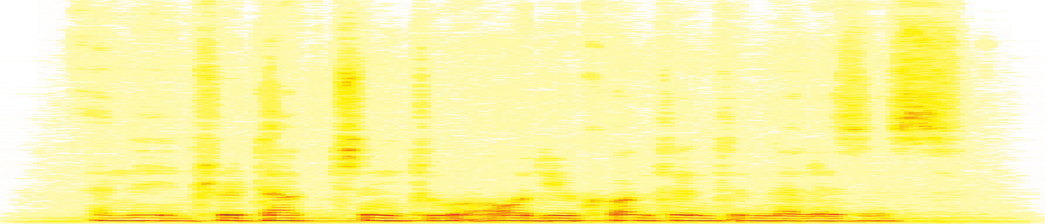
# Introduction to Audio Content Analysis

## Module 2.3: Fundamentals — Convolution

alexander lerch



Georgia Tech Center for Music Technology
College of Design

# introduction

## overview

**corresponding textbook section**

Chapter 2 — Fundamentals: pp. 14–18
Appendix A — Convolution Properties: pp. 181–183

- **lecture content**
  - LTI systems
  - convolution
  - filter examples
- **learning objectives**
  - basic understanding of linearity and time-invariance
  - basic understanding of the convolution operation
  - ability to implement simple filters

# introduction

## overview

**corresponding textbook section**

Chapter 2 — Fundamentals: pp. 14–18
Appendix A — Convolution Properties: pp. 181–183

- **lecture content**
  - LTI systems
  - convolution
  - filter examples

- **learning objectives**
  - basic understanding of linearity and time-invariance
  - basic understanding of the convolution operation
  - ability to implement simple filters

# systems

## introduction



a system:

- any process producing an output signal in response to an input signal

$$x(t) \longrightarrow \boxed{\text{System}} \longrightarrow y(t)$$

**name examples for systems in signal processing**

# systems
## introduction



a system:

- any process producing an output signal in response to an input signal

$$x(t) \longrightarrow \boxed{\text{System}} \longrightarrow y(t)$$

**name examples for systems in signal processing**

- filters, effects
- vocal tract
- room
- (audio) cable
- ...

# systems
## LTI systems



**LTI: Linear Time-Invariant Systems**

are a great model for many real-world systems

- **linearity**
  1. *homogeneity*: $f(ax) = af(x)$
  2. *superposition* (additivity): $f(x+y) = f(x) + f(y)$
- **time invariance**: $f\left(x(t-\tau)\right) = f(x)(t-\tau)$

# convolution
## introduction



### convolution

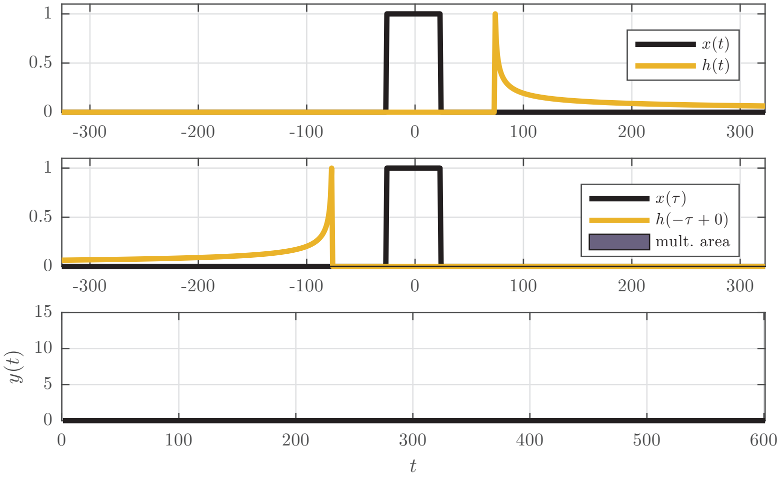
convolution operation describes the **output of an LTI system**:

$$y(t) = (x * h)(t) \quad := \quad \int\limits_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau$$

$$y(i) = (x * h)(i) \quad := \quad \sum_{j=-\infty}^{\infty} x(j)h(i-j)$$

# convolution

## animation




matlab source: matlab/animateConvolution.m

# convolution

## properties

- **identity**:

$$x(i) = \delta(i) * x(i)$$

- **commutativity**:

$$h(i) * x(i) = x(i) * h(i)$$

- **associativity**:

$$\big(g(i) * h(i)\big) * x(i) = g(i) * \big(h(i) * x(i)\big)$$

- **distributivity**:

$$g(i) * \big(h(i) + x(i)\big) = \big(g(i) * h(i)\big) + \big(g(i) * x(i)\big)$$

- **circularity**:

