Continuous and Discrete LTI Systems

Miguel Alfonso Mendez
von Karman Institute for Fluid Dynamics *

12 February 2020

This chapter reviews the fundamentals of continuous and discrete Linear Time-Invariant (LTI) systems with Single Input-Single Output (SISO). We start from the general notions of signals and systems, the signal representation problem and the related orthogonal bases in discrete and continuous forms. We then move to the key properties of LTI systems and discuss their eigenfunctions, the input-output relations in the time and frequency domains, the conformal mapping linking the continuous and the discrete formulations, and the modeling via differential and difference equations. Finally, we close with two important applications: (linear) models for time series analysis and forecasting and (linear) digital filters for multi-resolution analysis. This chapter contains seven exercises, the solution of which is provided in the book’s webpage1.

---

*mendez@vki.ac.be

1https://www.datadrivenfluidmechanics.com/download/book/chapter4.zip
Contents

1 Signals, Systems, Data and Modeling Fluid Flows 3

2 A note about notation and style 5

3 Signal and Orthogonal Bases 6
   3.1 Discrete Signals 6
   3.2 Continuous Signals 8

4 Convolutions and Eigenfunctions 10

5 Causal and Stable Systems 12

6 LTI Systems in their Eigenspace 13
   6.1 Laplace and Z Transforms 13
   6.2 Discrete and Continuous Frequencies 14
   6.3 The Convolution Theorem 15
   6.4 Differential and Difference Equations 16

7 Application I: Harmonic Analysis and Filters 18
   7.1 From Laplace to Fourier 19
   7.2 Multiresolution Analysis and Digital Filters 20

8 Application II: Time Series Analysis 24
   8.1 Stochastic LTI Systems 25
   8.2 Time Series Forecasting via LTI Systems 26

9 What’s next? 27
1 Signals, Systems, Data and Modeling Fluid Flows

A *signal* is any function that conveys information about a specific variable (e.g., velocity or pressure) of interest in our analysis. A signal can be a function of one or multiple variables, it can be continuous or discrete, and can have infinite or finite duration/extension (that is be non-null only within a range of its domain). Signals are produced by *systems*, usually in response to other signals or due to the interaction between different interconnected *subsystems*. In the most general setting, a system is any entity that is capable of manipulating an *input* signal and produce an *output* signal.

At such a level of abstraction, a vast range of problems in applied science falls within the framework of this chapter. For a fluid dynamicist, the flow past an airfoil is a system in which the inputs are the flow parameters (e.g., free stream-velocity and turbulence) and control parameters (e.g., the angle of attack), and the outputs are the drag and lift components of the aerodynamic force exchanged with the flow.

Any measurement chain is a system in which the input is the quantity *to be measured*, and the output is the quantity that *is measured*. A hot-wire anemometer, for example, is a complex system that measures the velocity of flow by measuring the heat loss by a wire which is heated by an electrical current. Any signal processing technique for denoising, smoothing and filtering is a system that takes in input the signal to be treated and outputs the signal with modified properties (e.g., with enhanced details or reduced noise).

Regardless of the number of subsystems composing a system, and whether the system is a physical system, a digital replica of it, or an algorithm in a computer program, the relations between input and output are governed by a *mathematical model*. The derivation of such models is instrumental for applications encompassing simulation, prediction or control. Models can be phrased with various degrees of sophistication, depending on the purposes for which these are developed. They usually take the form of Partial Differential Equations (PDEs), Ordinary Differential Equations (ODEs) or Difference Equations (DEs) or simple algebraic relation. Different models might have different ranges of *validity* (hence different degrees of generalization) and might involve different levels of complexity in their *validation*.

Models can be derived from two different routes, hinging on *data* and *experimentation* in various ways. The first route is that of *fundamental principles*, based on the division of a system into subsystems for which empirical observations have allowed to derive well established and validated relations. In the example of the hot-wire anemometer, the subsystems operate according to Newton’s cooling law for forced convection, the resistive heating governed by Joule’s law, the thermoelectric laws relating the wire resistance to its temperature, and the Ohm’s and Kirchhoff’s laws governing the electric circuits that are designed to indirectly measure the heat losses. Arguably, none of these laws were formulated with a hot-wire anemometer in mind. Yet, their range of validity is wide enough to accommodate *also* such an application: these laws generalize well.

In the example of the flow past an airfoil, the system is governed by Navier-Stokes equations, which incorporates other laws such as constitutive relations for the shear stresses (e.g., Newton’s law for a Newtonian fluid) and the heat fluxes (e.g., Fourier’s Law for conduction). These closure relations led to the notion of fluid properties such as dynamic viscosity and thermal conductivity and were also derived in simple experiments that did not target any specific application.
Because of their remarkable level of generalization, we tend to see these laws as simple mathematical representation of the ‘laws of nature’. Whether nature is susceptible to mathematical treatment is a question with deep philosophical aspects. As engineers, we accept the pragmatic view of relying on models and laws if these are validated and useful. This is the foundation of our scientific and technical background.

The second route is that of system identification or data-driven approach, based on the inference of a suitable model from a (usually large) set of input-output data. The model might be constrained to a certain parametric form (e.g., with a given order in the differential equations) or can be completely inferred from data. In the first case, we face a regression problem of identifying the parameters such that the model fits the data. In the second case, an algorithm proposes possible models (e.g., in Genetic Programming, see Chapter 14) or uses such a complex parametrization that an analytic form is not particularly interesting (e.g., in Artificial Neural Networks, see Chapter 3).

This paradigm is certainly not new. An excellent example of system identification is the method proposed by the Swedish Physicist Angström to measure thermal conductivity of a material in 1861 (Sundqvist, 1991). This method consists in using a long metal rod in which a harmonic heat wave is produced by periodically varying the temperature at one end. The spatial attenuation of the heat wave is modeled by the unsteady one-dimensional heat equation, leaving the thermal conductivity as an unknown. A simple formula can be derived for the harmonic response of the system, and the thermal conductivity is then identified by fitting the model to data.

Although the data-driven paradigm has a long history (Ljung, 2008), its capabilities have been significantly augmented and empowered by our increasing ability to generate enormous amounts of data and by the powerful tools popularized by the ongoing machine learning revolution. Some of these are described in Chapters 1 and 3. The big challenge (and the big opportunity) in big data is to combine physical modeling with the data-driven modeling. It is the author’s opinion that the combination of machine learning tools and the general framework of signals and systems can accommodate the formulation of many problems of applied science and, with the required caution, problems in fluid mechanics.

Caution is certainly required as the Navier-Stokes equations feature the entire spectra of complexities and challenges that a system analyst could think of. In most configuration of interest, fluid flows are nonlinear: a linear combination of input does not produce a predictable linear combination of outputs. It is thus not possible to predict the response of the system from a dictionary of known outputs. Fluid flows are often high dimensional and involve many scales: the amount of information required to identify the state of a flow system is tremendous. Fluid flows are often chaotic systems and are thus unpredictable: an infinitesimal change in initial or boundary conditions quickly yields very different instantaneous states of the system. Decades of fluid mechanics research aimed at models that are of reduced order, which include engineering closure laws (e.g., turbulence modeling) or treats the flows in terms of statistical quantities that are usually predictable.

While the Navier-Stokes equations are on the top of the ladder of complexity in the signal-system framework, this chapter treats systems that are at the bottom of the ladder: systems that are linear, single dimensional, deterministic and time invariant. These are a

2The harmonic response of a system is described in Section 7.
subclass of Linear Time (or Translation) Invariant systems (LTI). The interest in such a review is justified by three reasons. The first is that LTI systems are still often encountered in practice, both in physical systems\(^3\) and in most signal processing operations. The second is that many nonlinear systems can be treated reasonably well as linear systems if the range of operations is close enough fixed state; this is what lead to the remarkable success of many linear control methods described in Chapter 10. The third reason is that this chapter provides the background material to understand time-frequency analysis in Chapter 5 and the multi-resolution analysis underpinning the data decompositions in Chapter 8, the state-space models and the linear control theory in Chapter 10, and the system identification tools in Chapter 12.

Most of the material presented in this chapter can be found in classic textbooks on signal and systems Oppenheim et al. (1996); Ljung and Glad (1994); Hsu (2013), signal processing Williamson (1999); Ingle and Proakis (2011); Hayes (2011), orthogonal transforms Wang (2009), system identification Oppenheim (2015); Ljung (1999) or control theory Sigurd Skogestad (2005); Ogata (2009). We begin this chapter by introducing the relevant notation.

### 2 A note about notation and style

In their most general form, continuous and discrete LTI systems admit Multiple Inputs and respond with Multiple Outputs (MIMO systems) or have a Single Input and respond with a Single Output (SISO systems)\(^4\). This classification and the relevant notation is further illustrated in the block diagram in Figure 1.

In a continuous SISO system, input and outputs are denoted respectively as continuous functions \(u(t), y(t) \in \mathbb{R}\) with \(t \in \mathbb{R}\). Following the signal processing literature, for discrete systems these are denoted using an index notation, as \(u[k], y[k] \in \mathbb{R}\) with \(k \in \mathbb{Z}\). In this chapter, discrete and continuous signals are assumed to be linked by a sampling process, hence the time domain is discretized as \(t \rightarrow t_k = k\Delta t\) with an index \(k \in \mathbb{Z}\), sampling period \(\Delta t = 1/f_s\), and (constant) sampling frequency \(f_s\). Therefore, the notation \(u(t_k)\) is equivalent to the index notation \(u[k]\), but the second makes no link to the sampling process nor the time axis. In a MIMO system, both the inputs and the outputs are vectors \(u(t), u[k] \in \mathbb{R}^{n_I}\) and \(y(t), y[k] \in \mathbb{R}^{n_O}\), with \(n_I\) and \(n_O\) the number of inputs and outputs. MIMO systems are better treated in state-space representation presented in Chapter 10, hence this chapter only focuses on SISO systems.

