Adaptive Filtering Based on Projection Method

Masahiro Yukawa

Block Seminar in Elite Master Study Course SIM
December 6 – 8, 2010

The course consists of the following six lectures:

Lecture 1: Introductory Adaptive Filtering
Lecture 2: Basics of Vector Space
Lecture 3: Alternating Projections and NLMS/APA
Lecture 4: Set Theoretic Adaptive Filtering
Lecture 5: Fixed Point Theory of Nonexpansive Mapping
Lecture 6: Topics in Adaptive Filtering

LECTURE 1
Introductory Adaptive Filtering

1.1. Outline of Lecture 1

1.2. Introduction

In the middle of previous century, A. N. Kolmogorov and N. Wiener have independently established Theory of Linear Optimum Filter based on statistical approaches (in frequency domain and time domain, respectively) [**1**–**3**]. The theory has been regarded in digital communications as one of the greatest contributions, as well as Shannon's sampling theorem which is *Magna Carta* in the information age. A path to the adaptive filtering has been opened by Widrow and Hoff in 1960 with their pioneering work of the least mean square (LMS) algorithm [**4**]. Another particular algorithm that had already existed at that time is the recursive least squares (RLS) algorithm. It is mentioned in some literature that Gauss in the late of 18 century had already formulated

the recursive least squares solution (see e.g., [**5**, § 11.7, § 12.6]), although the original work of the RLS algorithm is often credited to Plackett [**6**] in modern times.

Lecture 1 aims to study the basics of adaptive filtering, including a linear system model, the Wiener filter, and the LMS and RLS algorithms with their properties. Throughout this series of lectures, we restrict our attention to real-valued cases for the sake of simplicity.

1.3. Notation

- $\mathbb{R}$: the set of all real numbers
- $\mathbb{N}$: the set of all nonnegative integers
- $\mathbb{N}^* := \mathbb{N} \setminus \{0\}$: the set of all positive integers
- $(\cdot)^\mathsf{T}$: vector (matrix) transpose
- $\mathcal{R}(\cdot)$: range space
- $\mathcal{N}(\cdot)$: null space
- $k \in \mathbb{N}$: time index
- $N \in \mathbb{N}^*$: filter length
- $u_k \in \mathbb{R}$: input signal at time k
- $d_k \in \mathbb{R}$: output signal (i.e., desired response) at time k
- $n_k \in \mathbb{R}$: measurement noise at time k
- $h \in \mathbb{R}$: filter coefficient
- $e_k(\boldsymbol{h}) \in \mathbb{R}$: an output error at time k, a function of a filter $\boldsymbol{h}$
- vectors are represented by bold-face lower-case letters (e.g., $\boldsymbol{a}$)
- matrices are represented by bold-face upper-case letters (e.g., $\boldsymbol{A}$)

1.4. Linear System Model

Let $(u_k)_{k\in\mathbb{N}} \subset \mathbb{R}$ be the input process and $(n_k)_{k\in\mathbb{N}} \subset \mathbb{R}$ the measurement noise process, where $k \in \mathbb{N}$ denotes the time index. We consider a simple linear system model:[1]

$$d_k := \sum_{i=1}^{N} u_{k-i+1} h_i^* + n_k, \ k \in \mathbb{N}, \tag{1.1}$$

where $h_i^* \in \mathbb{R}$, $i = 1, 2, \cdots, N$, stands for the impulse response of the system. In words, the output signal $d_k \in \mathbb{R}$ is a linear combination of the N consecutive input signals u_k, u_{k-1}, $\cdots$, u_{k-N+1} plus the noise

[1]The system model should have the minimum possible complexity according to *Occam's razor*.

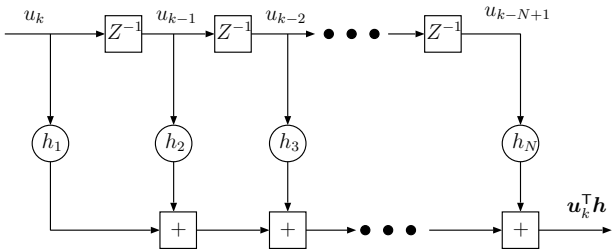


Fig. 1-1. A transversal filter structure for implementing digital filters.

n_k. By means of vector notation, it can simply be stated as follows:

$$d_k = \boldsymbol{u}_k^{\mathsf{T}}\boldsymbol{h}^* + n_k \in \mathbb{R},\ k \in \mathbb{N}, \tag{1.2}$$

where $\boldsymbol{u}_k := [u_k, u_{k-1}, \cdots, u_{k-N+1}]^{\mathsf{T}}$ and $\boldsymbol{h}^* := [h_1^*, h_2^*, \cdots, h_N^*]^{\mathsf{T}}$, which we respectively refer to as the input vector (at time $k \in \mathbb{N}$) and the *estimandum* (a system to be estimated). Unless otherwise stated, the input and output data are assumed available. Regarding the system described above, we may consider the following tasks:

task 1: estimate $\boldsymbol{h}^*$ (e.g., channel estimation), or
task 2: estimate d_k in the form of $\boldsymbol{u}_k^{\mathsf{T}}\boldsymbol{h} = \sum_{i=1}^{N} u_{k-i+1}h_i$ with a linear digital filter $\boldsymbol{h} := [h_1, h_2, \cdots, h_N]^{\mathsf{T}} \in \mathbb{R}^N$ (e.g., echo cancellation).

A transversal filter shown in Fig. 1-1 is commonly used to implement a digital filter. In the figure, Z^{-1} stands for a single delay; input is u_k and output is $\boldsymbol{u}_k^{\mathsf{T}}\boldsymbol{h}$. The two tasks above are related to each other, as clarified in Section 1.6.

1.5. Matrix-form of Wiener-Hopf Equations

We restrict our attention to the case of discrete time, which is particularly simpler compared to the continuous time case which has actually been treated by Wiener; the interested readers may refer to [**7**]. Apart from the Wiener's original philosophy, we present the Wiener-Hopf equations in a simplest possible way. Let us assume that the input and output processes, $(u_k)_{k\in\mathbb{N}}$ and $(d_k)_{k\in\mathbb{N}}$, are *jointly wide-sense stationary*

stochastic processes.[2] Consider **task 2** in the previous section. A natural requirement for a filter $\boldsymbol{h} \in \mathbb{R}^N$ would be suppressing maximally the error signal

$$e_k(\boldsymbol{h}) := \boldsymbol{u}_k^{\mathsf{T}}\boldsymbol{h} - d_k,\ \forall k \in \mathbb{N}. \tag{1.3}$$

How can we formalize the problem mathematically? As $e_k(\boldsymbol{h})$ can take negative values, it is nonsense to minimize $e_k(\boldsymbol{h})$ itself. One should minimize its magnitude. For the sake of convenience, a typical way is considering the squared error $e_k^2(\boldsymbol{h})$. Because we wish to suppress the error for any possible pair of input and output (or input and noise), we take *expectation*, namely an *ensemble average*, of the squared error. The resultant criterion

$$f_{\mathrm{MSE}}(\boldsymbol{h}) := E\left\{e_k^2(\boldsymbol{h})\right\} = E\left\{\left(d_k - \boldsymbol{u}_k^{\mathsf{T}}\boldsymbol{h}\right)^2\right\},\ \boldsymbol{h} \in \mathbb{R}^N, \tag{1.4}$$

is called the *mean squared error (MSE)*. The problem is now formalized simply as follows: minimize $f_{\mathrm{MSE}}(\boldsymbol{h})$. It is easily verified that

$$f_{\mathrm{MSE}}(\boldsymbol{h}) = \boldsymbol{h}^{\mathsf{T}}E\{\boldsymbol{u}_k\boldsymbol{u}_k^{\mathsf{T}}\}\boldsymbol{h} - 2\boldsymbol{h}^{\mathsf{T}}E\{\boldsymbol{u}_k d_k\} + E\{d_k^2\} \tag{1.5}$$

$$= \boldsymbol{h}^{\mathsf{T}}\boldsymbol{R}\boldsymbol{h} - 2\boldsymbol{h}^{\mathsf{T}}\boldsymbol{p} + E\{d_k^2\}, \tag{1.6}$$

where $\boldsymbol{R} := E\{\boldsymbol{u}_k\boldsymbol{u}_k^{\mathsf{T}}\} \in \mathbb{R}^{N\times N}$ is the autocorrelation matrix of the input and $\boldsymbol{p} := E\{\boldsymbol{u}_k d_k\} \in \mathbb{R}^N$ the cross-correlation vector between the input and the output. Therefore, a minimizer of f_{MSE} is characterized as a solution of the following partial differential equation for $\boldsymbol{h} =: [h_1, h_2, \cdots, h_N]^{\mathsf{T}}$:

$$\frac{\partial f_{\mathrm{MSE}}(\boldsymbol{h})}{\partial \boldsymbol{h}} := \left[\frac{\partial f_{\mathrm{MSE}}(\boldsymbol{h})}{\partial h_1}, \frac{\partial f_{\mathrm{MSE}}(\boldsymbol{h})}{\partial h_2}, \cdots, \frac{\partial f_{\mathrm{MSE}}(\boldsymbol{h})}{\partial h_N}\right]^{\mathsf{T}} \tag{1.7}$$

$$= 2\boldsymbol{R}\boldsymbol{h} - 2\boldsymbol{p} = \boldsymbol{0}, \tag{1.8}$$

leading to the following normal equation, so-called *Wiener-Hopf equations*:

$$\boldsymbol{R}\boldsymbol{h} = \boldsymbol{p}. \tag{1.9}$$

If $\boldsymbol{R}$ is nonsingular (as commonly assumed), (1.9) has the unique solution $\boldsymbol{R}^{-1}\boldsymbol{p}$, which is widely referred to as the *Wiener solution* (or

[2] A stochastic process is said to be *strictly stationary* if its statistical properties are invariant to a time shift. *Wide-sense stationarity* requires weaker conditions than the strict stationarity. The theory of Wiener filters has been established on the general assumption of jointly wide-sense stationarity, but the readers who are not familiar with stochastic processes may think that the input and output processes are assumed to have their statistical properties invariant to a time shift.

the *Wiener filter*). We have seen above that the Wiener-Hopf equations (1.9) have been derived by minimizing the MSE function, thus the Wiener filter is also called the *minimum MSE (MMSE) filter*.

1.6. Wiener Solution and its MSE

Rewrite (1.1) as

$$d_k := z_k + n_k \in \mathbb{R},\ k \in \mathbb{N}, \tag{1.10}$$

where $z_k := \boldsymbol{u}_k^{\mathsf{T}} \boldsymbol{h}^*$. Without any loss of generality, we assume the following:

- $E\{z_k^2\} > 0$,
- $\boldsymbol{p} \in \mathcal{R}(\boldsymbol{R})$ so (1.9) has a solution, and
- $\boldsymbol{p}_{n\boldsymbol{u}} := E\{n_k \boldsymbol{u}_k\} \in \mathcal{R}(\boldsymbol{R})$.

Let $\widetilde{\boldsymbol{h}} \in \mathbb{R}^N$ satisfy $\boldsymbol{R}\widetilde{\boldsymbol{h}} = \boldsymbol{p}_{n\boldsymbol{u}}$. We then obtain

$$\begin{aligned} f_{\mathrm{MSE}}(\boldsymbol{h}) &= E\{[(\boldsymbol{h} - \boldsymbol{h}^*)^{\mathsf{T}} \boldsymbol{u}_k - n_k]^2\} \\ &= (\boldsymbol{h} - \boldsymbol{h}^*)^{\mathsf{T}} \boldsymbol{R} (\boldsymbol{h} - \boldsymbol{h}^*) - 2(\boldsymbol{h} - \boldsymbol{h}^*)^{\mathsf{T}} \boldsymbol{p}_{n\boldsymbol{u}} + E\{n_k^2\} \\ &= (\boldsymbol{h} - \boldsymbol{h}^* - \widetilde{\boldsymbol{h}})^{\mathsf{T}} \boldsymbol{R} (\boldsymbol{h} - \boldsymbol{h}^* - \widetilde{\boldsymbol{h}}) - \widetilde{\boldsymbol{h}}^{\mathsf{T}} \boldsymbol{R} \widetilde{\boldsymbol{h}} + E\{n_k^2\} \\ &\geq E\{n_k^2\} - \widetilde{\boldsymbol{h}}^{\mathsf{T}} \boldsymbol{R} \widetilde{\boldsymbol{h}}. \end{aligned} \tag{1.11}$$

Here, the last inequality holds because of the positive semi-definiteness of $\boldsymbol{R}$.[3] The equation (1.11) indicates that a minimizer of f_{MSE} is characterized as such a vector $\boldsymbol{h}_{\mathrm{W}}$ that satisfies

$$\boldsymbol{h}_{\mathrm{W}} - \boldsymbol{h}^* - \widetilde{\boldsymbol{h}} \in \mathcal{N}(\boldsymbol{R}),\ \text{i.e.,}\ \boldsymbol{R}(\boldsymbol{h}_{\mathrm{W}} - \boldsymbol{h}^* - \widetilde{\boldsymbol{h}}) = \boldsymbol{0}. \tag{1.12}$$

As $\boldsymbol{0} \in \mathcal{N}(\boldsymbol{R})$, (1.12) assures that $\boldsymbol{h}_{\mathrm{W}} := \boldsymbol{h}^* + \widetilde{\boldsymbol{h}}$ is a Wiener solution (a solution to (1.9)) for any $\widetilde{\boldsymbol{h}}$ such that $\boldsymbol{R}\widetilde{\boldsymbol{h}} = \boldsymbol{p}_{n\boldsymbol{u}}$. If $\boldsymbol{R}$ is nonsingular, $\widetilde{\boldsymbol{h}} := \boldsymbol{R}^{-1}\boldsymbol{p}_{n\boldsymbol{u}}$ is unique and the Wiener solution is $\boldsymbol{h}_{\mathrm{W}} := \boldsymbol{h}^* + \boldsymbol{R}^{-1}\boldsymbol{p}_{n\boldsymbol{u}} = \boldsymbol{R}^{-1}\boldsymbol{p}$.

