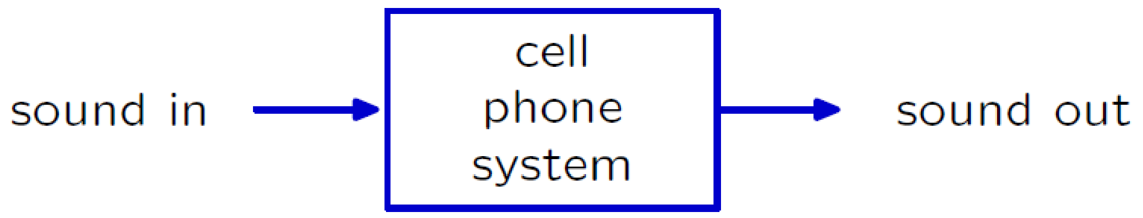


Example: Tanks

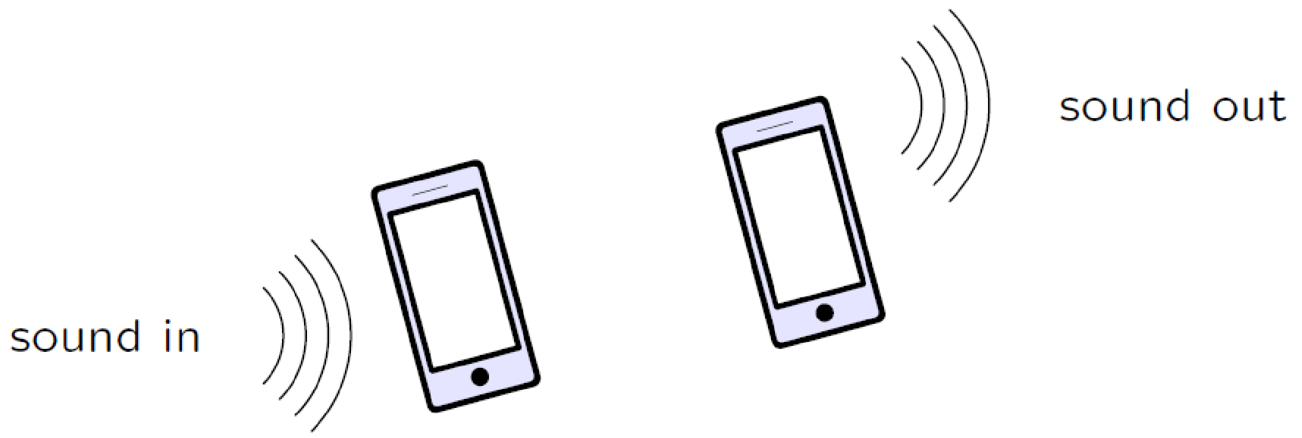


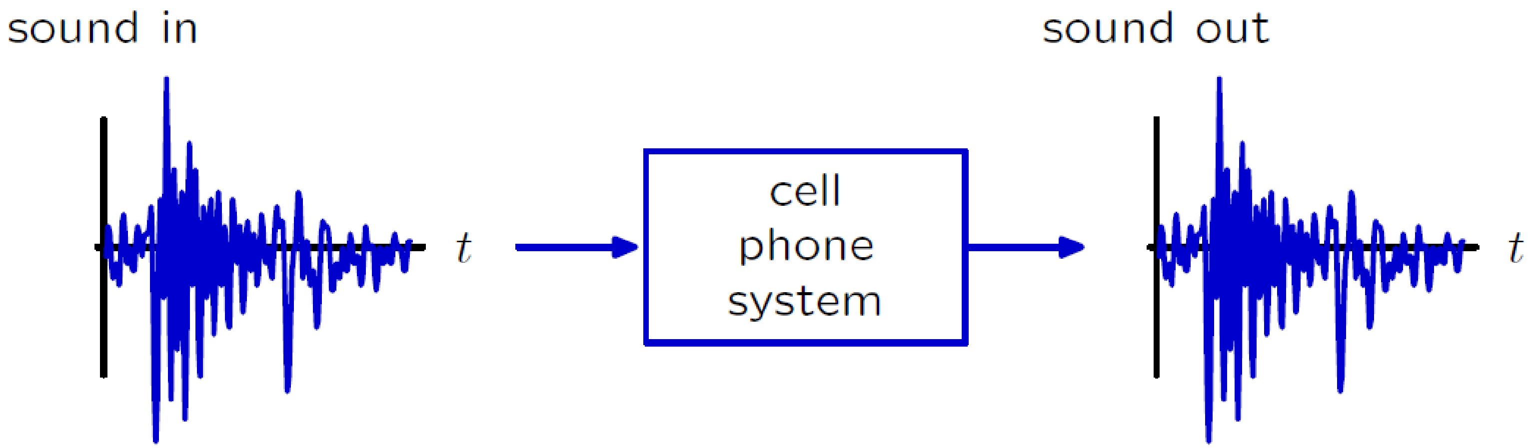
Example: Cell Phone System





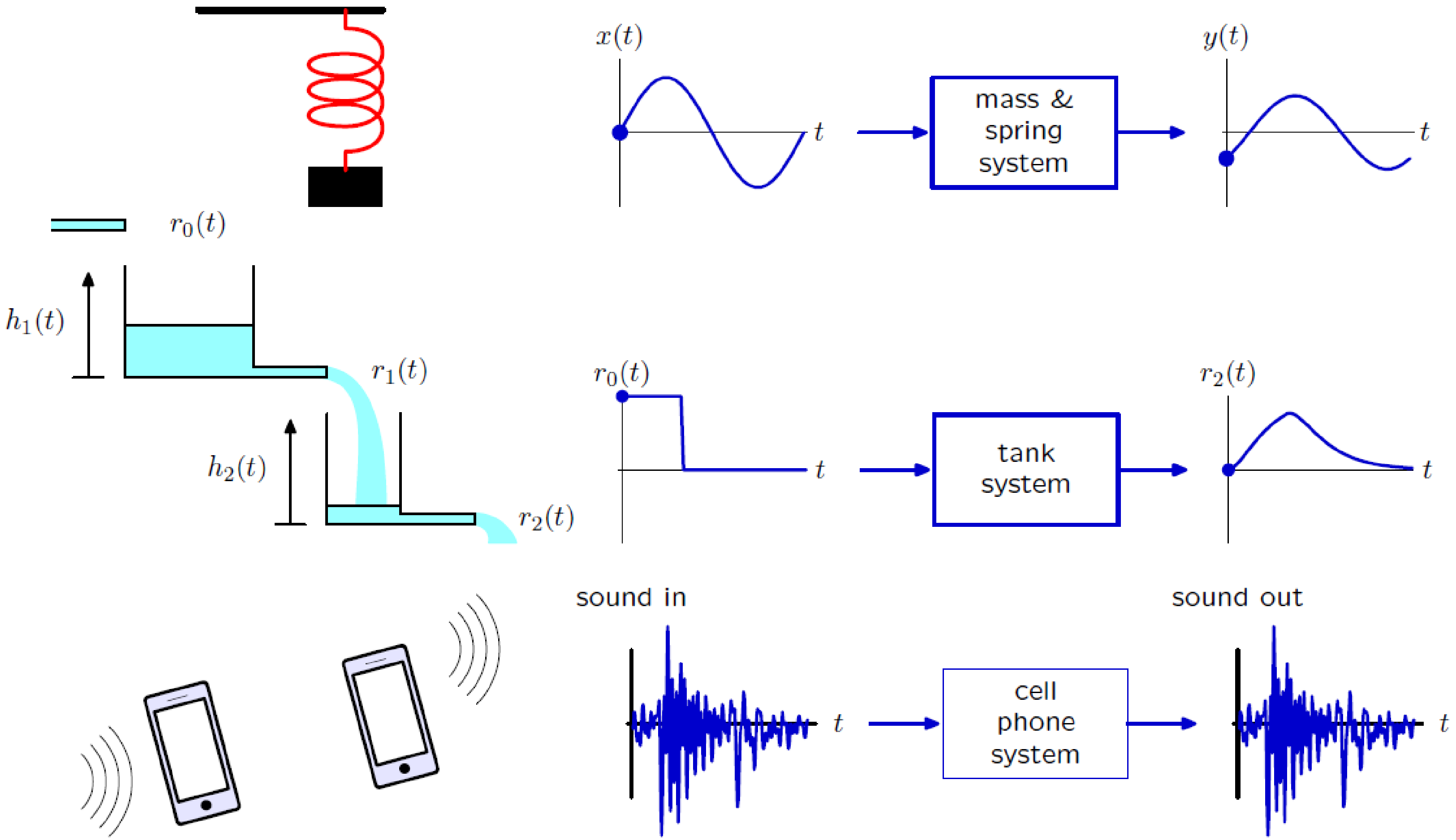
Example: Cell Phone System





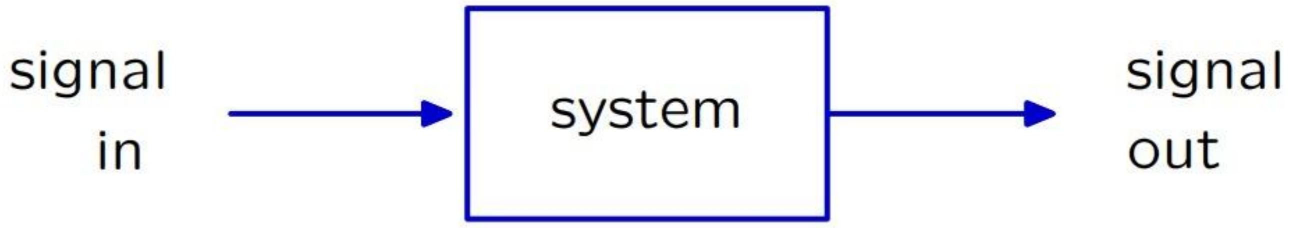
Signals and Systems: Widely Applicable