---

\(^3\)see the previous example of Angström’s system.

\(^4\)Or can be a combination of the two as in MISO/SIMO systems.
Signals and systems can contain a deterministic and a stochastic part. While the focus is mostly on the deterministic part, the treatment of stochastic signals is briefly recalled in section 8.

3 Signal and Orthogonal Bases

For reasons that will become clear in Section 4, it is convenient to represent a signal in a way that allows decoupling the contribution of every time instance. In other words, we seek to define a signal with respect to a very localized basis that allows for sampling the signal at a given time. With such a (unitary) basis, the sampling process can be done by direct comparison: for example, we say that a continuous signal \( u(t) \) has \( u(3) = 2 \) because at time \( t = 3 \) this signal equals two times the element of the basis that is unitary (in a sense to be defined) at \( t = 3 \) and zero elsewhere. Mathematically, this ‘comparison’ process is a correlation, the signal processing equivalent of the inner product. This notion is more easily introduced for discrete signals, considered in 3.2. Continuous signal are treated in 3.1.

3.1 Discrete Signals

Consider a discrete signal of finite duration \( u[k] = [1, 3, 0, 4, 5]^T \), i.e. \( k \in [0, \ldots, 4] \) which can be arranged as a column vector \( u \in \mathbb{R}^{5 \times 1} \). In many applications, a finite duration signal is assumed to be a special case of an infinite duration signal which is zero outside the available points. This is a common practice referred to as zero-padding that will be further discussed in Chapter 8. This signal can be written as a linear combination of shifted unitary impulses:

\[
\begin{bmatrix}
1 \\
0 \\
0 \\
0 \\
\delta[l-0]
\end{bmatrix}
+ 3
\begin{bmatrix}
0 \\
1 \\
0 \\
0 \\
\delta[l-1]
\end{bmatrix}
+ 0
\begin{bmatrix}
0 \\
1 \\
0 \\
0 \\
\delta[l-2]
\end{bmatrix}
+ 4
\begin{bmatrix}
0 \\
0 \\
1 \\
0 \\
\delta[l-3]
\end{bmatrix}
+ 5
\begin{bmatrix}
0 \\
0 \\
0 \\
1 \\
\delta[l-4]
\end{bmatrix}
\]

The elementary basis to describe such a signal is thus the set:

\[
\delta_k = \delta[l - k] = \begin{cases} 
0 & \text{if } l \neq k \\
1 & \text{if } l = k 
\end{cases} \quad \text{and} \quad \sum_{l=-\infty}^{l=\infty} \delta[l - k] = 1 .
\]

Note that two notations are introduced. \( \delta_k \) is a vector of the same size of \( u \) which is zero expect at \( l = k \), where it is equal to one; \( \delta[l - k] \) is a sequence of numbers collecting the same information. In this sequence, \( k \) is the index spanning the position of the impulse while the index \( l \) spans the time (shift) domain. Infinite duration signals are vectors of infinite length. For two such signals \( a[k], b[k] \) or vectors \( a, b \), the inner product for the ‘comparison procedure’ is

\footnote{Note that we will here use a ‘Python-like’ indexing, that is starting from \( k = 0 \) rather than \( k = 1 \).}
\[ \langle a, b \rangle = b^\dagger a = \sum_{k=0}^{n-1} a[k] \bar{b}[k]. \]  

(3)

where \( b^\dagger = b^T \) is the Hermitian transpose, with the superscript \( T \) denoting transposition and the over-line denoting conjugation. With such a basis, the value of the signal at a specific location \( u[k] = u_k \) can be written as

\[ u[k] = \sum_{l=-\infty}^{\infty} u[l] \delta[k-l]. \]

(4)

The operation on the left is a correlation: for a given location of the impulse \( k \), the inner product is performed over the index \( l \) spanning the entire length of the signal and the result is a scalar – the signal’s value at the index \( k \). The operation on the right is a discrete convolution and the result is a signal: the entire set of shifts will be spanned. We shall return to the algebra of this operation in Chapter 8.

Note the flipping of the indices \( \delta[l-k] \) in (2) to \( \delta[k-l] \) in (4). In the first case, the location of the impulse \( k \) is fixed and \( l \) spans the vector entries; in the second case, within the summation, the time domain \( k \) is fixed and \( l \) spans the possible locations of the delta functions.

The inner product in a vector space is the fundamental operation that allows for a rigorous definition of intuitive geometrical notions such as the length of a vector and the angle between two vectors. The length (\( l^2 \) norm) of a vector \( ||a|| \) and the cosine of the angle \( \beta \) between two vectors \( a \) and \( b \) of equal size are defined respectively as

\[ ||a|| = \sqrt{\langle a, b \rangle} = \sqrt{b^\dagger a} \quad \text{and} \quad \cos(\beta) = \frac{\langle a, b \rangle}{||a|| ||b||}. \]

(5)

In signal processing, the first quantity is the root of the signal’s energy, defined as \( \mathcal{E}\{y\} = ||y||^2 \) while the second is the normalized correlation between two signals.

Signals with finite energy are square-summable. If \( \cos \beta = 0 \), two vectors are orthogonal and two signals are uncorrelated; if \( \cos \beta = 1 \) two vectors are aligned and two signals are perfectly correlated.

Notice that the projection of a vector \( a \) onto a vector \( b \) (see Figure 2) is

\[ a_b = ||a|| \cos(\beta) = \frac{\langle a, b \rangle}{||b||}. \]

(6)

Hence if \( b \) is a basis vector of unitary length, the inner product equals the projection. When this occurs, as in most of the

\footnote{This distinction is irrelevant for a symmetric function such as \( \delta \), but it is essential in the general case: if \( \delta[k-l] \) is replace by \( \delta[l-k] \) in (4), the operation is called cross-correlation.}

\footnote{Note that the notion of energy is used in signal processing for the square of a signal independently of whether this is actually linked to physical energy.}

\[ \text{Figure 2: Projections and inner products.} \]
cases presented in what follows, the notion of *inner product* and *projection* are used interchangeably.

The basis of shifted impulses has a very special property: it is *orthonormal*. This means that the inner product of two basis elements (in this case the shifted delta functions) gives zero unless the same basis element is considered, in which case we recover its unitary norm (energy). We return to this property in Chapter ??_. Before moving to continuous signals, it is worth introducing another important signal that is linked to delta functions, namely the Heaviside step function. This is defined as

\[ u_S[k-l] = \begin{cases} 1 & \text{if } k \geq l \\ 0 & \text{if } l < k \end{cases} \text{ i.e. } \rightarrow u_S[k-l] = \sum_{r=-\infty}^{k} \delta[r-l] \quad (7) \]

The difference between two shifted step functions generates a box function \( u_{B_{a,b}} \) which is unitary in the range \( k = [a, b-1] \) and is zero outside. The difference between two step functions shifted by a single step is a delta function, i.e. \( \delta[k-l] = u_S[k-l] - u_S[k-l-1] \). In discrete setting, this is equivalent to a differentiation. Hence, the delta function is the derivative of the step function and the summation in (7) shows that the step function is an integral of the delta functions. The reader should close this subsection with an exercise on some distinctive features of discrete signals.

**Exercise 1: Continuous vs Discrete**

Consider the continuous signal \( y(t) = e^{-0.2t} \cos(2\pi t) \) where \( t \) is in seconds. Assuming that the signal is sampled at \( f = 100 \text{Hz} \), and that \( n_t = 320 \) points are collected from \( t = 0 \), what is the resulting discrete signal? Plot this signal and comment on its link with the continuous one.

### 3.2 Continuous Signals

The extension of the previously introduced notion to continuous signals brings several complications, a detailed resolution of which is out of the scope of this chapter. Interested readers are referred to Kaiser (2010) for a gentle introduction. The main difficulty arise from the need to define an inner product space which can generalize (3) for functions while allowing for a basis of impulses, i.e. functions that are zero at all but one point. This generalization is provided by the Hilbert space, within which the inner product of two complex-valued functions \( a(t), b(t) \in \mathbb{C} \) is

\[ \langle a, b \rangle = \int_{-\infty}^{\infty} a(t) \bar{b}(t) \, dt. \quad (8) \]

The notion of energy (norm), correlation and projection in (5)-(6) extends to continuous signals using the inner product\(^8\) in (8). The generalization of (4) requires the definition of a continuous delta function \( \delta(t) \) with the same properties as \( \delta[k] \): it is nonzero only at a given time and it is absolutely integrable. This can be constructed as the limit of a suitably chosen function having unity area over an infinitesimal time interval. An example

\( ^8 \)Note that the upper case is used for the norms of a continuous function, i.e. \( ||a(t)||_2 = \sqrt{\langle a, a \rangle} \), with the inner product in (8), is the \( L^2 \) norm of the function \( a(t) \).
is a normalized Gaussian $G(t, \mu, \sigma)$ which has unitary integral regardless of its standard deviation $\sigma$:

$$
\int_{-\infty}^{\infty} G(t, \mu = 0, \sigma)dt = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} e^{-\frac{t^2}{2\sigma^2}} dt = 1.
$$

This function gets narrower and taller, as $\sigma \to 0$, to the point in which it becomes infinite at $t = 0$ and null everywhere else, while still having unitary area. This is the definition of continuous Dirac delta function, which in its shifted form is:

$$
\delta(t - \tau) = \begin{cases} 
0 & \text{if } t \neq \tau \\
\infty & \text{if } t = \tau
\end{cases}
$$

This function is not an ordinary one, as its integration poses several technical difficulties. Without entering into details of measure and distribution theory Richards and Youn (1990), we shall accept this as a generalized function that serves well our purpose of sampling a continuous signal. From the definition in (10), it is easy to derive the sifting (sampling property):

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

The equivalence $y(\tau) = y(t)$ in the integral results from the product $y(\tau) \delta(t - \tau)$ being null everywhere but at $t$; then, in the last step it is possible to move $y(t)$ outside the integral as this is independent from the integration domain $\tau$.