In many scenarios, the noise is statistically orthogonal to the input; i.e., $\boldsymbol{p}_{n\boldsymbol{u}} = \boldsymbol{0}$. In this case, $\widetilde{\boldsymbol{h}} := \boldsymbol{0}$ satisfies $\boldsymbol{R}\widetilde{\boldsymbol{h}} = \boldsymbol{p}_{n\boldsymbol{u}}$, meaning that $\boldsymbol{h}^*$ is a Wiener solution. If in particular $\boldsymbol{R}$ is nonsingular, $\boldsymbol{h}^*$ is the unique Wiener solution. Also, $\boldsymbol{R}\widetilde{\boldsymbol{h}} = \boldsymbol{p}_{n\boldsymbol{u}} = \boldsymbol{0}$ implies

$$f_{\mathrm{MSE}}(\boldsymbol{h}) \geq E\{n_k^2\}. \tag{1.13}$$

In the remainder of this lecture notes, we assume (i) the nonsingularity of $\boldsymbol{R}$ and (ii) the orthogonality between the input and noise signals,

[3]A matrix $\boldsymbol{A} \in \mathbb{R}^{N \times N}$ is positive semi-definite if and only if $\boldsymbol{x}^{\mathsf{T}}\boldsymbol{A}\boldsymbol{x} \geq 0$ for all $\boldsymbol{x} \in \mathbb{R}^N$ [8]. The positive semi-definiteness of $\boldsymbol{R}$ can be verified by $\boldsymbol{x}^{\mathsf{T}}\boldsymbol{R}\boldsymbol{x} = \boldsymbol{x}^{\mathsf{T}} E\{\boldsymbol{u}_k \boldsymbol{u}_k^{\mathsf{T}}\}\boldsymbol{x} = E\{\boldsymbol{x}^{\mathsf{T}}\boldsymbol{u}_k \boldsymbol{u}_k^{\mathsf{T}}\boldsymbol{x}\} = E\{(\boldsymbol{u}_k^{\mathsf{T}}\boldsymbol{x})^2\} \geq 0,\ \forall \boldsymbol{x} \in \mathbb{R}^N$.

and we do not explicitly distinguish **task 1** and **task 2** as $\boldsymbol{h}^*$ is the optimal solution for both tasks.

The ratio between the signal power and the noise power at the output, *i.e., the signal to noise ratio (SNR)*, is defined as

$$\mathrm{SNR} := 10 \log_{10} \frac{E\{z_k^2\}}{E\{n_k^2\}}\ [\mathrm{dB}].$$

Instead of MSE itself, we employ the normalized MSE defined as

$$f_{\mathrm{NMSE}}(\boldsymbol{h}) := \frac{f_{\mathrm{MSE}}(\boldsymbol{h})}{E\{z_k^2\}} \geq 10^{-\mathrm{SNR}/10},\ \boldsymbol{h} \in \mathbb{R}^N. \tag{1.14}$$

The normalized MSE is convenient in numerical simulations to see how close the achievement of an algorithm to the theoretical lower bound.

1.7. Adaptive Filtering

We have seen that a Wiener solution can be obtain by solving the normal equations $\boldsymbol{R}\boldsymbol{h} = \boldsymbol{p}$, which is a simple linear problem although the difficulty to solve it depends on the condition of $\boldsymbol{R}$. However, the statistical information $\boldsymbol{R}$ and $\boldsymbol{p}$ is not available *a priori* and thus should be estimated from data samples in practice. One may think that we can collect a sufficiently large number of data samples to obtain reasonable estimates of $\boldsymbol{R}$ and $\boldsymbol{p}$, and then solve the corresponding normal equations. This is called *batch* processing. In real-time systems, it is required to give an output every time when one receives data. A straightforward way would be to update the estimates of $\boldsymbol{R}$ and $\boldsymbol{p}$ at each time instant and solve the corresponding normal equations again and again. Unfortunately, it is computationally expensive and is not affordable when the filter length N becomes large.

It is therefore desired to update a filter $\boldsymbol{h}_k$, $k \in \mathbb{N}$, iteratively in such a way that an 'error' hopefully becomes small as time goes by. This is called *adaptive* processing. As the filter coefficients $h_k^{(i)}$ with $\boldsymbol{h}_k =: [h_k^{(1)}, h_k^{(2)}, \cdots, h_k^{(N)}] \in \mathbb{R}^N$ change adaptively in time, the filtering process is called adaptive filtering (see Fig. 1-2). The remarkable advantages of the adaptive processing over the batch one include *low computational costs* and *adaptivity to the system changes*. An adaptive filtering algorithm undertakes the role of adjusting the coefficients $h_k^{(i)}$s. A bit more mathematically, an adaptive filtering algorithm generates a vector sequence $(\boldsymbol{h}_k)_{k \in \mathbb{N}} \subset \mathbb{R}^N$ in a recursive way. We present two classical approaches and summarize their advantages and disadvantages below.

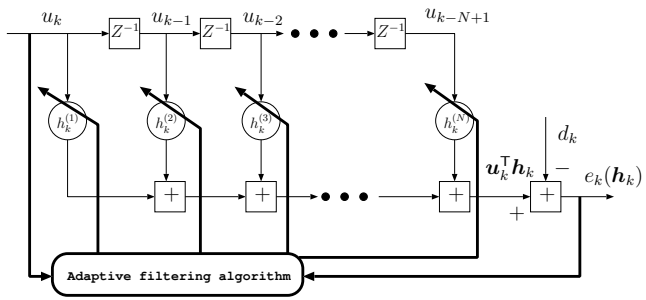


Fig. 1-2. The concept of adaptive filters implemented with the transversal filter structure.

1.8. Least Mean Square Algorithm

The Least Mean Square (LMS) algorithm is in short an instantaneous approximation of the gradient algorithm (also known as the steepest descent algorithm). The gradient method is an iterative method to minimize a differentiable function $f : \mathbb{R}^N \to \mathbb{R}$ by

$$\boldsymbol{h}_{k+1} := \boldsymbol{h}_k - \lambda \boldsymbol{\nabla} f(\boldsymbol{h}_k),\ k \in \mathbb{N}, \tag{1.15}$$

for an initial point $\boldsymbol{h}_0 \in \mathbb{R}^N$, where $\lambda > 0$ is the step size and $\boldsymbol{\nabla} f(\boldsymbol{h}_k)$ denotes the gradient of f at $\boldsymbol{h}_k$; $\boldsymbol{\nabla} f(\boldsymbol{h}) := \dfrac{\partial f(\boldsymbol{h})}{\partial \boldsymbol{h}}$, $\boldsymbol{h} \in \mathbb{R}^N$. Intuitively, $-\boldsymbol{\nabla} f(\boldsymbol{h}_k)$ gives a *steepest descent* direction in which the value of f is maximally decreased in the neighborhood of $\boldsymbol{h}_k$, and therefore (1.15) reduces the function value provided λ is sufficiently small. Referring to (1.6), the gradient of the MSE function is given by

$$\boldsymbol{\nabla} f_{\mathrm{MSE}}(\boldsymbol{h}) = 2\boldsymbol{R}\boldsymbol{h} - 2\boldsymbol{p}. \tag{1.16}$$

The gradient method to solve the Wiener-Hopf equations is thus given by

$$\boldsymbol{h}_{k+1} := \boldsymbol{h}_k - \lambda \boldsymbol{\nabla} f_{\mathrm{MSE}}(\boldsymbol{h}_k) = \boldsymbol{h}_k - 2\lambda(\boldsymbol{R}\boldsymbol{h}_k - \boldsymbol{p}),\ k \in \mathbb{N}. \tag{1.17}$$

Replacing $\boldsymbol{R}(:= E\{\boldsymbol{u}_k \boldsymbol{u}_k^\mathsf{T}\})$ and $\boldsymbol{p}(:= E\{\boldsymbol{u}_k d_k\})$ in $\boldsymbol{\nabla} f_{\mathrm{MSE}}(\boldsymbol{h})(= 2\boldsymbol{R}\boldsymbol{h} - 2\boldsymbol{p})$ respectively by instantaneous approximations $\boldsymbol{u}_k \boldsymbol{u}_k^\mathsf{T}$ and $\boldsymbol{u}_k d_k$, we obtain an instantaneous approximation of the gradient at each k:

$$\widehat{\boldsymbol{\nabla}}_k f_{\mathrm{MSE}}(\boldsymbol{h}) := 2\boldsymbol{u}_k \boldsymbol{u}_k^\mathsf{T}\boldsymbol{h} - 2\boldsymbol{u}_k d_k = 2(\boldsymbol{u}_k^\mathsf{T}\boldsymbol{h} - d_k)\boldsymbol{u}_k = 2e_k(\boldsymbol{h})\boldsymbol{u}_k. \tag{1.18}$$

Based on the replacement above, the LMS algorithm is given as follows:

$$\boldsymbol{h}_{k+1} := \boldsymbol{h}_k - \lambda \widehat{\boldsymbol{\nabla}}_k f_{\mathrm{MSE}}(\boldsymbol{h}_k) = \boldsymbol{h}_k - 2\lambda e_k(\boldsymbol{h}_k)\boldsymbol{u}_k,\ k \in \mathbb{N}, \tag{1.19}$$

where $\lambda > 0$ should be 'sufficiently' small for stability. An upper bound of λ depends on the definition of stability. A widely-used upper bound is $2/\sigma_{\boldsymbol{R}}^{(\max)}$ (which is required for stability of zero-order solutions of LMS filters; cf. [**7**, p. 306]), where $\sigma_{\boldsymbol{R}}^{(\max)}$ is the maximum eigenvalue of $\boldsymbol{R}$. By (1.19), one can easily see that $\boldsymbol{h}_k - \boldsymbol{h}_0$ at time $k \in \mathbb{N}$ is a linear combinations of $\boldsymbol{u}_0$, $\boldsymbol{u}_1$, $\cdots$, $\boldsymbol{u}_{k-1}$; i.e., $\boldsymbol{h}_k - \boldsymbol{h}_0 \in \mathrm{span}(\boldsymbol{u}_0, \boldsymbol{u}_1, \cdots, \boldsymbol{u}_{k-1})$. The way of replacing the gradient term in the gradient method by its approximation based on a single measurement is generically called *stochastic gradient method (or stochastic gradient descent method).*

1.9. Recursive Least Squares Algorithm

The Recursive Least Squares (RLS) algorithm is in short a method to solve *approximate* Wiener-Hopf equations at every time instant somewhat efficiently by a recursive formula based on the matrix inversion lemma. To be precise, $\boldsymbol{R}$ and $\boldsymbol{p}$ are approximated by sample averages respectively as $\boldsymbol{R}_{k+1} := \boldsymbol{u}_k \boldsymbol{u}_k^\mathsf{T} + \gamma \boldsymbol{R}_k$ and $\boldsymbol{p}_{k+1} := \boldsymbol{u}_k d_k + \gamma \boldsymbol{p}_k$ for initial estimates $\boldsymbol{R}_0 \in \mathbb{R}^{N\times N}$ and $\boldsymbol{p}_0 \in \mathbb{R}^N$. Here, $\gamma \in (0, 1)$ is called the *forgetting factor* and should be close to one (e.g., $\gamma = 0.99$) for stability. The RLS algorithm solves the normal equation $\boldsymbol{R}_{k+1}\boldsymbol{h} = \boldsymbol{p}_{k+1}$ at each time $k \in \mathbb{N}$ with the following recursive formula:

$$\boldsymbol{R}_{k+1}^{-1} = \gamma^{-1}\boldsymbol{R}_k^{-1} - \frac{\gamma^{-2}\boldsymbol{R}_k^{-1}\boldsymbol{u}_k\boldsymbol{u}_k^\mathsf{T}\boldsymbol{R}_k^{-1}}{1 + \gamma^{-1}\boldsymbol{u}_k^\mathsf{T}\boldsymbol{R}_k^{-1}\boldsymbol{u}_k}. \tag{1.20}$$