The Signals and Systems approach has broad applications: electrical, mechanical, optical, acoustic, biological, financial, ...



The System Abstraction

Many applications of signal processing can be formulated as systems that convert an input signal into an output signal.



Examples:

- **audio**: equalization, noise reduction, reverberation reduction, echo cancellation, pitch shift (auto-tune)
- **image:** smoothing, edge enhancement, unsharp masking, feature detection
- **video:** image stabilization, motion magnification

Example

Audio: Vocal removal

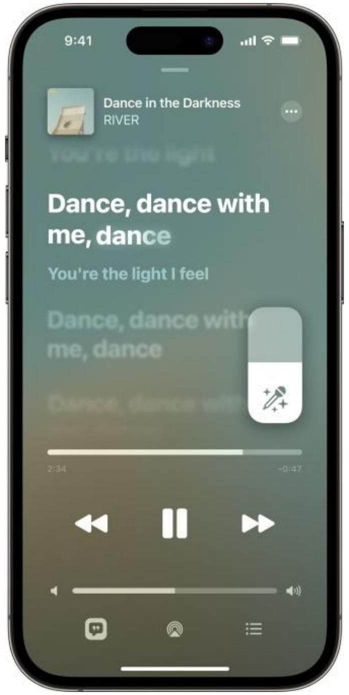
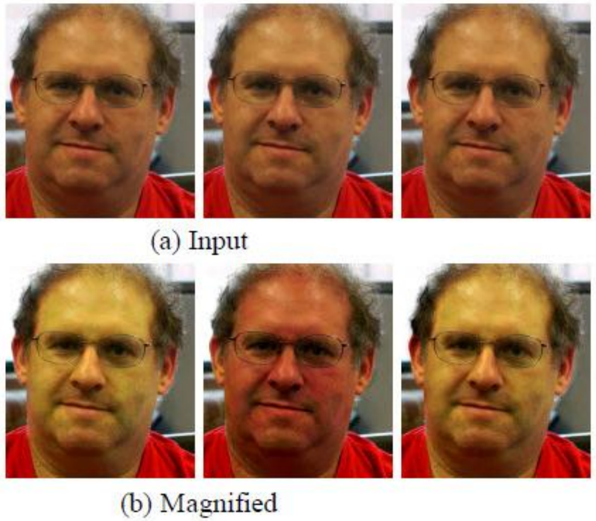