As for the discrete case, it is interesting to introduce the unitary step function and its link with the delta function as:

$$
u_S(t - t_0) = \begin{cases} 
1 & \text{if } t > t_0 \\
0 & \text{if } t < t_0
\end{cases}
$$
i.e. $\longrightarrow u_S(t - t_0) = \int_{-\infty}^{t} \delta(t - \tau) d\tau$

Notice that the step function is not defined at $t = t_0$. This definition is the continuous analogue of (7). To show that the delta function is the derivative of the step function we must introduce the notion of generalized derivative. For a continuous signal $u(t)$, denoting as $u'$ and $u^{(n)}$ its first and $n^{th}$ derivative, integration by part using an appropriate test function $\xi(t)$ gives

$$
\int_{-\infty}^{\infty} \xi(t) u^{(n)}(t) dt = (-1)^n \int_{-\infty}^{\infty} \xi^{(n)}(t) u(t) dt
$$

where the test function $\xi(t)$ is assumed to be continuous and differentiable at least $n$ times and is such that $\xi(t) \to 0$ for $t \to \pm\infty$. The first derivative of $u_S$ is

$$
\int_{-\infty}^{\infty} \xi(t) u_S(t - t_0) dt = -\int_{-\infty}^{\infty} \xi'(t) u_S(t - t_0) dt = -\int_{t_0}^{\infty} \xi'(t) = 
$$

$$
= \xi(t_0) - \xi(\infty) = \xi(t_0) = \int_{-\infty}^{\infty} \xi(t) \delta(t - t_0) dt
$$
The first equality results from direct application of (13); the second from \( u_S(t - t_0) \) being zero in \( t < t_0 \). After integration, the sifting property of the delta function (11) is used, and the result is obtained by equivalence of last step with the first, holding on the fact that \( \xi(t) \) is arbitrary.

4 Convolutions and Eigenfunctions

In a linear system, the input-output relation satisfy Homogeneity and Superposition. For continuous \( (S_c\{u(t)\} = y(t)) \) and discrete \( (S_d\{u[k]\} = y[k]) \) systems, these set

\[
\begin{align*}
\text{Homegeneity:} & \quad S_c\{au(t)\} = aS_c\{u(t)\} = ay(t) \quad \forall a \in \mathbb{C} \\
\quad S_d\{au[k]\} = aS_d\{u[k]\} = ay[k] \quad \forall a \in \mathbb{C}
\end{align*}
\] (15)

\[
\begin{align*}
\text{Superposition:} & \quad S_c\{\int_{-\infty}^{\infty} u(t)\,dt\} = \int_{-\infty}^{\infty} S_d\{u(t)\}\,dt = \int_{-\infty}^{\infty} y(t)\,dt \\
\quad S_d\{\sum_{n=1}^{N} u_n[k]\} = \sum_{n=1}^{N} S_d\{u_n[k]\} = \sum_{n=1}^{N} y_n[k]
\end{align*}
\] (16)

where we considered a finite summation of \( N \) inputs for the discrete case and an infinite summation of infinitesimally close inputs for the continuous one. Combining these properties, we see that a linear combination of inputs results in the same linear combination of outputs:

\[
\begin{align*}
\text{Superposition:} & \quad S_c\{\int_{-\infty}^{\infty} a(\tau)u(t,\tau)\,d\tau\} = \int_{-\infty}^{\infty} a(\tau)S_d\{u(t,\tau)\}\,d\tau = \int_{-\infty}^{\infty} a(\tau)y(t,\tau)\,d\tau \\
\quad S_d\{\sum_{n=1}^{N} a_n u_n[k]\} = \sum_{n=1}^{N} a_n S_d\{u_n[k]\} = \sum_{n=1}^{N} a_n y_n[k]
\end{align*}
\] (17)

A system is time-invariant (or translation-invariant, if time is replaced by space) if the response to the input does not change over time (or space), i.e.:

\[
\begin{align*}
\text{if} & \quad S_c\{u(t)\} = y(t) \quad \text{then} \quad S_c\{u(t - t_0)\} = y(t - t_0) \quad \forall t_0 \in \mathbb{R} \\
\text{if} & \quad S_d\{u[k]\} = y[k] \quad \text{then} \quad S_d\{u[k - k_0]\} = y[k - k_0] \quad \forall k_0 \in \mathbb{Z}
\end{align*}
\] (18)

A system is Linear Time/Translation Invariant (LTI) if it is both linear and time/-translation invariant. These two properties, combined with the signal representations in (10)-(11) make the analysis of LTI systems particularly simple: the response to any input can be fully characterized from the response to a single impulse.

Defining as \( h(t) = S_c\{\delta(t)\} \) the impulse response of a continuous system, the response to any input is
\[ y(t) = S_c\{u(t)\} = S_c\left\{ \int_{-\infty}^{\infty} u(t) \delta(t-\tau) d\tau \right\} = \int_{-\infty}^{\infty} u(t) S_c\{\delta(t-\tau)\} d\tau \]

\[ \rightarrow y(t) = \int_{-\infty}^{\infty} u(\tau) h(t-\tau) d\tau = \int_{-\infty}^{\infty} h(\tau) u(t-\tau) d\tau \]  

(19)

The last two integrals are equivalent forms of the convolution integral, hinging on its commutative property. Similarly, defining \( h[k] = S_d\{\delta[k]\} \) the impulse responses of a discrete system, the response to any input is

\[ y[k] = S_d\{u[k]\} = S_d\left\{ \sum_{l=-\infty}^{\infty} u[k] \delta[k-l] \right\} = \sum_{l=-\infty}^{\infty} u[k] S_d\{\delta[k-l]\} \]

\[ \rightarrow y[k] = \sum_{l=-\infty}^{\infty} u[l] h[k-l] = \sum_{l=-\infty}^{\infty} h[k] u[k-l] \]  

(20)

having introduced the discrete convolution and its commutative property.

Note that the system response obtained via (19) or (20) is independent from the initial state of a system. Such response is thus a particular solution, i.e. after the transitory from the initial condition vanishes.

A special case is produced when the input is an exponential of the form \( u(t) = e^{st} \) for continuous systems and \( u[k] = z^k \) for the discrete ones, with \( s, z \in \mathbb{C} \). In what follows, we write \( s \) in a Cartesian form as \( s = \sigma + j\omega \) and \( z \) in a polar form as \( z = \rho e^{j\theta} \); the convenience in this is evident in Section 7. For the moment, note that these two variables are the continuous and discrete complex frequencies.

In continuous systems, from (19), the output is

\[ y(t) = S_c\{e^{st}\} = \int_{-\infty}^{\infty} h(\tau) e^{-(t-\tau)s} d\tau = \left( \int_{-\infty}^{\infty} h(\tau) e^{-\tau s} d\tau \right) e^{st} = \lambda_c(s) e^{st} \]  

(21)

In discrete systems, from in (20), the output is

\[ y[k] = S_d\{z^k\} = \sum_{l=-\infty}^{\infty} h[l] z^{k-l} = \left( \sum_{l=-\infty}^{\infty} h[l] z^{-l} \right) z^k = \lambda_d(z) z^k \]  

(22)

In both cases, this result shows that LTI system responds to a complex exponential with the same input multiplied by a complex number (\( \lambda_d \) or \( \lambda_c \)). This number solely depends on the complex frequencies \( z \in \mathbb{C} \) and \( s \in \mathbb{C} \). Therefore, these special inputs functions are eigenfunctions of the LTI operators and their complex eigenvalues are:

\[ \lambda_c(s) = H(s) = \int_{-\infty}^{\infty} h(\tau) e^{-\tau s} d\tau \]  

(23a)

\[ \lambda_d(z) = H(z) = \sum_{l=-\infty}^{\infty} h[l] z^{-l} \]  

(23b)
These are respectively the Laplace transform and the Z-Transform of the impulse response. These are the transfer functions of the LTI systems and link input and output in the complex frequency domain. For time varying or nonlinear systems, the notion of transfer function is not useful. Finally, note that in discrete systems, the transfer function is a continuous function of \( z \).

### Exercise 2: Convolution with Impulse Responses

To identify a SISO system, a step function test is carried out. A regression analysis reveals that the response of the system to a step \( u_S \) is well described by the function

\[
y_S = \frac{1}{5} - \frac{1}{5} e^{-t} \left( \cos(2t) + \frac{1}{2} \sin(2t) \right)
\]  

Compute the response of the system to an exponential input \( u(t) = e^{-t} \) in the continuous and the discrete domains, assuming that the system can be sampled at \( f_s = 10 \) Hz or at \( f_s = 3 \) Hz for \( T = 5 \) s.

### 5 Causal and Stable Systems

In most applications of interest, signals are assumed to be null at time \( t < 0 \). This is important for the impulse response of systems that are causal, in which the impulse response is \( h(t) = 0 \) for \( t < 0 \) and \( h[k] = 0 \) for \( k < 0 \). This means that no output can be produced before the input, and hence the system is not anticipatory. Models of physical systems and online data processing must be causal. On the other hand, many data processing schemes operating offline are not causal (e.g., zero-phase filters described in section 8).