We would not describe the whole recursions of RLS, as it can be easily found in the literature [**5, 7**]. Instead, we derive its equivalent expression similar to (1.19). Right-multiplying both sides of (1.20) by $\boldsymbol{p}_{k+1}(:= \boldsymbol{u}_k d_k + \gamma \boldsymbol{p}_k)$ yields

$$\boldsymbol{R}_{k+1}^{-1}\boldsymbol{p}_{k+1} = \left(\gamma^{-1}\boldsymbol{R}_k^{-1} - \frac{\gamma^{-2}\boldsymbol{R}_k^{-1}\boldsymbol{u}_k\boldsymbol{u}_k^\mathsf{T}\boldsymbol{R}_k^{-1}}{1 + \gamma^{-1}\boldsymbol{u}_k^\mathsf{T}\boldsymbol{R}_k^{-1}\boldsymbol{u}_k}\right)(\boldsymbol{u}_k d_k + \gamma \boldsymbol{p}_k). \tag{1.21}$$

By letting $\boldsymbol{h}_{k+1} := \boldsymbol{R}_{k+1}^{-1}\boldsymbol{p}_{k+1}$ and $\boldsymbol{h}_k := \boldsymbol{R}_k^{-1}\boldsymbol{p}_k$, (1.20) becomes with simple manipulations

$$\boldsymbol{h}_{k+1} := \boldsymbol{h}_k - \lambda_k e_k(\boldsymbol{h}_k)\boldsymbol{R}_k^{-1}\boldsymbol{u}_k, \tag{1.22}$$

where $\lambda_k := (\boldsymbol{u}_k^\mathsf{T}\boldsymbol{R}_k^{-1}\boldsymbol{u}_k + \gamma)^{-1} \in (0, \gamma^{-1})$. As will be clarified later, a simple modification of (1.22) reveals that RLS can be interpreted as an iterative projection method onto hyperplanes with *time-dependent metric.*

Exercise 1. Derive (1.22).

1.10. Tradeoff Issues

The LMS algorithm has $O(N)$ complexity, hence it is simple to implement, and also it is robust to disturbance (The robustness has been theoretically proved based on H^∞ theory [**9**]). However, it is well known that LMS suffers from slow convergence when the input signal is correlated. In practice, LMS is robust as far as the step size parameter λ is chosen to be sufficiently small, which however results in slow convergence.

The RLS algorithm, on the other hand, exhibits very fast convergence even for highly correlated input signals. This however comes at the price of (i) N^2 computational complexity and (ii) poor tracking performance when the estimandum $\boldsymbol{h}^*$ changes abruptly. For improving the tracking performance of RLS, one needs to decrease the value of γ to forget quickly the information acquired before the change of $\boldsymbol{h}^*$. This however results in a large estimation error at steady state. This is because a small number of data are taken into account in computing an arithmetic average to estimate $\boldsymbol{R}$ and $\boldsymbol{p}$ and hence the estimates tend to become inaccurate. Moreover, if the γ value is too small, such as $\gamma := 0.9$, the algorithm tends to diverge.

To overcome the tradeoff issues mentioned above, a significant amount of efforts has been devoted. In this series of lectures, we focus on a direction of improving the LMS algorithm. The contents of the following lecture, basics of vector spaces, enable us to get a nice geometric interpretation of the improved algorithms, which greatly helps our understanding.

LECTURE 2
Basics of Vector Space

2.1. Outline of Lecture 2

2.2. Introduction

In Lecture 1, we have seen that an adaptive filtering algorithm generates a sequence of filters $(\boldsymbol{h}_k)_{k\in\mathbb{N}} \subset \mathbb{R}^N$ in a recursive manner. The fundamental question would naturally arise: *does the sequence* $(\boldsymbol{h}_k)_{k\in\mathbb{N}}$ *converge, or under what conditions does the sequence* $(\boldsymbol{h}_k)_{k\in\mathbb{N}}$ *converge?* To discuss the convergence issue, the notion of *vector spaces* — or more specifically *Hilbert spaces* — provides a convenient and reasonably general stage [**10**–**12**].

Why do we need to learn such abstract mathematics? If we solely want to discuss about a convergence property of a specific adaptive filtering algorithm such as LMS, it would be sufficient to define a 'distance' between $\boldsymbol{a} := [a_1, a_2, \cdots, a_N]^\mathsf{T}$ and $\boldsymbol{b} := [b_1, b_2, \cdots, b_N]^\mathsf{T}$ in $\mathbb{R}^N$ as $\|\boldsymbol{a}-\boldsymbol{b}\| := \sqrt{\sum_{n=1}^N (a_n - b_n)^2}$, and say that the filter sequence

$(\boldsymbol{h}_k)_{k\in\mathbb{N}} \subset \mathbb{R}^N$ converges to some $\hat{\boldsymbol{h}} \in \mathbb{R}^N$ when $\left\|\boldsymbol{h}_k - \hat{\boldsymbol{h}}\right\| \to 0$ as $k \to \infty$. However, is this a unique way to measure the closeness of $\boldsymbol{a}$ and $\boldsymbol{b}$? In fact, there are infinitely many ways to define a distance; e.g., $\|\boldsymbol{a}-\boldsymbol{b}\| := \sqrt{\sum_{n=1}^N w_n(a_n - b_n)^2}$ for an arbitrarily chosen $w_n > 0$, $n = 1, 2, \cdots, N$. The recent researches of adaptive filtering have shown that appropriate designs of a distance function yield efficient adaptation rules. If we perform an analysis for a specific distance function, we need to perform again an analysis when a different distance function is employed. An analysis based on Hilbert spaces eliminates such an exhausting repetition, although it is only a 'stone' on the mountain of benefits from the study of Hilbert spaces (cf. [**10**–**13**]).

Lecture 2 aims to present some definitions and basic results of vector spaces.

2.3. Vector Spaces

We repeat that we solely consider the real-valued cases throughout the lectures. Intuitively, a vector space is a set of elements enjoying the following properties: (i) it contains a *null (zero) vector*, (ii) each element is allowed to be *multiplied by any scalar* (i.e., any real number in this lecture), and (iii) each pair of elements is allowed to be *added* with each other. Accordingly, two operations are provided: addition and scalar multiplication.

Definition 2.1. A set X is said to be a *vector space* (or a *linear space*) if *addition* that associates any pair $(\boldsymbol{x}, \boldsymbol{y}) \in X \times X$ with $\boldsymbol{x}+\boldsymbol{y} \in X$ and *scalar multiplication* that associates any pair $(\alpha, \boldsymbol{x}) \in \mathbb{R} \times X$ with $\alpha\boldsymbol{x} \in X$ satisfy the following conditions, respectively.

(a) For any $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z} \in X$:
  A1. $\boldsymbol{x}+\boldsymbol{y} = \boldsymbol{y}+\boldsymbol{x}$ (commutative law)
  A2. $(\boldsymbol{x}+\boldsymbol{y})+\boldsymbol{z} = \boldsymbol{x}+(\boldsymbol{y}+\boldsymbol{z})$ (associative law)
  A3. There exists a null vector $\boldsymbol{0}$ such that $\boldsymbol{x}+\boldsymbol{0} = \boldsymbol{x}$, $\forall \boldsymbol{x} \in X$.
(b) For any $\boldsymbol{x}, \boldsymbol{y} \in X$ and any $\alpha, \beta \in \mathbb{R}$:
  M1. $\alpha(\boldsymbol{x}+\boldsymbol{y}) = \alpha\boldsymbol{x}+\alpha\boldsymbol{y}$ (distributive law 1)
  M2. $(\alpha+\beta)\boldsymbol{x} = \alpha\boldsymbol{x}+\beta\boldsymbol{x}$ (distributive law 2)
  M3. $(\alpha\beta)\boldsymbol{x} = \alpha(\beta\boldsymbol{x})$ (associative law)
  M4. $0\boldsymbol{x} = \boldsymbol{0}$, $1\boldsymbol{x} = \boldsymbol{x}$

We note that an element of a vector space is called a *vector*, or a *point*, and denoted by a bold-face lower-case letters throughout Lecture 2. The vector $(-1)\boldsymbol{x}$ is denoted by $-\boldsymbol{x}$ for convenience, and we have

$\boldsymbol{x}+(-\boldsymbol{x})=(1-1)\boldsymbol{x}=\boldsymbol{0}$ by (M2) and (M4): $-\boldsymbol{x}$ is called the additive inverse of $\boldsymbol{x}$.

Example 2.2.

(a) The simplest example of a vector space would be the set of real numbers $\mathbb{R}$. Both addition and scalar multiplication are defined in an ordinary way. It is readily verified that these ordinary operations satisfy the conditions (A1)–(A3) and (M1)–(M4). In other words, the concept of vector space is a generalization of $\mathbb{R}$ with the ordinary operations.

(b) A slight extension of $\mathbb{R}$ is the N dimensional Euclidean space $\mathbb{R}^N$ whose elements take the form of $[x_1, x_2, \cdots, x_N]^\mathsf{T}$ with real components x_i. As usual, addition and scalar multiplication are performed in a componentwise fashion. Thus, for any $\boldsymbol{x} := [x_1, x_2, \cdots, x_N]^\mathsf{T} \in \mathbb{R}^N$, $\boldsymbol{y} := [y_1, y_2, \cdots, y_N]^\mathsf{T} \in \mathbb{R}^N$, and $\alpha \in \mathbb{R}$, we have

$$\boldsymbol{x}+\boldsymbol{y} = [x_1+y_1, x_2+y_2, \cdots, x_N+y_N]^\mathsf{T} \tag{2.3}$$

$$\alpha\boldsymbol{x} = [\alpha x_1, \alpha x_2, \cdots, \alpha x_N]^\mathsf{T}. \tag{2.4}$$

(c) The set of all real-valued functions defined on $\mathbb{R}$, $X := \{\boldsymbol{f} : \mathbb{R} \to \mathbb{R}\}$, forms a vector space with the operations given as follows. For any $\boldsymbol{f}, \boldsymbol{g} \in X$ and $\alpha \in \mathbb{R}$, we have

$$\boldsymbol{f}+\boldsymbol{g} : \mathbb{R} \to \mathbb{R}, x \mapsto \boldsymbol{f}(x)+\boldsymbol{g}(x) \tag{2.5}$$

$$\alpha\boldsymbol{f} : \mathbb{R} \to \mathbb{R}, x \mapsto \alpha\boldsymbol{f}(x). \tag{2.6}$$

See [**10, 11**] for other examples of vector spaces.

2.4. Subspaces

Definition 2.7. A subset M of a vector space X is said to be a *subspace* (or a *linear subspace*) if M itself is a vector space under the same operations of addition and scalar multiplication defined on X.

The following theorem provides a simple way to check whether a given subset is a subspace.

Theorem 2.8. *A nonempty subset M of a vector space X is a subspace of X if and only if*

(a) $\boldsymbol{x}+\boldsymbol{y} \in M$, $\forall \boldsymbol{x}, \boldsymbol{y} \in M$, *and*
(b) $\alpha\boldsymbol{x} \in M$, $\forall \boldsymbol{x} \in M$, $\forall \alpha \in \mathbb{R}$.

The necessary and sufficient condition to be a subspace can also be expressed as follows: $\alpha\boldsymbol{x}+\beta\boldsymbol{y} \in M$, $\forall \boldsymbol{x}, \boldsymbol{y} \in M$, $\forall \alpha, \beta \in \mathbb{R}$.

Exercise 2. Show that the intersection of two subspaces of a vector space X is a subspace of X.

We are now interested in how 'large' a vector space, or a subspace, is. The 'size' of a space is related to the number of parameters that we need for specifying each element of the space; if we just say 'a space', it means either a vector space or a subspace. To discuss it clearly, the fundamental concepts of *bases* and *dimensionality* should be introduced. We need first to define *linear dependency* and *span*.

Definition 2.9. Given a space X, let $S := \{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_m\} \subset X$ for some $m \in \mathbb{N}^*$. A vector expressed in the form $\sum_{i=1}^m \alpha_i \boldsymbol{x}_i$, $\alpha_i \in \mathbb{R}$, is called a *linear combination* of $\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_m$. The set S is said to be *linearly independent* if $\sum_{i=1}^m \alpha_i \boldsymbol{x}_i = \boldsymbol{0} \Leftrightarrow \alpha_1 = \alpha_2 = \cdots = \alpha_m = 0$. Otherwise S is said to be *linearly dependent*. The set $\mathrm{span}(S) := \{\sum_{i=1}^m \alpha_i \boldsymbol{x}_i : \alpha_i \in \mathbb{R} \text{ for } i = 1, 2, \cdots, m\}$ is called the *span* of S. If $\mathrm{span}(S) = X$, it is said that S spans X.