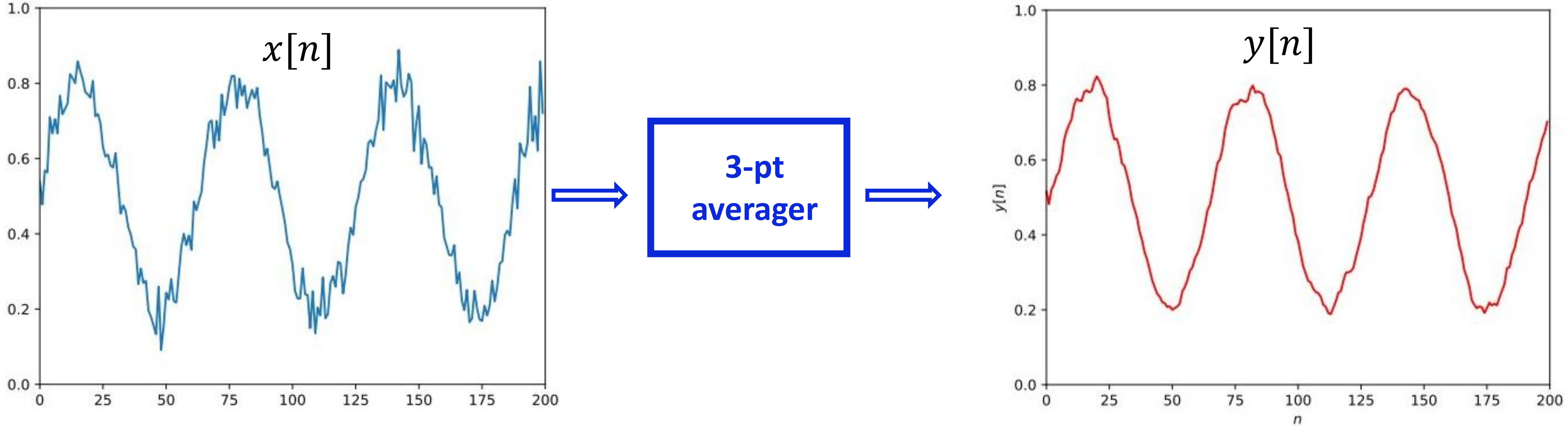
Image: Denoising

Video: Motion magnification



Example: Running Average

Noisy sensor data can be "smoothed" to reduce the impact of noise on the signal. For example, consider the following data on the left, consisting of a sinusoid corrupted with noise:



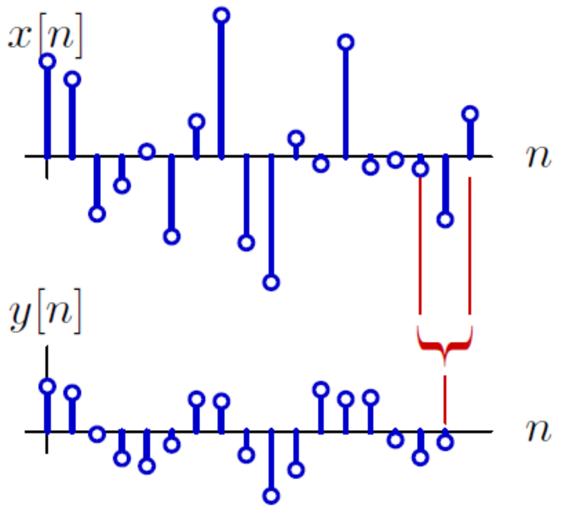
Consider the case where this signal is the input to a system described as "three point averager", whose output at time n is the average of three consecutive input samples:

$$y[n] = \frac{x[n-1] + x[n] + x[n+1]}{3}$$

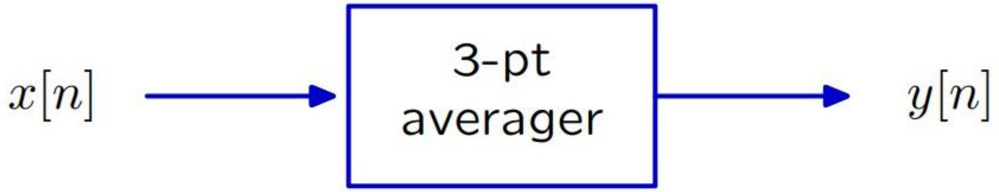
Example System: Three-point Averaging

The output at time *n* is average of inputs at times *n*-1, *n*, and *n*+1.

$$y[n] = \frac{1}{3}\Big(x[n-1] + x[n] + x[n+1]\Big)$$



Think of this process as a system with input $x[n]$ and output $y[n]$.