As anticipated in the previous exercise, the convolution integral and summations in (19)-(20) for causal systems become

\[
y(t) = S_c\{u(t)\} = \int_0^t h(\tau) u(t - \tau)d\tau = \int_0^t u(\tau) h(t - \tau)d\tau
\]

\[
y[k] = S_d\{u[k]\} = \sum_{l=0}^k h[l] u[k - l] = \sum_{l=0}^k u[l] h[k - l].
\]

The upper limit is replaced by \( t \) or \( k \) since signals and impulse responses are null for \( \tau > t \) or \( k > l \) and the lower one is replaced by 0 since both are null for \( t < 0 \) or \( l < 0 \). The continuous and discrete transfer functions in (23) become

\[
\lambda_c(s) = H(s) = \int_0^\infty h(\tau) e^{-\tau s}d\tau \quad \text{and} \quad \lambda_d(z) = H(z) = \sum_{n=0}^\infty h[n] z^{-n}
\]  

In the continuous case, \( t = 0 \) must be included in the integration; hence the lower bound is tuned to accommodate for any peculiarity occurring at \( t = 0 \) (notably an impulse). Nevertheless, to avoid the extra notational burden, the minus subscript in the lower bound is dropped in what follows.
Finally, another important class of interest is that of stable systems. The stability analysis of complex systems is a broad topic (see Chapters 10 and 13). Here, we limit the focus to bounded-input/bounded output (BIBO) stability. A system is BIBO stable if its response to any bounded input is a bounded output. This requires that the impulse response of continuous and discrete signals satisfy:

$$
\int_{0}^{\infty} |h(t)|dt < \infty \quad \text{and} \quad \sum_{k=0}^{\infty} |h[k]| < \infty. \quad (27)
$$

A different notion is that of asymptotic stability, which is related to the internal stability of a system. A system is asymptotically stable if every initial state, in the absence of inputs, produces a bounded response that converges to zero. In an LTI system, asymptotic stability implies BIBO stability, but the reverse is not true.

6 LTI Systems in their Eigenspace

In the previous section, we have seen that complex exponentials are eigenfunctions of LTI systems. Great insights on a system behavior can be obtained by projecting their input-output relation onto the system’s eigenfunctions. The projection of signals into complex exponentials leads to the Laplace transform in the continuous domain and the Z transform in the discrete domain. This section is divided into four subsections. We start with some definitions.

6.1 Laplace and Z Transforms

**Laplace Transforms.** Given a continuous signal and causal signal $u(t)$, the Laplace transforms are

$$
U(s) = \mathcal{L}\{u(t)\} = \int_{-\infty}^{\infty} u(t)e^{-st}dt = \int_{0}^{\infty} u(t)e^{-st}dt. \quad (28)
$$

The first integral is the bilateral transform; the second is the unilateral transform. As we here focus on causal signals (i.e. $u(t < 0) = 0$), these are identical. Nevertheless, these are different tools required for different purposes: the first is suitable for infinite duration signals, for which it can be linked to the Fourier Transform (Section 7); the second is developed for solving initial value problems, as it naturally handles initial conditions.

These integral converge, and hence the Laplace transforms exist, if $u(t)e^{-st} \to 0$ for $t \to \pm\infty$ (only $t \to +\infty$ for the unilateral). This requires that the signal is of exponential order, i.e. grows more slowly than a multiple of some exponential: $|u(t)| \leq Me^{\alpha t}$. If this is the case, the range of values $\mathbb{R}\{s\} > \alpha$ is the region of convergence (ROC) of the transform.

The reader is referred to Beerends et al. (2003), Wang (2009), Hsu (2013) for a review of all the properties of the Laplace transform; we here focus on the key operations enabled by this powerful tool and we omit formulation of the inverse Laplace transform, as it requires notions of complex variables theory that are out of the scope of this chapter.
The Python script Ex3.py to solve Exercise 3 provides the commands to compute both the transform and its inverse using the Python library SymPy.9

The key property of interest in this chapter is that of time derivation, which can be easily demonstrated using integration by parts. The bilateral ($\mathcal{L}^b$) and unilateral ($\mathcal{L}^u$) transforms of a time derivative are

$$
\mathcal{L}^b\{u'(t)\} = sU(s) \quad \mathcal{L}^u\{u'(t)\} = sU(s) - u(0). \tag{29}
$$

That is differentiation in the time domain corresponds to multiplication by $s$ in the frequency domain; similarly, one can show that integration in the time domain corresponds to division by $s$. Notice that no distinction between $\mathcal{L}^b$ and $\mathcal{L}^u$ is needed for a system initially at rest and the bilateral transform cannot handle initial conditions.

Finally, compare the inner product in (28) with (8), taking $a(t) = u(t)$. Note that the Laplace transform is a projection of the signal $u(t)$ onto an exponential basis $b_L(t, s) = e^{-\sigma+\omega}$. It is left as an exercise to show that the Laplace basis is not orthogonal, unless $\sigma = 0$. That is the basis of the continuous Fourier transform.

**Z Transform.** Given a discrete and causal signal $u[k]$, the Z transforms are:

$$
U(z) = \mathcal{Z}\{u[k]\} = \sum_{k=-\infty}^{+\infty} u[k]z^{-k} = \sum_{k=0}^{\infty} u[k]z^{-k}. \tag{30}
$$

The same distinction on bilateral or unilateral transforms, as well as their equivalence for causal signals, apply to the discrete transform. Observe that the Z transform is a continuous function of $z$. These summation converge, and hence the transforms exists, if $|u[k]z^{-k}| < 1$. The domain within which this occurs is the region of convergence (ROC) of the transforms, and this is typically within a domain such that $|z| > \alpha$. As in the previous section, we avoid a review of all the properties of this transform, and the formulation of its inverse (see Wang (2009); Hayes (2011)).

The Z-transform equivalent of the time differentiation and integration properties of the Laplace transform are the time-shifting properties:

$$
\mathcal{Z}\{u[k - 1]\} = z^{-1} U(z) \quad \text{and} \quad \mathcal{Z}\{u[k + 1]\} = z U(z). \tag{31}
$$

Finally, compare (30) with (3) taking $a[k] = u[k]$ to see that the Z-transform is a projection of the signal onto a basis of powers $b_Z[k] = \rho^{-k}e^{j\theta k}$. This basis is not orthogonal unless $\rho = 1$. That is the basis of the discrete Fourier transform.

The Z-transform is not suited for compression purposes, as the dimensionality of the problem in the transformed domain is increased: the original signal has a (finite) dimension $t_k$, but the same information in the frequency domain is mapped onto a continuous complex plane $z$.

### 6.2 Discrete and Continuous Frequencies

If a discrete signal is obtained by sampling a continuous one, the link between these two transforms reveals the important impact of the sampling process on the frequency domain.

---

*9see [https://www.sympy.org/en/index.html](https://www.sympy.org/en/index.html).*
The discretization \( t_k = k\Delta t \) creates a point-wise equivalence such that any signal (causal or not) can be equivalently written as

\[
u[k] = \sum_{l=-\infty}^{\infty} u[l]\delta[k-l] \Longleftrightarrow u(t_k) = \sum_{l=-\infty}^{\infty} u(t_l)\delta(t_k - l\Delta t).
\]

(32)

With \( e^{-st_k} = e^{-sk\Delta t} = z^k \), the Laplace transform of this discrete signal is

\[
\mathcal{L}\{u[k]\} = \int_{-\infty}^{\infty} \sum_{l=-\infty}^{\infty} u[l]\delta(t - l\Delta t)e^{-st} dt = \int_{-\infty}^{\infty} \sum_{l=-\infty}^{\infty} u[l]\delta(t - l\Delta t)z^{-l} dt = \sum_{l=-\infty}^{\infty} u[l]z^{-l} \int_{-\infty}^{\infty} \delta(t - l\Delta t) dt = Z\{u[k]\}
\]

(33)

The equivalence of these transforms relies on the change of variables \( z = e^{s\Delta t} \). This maps \( s \) onto the \( z \) while preserving angles in the two domains: such mapping is a conformal mapping. Introducing \( s = \sigma + j\omega \) and \( z = \rho e^{j\theta} \), shows that \( \rho = e^{\sigma\Delta t} \) and \( \theta = \omega\Delta t \).

Figure 3 shows several important features of this mapping. Observe that the imaginary axis \( \omega \) in the continuous domain is mapped onto an angular coordinate \( \theta \in [-\pi, \pi] \) or \( \theta \in [0, 2\pi] \). This offers yet another way of introducing the notion of aliasing and the Nyquist-Shannon sampling theorem (encountered in the exercise 1), arising from the fact that the frequency domain of a digital signal is periodic. Another important observation is that the left side of the complex domain \( s = \sigma < 0 \) is mapped inside the unit circle \( \rho = 1 \), while the axis \( s = \sigma = 0 \) is mapped on the unit circle.

![Figure 3: The conformal mapping linking the Laplace Transform and the Z transform.](image)

6.3 The Convolution Theorem

While the convolution integral and summations in (19) and (20) provide the input-output relation in the time domain using the impulse response, further insights can be obtained.
by analyzing this relation in the frequency domain. We here give, without proofs\(^\text{10}\), one of the most important results of signal processing, known as the convolution theorem: \textit{a convolution in the time domain is a multiplication in the frequency domain}. For continuous and discrete systems, this means:

\[
Y(s) := \mathcal{L}\{y(t)\} = \mathcal{L}\left\{\int_0^t h(\tau) u(t - \tau) d\tau\right\} = H(s) U(s)
\] (34a)

\[
Y(k) := \mathcal{Z}\{y[k]\} = \mathcal{Z}\left\{\sum_{l=0}^{k} h[l] u[k - l]\right\} = H(z) U(z).
\] (34b)

### 6.4 Differential and Difference Equations

Continuous LTI systems can be described in terms of (linear) differential equations while discrete LTI systems can be described in terms of (linear) difference equations. The Laplace and the Z-transform are powerful tools to solve these equations because of the properties in (29) and (31). In both cases, the equations have constant coefficients and are often acronymized as LCCDE.

**Differential Equations.** The general form of the LCCDE of a continuous SISO LTI systems with input \(u(t)\) and output \(y(t)\) reads

\[
\sum_{n=0}^{N_b} a_n y^{(n)}(t) = \sum_{n=0}^{N_f} b_n u^{(n)}(t)
\] (35)

where \(N_f \geq N_b\) is the order of the system\(^\text{11}\). The coefficients \(a_n\) are called \textit{feedback coefficients}; the coefficients \(b_n\) are \textit{feedforward coefficients}.