Exercise 3. Suppose $S := \{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_m\}$ is linearly dependent. Then show that there exists an element of S that can be expressed as a linear combination of the other elements of S.

Exercise 4. Show that $\mathrm{span}(S)$ is a subspace of X.

Definition 2.10. A subset S of a vector space X is said to be a *basis* of X if (i) it is linearly independent and (ii) it spans X. If a basis of X has a finite number of elements, X is said to be finite dimensional; otherwise, it is said to be infinite dimensional. For a finite dimensional space X with its basis S, $\dim(X) := |S|$ is said to be the dimension of X, where $|S|$ denotes the cardinality of S.

Note that any vector space has a basis.

Exercise 5. Show that any two bases of a finite-dimensional space have the same number of elements.

Definition 2.11. Let M be a subspace of a vector space X. Then, a translation of M by some $\boldsymbol{v} \in X$, defined as $V := M + \boldsymbol{v} := \{\boldsymbol{x}+\boldsymbol{v} : \boldsymbol{x} \in M\}$, is called a *linear variety*.

Proposition 2.12. *A subset V of a vector space is a linear variety if and only if $\alpha\boldsymbol{x}+(1-\alpha)\boldsymbol{y} \in V$, $\forall \boldsymbol{x}, \boldsymbol{y} \in V$, $\forall \alpha \in \mathbb{R}$.*

A vector expressed in the form $\sum_{i=1}^m \alpha_i \boldsymbol{x}_i$ with $\alpha_i \in \mathbb{R}$ satisfying $\sum_{i=1}^m \alpha_i = 1$ is called an *affine combination* of $\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_m$. The term $\alpha\boldsymbol{x}+(1-\alpha)\boldsymbol{y}$ in Proposition 2.12 is an affine combination of $\boldsymbol{x}$ and $\boldsymbol{y}$, and linear variety is also called *affine set*.

2.5. Limit of a Sequence of Real Numbers

Convergence of a sequence of vectors is defined based on convergence of real-number sequence, which is defined as follows.

Definition 2.13. Let $(a_n)_{n\in\mathbb{N}} \subset \mathbb{R}$ be a sequence of real numbers. Then the sequence is said to *converge* to $a \in \mathbb{R}$ if $|a_n - a| \to 0$ as $n \to \infty$; we express this as $\lim_{n\to\infty} a_n = a$, or $a_n \to a$, $n \to \infty$. A more rigorous definition is the following: the sequence $(a_n)_{n\in\mathbb{N}} \subset \mathbb{R}$ is said to *converge* to $a \in \mathbb{R}$ if for any $\epsilon > 0$ there exists $N(\epsilon) \in \mathbb{N}$ such that $|a_n - a| < \epsilon$ for all $n \geq N(\epsilon)$. If there exists $a \in \mathbb{R}$ to which $(a_n)_{n\in\mathbb{N}} \subset \mathbb{R}$ converges, then the sequence is said to be *convergent;* otherwise it is said to *diverge*.

The limit presented above can only be defined for convergent sequences. In the case that we do not know whether a sequence is convergent, we can use the notion of *limit superior* and *limit inferior*, both of which can be defined for any real-number sequences.

Definition 2.14. A set $S \subset \mathbb{R}$ of real numbers is said to be *bounded above* (or *bounded below*) if there exists $\alpha \in \mathbb{R}$ such that $x \leq \alpha$ for all $x \in S$ (or $x \geq \alpha$ for all $x \in S$). If S is bounded above (or below), such an α is called an *upper bound* (or *lower bound*) of S, and the smallest upper bound (or the largest lower bound) is called *supremum* (or *infimum*). We denote the *supremum* (or *infimum*) of S as $\sup_{x\in S}(x)$ (or $\inf_{x\in S}(x)$).

Proposition 2.15. *A real-number sequence $(a_n)_{n\in\mathbb{N}} \subset \mathbb{R}$ is said to be monotonically increasing (or decreasing) if $a_n \leq a_{n+1}$ (or $a_{n+1} \leq a_n$) for any $n \in \mathbb{N}$. A monotonically increasing (or decreasing) sequence is convergent if and only if it is bounded above (or below). In particular, if $(a_n)_{n\in\mathbb{N}} \subset \mathbb{R}$ is monotonically increasing sequence bounded above, then $\lim_{n\to\infty} a_n = \sup a_n$. If in contrast $(a_n)_{n\in\mathbb{N}} \subset \mathbb{R}$ is monotonically decreasing sequence bounded below, then $\lim_{n\to\infty} a_n = \inf a_n$.*

Definition 2.16. Let $(a_n)_{n\in\mathbb{N}} \subset \mathbb{R}$ be an arbitrary sequence of real numbers. Then, the limit superior of $(a_n)_{n\in\mathbb{N}}$ is defined as follows.

(a) If $(a_n)_{n\in\mathbb{N}}$ is not bounded above, $\limsup_{n\to\infty} a_n := +\infty$.
(b) If $(a_n)_{n\in\mathbb{N}}$ is bounded above,

$$\limsup_{n\to\infty} a_n := \begin{cases} \lim_{n\to\infty} \tilde{a}_n & \text{if } (\tilde{a}_n)_{n\in\mathbb{N}} \text{ is bounded below} \\ -\infty & \text{otherwise,} \end{cases} \tag{2.17}$$

where $\tilde{a}_n := \sup\{a_i : i \geq n\}$. Note that $(\tilde{a}_n)_{n\in\mathbb{N}}$ is monotonically decreasing with a reference to Proposition 2.15.

The limit inferior of $(\tilde{a}_n)_{n\in\mathbb{N}}$ is defined as $\liminf_{n\to\infty} a_n := -\limsup_{n\to\infty} -a_n$. If $\limsup_{n\to\infty} a_n = \liminf_{n\to\infty} a_n = a$ for some $a \in \mathbb{R}$, then $\lim_{n\to\infty} a_n = a$.

2.6. Metric Space

For a vector space X, an introduction of the following operations brings us rich results.

- *Metric distance* $d(\boldsymbol{x}, \boldsymbol{y})$: a distance (closeness) of two vectors $\boldsymbol{x}, \boldsymbol{y} \in X$.
- *Norm* $\|\boldsymbol{x}\|$: a length of a vector $\boldsymbol{x} \in X$. A distance of two vectors $\boldsymbol{x}, \boldsymbol{y} \in X$ can be measured by $d(\boldsymbol{x}, \boldsymbol{y}) := \|\boldsymbol{x} - \boldsymbol{y}\|$, which is called *the metric induced by the norm*.
- *Inner product* $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$: similarity of two vectors $\boldsymbol{x}, \boldsymbol{y} \in X$. A length of a vector $\boldsymbol{x} \in X$ can be measured by $\|\boldsymbol{x}\| := \sqrt{\langle \boldsymbol{x}, \boldsymbol{x} \rangle}$, which is called *the norm induced by the inner product*.

A vector space equipped with a metric, a norm, or an inner product is called a metric space, a normed space, or an inner product space. We can see that, if X is equipped with an inner product, a norm is induced automatically, and accordingly a metric is induced. Hence, an inner product space has more operations available than a normed space or a metric space, and — more importantly — it has a nice geometric property as will be seen later. Nevertheless, we start with discussing about metric spaces because 'metric' is the minimum notion to define *convergence* of a vector sequence formally.

Definition 2.18. Let X be a vector space. A mapping $d : X \times X \to [0, \infty)$ is said to be a *metric* on X if the following conditions are satisfied for any $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z} \in X$.

(a) $d(\boldsymbol{x}, \boldsymbol{y}) \geq 0$ and $d(\boldsymbol{x}, \boldsymbol{y}) = 0 \Leftrightarrow \boldsymbol{x} = \boldsymbol{y}$.
(b) $d(\boldsymbol{x}, \boldsymbol{y}) = d(\boldsymbol{y}, \boldsymbol{x})$ (symmetry).
(c) $d(\boldsymbol{x}, \boldsymbol{z}) \leq d(\boldsymbol{x}, \boldsymbol{y}) + d(\boldsymbol{y}, \boldsymbol{z})$ (triangle inequality).

Example 2.19.

(a) In the vector space $\mathbb{R}$, the most natural distance between two real values $\boldsymbol{x}$ and $\boldsymbol{y}$ would be an absolute value of $\boldsymbol{x} - \boldsymbol{y}$. Namely, $d(\boldsymbol{x}, \boldsymbol{y}) := |\boldsymbol{x} - \boldsymbol{y}|$, $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}$, satisfies the conditions (a)–(c) of metric.
(b) In $\mathbb{R}^N$, a metric distance between $\boldsymbol{x} := [x_1, x_2, \cdots, x_N]^\mathsf{T}$ and $\boldsymbol{y} := [y_1, y_2, \cdots, y_N]^\mathsf{T}$ can be defined as follows.
  (i) $d(\boldsymbol{x}, \boldsymbol{y}) := \sqrt{\sum_{n=1}^{N} (x_n - y_n)^2}$. This is called the *Euclidean distance*.

(ii) $d(\boldsymbol{x},\boldsymbol{y}) := \left(\sum_{n=1}^N |x_n - y_n|^p\right)^{1/p}$, where $p \in (0,\infty)$. This is called the *Minkowski distance*. If in particular $p = 1$, it is called the *Manhattan distance* etc.
(iii) $d(\boldsymbol{x},\boldsymbol{y}) := \max_{n=1}^N |x_n - y_n|$. This is called the *Chebyshev distance*, and it can be obtained by taking a limit of the *Minkowski distance* as $p \to \infty$.
(iv) $d(\boldsymbol{x},\boldsymbol{y}) := |S(\boldsymbol{x},\boldsymbol{y})|$ with $S(\boldsymbol{x},\boldsymbol{y}) := \{n \in \{1,2,\cdots,N\} : x_n \neq y_n\}$. This is called the *Hamming distance*.

Definition 2.20. Let X be a metric space. Then a sequence $(\boldsymbol{a}_n)_{n\in\mathbb{N}} \subset X$ is said to be *convergent* if there exists a point $\boldsymbol{a} \in X$ such that $\lim_{n\to\infty} d(\boldsymbol{a}_n,\boldsymbol{a}) = 0$. This is denoted as $\lim_{n\to\infty} \boldsymbol{a}_n = \boldsymbol{a}$, or $\boldsymbol{a}_n \to \boldsymbol{a}$ as $n \to \infty$.

It should be remarked that the convergence of $(\boldsymbol{a}_n)_{n\in\mathbb{N}} \subset X$ is defined through the convergence of the real-number sequence $(d(\boldsymbol{a}_n,\boldsymbol{a}))_{n\in\mathbb{N}}$ to the real number 0.

Definition 2.21. Let X be a metric space, and S be a subset of X.

(a) $\boldsymbol{x} \in S$ is said to be an *interior point* of S if there exists $\epsilon > 0$ such that $S \supset B(\boldsymbol{x},\epsilon) := \{\boldsymbol{y} \in X : d(\boldsymbol{x},\boldsymbol{y}) < \epsilon\}$. $B(\boldsymbol{x},\epsilon)$ is called an *open ball* centered at $\boldsymbol{x}$ with the radius ϵ.
(b) S is said to be *open* if the set of all interior points of S coincides with S itself. The entire space X and $\emptyset$ are regarded to be open.
(c) S is said to be *closed* if the complement of S, i.e. $X \setminus S := \{\boldsymbol{x} \in X : \boldsymbol{x} \notin S\}$, is open. The entire space X and $\emptyset$ are regarded to be closed.
(d) The minimum closed set containing S is called the *closure* of S and is denoted by $\bar{S}$.

Proposition 2.22.

(a) *The intersection of a finite number of open sets are open; the union of an arbitrary collection of open sets are open.*
(b) *The intersection of an arbitrary collection of closed sets are closed; the union of a finite number of open sets are open.*

The notions of convergence and closedness are connected by the following proposition.

Proposition 2.23. *A nonempty subset S of a metric space X is closed if and only if a convergent sequence $(\boldsymbol{x}_n)_{n\in\mathbb{N}} \subset S$ has its limit in S.*

Definition 2.24. Let X and Y be metric spaces with d_X and d_Y, respectively. A mapping $f : X \to Y$ is said to be *continuous* if $\boldsymbol{x} \to \boldsymbol{\xi}$ implies $f(\boldsymbol{x}) \to f(\boldsymbol{\xi})$. More precisely, the condition is replaced by the following: if for any $\epsilon > 0$ there exists $\delta > 0$ such that $d_X(\boldsymbol{x},\boldsymbol{\xi}) < \delta$, $\boldsymbol{x} \in X$, implies $d_Y(f(\boldsymbol{x}), f(\boldsymbol{\xi})) < \epsilon$.

