Digital Signal Processing

GuoHui

Xiamen University

Collaborator:

GuoHui

Xiamen University

October 20, 2025

Abstract

These are notes on **DSP**.

Contents

1 Introduction

1.1 Overview

Some prerequisite:

In order to study this course better, you need to have some mathmatical maturity and a foundation of calculus.

The textbook we use is Discrete-Time Signal Processing, 3rd edition written by A. V. Oppenheim.

1.2 Some Suggestions

In the first subsection, let's give the following suggestions.

No reading, No learning.

After we enter a high level of learning, we mainly depend on reading, because no one will tell us the key points. The reason for setting up a series of courses is to make our learning more targeted.

No writing, No reading.

What is effective reading? You must write something in the paper, such as the math notations. Only in that way will you brain not skip many details.

No data, No truth.

In the vast majority of scientific fields, we must apply theroies and methods to real data rather than relying on simulation.

No Analytic, No understanding.

If you say, "You have well understood something", you must provide a detailed analysis using symbolic language.

No programming, No coginition.

Finally, if you wish to have an intutive coginition of something, the typical approach is to use a series of visualization methods (such as pictures, tables...) to present the images in your mind.

1.3 Digital vs Analog

There are the following differences between the two.

First, digital means programmable. Analog rely on hardware, digital are true software, which is very flexible and easy to change.

Second, in the digital world we use bit to describe precision. And digital has a high and controllable precision.

Third, storing, digital has a high storage density and compressible.

Finally, digital's cost is low, because it's manufactured on a silicon wafer.

1.4 Course Arrangement

The digital signal processing course has the following chapters.

Chapter1: Preliminary for Digital.

In this chapter, we will introduce Discrete Signal and System from the following two perspectives:

- Time domain;
- Frequency domain.

Chapter2: How to obtain digital signal.

In this chapter, we will introduce:

- A/D: Using sampling to help us move from the real word (continuous) to digital word.
- D/A: Return journey.

Chapter3: How to process digital signal.

In this chapter, we have two basic tools for recoginzing signals:

- Fourier Transform.
- Z transform.

Also we will introduce Linear Filtering;

- Representation: Discrete Convolution.
- Implemention: Time domain $\longleftrightarrow$ Frequency domain.
- Architecture: from software $\longrightarrow$ chip.

Chapter4: How to improve performance.

We have the following pursuit:

- Faster: FFT.
- More accurate
  - Quantization Noise.
  - Finite word length.

Chapter5: How to design linear filters.

We will introduce the following classic filters:

- FIR.
- IIR.
- Hilbert transform.

Chapter6: How to extend linear filters.

We will introduce:

- Real signal $\longrightarrow$ Complex signal.
- Multirate filters.

Chapter7: How to apply linear filters.

1.5 History of DSP

Now let's briefly review the history of DSP.

1730, Fourier, Fourier Transform.

In the process of studying the heat conduction, Fourier discovered that:

$$\text{any complex motion} \longleftrightarrow \text{linear combination of simple harmonic motion}$$

and that is the Fourier Transform.

1900, Marconi, Electromagnetic waves are simple harmonics

This discovery extends Fourier Transform form thermal to electromagnetic.

1930, Bode, Control Science.

In the field of control science, the first step is perception. Perception is sampling to get data, and extract useful information from the data.

1940, Word War 2.

During this period, two important discoveries were made:

- Wiener Filtering: the foundation of Optimal Filtering (Stochastic Process required).
- Nyquist-Shannon Sampling Theorem.

1950, The Apollo Moon-Landing Project.

This plan gave rise to Kalman Filtering, another methods of Optimal Filtering.

1960, Cooley and Tukey, FFT.

This is the origin of signal processing.

1980, DSP (processor).

A special CPU designed for signal processing task.

2 Discrete Time Signal and System

2.1 Conventions

Considering the continuous signal $x(t)$, we can find it has two elements:

- Independent Variable: It can be time or space, here is time t.
- Dependent Variable.

Therefore, we can draw a conclusion: Signal is a function.

However, a mathmatical function often has analytical expression, for example:

$$x(t) = 3t^2 \tag{1}$$

But signals often don't have explicit analytical expressions, so we need to process the signal to get information.