The LCCDE provides an \textit{implicit} representation of a system since the input-output relation can be revealed only by solving the equation. Introducing the Laplace transform in a LCCDE is an operation similar to the Galerkin projection underpinning Reduced Order Modeling (ROM, see Chapters 1 and 14). Recalling that the Laplace transform is a projection onto the basis \(b_L(t)\), (35) leads to:

\[
\left\langle \sum_{n=0}^{N_b} a_n y^{(n)}(t), b_L(t) \right\rangle = \left\langle \sum_{n=0}^{N_f} b_n u^{(n)}(t), b_L(t) \right\rangle \rightarrow \sum_{n=0}^{N_b} a_n \mathcal{L}\{y^{(n)}(t)\} = \sum_{n=0}^{N_f} b_n \mathcal{L}\{u^{(n)}(t)\} \rightarrow Y(s) \left(\sum_{n=0}^{N_b} a_n s^{(n)}\right) = U(s) \left(\sum_{n=0}^{N_f} b_n s^{(n)}\right) \rightarrow
\]

\[
H(s) = \frac{Y(s)}{U(s)} = \frac{\sum_{n=0}^{N_f} b_n s^{(n)}}{\sum_{n=0}^{N_b} a_n s^{(n)}} = \frac{b_{N_f}}{a_{N_b}} \frac{\prod_{n=0}^{N_f} (s - z_n)}{\prod_{n=0}^{N_b} (s - p_n)}.
\]

\(^{10}\text{See Wang (2009) for more details.}\)

\(^{11}\text{The condition } N_f \geq N_b \text{ is necessary to ensure that the system is \textit{realizable}, that is both stable and causal. More about this in the next footnote.}\)
The transfer function of LTI systems is a polynomial rational function of $s$, with the coefficients of the polynomials being the coefficients of the LCCDE. In the factorized form, $z_n$ and $p_n$ are respectively the zeros and the poles of the system. Note that since the coefficients $a_n$, $b_n$ are real, these can either be purely real or appear in complex conjugate pairs. These coefficients have a straightforward connection with the LCCDE, which can immediately be recovered from the transfer function. The zeros $z_n$ are associated to inputs $e^{z_n t}$ in which the transfer function is null and thus leads to no output; the poles $p_n$ correspond to resonances, inputs $e^{p_n t}$ in which the transfer function is infinite and leads to the blow-up of the system.

The poles are eigenvalues of the matrix $A$, advancing a linear system in its state-space representation (see Chapters 10 and 12). In a stable system, poles are located in regions of the $s$-plane that are ‘not-accessible’ by any input, that is outside the ROC of the transfer function. Defining the ROC of $H(s)$ as $\mathbb{R}\{s\} > \alpha$, and observing that the poles are by definition outside the ROC, stability is guaranteed if $\alpha = 0$, i.e. if the ROC includes the imaginary axis. This is equivalent to impose that all the poles are located in the left side of the s-plane, i.e. $\mathbb{R}\{p_n\} < 0 \forall n \in [0, N_f]$. This can also be derived from the BIBO stability condition in (27).

**Difference Equations.** In the discrete case, the general form of LCCDE associated to SISO LTI systems with input $u[t]$ and output $y[t]$ reads

$$
\sum_{n=0}^{N_b} a_n y[k - n] = \sum_{n=0}^{N_f} b_n u[k - n], \text{ i.e. } y[k] = \sum_{n=0}^{N_b} b^*_n u[k - n] - \sum_{n=1}^{N_f} a^*_n y[k - n].
$$

The order of the system is $\max(N_b, N_f)$. The form on the right plays a fundamental role in filter implementation, time series analysis and system identification and is known as recursive form of the difference equation. Note that the feedback and feedforward coefficients in the recursive form are simply $a^*_n = a_n/a_0$ and $b^*_n = b_n/a_0$ respectively, hence the coefficient $b^*_0$ is the static gain of the system.

As for the continuous case, projecting (37) onto the $Z$ basis $b_Z(t)$ via $Z$ transform and using (31), yield the transfer function of a discrete system:

---

12 A transfer function that has more zeros than poles (i.e. $N_f > N_b$) is said to be improper. In this case, $\lim_{s \to \infty} |H(s)| = +\infty$, which violates stability: this implies that at large frequencies, a finite input can produce an infinite output. Moreover, after the polynomial division, the transfer function brings polynomial terms in $s$. The inverse Laplace transform of these are (generalized) derivatives of the delta functions; hence the corresponding impulse response $h(t) = \mathcal{L}^{-1}\{H(s)\}$ violates causality.

13 Note that in (37) the restriction $N_b \geq N_f$ is not needed to enforce causality: by construction, the output $y[k]$ only depends on past information.
\begin{equation}
\left\langle \sum_{n=0}^{N_b} a_n y[k-n], b z[k] \right\rangle = \left\langle \sum_{n=0}^{N_f} b_n u[k-n], b z[k] \right\rangle \rightarrow \sum_{n=0}^{N_b} a_n z \left\{ y[k-n] \right\} = \sum_{n=0}^{N_f} b_n z \left\{ u[k-n] \right\} \rightarrow Y(z) \left( \sum_{n=0}^{N_b} a_n z^{-n} \right) = U(z) \left( \sum_{n=0}^{N_f} b_n z^{-n} \right) \rightarrow
\end{equation}

where \( \zeta_n \) and \( \pi_n \) are respectively the zeros and poles of the discrete transfer function. Observe that the factored form of the discrete transfer function is usually given in terms of polynomials of \( z^{-1} \) rather than \( z \).

The link between zero and poles in continuous and discrete domains is given by the conformal mapping in Figure 3. In the absence of inputs, the poles control the evolution of a linear system from its initial condition (i.e. the homogeneous solution of the LCCDE). The Dynamic Mode Decomposition (DMD) introduced in Chapter 7 is a powerful tool to identify the poles \( \pi_n \) of a system from data, and to build linear Reduced-Order Models by projecting the data onto the basis of eigenfunctions \( z^{\pi_n k} \).

Finally, in analogy with the continuous case, a discrete system is stable if its poles are outside the ROC of the transfer function. Defining the ROC of \( H(z) \) as \( |z| \geq \alpha \), one sees that this occurs if \( \alpha = 1 \): the ROC include the unit circle and hence all the poles have \( |\pi_n| < 1 \). This can be derived from the BIBO stability condition in (30).

**Exercise 3: Transfer Function Analysis**

Consider the system in Exercise 2. Compute the transfer function and the system output from the frequency domain, then identify the LCCDE governing the input-output relation. Then, assuming that a discrete system is obtained sampling the continuous domain, derive a recursive formula that mimics the input-output link of the continuous system. Test your result for a sampling frequency of \( f_s = 3Hz \) and \( f_s = 10Hz \).

### 7 Application I: Harmonic Analysis and Filters

BIBO stability guarantees that the output produced by a stationary input is also stationary. It is thus interesting to consider only the portion of the complex planes \( s \) and \( z \) associated to infinite duration signals, i.e. \( s = j\omega \) and \( z = e^{j\theta} \). These correspond to harmonic eigenvalues of the LTI system, hence lead to an harmonic response. From Laplace and Z transforms, we move to continuous and discrete Fourier transforms in section 7.1. A system that manipulates the harmonic content of a signal is a filter; these are introduced Section 7.2 along with their fundamental role in multi-resolution decompositions.
7.1 From Laplace to Fourier

Consider the bilateral Laplace and Z transform of a signal along the imaginary axis $s = j\omega$ and the unitary circle $z = e^{j\theta}$ respectively:

$$U(j\omega) = \mathcal{F}_C\{u(t)\} = \int_{-\infty}^{\infty} u(t)e^{-j\omega t}dt \quad (39a)$$

$$U(j\theta) = \mathcal{F}_D\{u[k]\} = \sum_{k=-\infty}^{+\infty} u[k]e^{-j\theta k} \quad (39b)$$

These are the continuous (CT) and the Discrete (Time) Fourier Transforms (DTFT). Both are continuous functions, with the second being periodic of period $2\pi$ because of the conformal mapping introduced in Figure 3. Comparing these to (28) and (30) shows that the bilateral Laplace and Z transforms are the Fourier transforms of $u(t)e^{-\alpha t}$ and $u[k]\rho^{-k}$. Without these exponentially decaying modulations, the conditions for convergence are more stringent: signals must be absolutely integrable and absolutely summable.

The main consequence is that infinite duration stationary signals do not generally admit a Fourier transform. This explains why the manipulations of these signals by an LTI system are better investigated in terms of some of their statistical properties, such as autocorrelation or autocovariance, as illustrated in section 8. A special exception are periodic signals, for which (39a) and (39b) lead to Fourier series, and the problem of convergence becomes less stringent.

In stable continuous and discrete LTI systems, satisfying (30), the impulse response always admit Fourier transform: these can be obtained by replacing $s = j\omega$ and $z = e^{j\theta}$ in the transfer function. This leads to the frequency transfer function, which are complex functions of real numbers $(\omega$ or $\theta)$, customarily represented by plotting $\log(|H(x)|)$ and $\arg(H(x))$ versus $\log(x)$ in a Bode plot, with $x = \omega$ or $x = \theta$. The modulus of the frequency transfer function is the amplitude response; its argument is the phase response.

If the Fourier transform (or series) exist for both inputs and output, the properties of the Laplace and Z transform applies: the harmonic contents of the output is $Y(j\omega) = H(j\omega)U(j\omega)$ in the continuous domain; $Y(j\theta) = H(j\theta)U(j\theta)$ in the discrete one.