We mention that in the general stage of metric spaces we can state the *Banach-Picard fixed-point theorem of contractive mappings*, which is the simplest results of fixed point theory. We would postpone this topic to Lecture 5 for reaching the Hilbert spaces through a shortest path.

2.7. Normed Space

Norm — which is a tool to measure a 'length' of a vector — is defined as follows.

Definition 2.25. Let X be a vector space. A mapping $\|\cdot\| : X \to [0,\infty)$ is said to be a *norm* on X if the following conditions are satisfied for any $\boldsymbol{x},\boldsymbol{y} \in X$ and any $\alpha \in \mathbb{R}$.

(a) $\|\boldsymbol{x}\| \geq 0$ and $\|\boldsymbol{x}\| = 0 \Leftrightarrow \boldsymbol{x} = \boldsymbol{0}$.
(b) $\|\boldsymbol{x}+\boldsymbol{y}\| \leq \|\boldsymbol{x}\| + \|\boldsymbol{y}\|$ (triangle inequality).
(c) $\|\alpha\boldsymbol{x}\| = |\alpha|\,\|\boldsymbol{x}\|$.

Example 2.26.

(a) In the vector space $\mathbb{R}$, $\|\boldsymbol{x}\| := |\boldsymbol{x}|$, $\boldsymbol{x} \in \mathbb{R}$, is a norm.
(b) In $\mathbb{R}^N$, we can defined a norm of any $\boldsymbol{x} := [x_1, x_2, \cdots, x_N]^\mathsf{T}$ as follows.
  (i) $\|\boldsymbol{x}\|_p := (\sum_{n=1}^N |x_n|^p)^{1/p}$, where $p \in [1,\infty)$. This is called the ℓ_p *norm*. In particular, $p = 2$ gives the *Euclidean norm* (or the ℓ_2 norm), and $p = 1$ gives the ℓ_1 norm which has been used as a regularization term to promote the *parsity* of estimates.
  (ii) $\|\boldsymbol{x}\|_p := (\sum_{n=1}^N w_n |x_n|^p)^{1/p}$, where $p \in [1,\infty)$ and $w_n \in (0,\infty)$, $n = 1,2,\cdots,N$. This is called the *weighted* ℓ_p *norm*.
  (iii) $\|\boldsymbol{x}\|_{\boldsymbol{Q}} := (\boldsymbol{x}^\mathsf{T}\boldsymbol{Q}\boldsymbol{x})^{1/2}$ for a symmetric positive definite matrix $\boldsymbol{Q} \in \mathbb{R}^{N\times N}$. This is called a *quadratic norm*, and $\boldsymbol{Q} := \boldsymbol{I}$ gives the Euclidean norm (or the ℓ_2 norm).
  (iv) $\|\boldsymbol{x}\|_\infty := \max_{n=1}^N |x_n|$. This is called the *Chebyshev norm* (or the ℓ_∞ norm).
(c) The ℓ_p norm for $p \in [1,\infty)$ in (b-i) and ℓ_∞ norm in (b-iv) can be extended to a sequence of infinitely-many real numbers $\boldsymbol{x} := (x_n)_{n\in\mathbb{N}}$ with $x_n \in \mathbb{R}$, and the corresponding spaces are respectively called the ℓ_p *space* and the ℓ_∞ *space*.

(i) The ℓ_p *space* consists of all such sequences $\boldsymbol{x} := (x_n)_{n\in\mathbb{N}}$ of real numbers that satisfy $\sum_{n\in\mathbb{N}} |x_n|^p < \infty$. The norm of $\boldsymbol{x} := (x_n)_{n\in\mathbb{N}}$ in ℓ_p is defined as $\|\boldsymbol{x}\|_p := (\sum_{n\in\mathbb{N}} |x_n|^p)^{1/p}$.
(ii) The ℓ_∞ *space* consists of all bounded sequences $\boldsymbol{x} := (x_n)_{n\in\mathbb{N}}$; i.e., $\sup_{n\in\mathbb{N}} |x_n| < \infty$. The norm of $\boldsymbol{x} := (x_n)_{n\in\mathbb{N}}$ in ℓ_∞ is defined as $\|\boldsymbol{x}\|_\infty := \sup_{n\in\mathbb{N}} |x_n|$.

Remark. In Example 2.26.(b-i), $p \in (0,1)$ does *not* give a norm because it violates the triangle inequality, but the function $d(\boldsymbol{x},\boldsymbol{y}) := \|\boldsymbol{x}-\boldsymbol{y}\|_p$ defines a metric; cf. Example 2.19.(b-ii). Also $\|\boldsymbol{x}\|_0 := |S(\boldsymbol{x})|$ with $S(\boldsymbol{x}) := \{n \in \{1,2,\cdots,N\} : x_n \neq 0\}$ does *not* define a norm, but the function $d(\boldsymbol{x},\boldsymbol{y}) := \|\boldsymbol{x}-\boldsymbol{y}\|_0$ defines a metric; cf. Example 2.19.(b-iv). The function $\|\cdot\|_0$ has been used as a sparseness measure, for example, in *compressed sensing*, and it is referred to as the ℓ_0 norm for convenience. The function $\|\cdot\|_p$ for $p \in (0,1)$ has been studied as an alternative to the ℓ_0 and ℓ_1 norms, and it is referred to as the ℓ_p norm.

Exercise 6. Given a norm $\|\cdot\|$ defined on a vector space X, show that $d(\boldsymbol{x},\boldsymbol{y}) := \|\boldsymbol{x}-\boldsymbol{y}\|$, $\boldsymbol{x},\boldsymbol{y} \in X$, is a metric; this means that a normed space can be regarded as a metric space with $d(\boldsymbol{x},\boldsymbol{y}) := \|\boldsymbol{x}-\boldsymbol{y}\|$.

Theorem 2.27. *Any finite dimensional subspace in a normed space is closed.*

Note that the closure of any subspace (of a possibly infinite dimension) in a normed space is a closed subspace.

Definition 2.28. Let X be a normed space. Then a sequence $(\boldsymbol{a}_n)_{n\in\mathbb{N}} \subset X$ is said to be *strongly convergent* (or *convergent in the norm*) if there exists a point $\boldsymbol{a} \in X$ such that $\lim_{n\to\infty} \|\boldsymbol{a}_n - \boldsymbol{a}\| = 0$. This is denoted as $\lim_{n\to\infty} \boldsymbol{a}_n = \boldsymbol{a}$, or $\boldsymbol{a}_n \to \boldsymbol{a}$ as $n \to \infty$.

Exercise 7. Let X be a normed space.

(a) Show that $\|\boldsymbol{x}\| - \|\boldsymbol{y}\| \leq \|\boldsymbol{x}-\boldsymbol{y}\|$, $\forall \boldsymbol{x},\boldsymbol{y} \in X$.
(b) Show that $\|\cdot\|$ is a continuous function (hint: use Exercise 7(a)).

Definition 2.29. Let $\|\cdot\|_{\mathrm{a}}$ and $\|\cdot\|_{\mathrm{b}}$ be norms on a vector space X. The norms $\|\cdot\|_{\mathrm{a}}$ and $\|\cdot\|_{\mathrm{b}}$ are said to be *equivalent* if there exist $\alpha,\beta \in (0,\infty)$ such that $\alpha \|\boldsymbol{x}\|_{\mathrm{a}} \leq \|\boldsymbol{x}\|_{\mathrm{b}} \leq \beta \|\boldsymbol{x}\|_{\mathrm{a}}$, $\forall \boldsymbol{x} \in X$.

It is known for instance that for any norm $\|\cdot\|$ on $\mathbb{R}^N$ there exists a quadratic norm $\|\cdot\|_{\boldsymbol{Q}}$ such that $\|\boldsymbol{x}\|_{\boldsymbol{Q}} \leq \|\boldsymbol{x}\| \leq \sqrt{N}\,\|\boldsymbol{x}\|_{\boldsymbol{Q}}$ [**14**]. The following theorem is of particular importance.

Theorem 2.30. *On a finite-dimensional vector space, any norm is equivalent to any other norm. (Note: this is not true for infinite-dimensional vector spaces.)*

Theorem 2.30 guarantees that in finite dimensional cases convergence of a vector sequence in a certain norm implies convergence of the sequence in any other norm.

2.8. Inner Product Space

Now we arrive at a stage which is closely related to our problem of adaptive filtering, which is basically an *estimation* problem. To estimate (or approximate) something, the common principle behind a significant amount of methods is *orthogonal projection* [**15**]; it will be discussed in detail in Section 2.10. The reader should already have an intuitive idea about the *orthogonality* by elementary geometry. However, how can we define the *orthogonality* in a general vector space? In normed spaces, the important concept of orthogonality is not necessarily available, but it is available in inner product spaces. (An inner product space is also called a *pre-Hilbert space*.)

Inner product — which is a tool to measure 'similarity' of two vectors — is defined as follows.

Definition 2.31. Let X be a vector space. A mapping $\langle\cdot,\cdot\rangle : X \times X \to \mathbb{R}$ is said to be an *inner product* on X if the following conditions are satisfied for any $\boldsymbol{x},\boldsymbol{y},\boldsymbol{z} \in X$ and any $\alpha,\beta \in \mathbb{R}$.

(a) $\langle \boldsymbol{x},\boldsymbol{y}\rangle = \langle \boldsymbol{y},\boldsymbol{x}\rangle$ (symmetry).
(b) $\langle \boldsymbol{x},\boldsymbol{x}\rangle \geq 0$ and $\langle \boldsymbol{x},\boldsymbol{x}\rangle = 0 \Leftrightarrow \boldsymbol{x} = \boldsymbol{0}$.
(c) $\langle \alpha\boldsymbol{x} + \beta\boldsymbol{y},\boldsymbol{z}\rangle = \alpha\langle \boldsymbol{x},\boldsymbol{z}\rangle + \beta\langle \boldsymbol{y},\boldsymbol{z}\rangle$.

Example 2.32.

(a) In the vector space $\mathbb{R}$, $\langle \boldsymbol{x},\boldsymbol{y}\rangle := \boldsymbol{x}\boldsymbol{y}$, $\boldsymbol{x},\boldsymbol{y} \in \mathbb{R}$, is an inner product.
(b) In $\mathbb{R}^N$, $\langle \boldsymbol{x},\boldsymbol{y}\rangle_{\boldsymbol{Q}} := \boldsymbol{x}^{\mathsf{T}}\boldsymbol{Q}\boldsymbol{y}$, $\boldsymbol{x},\boldsymbol{y} \in \mathbb{R}^N$, defines an inner product, where $\boldsymbol{Q} \in \mathbb{R}^{N\times N}$ is a symmetric positive definite matrix. In particular, $\boldsymbol{Q} := \boldsymbol{I}$ produces the standard inner product that induces the Euclidean norm (or the ℓ_2 norm).
(c) The ℓ_2 space in Example 2.26.(c-i) becomes a pre-Hilbert space with the inner product defined as follows: $\langle \boldsymbol{x},\boldsymbol{y}\rangle := \sum_{n\in\mathbb{N}} x_n y_n$, $\boldsymbol{x} := (x_n)_{n\in\mathbb{N}}$, $\boldsymbol{y} := (y_n)_{n\in\mathbb{N}}$. The Cauchy-Schwarz inequality ensures that the inner product takes a finite value (see Proposition 2.33 below).

Exercise 8. Given an inner product $\langle \cdot, \cdot \rangle$ defined on a vector space X, show that $\|\boldsymbol{x}\| := \sqrt{\langle \boldsymbol{x}, \boldsymbol{x} \rangle}$, $\boldsymbol{x} \in X$, is a norm; this means that an inner product space can be regarded as a normed space with $\|\boldsymbol{x}\| := \sqrt{\langle \boldsymbol{x}, \boldsymbol{x} \rangle}$.

In the following, $\|\cdot\|$ employed in an inner product space stands for the induced norm unless otherwise stated.

Proposition 2.33. *Let X be an inner product space. Then, the following hold for any $\boldsymbol{x}, \boldsymbol{y} \in X$.*

(a) $|\langle \boldsymbol{x}, \boldsymbol{y} \rangle| \leq \|\boldsymbol{x}\| \, \|\boldsymbol{y}\|$ *(The Cauchy-Schwarz inequality); equality holds if and only if $\boldsymbol{x}$ and $\boldsymbol{y}$ are linearly dependent.*

(b) $\|\boldsymbol{x} + \boldsymbol{y}\|^2 + \|\boldsymbol{x} - \boldsymbol{y}\|^2 = 2\|\boldsymbol{x}\|^2 + 2\|\boldsymbol{y}\|^2$ *(Parallelogram law).*

By Proposition 2.33(b), we see that induced norms satisfy the parallelogram law. The following theorem tells us more: any norm satisfying the parallelogram law can be induced by an inner product defined with the norm.