Now let's introduce continuous-time signal and discrete-time signal:

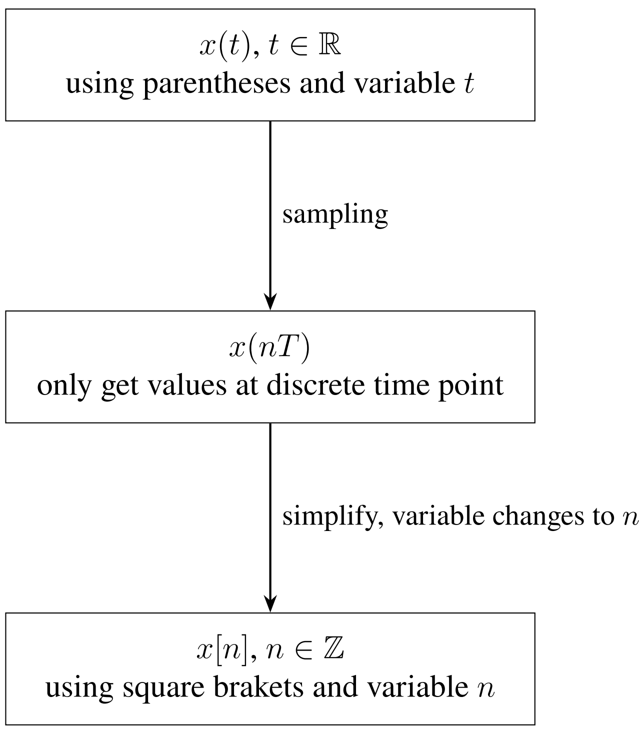

Figure 1: Signals

2.2 Special Signals

Here we introduce some signals.

First, Impulse siganl. Definition:

$$\delta[n] = \begin{cases} 1, & n = 0 \\ 0, & n \neq 0 \end{cases} \tag{2}$$

The most impotant point about this signal is every signal $x[n]$ can be decomposed into the linear combination of shifted impulse $\delta[n-k]$:

$$x[n] = \sum_{k=-\infty}^{+\infty} x[k] \underbrace{\delta[n-k]}_{\text{fundamental element}}, k \in \mathbb{Z} \tag{3}$$

And this formula leads to an important idea:

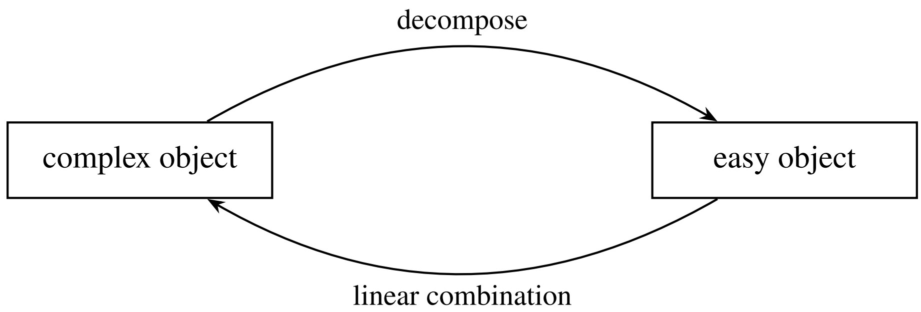

Figure 2: An important Idea

This idea tells us: Comples objects can be decomposed into the linear combination of many easy objects.

Second, Unit step. Definition:

$$u[n] = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases} \tag{4}$$

So what is the relation of $u[n]$ and $\delta[n]$? We can easily get:

$$\begin{aligned} \delta[n] &= u[n] - u[n-1] \\ u[n] &= \sum_{k=-\infty}^{n} \delta[k] \end{aligned} \tag{5}$$

Finally, we should introduce an important function:

$$x_\omega(t) = \mathbf{e}^{\mathbf{j}\omega t} \tag{6}$$

The subscript of x is ω, means the frequency.

Why do we introduce it? Because if we perform a linear transform on this function, the frequency doesn't change.

And we can give the discrete-time version:

$$x_\omega[n] = \mathbf{e}^{\mathbf{j}\omega n} \tag{7}$$

As $n \in \mathbb{Z}$, you can get an interesting property:

$$\begin{aligned} x_{\omega+2\pi}[n] &= \mathbf{e}^{\mathbf{j}(\omega+2\pi)n} \\ &= \mathbf{e}^{\mathbf{j}\omega n} \end{aligned} \tag{8}$$

This means, in the frequency domain, this function has periodicity, and the period is 2π.