Discrete signals of finite duration $u(t_k) = \mathbf{u} \in \mathbb{R}^{n_t \times 1} = u[k]$, with $k \in [0,n_t-1]$, are usually extended to infinite duration signals assuming periodic boundary conditions. The frequency domain is thus discretized into bins $\theta_n = n\Delta\theta$ with $n \in [0,n_f-1]$ and $\Delta\theta = 2\pi/n_F$. The mapping to the continuous frequency domain, from Fig. 3, gives $f_n = 2\pi\omega_n = n f_s/n_F$, with $f_s = 1/\Delta t$ the sampling frequency. With both time and frequency domain discretized, the Fourier pair are usually written as

$$U[n] = \frac{1}{\sqrt{n_t}} \sum_{k=0}^{n_t-1} u[k]e^{-2\pi j n k / n_F} \leftrightarrow u[k] = \frac{1}{\sqrt{n_t}} \sum_{n=0}^{n_f-1} U[n]e^{2\pi j n k / n_F}. \quad (40)$$

---

14This condition is sufficient but not necessary: some non-square integral functions do admit a Fourier transform. Important examples are the constant function $u(t) = 1$ or the step function $u_s(t)$. Moreover, note that the Fourier Transform can be obtained from the Laplace and Z transform, only for signals that are absolutely integrable or summable. For instance, the Laplace transform of $e^{\alpha t}$ with $\alpha > 0$, has ROC $s > \alpha$ while the Fourier transform does not exist.

15These are often called real frequencies as opposed to the complex frequencies $s$ and $z$. 
The equations in (49) are respectively the Discrete Fourier Transform (DFT) and its inverse. Note that the normalization \( \frac{1}{\sqrt{n}} \) is used for later convenience: we see in Chapter 8 that (49) can be written as matrix multiplications with the columns of the matrix being orthonormal vectors. Finally, if \( n_f = n_t \) and \( n_t \) is a power of 2, this multiplication can be performed using the famous FFT (Fast Fourier Transform) algorithm (see Loan (1992)), reducing the computational cost from \( n_t^2 \) to \( n_t \log_2(n_t) \). An excellent review of the DFT is provided by Smith (2007b).

Exercise 4: Frequency Response Analysis

Consider the discrete system derived in Exercises 3, but now assume that the static gain is unitary. Compute the frequency transfer function of this system and show that this can be seen as a low-pass filter. Study how the frequency response changes if the coefficients \( a_1 \) or \( a_2 \) are set to zero. Finally, derive the system that should have the complementary transfer function and show its amplitude response. Is this response also complementary?

7.2 Multiresolution Analysis and Digital Filters

Filters are at the center of most signal processing applications, and the theory behind their design is a vast subject (Smith, 2007a). Among the essential applications discussed in this book are feedback control design (Chapter 10), Multiresolution Analysis (MRA) and wavelet decomposition (Chapter 5) and Multiscale Proper Orthogonal Decomposition (Chapter 8).

In feedback control, a controller manipulates the feedback coefficients of an LCCDE by introducing a control input which is function of the output. Therefore, filter design methods can be used to design the actuation such that the transfer function of the controlled system rejects certain disturbances (see Bode design methods (Distefano, 2013)).

In MRA, filters are used to decompose signals. While the DFT represents a signal as a linear combination of harmonics, MRA represents it as a combination of frequency bands called scales. A packet of similar frequencies can be assembled into bases called wavelets, hence the connection to Chapter 5.

The MRA partitions the spectra of signal into \( n_M \) scales, each taking a portion of the signal's content in bands \([0, f_1], [f_1, f_2], \ldots [f_{M-1}, f_s/2]\). In Chapter 8, these are identified by a frequency splitting vector \( F_V = [f_1, f_2, f_3 \ldots f_{M-1}] \).

The MRA of a discrete signal can be written as:

\[
    u[k] = \sum_{m=1}^{M} s_m[k] = F^{-1} \left\{ \sum_{m=1}^{M} H_m(f_n)U(f_n) \right\} \text{ with } \sum_{m=1}^{M} |H_m(f_n)| = 1 , \quad (41)
\]

where \( s_m \) is the portion of the signal in the scale \( m \), within the frequency range \( f_n \in [f_{m-1}, f_m] \) and \( H_m(f_n) \) is the transfer function of the filter that isolates that portion. Therefore, \( |H_m(f_n)| \approx 1 \) for \( f_n \in [f_{m-1}, f_m] \) and \( |H_m(f_n)| \approx 0 \) otherwise. The assumption on the right enables a lossless decomposition.

The MRA requires the definition of one low pass filter for the range \([0, f_1]\), one high-pass filter for the range \([f_{M-1}, f_s/2]\) and \( M - 2 \) bandpass filters. Because these are
complementary, all these filters can be obtained from a set of low pass filters, as described at the end of this section. Therefore, to learn MRA, one should first learn how to construct a low pass filter with a given cut-off frequency $f_c$.

We now focus on the two main families of filters and the most common design methods. Let us consider a specific example, with $f_s = 2kHz$ and $f_c = 200Hz$. In the digital frequency domain, we map the sampling frequency to $\theta_s = 2\pi$ and the cut-off to $\theta_c = \pi/5$.

The transfer function of the ideal low pass filter is

$$H_{id}(j\theta) = \begin{cases} e^{-j\alpha_d\theta} & \text{if } |\theta| \leq \theta_c \\ 0 & \text{if } \theta_c < |\theta| \leq \pi \end{cases}$$

(42)

This leads to the impulse response

$$h_{id}[k] = \frac{1}{2\pi} \int_{-\pi}^{\pi} H(j\theta)e^{-jk\theta}d\theta = \frac{\sin[k - \alpha_d]\theta_c}{\pi(k - \alpha_d)}.$$ 

(43)

The need for a delay $\alpha$ is evident after introducing FIR filters. Notice that having a linear phase $\alpha\theta$ delays the input without distorting its waveform. The modulus of this frequency response and a portion of its transfer function are shown in Figure 4a) with continuous black curves.

\[\text{Figure 4: Fig (a): Absolute value of the frequency response of an ideal (black continuous), a IIR (red dash-dotted) and a FIR (blue dashed) low pass filter; Fig (b): impulse responses associated to the frequency responses on the left. In the case of the FIR, the response is shifted backward by $\alpha_d$.}\]

Such ideal filter is not realizable in the time domain because its impulse response is not causal ($h[k] \neq 0$ for $k < 0$) and is not absolutely summable. The ideal constraints must be relaxed. The most popular categories are IIR and FIR filters.

1) Infinite Impulse Response Filters (IIR). These filters are based on a continuous function that mimics the ideal low pass filter. The most common are Butterworth, Chebyshev and Elliptic Filters (Hayes, 2011; Oppenheim and Schafer, 2009). These filters have no zeros and $N$ poles, with $N$ the filter order, equally spaced around the unit circle.

Once these poles are computed, the continuous frequency response function can be readily obtained in its factor form and the last step consist in identifying the associated
recursive formula as in exercise 4. However, note that mapping from $s$ to $z$ is usually performed using the bilinear transform\(^{16}\) rather than the standard mapping $z = e^{\Delta t s}$ that is used in Exercise 4, since this has the advantage of mapping $f_s/2$ to $\pi$ and thus prevent aliasing. The nonlinearity in the bilinear transform results in a wrapping of higher frequencies so the correct cut-off frequency should first be pre-warped to account for the distortion in the frequency calculation\(^{17}\). Software packages such as SciPy in PYTHON or MATLAB offer the functions BUTTERWORTH to design a Butterworth filter with given order and cut-off frequency (see PYTHON script Ex5.py).

The red curves in Figure 4a) show the amplitude response and the impulse response of a Butterworth filter of order 11. The main advantage of these filters is their capability of well approximating the ideal filter using a limited order, which requires storing few coefficients in their recursive formulation. On the other hand, these filters tend to become unstable as the order increases, (and the poles approach the unit circle). Moreover, their phase delay is generally not constant, and this potentially introduces phase distortion. Finally, note that since the impulse response of these filters is infinite, these cannot be implemented in the time domain via simple convolution, but via the recursive solution of the filter’s LCCDE.

2) Finite Impulse Response Filters (FIR). These filters are constructed in the discrete domain and have no poles (no feedback coefficients in their recursive formulation). This leads to a finite impulse response. The classic design method is the windowing technique, which consists in multiplying the impulse response of the ideal filter in (42) by a window $w[k]$ which is zero outside the interval $0 \leq k \leq N$, with $N$ the filter order.

Taking $N$ as an odd number, these windows are symmetric about the midpoint, i.e. $w[k] = w[N - k]$; this results in the lag $\alpha = (N - 1)/2$ in the output with respect to the input. The need for a lag in the ideal filter in (42)-(43) is now clear: if $\alpha = 0$, the windowed impulse response is centered in 0, and the filter is non-causal. If the filtering is performed ‘offline’, it is possible to obtain a zero-phase filter by centering the windowed impulse response in $k = 0$.

Common functions are the Hanning, Hamming or Blackman and Kaiser windows. The windowing in the time domain corresponds to a convolution in the frequency domain between the ideal filter and the Fourier transform of the window function. This smooths the transition from the band-pass to the band-stop region. Software packages as SciPy in PYTHON or MATLAB offer the functions FIRWIN and FIR1 to design FIR filters with a given order, cut-off frequency and window function.

In addition to the linear phase response, these filters are also always stable because of the lack of poles. Moreover, the finite length of their impulse response enables their implementation via convolution. Note, however, that FIR filters require much larger order to achieve performances comparable with IIR filters. Figure 4b shows, in the dashed blue line, the amplitude function and the corresponding impulse response (shifted by $(N - 1)/2$)

\(^{16}\)which reads:

$$s = \frac{2}{\Delta t} \left( \frac{1 - z^{-1}}{1 + z^{-1}} \right) \leftrightarrow z = \frac{1 + \Delta t/2}{1 - \Delta t/2} s$$

\(^{17}\)The pre-warp can be achieved using $f'_c = f_c / \pi \tan(\pi f_c / f_s)$. Therefore, if the desired cut-off frequency is $f_c = 200Hz$ with a sampling frequency $f_s = 1000Hz$, the filter should target a cut-off frequency of $f'_c = 231.26Hz$ to compensate for the warping due to the bilinear transform.
of a FIR filter designed using a Hamming window of order $N = 111$. The highest the filter order, the larger is the window multiplying the ideal impulse response, the more this filter approach the ideal one. On the other hand, increasing the filter order increases the sensitivity of the filter to the finite duration of the signal.