Theorem 2.34. *Suppose that the norm $\|\cdot\|$ equipped in a normed space X satisfy the parallelogram law. Then the operator $\langle \cdot, \cdot \rangle : X \times X \to \mathbb{R}$ defined by*

$$\langle \boldsymbol{x}, \boldsymbol{y} \rangle := \frac{1}{4} \left(\|\boldsymbol{x} + \boldsymbol{y}\|^2 - \|\boldsymbol{x} - \boldsymbol{y}\|^2 \right), \ \boldsymbol{x}, \boldsymbol{y} \in X, \tag{2.35}$$

satisfies the conditions of inner product, and $\|\boldsymbol{x}\| = \sqrt{\langle \boldsymbol{x}, \boldsymbol{x} \rangle}$, $\boldsymbol{x} \in X$.

Exercise 9. On an inner product space X, let $(\boldsymbol{x}_n)_{n \in \mathbb{N}} \subset X$ and $(\boldsymbol{y}_n)_{n \in \mathbb{N}} \subset X$ satisfy $\lim_{n \to \infty} \boldsymbol{x}_n = \boldsymbol{x} \in X$ and $\lim_{n \to \infty} \boldsymbol{y}_n = \boldsymbol{y} \in X$. Then, show that $\lim_{n \to \infty} \langle \boldsymbol{x}_n, \boldsymbol{y}_n \rangle = \langle \boldsymbol{x}, \boldsymbol{y} \rangle$.

Definition 2.36. Let X be an inner product space. Then, $\boldsymbol{x} \in X$ and $\boldsymbol{y} \in X$ are said to be orthogonal if $\langle \boldsymbol{x}, \boldsymbol{y} \rangle = 0$; this is symbolized by $\boldsymbol{x} \perp \boldsymbol{y}$. If $\boldsymbol{x}$ is orthogonal to any vector in a set $S \subset X$, then $\boldsymbol{x}$ is said to be orthogonal to S (written as $\boldsymbol{x} \perp S$).

The well-known Pythagorean theorem in elementary geometry holds true in inner product spaces.

Lemma 2.37. *Given $\boldsymbol{x}$ and $\boldsymbol{y}$ in an inner product space X, $\boldsymbol{x} \perp \boldsymbol{y}$ implies $\|\boldsymbol{x} + \boldsymbol{y}\|^2 = \|\boldsymbol{x}\|^2 + \|\boldsymbol{y}\|^2$.*

We are almost ready to discuss about the orthogonal projection theorem, as we have formally defined the orthogonality. We however need a final step — that is *completeness* of a space — to ensure existence of orthogonal projection.

2.9. Hilbert Space

For a while, we get back to metric spaces, which is more general than normed spaces and inner product spaces.

Definition 2.38. In a metric space X, a sequence $(\boldsymbol{x}_n)_{n \in \mathbb{N}} \subset X$ is said to be a *Cauchy sequence* if $d(\boldsymbol{x}_n, \boldsymbol{x}_m) \to 0$ as $n, m \to \infty$.

Lemma 2.39. *In a metric space X, the following hold.*

(a) *A convergent sequence has its unique limit.*

(b) *A convergent sequence is a Cauchy sequence.*

(c) *A Cauchy sequence is bounded.*

Definition 2.40. A metric space X is said to be *complete* if every Cauchy sequence $(\boldsymbol{x}_n)_{n \in \mathbb{N}} \subset X$ has a limit in X; i.e., there exists $\boldsymbol{x} \in X$ such that $\lim_{n \to \infty} \boldsymbol{x}_n = \boldsymbol{x}$.

Example 2.41. The spaces $\mathbb{R}$ with the metric in Example 2.19(a) and $\mathbb{R}^N$ with the metric in Example 2.19(b-i), (b-ii), or (b-iii) are complete metric spaces.

Because a normed space or an inner product space is a special kind of metric space, the concepts of convergence, closedness, completeness, etc., apply in these spaces.

Definition 2.42. A complete normed space is said to be a *Banach space*.

Example 2.43. Each space with each norm in Example 2.26 ($\mathbb{R}$, $\mathbb{R}^N$, ℓ_p for $p \in [1, \infty)$, ℓ_∞) is a Banach space. Indeed, $\mathbb{R}^N$ is a Banach space for any norm (cf. Theorem 2.30).

Definition 2.44. A complete inner product space is said to be a *Hilbert space*. We use $\mathcal{H}$ to denote a *Hilbert space*.

Example 2.45. Each space with each inner product in Example 2.32 ($\mathbb{R}$, $\mathbb{R}^N$, ℓ_2) is a Hilbert space.

In an infinite dimensional Hilbert space, it is not always easy to prove the strong convergence (see Definition 2.28). In the case that the strong convergence is difficult (or not able) to prove, the *weak convergence* is discussed as an intermediate step (or an alternative goal).

Definition 2.46. A sequence $(\boldsymbol{x}_n)_{n \in \mathbb{N}}$ in a Hilbert space $\mathcal{H}$ is said to *weakly convergent* if there exists $\boldsymbol{x} \in \mathcal{H}$ such that $\lim_{n \to \infty} \langle \boldsymbol{x}_n - \boldsymbol{x}, \boldsymbol{y} \rangle = 0$ for every $\boldsymbol{y} \in \mathcal{H}$. This is denoted as $\boldsymbol{x}_n \rightharpoonup \boldsymbol{x}$ as $n \to \infty$; $\boldsymbol{x}$ is called a *weak limit*.

Theorem 2.47. *For a sequence $(\boldsymbol{x}_n)_{n\in\mathbb{N}}$ in a Hilbert space $\mathcal{H}$, the following hold.*

(a) *If $(\boldsymbol{x}_n)_{n\in\mathbb{N}}$ is weakly convergent, it has a unique limit.*
(b) *Strong convergence of $(\boldsymbol{x}_n)_{n\in\mathbb{N}}$ implies its weak convergence to the same point.*
(c) *If $\mathcal{H}$ has a finite dimension, then weak convergence of $(\boldsymbol{x}_n)_{n\in\mathbb{N}}$ implies its strong convergence to the same point.*

Theorem 2.47.(b) and (c) suggests that there is no need to distinguish the two notions of convergence in finite dimensional cases.

Exercise 10. Show examples of weakly convergent sequence that is *not* strongly convergent.

2.10. Orthogonal Projection Theorem

In a Hilbert space $\mathcal{H}$, consider the following optimization problem: find the *best approximating* point of $\boldsymbol{x} \in \mathcal{H}$ in a closed subspace M of $\mathcal{H}$. More particularly, find a vector $\boldsymbol{m} \in M$ 'closest' to $\boldsymbol{x}$ in the sense of minimizing $\|\boldsymbol{x} - \boldsymbol{m}\|$. The following theorem provides important insight into the best approximation problem.

Theorem 2.48. *Let X be a Hilbert space, M a closed subspace of X, and $\boldsymbol{x} \in X$ chosen arbitrarily. Then, there exists a unique point $\boldsymbol{m}_0 \in M$ such that $\|\boldsymbol{x} - \boldsymbol{m}_0\| \leq \|\boldsymbol{x} - \boldsymbol{m}\|$, $\forall \boldsymbol{m} \in M$. Moreover, $\boldsymbol{m}_0$ is the unique minimizer if and only if $\boldsymbol{x} - \boldsymbol{m}_0 \perp M$. The $\boldsymbol{m}_0$ is called the orthogonal projection of $\boldsymbol{x}$ onto M, and we denote it as $P_M(\boldsymbol{x}) := \operatorname{argmin}_{\boldsymbol{m}\in M} \|\boldsymbol{x} - \boldsymbol{m}\|$.*

Definition 2.49. Given a subset S of a Hilbert space $\mathcal{H}$, $S^\perp := \{\boldsymbol{x} : \boldsymbol{x} \perp S\}$ is said to be the *orthogonal complement* of S.

Definition 2.50. A vector space X is said to be the *direct sum* of two subspaces M_1 and M_2 if every vector $\boldsymbol{x} \in X$ has a unique decomposition in the form of $\boldsymbol{x} = \boldsymbol{m}_1 + \boldsymbol{m}_2$ where $\boldsymbol{m}_1 \in M_1$ and $\boldsymbol{m}_2 \in M_2$. In this case, we write $X = M_1 \oplus M_2$.

Proposition 2.51. *For any subset S of a Hilbert space, $S^\perp$ is a closed subspace.*

Theorem 2.52. *If M is a closed subspace of a Hilbert space $\mathcal{H}$, then $\mathcal{H} = M \oplus M^\perp$ and $M = M^{\perp\perp}$. In fact, any $\boldsymbol{x} \in \mathcal{H}$ can be decomposed uniquely as $\boldsymbol{x} = P_M(\boldsymbol{x}) + P_{M^\perp}(\boldsymbol{x})$. This is called orthogonal decomposition.*

Proposition 2.53. *For any closed subspace M of a Hilbert space $\mathcal{H}$, the following statements hold.*

(a) *For any $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{H}$,*

$$\langle \boldsymbol{x}, P_M(\boldsymbol{y})\rangle = \langle P_M(\boldsymbol{x}), \boldsymbol{y}\rangle = \langle P_M(\boldsymbol{x}), P_M(\boldsymbol{y})\rangle\,. \tag{2.54}$$

(b) *For any $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{H}$ and any $\alpha, \beta \in \mathbb{R}$,*

$$P_M(\alpha\boldsymbol{x} + \beta\boldsymbol{y}) = \alpha P_M(\boldsymbol{x}) + \beta P_M(\boldsymbol{y}). \tag{2.55}$$

Definition 2.56. A subset S of a Hilbert space is said to be *orthogonal* if it does not contain the null vector and each pair of its elements is orthogonal. If in addition each of its elements has unit norm, S is said to be *orthonormal*.

Proposition 2.57. *An orthogonal set is linearly independent.*

LECTURE 3
Alternating Projections and NLMS/APA

3.1. Outline of Lecture 3

3.2. Introduction

In this lecture, we will see how the orthogonal projection theorem is exploited for engineering problems including adaptive filtering. Indeed the projection theorem plays a role to give a natural link the two notions: *algebra* and *geometry*. To make the projection theorem useful for real-world applications, we learn the calculus first. We start with the projection onto a one-dimensional subspace. Despite its simplicity, it provides a plane explanation of the *Fourier series expansion* and the *Gram-Schmidt orthonormalization procedure*. We then proceed to the projection onto a multi-dimensional subspace, a special type of infinite-dimensional subspace, and a linear variety. Once we learn the calculus, it is time to use it. We learn several types of projection methods which are classified into two categories: *serial methods* and *parallel methods*. For instance, the methods of the projections onto subspaces

(or linear varieties) can be applied to solve systems of linear equations which we often encounter in engineering problems. We also learn the known results about the rate of convergence for alternating projections (i.e., serial methods), which is based on the "angle" between subspaces. The contents of Sections 3.3–3.8 allow easy access to two projection-based adaptive filtering algorithms: the *Normalized Least Mean Square (NLMS) algorithm* and the *Affine Projection Algorithm (APA)*. In particular, we learn the geometric properties of the algorithms.

3.3. Projection onto One Dimensional Subspace

Let us start by considering a simple approximation problem: given a nonzero vector $\boldsymbol{y}$ and an arbitrary vector $\boldsymbol{x} \in \mathcal{H}$ in a Hilbert space $\mathcal{H}$, find a vector $\hat{\boldsymbol{x}} \in M_{\boldsymbol{y}} := \mathrm{span}(\{\boldsymbol{y}\})$ which is closest to $\boldsymbol{x}$. By the definition of span, $\hat{\boldsymbol{x}}$ can be expressed as $\hat{\boldsymbol{x}} = \alpha \boldsymbol{y}$ for some $\alpha \in \mathbb{R}$. By Theorem 2.48, it should be satisfied that $\langle \boldsymbol{x} - \hat{\boldsymbol{x}}, \beta \boldsymbol{y} \rangle = 0$, $\forall \beta \in \mathbb{R}$, thus $\langle \boldsymbol{x} - \hat{\boldsymbol{x}}, \boldsymbol{y} \rangle = \langle \boldsymbol{x} - \alpha \boldsymbol{y}, \boldsymbol{y} \rangle = 0$. This implies $\alpha = \langle \boldsymbol{x}, \boldsymbol{y} \rangle / \|\boldsymbol{y}\|^2$, hence it follows that

$$P_{M_{\boldsymbol{y}}}(\boldsymbol{x}) = \hat{\boldsymbol{x}} = \left\langle \boldsymbol{x}, \frac{\boldsymbol{y}}{\|\boldsymbol{y}\|} \right\rangle \frac{\boldsymbol{y}}{\|\boldsymbol{y}\|}. \tag{3.1}$$

The following example shows that the projection onto a one-dimensional subspace is the fundamental tool to construct the well-known Fourier series.