What frequency does 2π here correspond to in the physical world? You will get the answer in the following chapter.

2.3 Discrete-Time System

We use the following diagram to represent the system and it's function:

$$\{x[n]\}_{n=-\infty}^{+\infty} \longrightarrow \boxed{\text{T}} \longrightarrow \{y[n]\}_{n=-\infty}^{+\infty}$$

Figure 3: $y[n] = T(\{x[n]\})$

Why does we use $\{x[n]\}_{n=-\infty}^{+\infty}$ and $\{y[n]\}_{n=-\infty}^{+\infty}$ to represent the input and output? Because the sysetem is divided into:

- Memory: $y[n]$ (output at time point n) depends on current and past inputs.
- Memoryless: $y[n]$ (output at time point n) only depends on current time point input $x[n]$.

So when we express the function of a system using symbols, a more rigorous way of writiing it is :

$$y[n] = T(\{x[n]\}) \tag{9}$$

This means the system T operates on a sequence of inputs, not only one input at time n. But for simplicity, we write it:

$$y[n] = T(x[n]) \tag{10}$$

Next, we introduce some typical systems.

Delay Device:

$$y[n] = x[n - n_d] \tag{11}$$

n_d is a scalar.

How should we understand the function of the system? We can give an example.

Suppose n_d is 2, and we want to get the output at time 3, that is $y[3]$. According to the formula, we have:

$$\begin{aligned} y[3] &= x[3 - n_d] \\ &= x[1] \end{aligned} \tag{12}$$

This means, if we want to get the current ouput, we need to use the past input. So this system is **memory**.

Integrator:

$$y[n] = \sum_{k=-\infty}^{n} x[k] \tag{13}$$

This formula leads to an important idea:

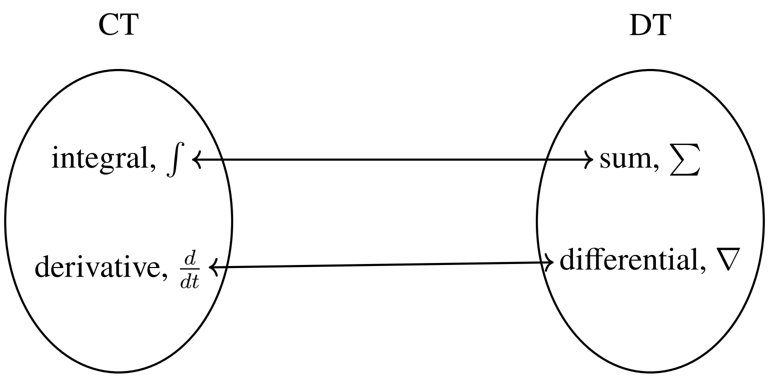

Figure 4: Continuous and Discrete Time Operations

Obviously, Integrator is **memory**.

Square Device:

$$y[x] = x^2[n] \tag{14}$$

Obviously, Integrator is **memoryless**.

2.4 Linearity

Linearity is an important property of systems, and also the key to our research.

Let's give the definition from the surface. If a system T is a linear system, then:

- Additivity: $T(x[n] + y[n]) = T(x[n]) + T(y[n])$.
- Scaling property: $T(\alpha x[n]) = \alpha T(x[n])$.

(we don't know it's "memory" or "memoryless").

This definition just gives us the surface input-output relation, but the system is still a black box. Now let's analyze the internal structure of the system using An important Idea.

Suppose T is a linear system, look at the following operations:

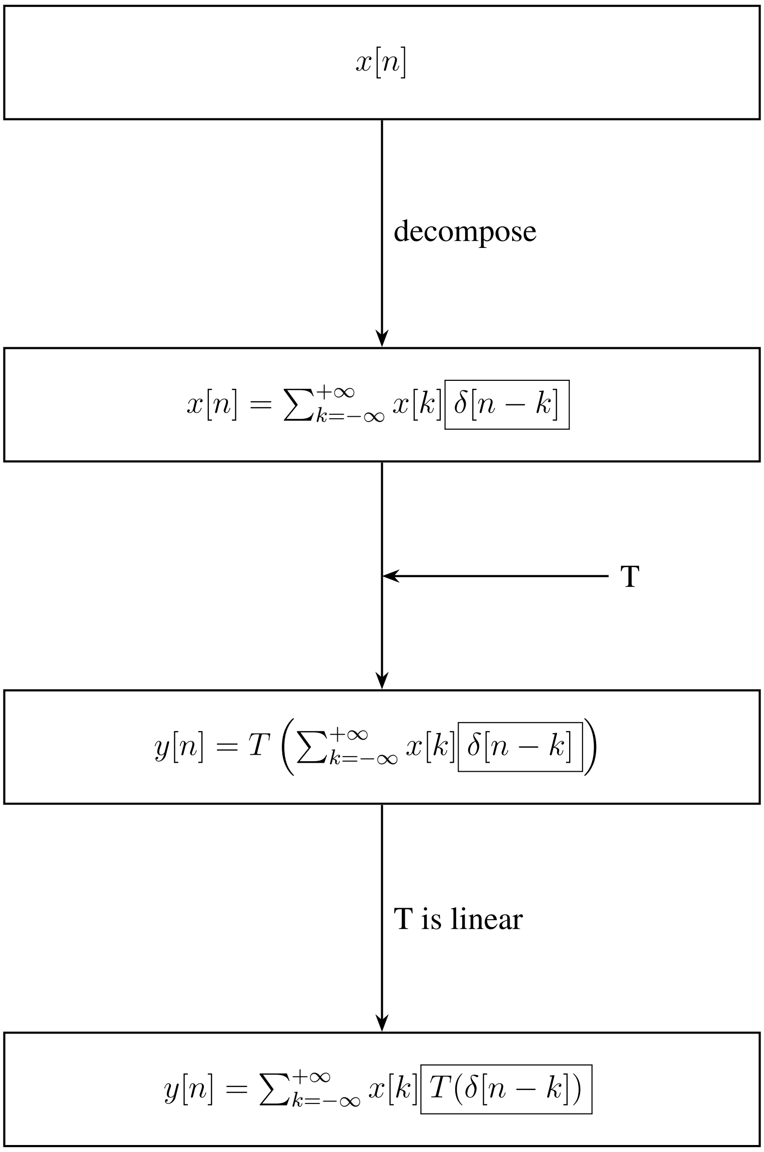

Figure 5: System Process

So we realize the following function:

$$
\begin{tikzcd}
x[n] \arrow[rr, "\text{linear system T}"] \arrow[d, "\substack{\text{decompose} \\ \text{using } \delta[n-k]}"] & & y[n] \arrow[d, "\substack{\text{decompose} \\ \text{using } T(\delta[n-k])}"] \\
\sum_{k=-\infty}^{+\infty} x[k] \boxed{\delta[n-k]} & & \sum_{k=-\infty}^{+\infty} x[k] \boxed{T\left(\delta[n-k]\right)}
\end{tikzcd}
$$

Figure 6: Summary of Linear System Derivation

That is to say, we decompose the input $x[n]$ into the linear combination of impulses. And as the system T is linear, the output can also be decomposed into the linear combinaton, and the elements is impulse response:

$$h(n,k) = T(\delta[n-k]) \tag{15}$$

2.5 Understanding Impulse Response

In the above subsection, we get the definition of **Impulse Response**:

$$h(n,k) = T(\delta[n-k]) \tag{16}$$

We give the following obvious understanding:

- $h(n,k)$ is a scalar, when n and k are all determined.
- $h(n,k)$ is a function of variable n and k.

Why do we call it impulse response?

Suppose our input is an impulse at time 2:

$$x[n] = \delta[n-2] \tag{17}$$

You can get:

$$y[n] = T(\delta[n-2]) = h(n,2) \tag{18}$$

Here, both $\delta[n-1]$ and $h(n,2)$ are signals, not a scalar:

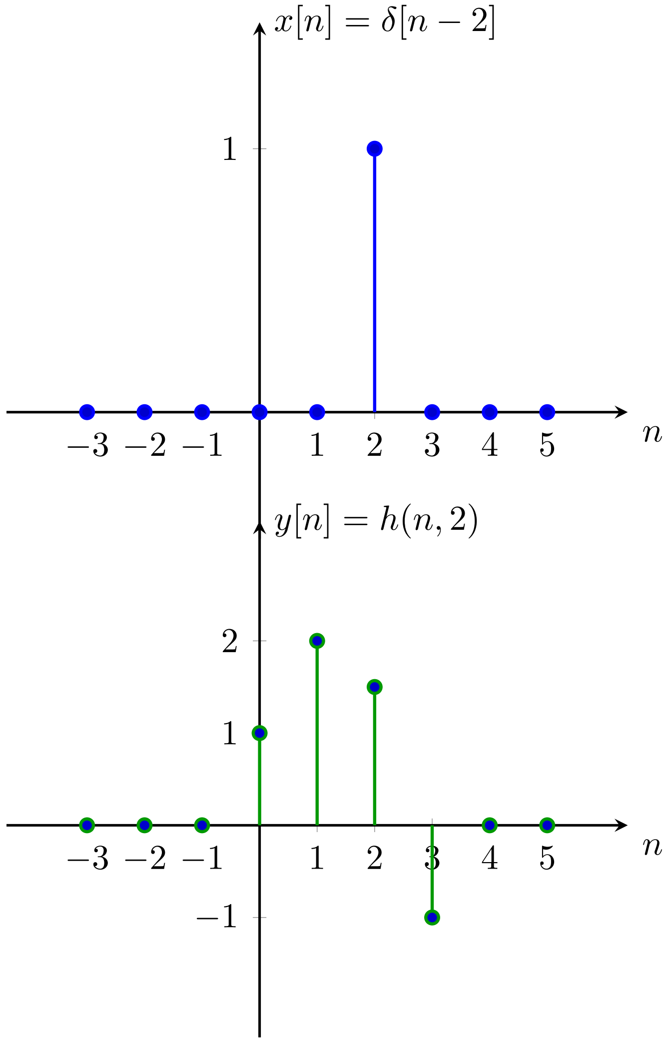

Figure 7: Impulse Response

If our input becomes $\delta[n-2] + \delta[n-3]$, what will happen? Our output will become:

$$y[n] = h(n,2) + h(n,3) \tag{19}$$

And if we pay attention to the output at time point 3 or $y[3]$, it's form is:

$$y[3] = h(3,2) + h(3,3) \tag{20}$$

So, in a summary:

- $h(n,2)$ means, give an impulse at time 2, the output containing all the time points that we get.
- $h(3,2)$ means, give an impulse at time 2, the output at time 3 that we get.

2.6 Linearity-Matrix Perspective

Matrix is also a linear transform.

Suppose we have the following matrix T:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \tag{21}$$

We use this matrix to operate on the following column vector:

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \tag{22}$$

The results are:

$$\begin{bmatrix} \boxed{\begin{matrix} a_{11} \\ a_{21} \\ a_{31} \end{matrix}} & \boxed{\begin{matrix} a_{12} \\ a_{22} \\ a_{32} \end{matrix}} & \boxed{\begin{matrix} a_{13} \\ a_{23} \\ a_{33} \end{matrix}} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = x_1 \begin{bmatrix} a_{11} \\ a_{21} \\ a_{31} \end{bmatrix} + x_2 \begin{bmatrix} a_{12} \\ a_{22} \\ a_{32} \end{bmatrix} + x_3 \begin{bmatrix} a_{13} \\ a_{23} \\ a_{33} \end{bmatrix} \tag{23}$$

How can we using the idea above to explain this?

Actually, we perfrom the following operations:

$$\begin{aligned} T \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} &= T(x_1 \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + x_2 \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + x_3 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}) \\ &= x_1 T(\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}) + x_2 T(\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}) + x_3 T(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}) \end{aligned} \tag{24}$$

We have the similar "impulse":

$$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{25}$$

And we have the similar "impulse response":

$$T(\begin{bmatrix}1\\0\\0\end{bmatrix}) = \begin{bmatrix}a_{11}\\a_{21}\\a_{31}\end{bmatrix}, \quad T(\begin{bmatrix}0\\1\\0\end{bmatrix}) = \begin{bmatrix}a_{12}\\a_{22}\\a_{32}\end{bmatrix}, \quad T(\begin{bmatrix}0\\0\\1\end{bmatrix}) = \begin{bmatrix}a_{13}\\a_{23}\\a_{33}\end{bmatrix} \tag{26}$$

From this example: "impulse response" is a nature of the system itself, has nothing to do with the input.