A FIR formulation make the calculation of complementary filters particularly simple thanks to the constant phase response, solely linked to the filter order. To illustrate this, consider a signal $u[k]$, and its DFT $U[n]$. Let us low-pass filter this signal to obtain $u_L[k]$ using a FIR filter with frequency transfer function $H_L = |H_L|e^{-i\alpha k}$. The filter operation in the frequency domain reads $U_L = |H_L|e^{-i\alpha \theta}U$. Let $U_H = |H_H|e^{-i\alpha \theta}U$ denote the high pass filter that gives the signal $u_H[k]$ having complementary spectra (i.e. $|H_L| + |H_H| = 1$) and same order and thus same phase delay. Using the shifting properties of the Fourier transforms, the link between the low-pass and the high-pass counter parts in the time and frequency domain sets

\[
\begin{align*}
\text{k Domain} & : u_H[k] = u[k + \alpha] - u_L[k] \\
\text{\theta Domain} & : H_H(i\theta)U(i\theta) = U(i\theta)e^{i\alpha \theta} - H_L(i\theta)U(i\theta) \\
& \rightarrow H_H(i\theta) = e^{i\alpha \theta} - H_L(i\theta).
\end{align*}
\]

Note that the backward shifting $u[k + \alpha]$ in the time domain cancels the phase delay produced by the low pass filter before performing the subtraction. It is easy to show that because of the linearity of the convolution, the impulse responses of complementary high-pass ($h_H$) and low-pass ($h_L$) filters are linked by\(^1\) $\delta[n] = h_L[n] + h_H[n]$.

Finally, we close with the practical implementation of MRA in ‘off-line’ conditions, for which it is possible to release the constraints of causality and use zero-phase filters. These are usually implemented by operating on the signal twice (first on $u[k]$ and then on $u[-k]$), to artificially cancel the phase delay of the operation. In SCIPY and in MATLAB this is performed using the function \texttt{FILTILT}.

If the phase delay is canceled, complementary filters can be computed by taking differences of the frequency transfer functions (which become real functions). Therefore, if the first scale with band-pass $[0, f_1]$ is identified by the frequency transfer function $H_1(f) = H_L(f, f_1)$, the second scale with band-pass $[f_1, f_2]$ is identified by a filter with transfer function $H_2(f) = H_L(f, f_2) - H_L(f, f_1)$. The transfer function of the general band-pass filter is $H_m(f) = H_L(f, f_m) - H_L(f, f_{m-1})$ while the last scale is identified by the high-pass filter with $H_M(f) = 1 - H_L(f, f_M)$.

This set of cascaded filters is known as filter bank and is at the heart of the pyramid algorithm for computing the discrete wavelet transform (Mallat, 2009; Strang, 1996), where it is combined with sub-sampling at each scale. The general architecture of this decomposition is summarized in Figure 4. Observe that at the limit at which all the frequency bands become unitary, the MRA becomes a DFT.

\(^1\)Note that this is not the only method to obtain an high pass filter from a low-pass filter: another approach is to reverse the frequency response $H_L$, flipping it from left to right about the frequency $f_s/4$ for $f > 0$ and from right to left about $-f_s/4$ for $f < 0$ (Smith, 1997). The impulse response of the resulting high pass filter is $h_H[n] = (-1)^n h_L[n]$. The two methods are equivalent if the cut-off frequency separating the transition bands is $f_s/4$. This is the case encountered when performing MRA via dyadic wavelets as discussed in Chapter 5.
Consider the synthetic signal

\[ u(t_k) = a_1 \sin(2\pi f_1 t_k) e^{-\frac{(t_k-\tau_1)^2}{b_1}} + a_2 \sin(2\pi f_2 t_k) e^{-\frac{(t_k-\tau_2)^2}{b_2}} + a_3 \sin(2\pi f_3 t_k) + \mathcal{N}(0, 0.2) \]  

(45)

with \( a_1 = 2, a_2 = a_3 = 1, f_{1,2,3} = 1,20,90 \) (in Hz), \( \tau_1 = 1, \tau_2 = 2.2 \) (in s) and \( b_1 = b_2 = 0.05 \). \( \mathcal{N}(0, 0.2) \) denotes Gaussian noise with zero mean and standard deviation 0.2. The signal is sampled over \( n_t = 4096 \) points at a sampling frequency of \( f_s = 1kHz \).

1) Compute ‘by hand’ the impulse response \( h_L \) of a low pass FIR filter of order \( N_O = 5 \), with cut off frequency \( f_c = 10Hz \) using a Hamming window. Check your answers using Python’s function \texttt{firwin}.

2) Prepare a filter \( h_L \) of order 511 using \texttt{firwin}. Apply this to the signal using four methods. 1) direct convolution, 2) FFT-based convolution with \texttt{SCIPY’s function fftconvolve}, 3) FFT-based convolution using \texttt{FFT} and \texttt{IFFT} and 4) a recursive implementation solving the filter’s LCCDE.

3) Construct a filter bank with frequency splitting vector \( F_V = [10, 70, 100, 300] \) and show the portions of the signal within the identified scales. Use filters with Hamming window and an order of \( N_O = 511 \).

### Exercise 5: Multi-Resolution Analysis

8 Application II: Time Series Analysis

LTI systems are the simplest model in time series analysis and forecasting. In these applications, treating signals and system as fully deterministic is too optimistic, and it is thus essential to consider stochastic signals: predictions have a certain probability range (Guidorzi, 2003; Brockwell and Davis, 2010). This section briefly reviews the main
features of stochastic signals and systems in 8.1. Section 8.2 reviews the basic tools for forecasting, using classic linear regression. Only the discrete domain is considered. More advanced techniques are discussed in Chapter 12.

8.1 Stochastic LTI Systems

A stochastic signal (or the stochastic portion of a signal) is a member of an ensemble of signals characterized by a set of probability density functions. For a comprehensive review of stochastic signals, the reader is referred to classic textbooks (Ljung, 1999; Oppenheim, 2015; Hsu, 2013). Here, we briefly recall how LTI systems manipulate stochastic signals.

The notion and the role of the impulse response remains the same as for deterministic signals: given an input (stochastic) discrete signal, the response of the system is governed by the convolution sum in (21). On the other hand, the notion of frequency spectra requires some adaptation, as stochastic signals do not generally admit a Fourier transform and focus must be placed on properties that are deterministic also in a stochastic signal. These are the statistical properties.

Accordingly, the time-invariance in LTI systems is extended in terms of invariance of the statistical properties. This is linked to the notion of stationarity. Stationarity can be weak or strong. A stochastic signal $v[k]$ is stationary in a strict sense (strong stationarity) if its distributions remain invariant over time. Weak stationarity (or stationarity in a wide-sense) requires that only its time average $\mu_v$ and autocorrelation $r_{vv}[m]$ of a signal $v$ are time-invariant. These are defined as follows

$$\mu_v = \mathbb{E}\{v[k]\} \quad \text{and} \quad r_{vv}[m] = \mathbb{E}\{v[k]v[k+m]\}, \quad (46)$$

with $\mathbb{E}$ the expectation operator.

In the analysis of LTI system’s response to stochastic signals, the link between a specific input and the corresponding output is not particularly interesting. Instead, we focus on the link between the statistical properties of the input and the output. In particular, we consider how the properties in (46) are manipulated. Let $y_v[k]$ be the response of the system to the stochastic signal $u_v[k]$. The expected (time average of the) output $\mu_y$ is:

$$\mu_y = \mathbb{E}\left\{\sum_{l=-\infty}^{\infty} h[l]u_v[k-l]\right\} = \sum_{l=-\infty}^{\infty} h[l]\mathbb{E}\{u_v[k-l]\} = \mu_v \sum_{l=-\infty}^{\infty} h[k]. \quad (47)$$

This is a direct application of the homogeneity (16) and superposition (17). We thus see that in a BIBO stable system (satisfying (27)) the output average is finite if the input average is finite. The input/output relation for the the autocorrelation function has a more involved derivation, here omitted (Oppenheim et al., 1996). Given $r_{uu}$ and $r_{yy}$ the input and the output autocorrelations, one retrieves:

$$r_{yy}[m] = \sum_{l=-\infty}^{\infty} r_{uu}[m-l]r_{hh}[l] \quad \text{where} \quad r_{hh}[k] = \sum_{l=-\infty}^{\infty} h[k]h[l+k]. \quad (48)$$

The sequence $r_{hh}[k]$ is the autocorrelation of the impulse response and operation on the left is a convolution. In words: the autocorrelation of the output is the convolution

\[^{19}\text{We also see why a high pass filter has } \sum_k h[k] = 0 \text{ while a low pass filter has } \sum_k h[k] = 1.\]
of the autocorrelation of the input with the autocorrelation of the impulse response. This equation extends the convolution link in (21) to the autocorrelation functions. These functions admit Fourier transform, so the convolution theorem can be used to see the link in the frequency domain:

\[ R_{yy}(j\omega) = C_{hh}(j\omega)R_{uu}(j\omega), \] (49)

where \( R_{yy}(j\omega) \), \( C_{hh}(j\omega) \) and \( R_{uu}(j\omega) \) are the Fourier transform of \( r_{yy}[k] \), \( r_{hh}[k] \) and \( r_{uu}[k] \) respectively. These are the power-spectral densities of \( y \), \( h \) and \( u \). Hence we see that an LTI system acts on the frequency content of the autocorrelation function of a stochastic signal.