Example 3.2. Let $\boldsymbol{u}_1 \in \mathcal{H}$ be a unit vector (i.e., a vector with its norm equal to unity). Then, the projection onto $M_1 := \mathrm{span}(\{\boldsymbol{u}_1\})$ is given by

$$P_{M_1}(\boldsymbol{x}) = \langle \boldsymbol{x}, \boldsymbol{u}_1 \rangle \, \boldsymbol{u}_1. \tag{3.3}$$

Theorem 2.52 tells us that

$$\boldsymbol{x} = P_{M_1}(\boldsymbol{x}) + P_{M_1^\perp}(\boldsymbol{x}) = \langle \boldsymbol{x}, \boldsymbol{u}_1 \rangle \, \boldsymbol{u}_1 + P_{M_1^\perp}(\boldsymbol{x}), \tag{3.4}$$

hence

$$P_{M_1^\perp}(\boldsymbol{x}) = \boldsymbol{x} - \langle \boldsymbol{x}, \boldsymbol{u}_1 \rangle \, \boldsymbol{u}_1, \tag{3.5}$$

which expresses the *approximation error* orthogonal to $\boldsymbol{u}_1$ (and M_1). Next, pick up another unit vector $\boldsymbol{u}_2 \in M_1^\perp$ (i.e., $\boldsymbol{u}_2 \perp \boldsymbol{u}_1$). The projection of the approximation error $P_{M_1^\perp}(\boldsymbol{x})$ onto $M_2 := \mathrm{span}(\{\boldsymbol{u}_2\})$

is given by

$$P_{M_2}(P_{M_1^\perp}(\boldsymbol{x})) = \left\langle P_{M_1^\perp}(\boldsymbol{x}), \boldsymbol{u}_2 \right\rangle \boldsymbol{u}_2 \tag{3.6}$$

$$= \langle \boldsymbol{x} - \langle \boldsymbol{x}, \boldsymbol{u}_1 \rangle \boldsymbol{u}_1, \boldsymbol{u}_2 \rangle \boldsymbol{u}_2 \tag{3.7}$$

$$= \langle \boldsymbol{x}, \boldsymbol{u}_2 \rangle \boldsymbol{u}_2. \tag{3.8}$$

In analogy with (3.4), we have

$$P_{M_1^\perp}(\boldsymbol{x}) = P_{M_2}(P_{M_1^\perp}(\boldsymbol{x})) + P_{M_2^\perp}(P_{M_1^\perp}(\boldsymbol{x})) \tag{3.9}$$

$$= \langle \boldsymbol{x}, \boldsymbol{u}_2 \rangle \boldsymbol{u}_2 + P_{M_2^\perp}(P_{M_1^\perp}(\boldsymbol{x})) \tag{3.10}$$

$$= \langle \boldsymbol{x}, \boldsymbol{u}_2 \rangle \boldsymbol{u}_2 + P_{M_{1:2}^\perp}(\boldsymbol{x}), \tag{3.11}$$

where $M_{1:2} := \mathrm{span}(\{\boldsymbol{u}_1, \boldsymbol{u}_2\})$ (show that $P_{M_2^\perp}(P_{M_1^\perp}(\boldsymbol{x})) = P_{M_{1:2}^\perp}(\boldsymbol{x})$ by using Lemma 2.37). By (3.4) and (3.11), it follows that

$$\boldsymbol{x} = \sum_{i=1}^{2} \langle \boldsymbol{x}, \boldsymbol{u}_i \rangle \boldsymbol{u}_i + P_{M_{1:2}^\perp}(\boldsymbol{x}). \tag{3.12}$$

By continuing this procedure, we obtain

$$\boldsymbol{x} = \sum_{i=1}^{n} \langle \boldsymbol{x}, \boldsymbol{u}_i \rangle \boldsymbol{u}_i + P_{M_{1:n}^\perp}(\boldsymbol{x}), \tag{3.13}$$

where $M_{1:n} := \mathrm{span}(\{\boldsymbol{u}_i\}_{i=1}^n)$ with $n \in \{1, 2, \cdots, N\}$ if $\mathcal{H}$ has finite ($N \in \mathbb{N}^*$) dimension, otherwise $n \in \mathbb{N}^*$. By Theorem 2.52, (3.13) implies

$$P_{M_{1:n}}(\boldsymbol{x}) = \sum_{i=1}^{n} \langle \boldsymbol{x}, \boldsymbol{u}_i \rangle \boldsymbol{u}_i, \tag{3.14}$$

from which immediate conclusion is the following.

- $\sum_{i=1}^n \langle \boldsymbol{x}, \boldsymbol{u}_i \rangle \boldsymbol{u}_i$ is a best approximation of $\boldsymbol{x}$ in the subspace $M_{1:n}$ for any n.
- In a finite dimensional case with dimension N, we have $M_{1:N} = \{0\}$, hence $\boldsymbol{x} = \sum_{i=1}^N \langle \boldsymbol{x}, \boldsymbol{u}_i \rangle \boldsymbol{u}_i$ gives an expansion of $\boldsymbol{x}$ as a series of the orthonormal vectors $\boldsymbol{u}_1, \boldsymbol{u}_2, \cdots, \boldsymbol{u}_N$.

In an infinite dimensional case, the following interesting result is known.

- There exists $\hat{\boldsymbol{x}} \in \overline{\mathrm{span}(\{\boldsymbol{u}_i\}_{i=1}^\infty)}$ such that $\lim_{n\to\infty} \sum_{i=1}^n \langle \boldsymbol{x}, \boldsymbol{u}_i \rangle \boldsymbol{u}_i = \hat{\boldsymbol{x}}$; in this case $\boldsymbol{x} - \hat{\boldsymbol{x}} \perp \overline{\mathrm{span}(\{\boldsymbol{u}_i\}_{i=1}^\infty)}$. This is a generalization of the theory of Fourier series, and the coefficients $\langle \boldsymbol{x}, \boldsymbol{u}_i \rangle$ are called *Fourier coefficients*.

Example 3.15. Let $(\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots)$ be a countable or finite sequence of linearly independent vectors in an inner product space X. Then, we can construct an orthonormal sequence $(\boldsymbol{u}_1, \boldsymbol{u}_2, \cdots)$ such that $M_{1:n} := \mathrm{span}(\{\boldsymbol{u}_i\}_{i=1}^n) = \mathrm{span}(\{\boldsymbol{v}_i\}_{i=1}^n)$ for any $n = 1, 2, \cdots$ as follows.

Step 1: Normalize the first vector $\boldsymbol{v}_1$ by $\boldsymbol{u}_1 := \boldsymbol{v}_1 / \|\boldsymbol{v}_1\|$.

Step 2: Project the second vector $\boldsymbol{v}_2$ onto $M_{1:1}^\perp$ by $P_{M_{1:1}^\perp}(\boldsymbol{v}_2) = \boldsymbol{v}_2 - P_{M_{1:1}}(\boldsymbol{v}_2) = \boldsymbol{v}_2 - \langle \boldsymbol{v}_2, \boldsymbol{u}_1 \rangle \boldsymbol{u}_1$ [see Theorem 2.52 and (3.14)] and then normalize it as

$$\boldsymbol{u}_2 := P_{M_{1:1}^\perp}(\boldsymbol{v}_2) / \left\| P_{M_{1:1}^\perp}(\boldsymbol{v}_2) \right\|.$$

Step 3: Project the third vector $\boldsymbol{v}_3$ onto $M_{1:2}^\perp$ by $P_{M_{1:2}^\perp}(\boldsymbol{v}_3) = \boldsymbol{v}_3 - \sum_{i=1}^2 \langle \boldsymbol{v}_3, \boldsymbol{u}_i \rangle \boldsymbol{u}_i$ and then normalize it as

$$\boldsymbol{u}_3 := P_{M_{1:2}^\perp}(\boldsymbol{v}_3) / \left\| P_{M_{1:2}^\perp}(\boldsymbol{v}_3) \right\|.$$

$\vdots$

Step n: Project the third vector $\boldsymbol{v}_n$ onto $M_{1:n-1}^\perp$ by $P_{M_{1:n-1}^\perp}(\boldsymbol{v}_n) = \boldsymbol{v}_n - \sum_{i=1}^{n-1} \langle \boldsymbol{v}_n, \boldsymbol{u}_i \rangle \boldsymbol{u}_i$ and then normalize it as

$$\boldsymbol{u}_n := P_{M_{1:n-1}^\perp}(\boldsymbol{v}_n) / \left\| P_{M_{1:n-1}^\perp}(\boldsymbol{v}_n) \right\|.$$

$\vdots$

Exercise 11. Show in Example 3.15 that $\boldsymbol{u}_i \perp \boldsymbol{u}_j$ for $i \neq j$, and $\mathrm{span}(\{\boldsymbol{u}_i\}_{i=1}^n) = \mathrm{span}(\{\boldsymbol{v}_i\}_{i=1}^n)$ for any n. The *orthonormalization* procedure in Example 3.15 is widely known as the *Gram-Schmidt procedure*.

3.4. Projection onto Multi-Dimensional Subspace

Now consider a bit more general approximation problem: given nonzero vectors $\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_n$ ($n \in \mathbb{N}^*$) and an arbitrary vector $\boldsymbol{x} \in \mathcal{H}$ in a Hilbert space $\mathcal{H}$, find a vector $\hat{\boldsymbol{x}} \in \mathrm{span}(\{\boldsymbol{y}_i\}_{i=1}^n)$ which is closest to $\boldsymbol{x}$. By the definition of span, $\hat{\boldsymbol{x}}$ can be expressed as $\hat{\boldsymbol{x}} = \sum_{i=1}^n \alpha_i \boldsymbol{y}_i$ for some $\alpha_i \in \mathbb{R}$. By Theorem 2.48, it should be satisfied that $\boldsymbol{x} - \hat{\boldsymbol{x}} \perp \mathrm{span}(\{\boldsymbol{y}_i\}_{i=1}^n)$, or equivalently $\left\langle \boldsymbol{x} - \hat{\boldsymbol{x}}, \sum_{j=1}^n \beta_j \boldsymbol{y}_j \right\rangle = \sum_{j=1}^n \beta_j \langle \boldsymbol{x} - \hat{\boldsymbol{x}}, \boldsymbol{y}_j \rangle = 0$, $\forall \beta_j \in \mathbb{R}$, which is satisfied if and only if

$\langle \boldsymbol{x} - \hat{\boldsymbol{x}}, \boldsymbol{y}_j \rangle = 0$, $\forall j \in \{1, 2, \cdots, n\}$. Observe that

$$\langle \boldsymbol{x} - \hat{\boldsymbol{x}}, \boldsymbol{y}_j \rangle = \left\langle \boldsymbol{x} - \sum_{i=1}^{n} \alpha_i \boldsymbol{y}_i, \boldsymbol{y}_j \right\rangle \tag{3.16}$$

$$= \langle \boldsymbol{x}, \boldsymbol{y}_j \rangle - \sum_{i=1}^{n} \alpha_i \langle \boldsymbol{y}_i, \boldsymbol{y}_j \rangle \tag{3.17}$$

$$= \langle \boldsymbol{x}, \boldsymbol{y}_j \rangle - \left[\langle \boldsymbol{y}_1, \boldsymbol{y}_j \rangle, \cdots, \langle \boldsymbol{y}_n, \boldsymbol{y}_j \rangle \right] \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{bmatrix}. \tag{3.18}$$

This implies that the condition $\langle \boldsymbol{x} - \hat{\boldsymbol{x}}, \boldsymbol{y}_j \rangle = 0$, $\forall j \in \{1, 2, \cdots, n\}$, can be expressed in the following matrix form:

$$\begin{bmatrix} \langle \boldsymbol{y}_1, \boldsymbol{y}_1 \rangle & \langle \boldsymbol{y}_2, \boldsymbol{y}_1 \rangle & \cdots & \langle \boldsymbol{y}_n, \boldsymbol{y}_1 \rangle \\ \langle \boldsymbol{y}_1, \boldsymbol{y}_2 \rangle & \langle \boldsymbol{y}_2, \boldsymbol{y}_2 \rangle & \cdots & \langle \boldsymbol{y}_n, \boldsymbol{y}_2 \rangle \\ \vdots & & \ddots & \vdots \\ \langle \boldsymbol{y}_1, \boldsymbol{y}_n \rangle & \langle \boldsymbol{y}_2, \boldsymbol{y}_n \rangle & \cdots & \langle \boldsymbol{y}_n, \boldsymbol{y}_n \rangle \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{bmatrix} = \begin{bmatrix} \langle \boldsymbol{x}, \boldsymbol{y}_1 \rangle \\ \langle \boldsymbol{x}, \boldsymbol{y}_2 \rangle \\ \vdots \\ \langle \boldsymbol{x}, \boldsymbol{y}_n \rangle \end{bmatrix}. \tag{3.19}$$

(3.19) is called *normal equations* for the (norm) minimization problem. The transpose of the $n \times n$ matrix appearing in the right hand side of (3.19) is referred to as the *Gram* matrix of $\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_n$; the Gram matrix is symmetric as we consider the real-valued case. The projection for $n = 1$ given in (3.1) is readily reproduced as a special case.