2.7 Time-Invariant

Supppose there is an arbitrary system T, and we have an input-output combination:

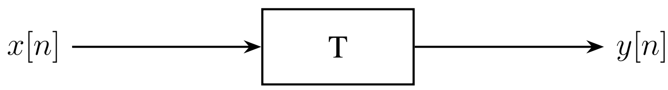

Figure 8: $y[n] = T(\{x[n]\})$

If system T is time-invariant, that is to say:

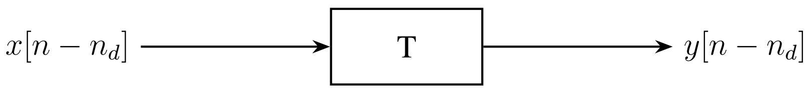

Figure 9: Time-Invariant System T

Here n_d is a scalar.

So in a word, if one system T is time-invariant, then the output delay is the same as the input delay.

2.8 LTI System

If a system simultaneously satisfies linearity and time invariance, then we call the system **LTI System**.

In the above subsection, if a system T satisfies linearity, then we can give **Impulse Response**:

$$h(n, k) = T(\delta[n - k]) \tag{27}$$

Here $h(n, k)$ is a function of variable n and k, and we just know this function has two variables.

Our problem is, if we give the linear system T the time invariance, what form should $h(n, k)$ meet?

Before the resoning begins, why do we focus on $h(n, k)$? Because here we research the nature of the system, and impulse response is the core of this problem.

Particularly, we give the following definition:

$$h[n] = h(n,0) = T(\delta[n]) \tag{28}$$

$h[n]$ means the impulse response of T when we give an impulse at time 0.

This is equivalent to: We give T an input $x[n] = \delta[n]$, the output is $y[n] = h[n]$.

Now give our input delay of k:

$$x[n] = \delta[n] \quad \xrightarrow{\text{delay} \quad k} \quad \hat{x}[n] = \delta[n-k] \tag{29}$$

Suppose system T is time-invariant:

$$\hat{y}[n] = y[n-k] = h[n-k] \tag{30}$$

As:

$$h(n,k) = T(\delta[n-k]) \tag{31}$$

So:

$$h[n-k] = h(n,k) \tag{32}$$

2.9 Convolution

Now we know, if a system T satisfies linearity and time-invariance, give it an input $x[n]$, output is:

$$y[n] = \sum_{k=-\infty}^{+\infty} x[k]h[n-k] \tag{33}$$

We can give the definition of two signals' $*$ oepration:

$$x[n] * h[n] = \sum_{k=-\infty}^{+\infty} x[k]h[n-k] \tag{34}$$

And we can say: If a system is LTI system, then:

$$\text{output} = \text{input} * \text{impulse} \quad \text{response} \tag{35}$$

2.10 An Example of Convolution

Suppose T is a LTI system, and the impulse response is:

$$h[n] = u[n] - u[n-N] \tag{36}$$

here $u[n]$ is unit step, N is a positive integer. We give it an input:

$$x[n] = a^n u[n] \tag{37}$$

here $|a| < 1$.

Calsulate the output:

$$y[n] = x[n] * h[n] = \sum_{k=-\infty}^{+\infty} x[k]h[n-k] \tag{38}$$

Method1. Analytical Calculations.

Expand the convolution formula:

$$\begin{aligned} y[n] &= \sum_{k=-\infty}^{+\infty} x[k]h[n-k] \\ &= \sum_{k=-\infty}^{+\infty} a^k u[k]u[n-k] - \sum_{k=-\infty}^{+\infty} a^k u[k]u[n-N-k] \end{aligned} \tag{39}$$

As the unit step $u[n]$ gives 0 when $n < 0$, and k is summation variable, we need to detemine which part of k gives a non-zero result.

For the part $\sum_{k=-\infty}^{+\infty} a^k u[k]u[n-k]$, if it's non-zero, then:

$$k \in [0, n] \tag{40}$$

For the part $\sum_{k=-\infty}^{+\infty} a^k u[k]u[n-N-k]$, if it's non-zero, then:

$$k \in [0, n-N] \tag{41}$$

Then because our main variable is n, we need to have a segmented discussion about different range of n, and use the (40) and (41) to find whether the part is 0 or not.