### 8.2 Time Series Forecasting via LTI Systems

Consider the explicit form of the LCCDE of a LTI in (37) and assume that the input signal \((u)\) has both a deterministic \((u_d)\) and a stochastic \((u_s)\) part (i.e. \( u = u_d + u_s \)). Because of homogeneity, the output of the LTI system also has a deterministic \((y_d)\) and a stochastic part \((y_s)\). We could split these as follows:

\[
y[k] = y_d[k] + y_s[k] = \sum_{n=0}^{N_b} b_n^* u_d[k-n] - \sum_{n=1}^{N_f} a_n^* y_d[k-n] + y_s.
\] (50)

Many models can be obtained depending on the assumptions on \( u_s \), and hence \( y_s \) (Guidorzi, 2003; Nielsen, 2019; Brockwell and Davis, 2010). For example, the stochastic part \( y_s \) can be taken as white noise with zero average or as the output of a moving average filtering\(^{20}\) of white noise. Any other filter can be used to allow controlling and/or modeling the frequency content of the stochastic contribution using (48) and (49).

To illustrate the main steps of time series forecasting, let us consider the simplest approach of \( y_s \) being white noise. If the system is known (i.e. the coefficients \( a_n^* \) and \( b_n^* \) are known), the recursive equation (50) can be written as a matrix multiplication\(^{21}\).

Assume that we have collected \( n_t \) samples of the input \( u \). Let \( u_d^{(-l)}, y_d^{(-l)} \in \mathbb{R}^{n_t \times 1} \) be the vectors collecting the deterministic inputs and outputs shifted backward by a lag \( l \). Then, matrix form of (50) is

\[
\begin{bmatrix} y^{(0)} \\ \vdots \\ y^{(-N_b)} \end{bmatrix} = \begin{bmatrix} u^{(0)} & u^{(-1)} & \ldots & u^{(-N_f)} & y^{(-1)} & \ldots & y^{(-N_b)} \end{bmatrix} \begin{bmatrix} b_0 \\ \vdots \\ b_{N_b} \\ a_1 \\ \vdots \\ a_{N_f} \end{bmatrix} + y_s.
\] (51)

We define \( H := [u^{(0)}, \ldots, u^{(-N_f)}, y^{(-1)} \ldots y^{(-N_b)}] \) the Hankel matrix of the LTI system and \( w = [b_0, \ldots, b_{N_b}, a_1 \ldots a_{N_f}]^T \) the vector of coefficients. The LTI system’s output is

\[
y^{(0)} = Hw + y_s
\] (52)

\(^{20}\)A ‘moving average’ filter is a filter with constant impulse response

\(^{21}\)This needs to be evaluated from from the first to the last entry of \( y^{(0)} \).
Time series forecasting via LTI systems begins with system identification. An excellent tutorial on the topic is provided by Semeraro and Mathelin (2016). The first goal is to identify the set of coefficients $w$ from the input/output vectors. The stochastic part is considered as noise and the determinist part is our expectation. We thus seek to solve the system $y(0) \approx Hw$. Like most regression problems, this problem is ill-posed: $H$ is rectangular and there is no guarantee that a unique solution exists. Like all linear regression problems, the solution is found by minimizing a regularized cost function of the form

$$J(w) = ||y^{(0)} - Hw||_2^2 + \alpha R(w),$$

with $\alpha \in \mathbb{R}$ acting as a smoothing parameter and $R(w)$ a regularizing function\footnote{Readers familiar with Lagrangian multipliers should recognize in (53) an augmented cost function with $\alpha$ the Lagrangian multiplier.} Classic choices are $R(w) = ||w||_2$ ($l^2$ penalty) or $R(w) = ||w||_1$ ($l^1$ penalty) or a combination of the two. The first is known as Tikhonov regularization, the second as LASSO regularization and the third as Elastic Net. These classic tools from machine learning (Bishop, 2016; Vladimir Cherkassky, 2008; Murphy, 2012) are also employed in Chapter 12.

The reader should notice that the regression method can be generalized easily: one could replace the predictive equation (52) by a more complex model (e.g. an Artificial Neural Network or templates of polynomial nonlinearities as in Chapter 12), and minimize a cost function like (53) using an arsenal of optimization strategy.

**Exercise 6: Time Series forecasting via LTI Systems**

Generate a synthetic stochastic system with coefficients $(a_1, a_2) = (-1/2, 0)$ and $(b_0, b_1, b_2) = (2, 1.8, -1.2)$. Consider a deterministic input signal $u[k] = u(t_k) = \sin(8\pi t_k) + 3 \exp(-(t_k - 3)^2/2)$ with $t_k = k\Delta t$, $\Delta t = 0.02$ and $k \in [0, 1999]$. The stochastic contribution $y_s[k]$ is white noise in $[0, 1]$. Use the set of 2000 points in input and output to identify the set of coefficients from the data, by solving (53) with a Tikhonov regularization. Consider $\alpha = 1$, and recall that the minimization of $J(w)$ in this kind of regression (known also as Ridge regression) has a simple solution:

$$w = (H^T H + \alpha I)^{-1} H y^{(0)},$$

with $I$ the identity matrix. Is the expected result recovered? is the identified LTI system capable of predicting what happens if $t_k > 5$? what happens for $\alpha = 0$ (i.e. ordinary least square)?

9 What’s next?

This chapter reviewed the fundamentals of signals and systems and presented Linear Time Invariant (LTI) systems in case of Single Input Single Output (SISO). We have seen that the input-output relation can be derived from knowledge of the impulse response of a...
system and via convolution integral. It was shown that complex exponentials are eigenfunctions of these systems and that important transforms can be derived by projecting input and output signals onto these eigenfunctions. In the eigenspace of the LTI systems, convolutions become multiplications.

Chapter 10 presents LTI systems the so-called state-space representation, which is more common in dynamical system theory and which allows for straightforward generalization to MIMO systems. Chapter 11 reviews the analysis of nonlinear systems, while Chapter 12 describes the system identification more broadly, considered also nonlinear systems.

This chapter reviewed the link between the continuous and the discrete world and the impact of the discretization on the eigenfunctions of an LTI system. Special values of the complex frequencies, called poles, yields infinite response of a system and are linked to the notion of stability, reviewed in Chapter 13. The reader should recognized that the identification of these poles from large datasets is the essence of the Dynamic Mode Decomposition described in Chapter 7.

Finally, this chapter also introduced the fundamentals of MRA, which well complements wavelet theory in Chapter 5 and Chapter 8 on the multiscale Proper Orthogonal Decomposition.

References

Beerends, R. J., ter Morsche, H. G., van den Berg, J. C., and van de Vrie, E. M. (2003). *Fourier and Laplace Transforms*. Cambridge University Press.

Bishop, C. M. (2016). *Pattern Recognition and Machine Learning*. Springer New York.

Brockwell, P. J. and Davis, R. A. (2010). *Introduction to Time Series and Forecasting (Springer Texts in Statistics)*. Springer.

Distefano, J. (2013). *Schaum’s Outline of Feedback and Control Systems*. McGraw-Hill Education - Europe.

Guidorzi, R. (2003). *Multivariable System Identification. From observations to models*. Bononia University Press.

Hayes, M. (2011). *Schaums Outline of Digital Signal Processing*. McGraw-Hill Education - Europe.

Hsu, H. (2013). *Schaum’s Outline of Signals and Systems, 3rd Edition (Schaum’s Outlines)*. McGraw-Hill Education.

Ingle, V. K. and Proakis, J. G. (2011). *Digital Signal Processing Using MATLAB*. Cengage Learning.

Kaiser, G. (2010). *A friendly guide to wavelets*. Springer Science & Business Media.

Ljung, L. (1999). *System Identification: Theory for the User*. Prentice Hall.

Ljung, L. (2008). Perspectives on system identification. *IFAC Proceedings Volumes*, 41(2):7172–7184.

*VKI*
Ljung, L. and Glad, T. (1994). *Modeling of Dynamic Systems*. Prentice Hall.

Loan, C. V. (1992). *Computational Frameworks for the Fast Fourier Transform*. Society for Industrial and Applied Mathematics.

Mallat, S. (2009). *A Wavelet Tour of Signal Processing*. Elsevier LTD, Oxford.

Murphy, K. P. (2012). *Machine Learning*. MIT Press Ltd.

Nielsen, A. (2019). *Practical Time Series Analysis*. O’Reilly UK Ltd.

Ogata, K. (2009). *Modern Control Engineering (5th Edition)*. Pearson.

Oppenheim, A. V. (2015). *Signals, Systems and Inference*. Pearson Education.

Oppenheim, A. V. and Schafer, R. W. (2009). *Discrete-Time Signal Processing (3rd Edition) (Prentice-Hall Signal Processing Series)*. Pearson.

Oppenheim, A. V., Willsky, A. S., and with S. Hamid (1996). *Signals and Systems (2nd Edition)*. Pearson.

Richards, J. I. and Youn, H. K. (1990). *The Theory of Distributions*. Cambridge University Press.

Semeraro, O. and Mathelin, L. (2016). An open-source toolbox for data-driven linear system identification.

Sigurd Skogestad, I. P. (2005). *Multivariable Feedback Control*. John Wiley & Sons.

Smith, J. O. (2007a). *Introduction to Digital Filters: with Audio Applications*. W3K Publishing.

Smith, J. O. (2007b). *Mathematics of the Discrete Fourier Transform (DFT): with Audio Applications*. W3K Publishing.

Smith, S. W. (1997). *The Scientist & Engineer’s Guide to Digital Signal Processing*. California Technical Pub.

Strang, G. (1996). *Wavelets and Filter Banks*. Wellesley-Cambridge Press, U.S.

Sundqvist, B. (1991). Thermal diffusivity measurements by Ångström method in a fluid environment. *International Journal of Thermophysics*, 12(1):191–206.

Vladimir Cherkassky, F. M. M. (2008). *Learning from Data*. John Wiley & Sons.

Wang, R. (2009). *Introduction to Orthogonal Transforms*. Cambridge University Press.

Williamson, D. (1999). *Discrete-time Signal Processing*. Springer London.