  $h(i)$ periodic $\Rightarrow y(i) = h(i) * x(i)$ periodic

# convolution
## properties

- **identity**:
$$x(i) = \delta(i) * x(i)$$
- **commutativity**:
$$h(i) * x(i) = x(i) * h(i)$$
- **associativity**:
$$(g(i) * h(i)) * x(i) = g(i) * (h(i) * x(i))$$
- **distributivity**:
$$g(i) * (h(i) + x(i)) = (g(i) * h(i)) + (g(i) * x(i))$$
- **circularity**:
$h(i)$ periodic $\Rightarrow y(i) = h(i) * x(i)$ periodic

# convolution

## properties

- **identity**:
$$x(i) = \delta(i) * x(i)$$
- **commutativity**:
$$h(i) * x(i) = x(i) * h(i)$$
- **associativity**:
$$\left(g(i) * h(i)\right) * x(i) = g(i) * \left(h(i) * x(i)\right)$$
- **distributivity**:
$$g(i) * \left(h(i) + x(i)\right) = \left(g(i) * h(i)\right) + \left(g(i) * x(i)\right)$$
- **circularity**:
$h(i)$ periodic $\Rightarrow y(i) = h(i) * x(i)$ periodic

# convolution
## properties

Georgia Tech Center for Music Technology College of Design

- **identity**:
$$x(i) = \delta(i) * x(i)$$
- **commutativity**:
$$h(i) * x(i) = x(i) * h(i)$$
- **associativity**:
$$\big(g(i) * h(i)\big) * x(i) = g(i) * \big(h(i) * x(i)\big)$$
- **distributivity**:
$$g(i) * \big(h(i) + x(i)\big) = \big(g(i) * h(i)\big) + \big(g(i) * x(i)\big)$$
- **circularity**:
$h(i)$ periodic $\Rightarrow y(i) = h(i) * x(i)$ periodic

# convolution

## properties

- **identity**:
$$x(i) = \delta(i) * x(i)$$
- **commutativity**:
$$h(i) * x(i) = x(i) * h(i)$$
- **associativity**:
$$\big(g(i) * h(i)\big) * x(i) = g(i) * \big(h(i) * x(i)\big)$$
- **distributivity**:
$$g(i) * \big(h(i) + x(i)\big) = \big(g(i) * h(i)\big) + \big(g(i) * x(i)\big)$$
- **circularity**:
$h(i)$ periodic $\Rightarrow y(i) = h(i) * x(i)$ periodic

# filter

## example 1: Moving Average

Georgia Tech Center for Music Technology College of Design

$$y(i) = \sum_{j=0}^{\mathcal{J}-1} b(j) \cdot x(i-j)$$

- replaces current sample with average of $\mathcal{J}$ samples
- smooths a signal (low pass)
- IR: rectangular
- linear phase, but inefficient for many coefficients
- Finite Impulse Response (FIR)

# filter

## example 2: Single-Pole



$$y(i) = (1-\alpha)\cdot x(i) + \alpha \cdot y(i-1)$$

- **recursive system**: output depends on previous *output*
- the larger alpha, the less the current input is taken into account (low pass)
- alpha from $0 \dots 1$
- efficient, but non-linear phase
- Infinite Impulse Response (IIR)

# summary
## lecture content

- **LTI system**
  - good model for many real-world system
  - linear (homogeneity, superposition) and time-invariant
  - impulse response reflects all system properties
- **convolution**
  - operation that computes the output of an LTI system from the input
- **low pass filters**
  - often used to smooth a signal
  - typical examples are moving average (FIR) and single pole (IIR)