Linear, Time-Invariant(LTI) System

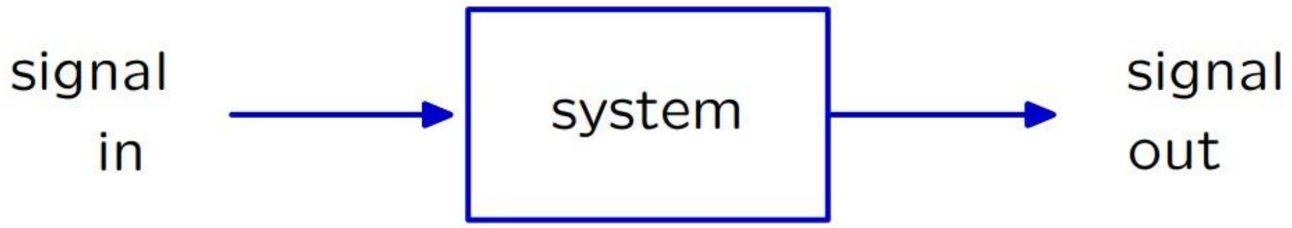
Arbitrary systems are arbitrarily difficult to describe.

Fortunately, many useful systems have two important properties:

- **Linearity** (additivity and homogeneity)
- **Time invariance**

In 6.300, we will focus on systems that have both of these properties, which are called **LTI systems**.

Multiple Representation of an LTI System



We can represent a system in the following three ways:

- Difference (differential) Equation: represent system by algebraic constraints on samples

$$y[n] = \frac{x[n-1] + x[n] + x[n+1]}{3}$$

- Convolution: represent a system by its unit-sample response
- Filter: represent a system by its amplification or attenuation of frequency components

Additivity

A system is additive if its response to a sum of inputs is equal to the sum of its responses to each input taken one at a time.

Given $$x_1[n] \longrightarrow \boxed{\text{system}} \longrightarrow y_1[n]$$

And $$x_2[n] \longrightarrow \boxed{\text{system}} \longrightarrow y_2[n]$$

the system is additive if

$$x_1[n] + x_2[n] \longrightarrow \boxed{\text{system}} \longrightarrow y_1[n] + y_2[n]$$

is true for all possible inputs.

Homogeneity

A system is homogeneous if multiplying its input by a constant multiplies its output by the same constant.

Given

$$x_1[n] \longrightarrow \boxed{\text{system}} \longrightarrow y_1[n]$$

the system is homogeneous if

$$\alpha x_1[n] \longrightarrow \boxed{\text{system}} \longrightarrow \alpha y_1[n]$$

is true for all α and all possible inputs.

Linearity

A system is linear if its response to a weighted sum of inputs is equal to the weighted sum (i.e. superposition) of its responses to each of the inputs.

Given $$x_1[n] \longrightarrow \boxed{\text{system}} \longrightarrow y_1[n]$$

And $$x_2[n] \longrightarrow \boxed{\text{system}} \longrightarrow y_2[n]$$

the system is linear if

$$\alpha x_1[n] + \beta x_2[n] \longrightarrow \boxed{\text{system}} \longrightarrow \alpha y_1[n] + \beta y_2[n]$$

is true for all α and β and all possible inputs.

Time-Invariance

A system is time-invariant if delaying the input to the system simply delays the output by the same amount of time.

Given

$$x[n] \longrightarrow \boxed{\text{system}} \longrightarrow y[n]$$

the system is time-invariant if

$$x[n-n_0] \longrightarrow \boxed{\text{system}} \longrightarrow y[n-n_0]$$

is true for all n_0 and for all possible inputs.

Check yourself (I)

Consider a system represented by the following difference equation:

$$y[n] = x[n] + x[n-1] \qquad \text{(for all n)}$$

is this system linear?

Check yourself (II)

Consider a system represented by the following difference equation:

$$y[n] = x[n] + 1 \qquad \text{(for all n)}$$

is this system linear?

Check yourself (III)

Consider a system represented by the following difference equation:

$$y[n] = x[n] \times x[n-1] \qquad \text{(for all n)}$$

is this system linear?

Participation question for Lecture

Check yourself (IV)

Consider a system represented by the following difference equation:

$$y[n] = nx[n] \qquad \text{(for all n)}$$

is this system linear?