For $n \in (-\infty, 0)$, the range of k cann't meet (40) or (41), so both part are 0:

$$y[n] = 0 \tag{42}$$

For $n \in [0, N-1]$, the range of k can meet (40), but cann't meet (41):

$$\begin{aligned} y[n] &= \sum_{k=-\infty}^{+\infty} a^k u[k] u[n-k] \\ &= \sum_{k=0}^{n} a^k \end{aligned} \tag{43}$$

For $n \in [N, \infty]$, the range of k can meet (40), and can meet (41):

$$\begin{aligned} y[n] &= \sum_{k=-\infty}^{+\infty} a^k u[k] u[n-k] - \sum_{k=-\infty}^{+\infty} a^k u[k] u[n-N-k] \\ &= \sum_{k=0}^{n} a^k - \sum_{k=0}^{n-N} a^k \end{aligned} \tag{44}$$

Graphical Solution

The graph of $h[n]$ and $x[n]$ are as follows, here we choose $a = 1/2$:

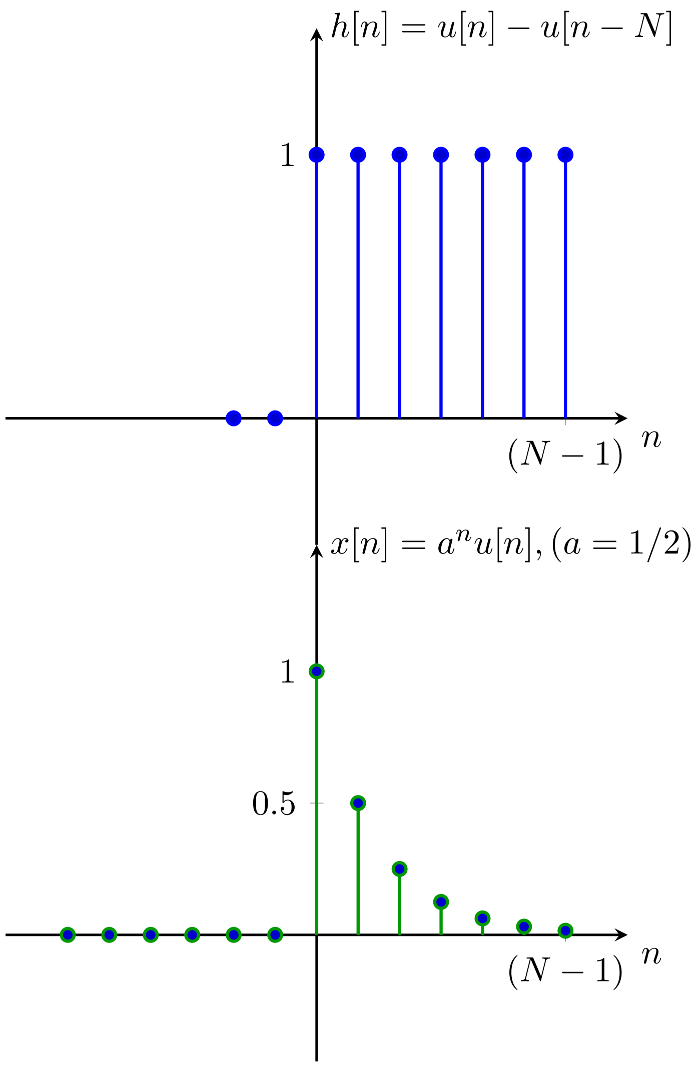

Figure 10: $h[n]$ and $x[n]$

As the summation variable is k, we perform the following operations:

$$h[n-k] = h[-(k-n)] \tag{45}$$

How can we get $h[-(k-n)]$? We can use $h[-k]$ and shift n units:

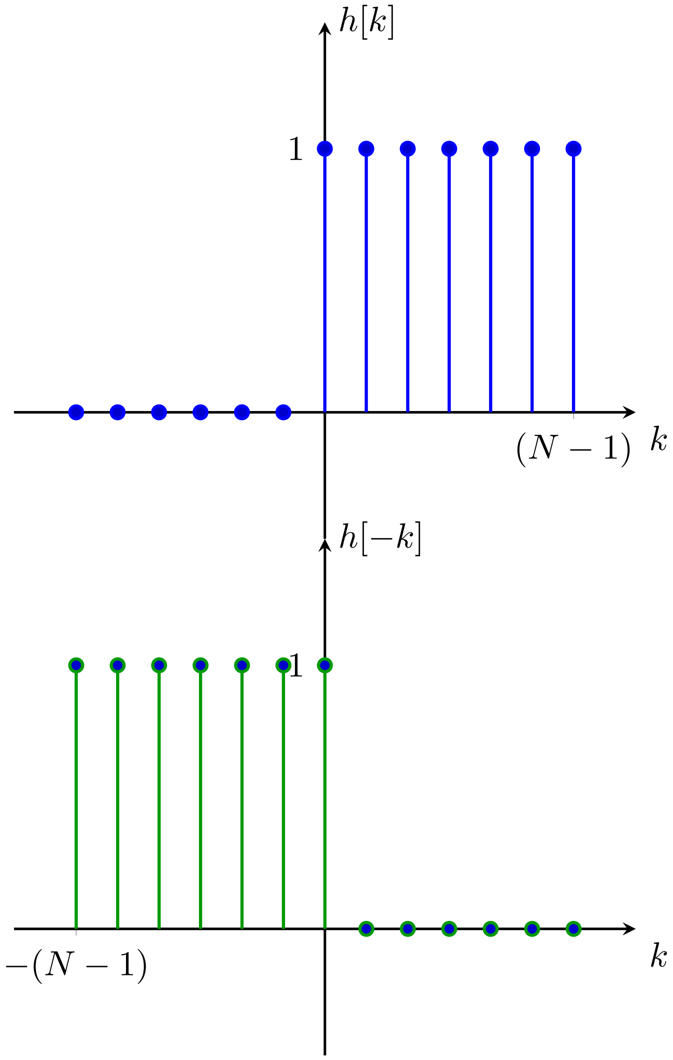

Figure 11: Top: The signal $h[n]$ and time-reversed signal $h[-n]$.

Then shift $h[-k]$ by n units, and perform corresponding multiplication:

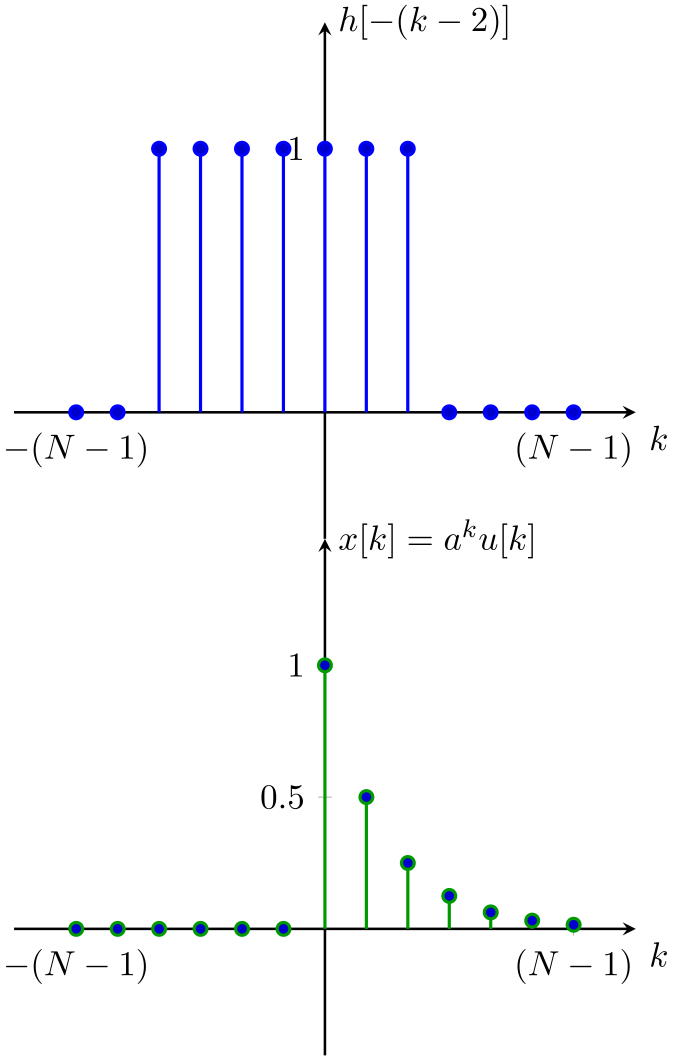

Figure 12: $y[n] = \sum x[k]h[n-k]$ at $n = 2$.

3 New