Check yourself (V)

Consider a system represented by the following difference equation:

$$y[n] = nx[n] \qquad \text{(for all n)}$$

is this system time-invariant?

Represent a LTI system with Difference Equations

A system is linear and time-invariant if it can be expressed in terms of a linear difference equation with constant coefficients of the following form:

$$\sum_m C_m\, y[n-m] = \sum_k d_k\, x[n-k]$$

e.g. 3-pt averager:

$$y[n] = \frac{x[n-1] + x[n] + x[n+1]}{3}$$

Linearity: weighted sum of outputs equal to weighted sum of inputs

$$\sum_m C_m\,(\alpha y_1[n-m] + \beta y_2[n-m]) = \sum_k d_k \cdot \alpha x_1\,[n-k] + \sum_k d_k \cdot \beta x_2\,[n-k]$$

Time invariance: delaying an input delays its output

$$\sum_m C_m\, y[n-n_0-m] = \sum_k d_k\, x[n-n_0-k]$$

The three-point averager is a linear, time-invariant system, but the system $y[n] = x[n] + 1$ is not a LTI system.

Linear Differential Equations with Constant Coefficients

If a continuous-time system can be described by a linear differential equation with constant coefficients as follows, then the system is LTI.

$$\sum_l c_l \frac{d^l}{dt^l} y(t) = \sum_m d_m \frac{d^m}{dt^m} x(t)$$

Additivity: output of sum is sum of outputs.

$$\sum_l c_l \frac{d^l}{dt^l} (y_1(t) + y_2(t)) = \sum_m d_m \frac{d^m}{dt^m} (x_1(t) + x_2(t))$$

Homogeneity: scaling an input scales its output.

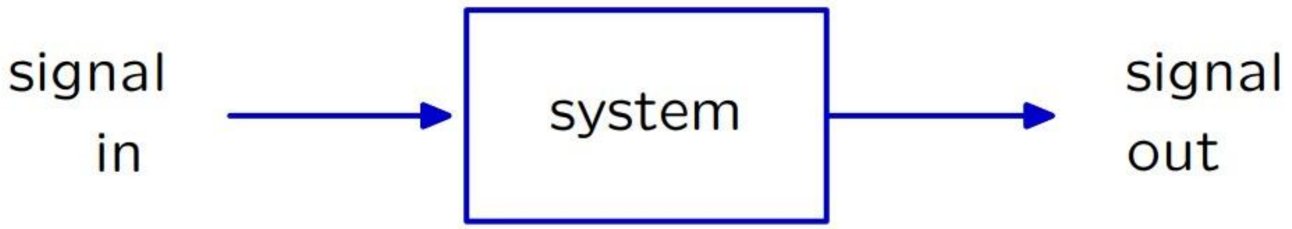
$$\sum_l c_l \frac{d^l}{dt^l} (\alpha y(t)) = \sum_m d_m \frac{d^m}{dt^m} (\alpha x(t))$$

Time invariance: delaying an input delays its output

$$\sum_l c_l \frac{d^l}{dt^l} y(t - t_0) = \sum_m d_m \frac{d^m}{dt^m} x(t - t_0)$$

Multiple Representation of LTI Systems

Next: Representing a system by its unit-sample response

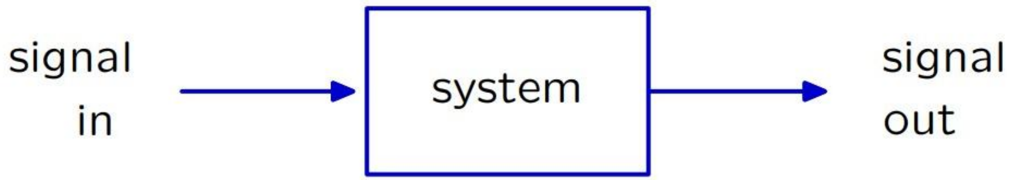


✓ Difference (differential) Equation: represent system by algebraic constraints on samples

- **Convolution: represent a system by its unit-sample response**
- Filter: represent a system as by its amplification or attenuation of frequency components

Summary

The concept of "system" to represent the process/method to manipulate signals:



Linear, Time-Invariant Systems

Three ways of representing a LTI system:

- Difference (differential) Equation: represent system by algebraic constraints on samples
- Convolution: represent a system by its unit-sample response
- Filter: represent a system as by its frequency response