Exercise 12. Show that the normal equations (3.19) are uniquely solvable if and only if $\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_n$ are linearly independent. Note that existence of a solution to the normal equations is guaranteed by Theorem 2.48. Note also that, in the case that $\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_n$ are linearly dependent, the multiplicity of the solutions of the normal equations corresponds to the multiplicity of the expressions of the (unique) projection $\hat{\boldsymbol{x}}$ as a linear combination of $\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_n$.

Example 3.20. Consider the case that $\mathcal{H} := \mathbb{R}^N$ with the standard inner product (i.e., $\langle \boldsymbol{x}, \boldsymbol{y} \rangle := \boldsymbol{x}^\mathsf{T} \boldsymbol{y}$) and $\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_n$ ($n \leq N$) are linearly independent. Let $\boldsymbol{Y} := [\boldsymbol{y}_1 \boldsymbol{y}_2 \cdots \boldsymbol{y}_n] \in \mathbb{R}^{N \times n}$ and $\boldsymbol{\alpha} := [\alpha_1, \alpha_2, \cdots, \alpha_n]^\mathsf{T}$. Then, (3.19) becomes $\boldsymbol{Y}^\mathsf{T} \boldsymbol{Y} \boldsymbol{\alpha} = \boldsymbol{Y}^\mathsf{T} \boldsymbol{x}$; the Gram matrix can be expressed as $\boldsymbol{Y}^\mathsf{T} \boldsymbol{Y}$. The linear independency of $\boldsymbol{y}_i$s surely implies the nonsingularity of $\boldsymbol{Y}^\mathsf{T} \boldsymbol{Y}$ (cf. Exercise 12), thus we obtain $\boldsymbol{\alpha} = (\boldsymbol{Y}^\mathsf{T} \boldsymbol{Y})^{-1} \boldsymbol{Y}^\mathsf{T} \boldsymbol{x}$. The solution of the approximation problem is given by $\hat{\boldsymbol{x}} = \sum_{i=1}^{n} \alpha_i \boldsymbol{y}_i = \boldsymbol{Y} \boldsymbol{\alpha} = \boldsymbol{Y} (\boldsymbol{Y}^\mathsf{T} \boldsymbol{Y})^{-1} \boldsymbol{Y}^\mathsf{T} \boldsymbol{x}$.

3.5. Projection onto Infinite Dimensional Subspace

In Section 3.4, we have seen that the projection onto a finite dimensional subspace can be computed by solving a system of linear equations called normal equations. What about the case that a subspace has infinite dimension? Fortunately, there is a certain important class of such problems that can be solved in an analogous way. The relation $\boldsymbol{x} = P_M(\boldsymbol{x}) + P_{M^\perp}(\boldsymbol{x})$ for any closed subspace M (see Theorem 2.52) implies the *duality* of the two problems: (i) compute $P_M(\boldsymbol{x})$ and (ii) compute $P_{M^\perp}(\boldsymbol{x})$. Namely, once we solve one of the problems, we can immediately solve the other.

The simplest case that the projection onto an infinite dimensional subspace can be computed *as easy as finite dimensional cases* would be the following. Let $\mathcal{H}$ be an infinite-dimensional Hilbert space, and define $M := \{\boldsymbol{x} \in \mathcal{H} : \langle \boldsymbol{x}, \boldsymbol{a} \rangle = 0\}$ for a given nonzero vector $\boldsymbol{a} \in \mathcal{H}$. Since $\langle \boldsymbol{x}, \boldsymbol{a} \rangle = 0 \Leftrightarrow \boldsymbol{x} \perp \mathrm{span}(\{\boldsymbol{a}\})$, we have $M = \mathrm{span}^\perp(\{\boldsymbol{a}\})$ which is a closed subspace. By $M^\perp = \mathrm{span}^{\perp\perp}(\{\boldsymbol{a}\}) = \mathrm{span}(\{\boldsymbol{a}\})$, the projection of any $\boldsymbol{x} \in \mathcal{H}$ onto M can be expressed as $P_M(\boldsymbol{x}) = \boldsymbol{x} - P_{\mathrm{span}(\{\boldsymbol{a}\})}(\boldsymbol{x})$, which can be easily computed by using (3.1). Note that the closed subspace M has infinite dimension, because assuming M has finite dimension, say $N \in \mathbb{N}^*$, implies $\mathcal{H} = M \oplus \mathrm{span}(\{\boldsymbol{a}\})$ also has finite dimension $N + 1$, yielding contradiction.

A slightly more general case can be considered by giving M in the following form: $M := \{\boldsymbol{x} \in \mathcal{H} : \langle \boldsymbol{x}, \boldsymbol{a}_i \rangle = 0, \forall i = 1, 2, \cdots, n\}$ for given nonzero vectors $\boldsymbol{a}_i \in \mathcal{H}$; $n = 1$ gives the previous case. In this case, $\langle \boldsymbol{x}, \boldsymbol{a}_i \rangle = 0, \forall i \in \{1, 2, \cdots, n\} \Leftrightarrow \boldsymbol{x} \perp \mathrm{span}(\{\boldsymbol{a}_i\}_{i=1}^n)$, hence $M = \mathrm{span}^\perp(\{\boldsymbol{a}_i\}_{i=1}^n)$ which is a closed subspace. The projection of any $\boldsymbol{x} \in \mathcal{H}$ onto M can be expressed as $P_M(\boldsymbol{x}) = \boldsymbol{x} - P_{\mathrm{span}(\{\boldsymbol{a}_i\}_{i=1}^n)}(\boldsymbol{x})$, where $P_{\mathrm{span}(\{\boldsymbol{a}_i\}_{i=1}^n)}(\boldsymbol{x})$ can be obtained by solving the normal equations presented in Section 3.4. In analogy with the previous case, we can show that the closed subspace M has infinite dimension.

3.6. Projection onto Linear Variety

A linear variety V in a Hilbert space $\mathcal{H}$ is a translation of a subspace $M \subset \mathcal{H}$ (see Definition 2.11); i.e., $V = M + \boldsymbol{v}$ for some $\boldsymbol{v} \in \mathcal{H}$. When the underlying subspace M is closed, V is said to be a *closed linear variety*. Existence and uniqueness of *projection onto a closed linear variety* can be verified essentially by Theorem 2.48. To see this, let us consider the following approximation problem: given any point $\boldsymbol{x} \in \mathcal{H}$ find its closest point $\hat{\boldsymbol{x}} \in V$, or more specifically, find $\hat{\boldsymbol{x}} \in \mathrm{argmin}_{\boldsymbol{y} \in V} \|\boldsymbol{x} - \boldsymbol{y}\|$, if such $\hat{\boldsymbol{x}}$ exists. Express $\hat{\boldsymbol{x}} \in V$ and $\boldsymbol{y} \in V$ respectively as $\hat{\boldsymbol{x}} = \boldsymbol{v} + \hat{\boldsymbol{x}}_M$

and $\boldsymbol{y} = \boldsymbol{v} + \boldsymbol{y}_M$, where $\hat{\boldsymbol{x}}_M \in M$ and $\boldsymbol{y}_M \in M$. Then, we can verify that $\hat{\boldsymbol{x}}_M = \operatorname{argmin}_{\boldsymbol{y}_M \in M} \|\boldsymbol{x} - (\boldsymbol{v} + \boldsymbol{y}_M)\| = P_M(\boldsymbol{x} - \boldsymbol{v})$, thus $\hat{\boldsymbol{x}} = P_M(\boldsymbol{x} - \boldsymbol{v}) + \boldsymbol{v}$. Note that $(\boldsymbol{x} - \boldsymbol{v}) - \hat{\boldsymbol{x}}_M = \boldsymbol{x} - \hat{\boldsymbol{x}} \perp M$.

Proposition 3.21. *Given a closed subspace M in a Hilbert space $\mathcal{H}$, define a linear variety as $V := M + \boldsymbol{v}$ for some $\boldsymbol{v} \in \mathcal{H}$. Then, given any $\boldsymbol{x} \in \mathcal{H}$, there exists a unique point $\hat{\boldsymbol{x}} \in V$ such that $\|\boldsymbol{x} - \hat{\boldsymbol{x}}\| \leq \|\boldsymbol{x} - \boldsymbol{y}\|$, $\forall \boldsymbol{y} \in V$. Moreover, $\hat{\boldsymbol{x}}$ is the unique minimizer if and only if $\boldsymbol{x} - \hat{\boldsymbol{x}} \perp M$. The $\hat{\boldsymbol{x}} \in V$ is called the orthogonal projection of $\boldsymbol{x}$ onto V, and we denote it as $P_V(\boldsymbol{x}) := \operatorname{argmin}_{\boldsymbol{y} \in V} \|\boldsymbol{x} - \boldsymbol{y}\|$. The projection has the following expressions.*

(a) $P_V(\boldsymbol{x}) = P_M(\boldsymbol{x} - \boldsymbol{v}) + \boldsymbol{v}$.
(b) $P_V(\boldsymbol{x}) = P_M(\boldsymbol{x}) + P_V(\boldsymbol{0})$.

Proposition 3.21(a) can be exploited, for instance, when V has the form of $V := \boldsymbol{v} + \operatorname{span}(\{\boldsymbol{a}_i\}_{i=1}^n)$ with $\boldsymbol{v} \in \mathcal{H}$ and $\boldsymbol{a}_1, \cdots, \boldsymbol{a}_n \in \mathcal{H} \setminus \{\boldsymbol{0}\}$ given *a priori*. In practice M or $M^\perp$ should have reasonably low dimension so that either P_M or $P_{M^\perp}$ has affordable computational costs. If however $\boldsymbol{v}$ is not given explicitly, we may use Proposition 3.21(b). Finding $P_V(\boldsymbol{0}) = \operatorname{argmin}_{\boldsymbol{y} \in V} \|\boldsymbol{y}\|$ is a *minimum norm problem*; i.e., find a vector in V that has minimum norm. This is an important problem, but before discussing it, we give a proof of Proposition 3.21(b) below.

First of all, we prove the following lemma.

Lemma 3.22. *For any $\boldsymbol{x} \in V$, it holds that $P_{M^\perp}(\boldsymbol{x}) = P_V(\boldsymbol{0})$. This suggests that (i) any $\boldsymbol{x} \in V$ can be decomposed as $\boldsymbol{x} = P_M(\boldsymbol{x}) + P_V(\boldsymbol{0})$ and (ii) $M^\perp \cap V = \{P_V(\boldsymbol{0})\}$.*

Fix $\boldsymbol{x} \in V$ arbitrarily. Then, any $\boldsymbol{y} \in V$ can be expressed as $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{z}$ for some $\boldsymbol{z} \in M$. Hence it follows that

$$P_V(\boldsymbol{0}) = \operatorname*{argmin}_{\boldsymbol{y} \in V} \|\boldsymbol{y}\| = \boldsymbol{x} + \operatorname*{argmin}_{\boldsymbol{z} \in M} \|\boldsymbol{x} + \boldsymbol{z}\| \tag{3.23}$$

$$= \boldsymbol{x} + \operatorname*{argmin}_{\boldsymbol{z} \in M} \|P_M(\boldsymbol{x}) + P_{M^\perp}(\boldsymbol{x}) + \boldsymbol{z}\|^2 \tag{3.24}$$

$$= \boldsymbol{x} + \operatorname*{argmin}_{\boldsymbol{z} \in M} \left(\|P_M(\boldsymbol{x}) + \boldsymbol{z}\|^2 + \|P_{M^\perp}(\boldsymbol{x})\|^2\right) \tag{3.25}$$

$$= \boldsymbol{x} - P_M(\boldsymbol{x}) = P_{M^\perp}(\boldsymbol{x}). \tag{3.26}$$

Let us now prove Proposition 3.21(b). Fix $\boldsymbol{x} \in V$ arbitrarily. Since any $\boldsymbol{y} \in V$ can be decomposed as $\boldsymbol{y} = P_M(\boldsymbol{y}) + P_V(\boldsymbol{0})$ by Lemma 3.22,

we can verify

$$P_V(\boldsymbol{x}) = \operatorname*{argmin}_{\boldsymbol{y} \in V} \|\boldsymbol{x} - \boldsymbol{y}\| \tag{3.27}$$

$$= P_V(\boldsymbol{0}) + \operatorname*{argmin}_{P_M(\boldsymbol{y}) \in M} \|\boldsymbol{x} - [P_M(\boldsymbol{y}) + P_V(\boldsymbol{0})]\| \tag{3.28}$$

$$= P_V(\boldsymbol{0}) + \operatorname*{argmin}_{\boldsymbol{z} \in M} \|P_M(\boldsymbol{x}) + P_{M^\perp}(\boldsymbol{x}) - [\boldsymbol{z} + P_V(\boldsymbol{0})]\|^2 \tag{3.29}$$

$$= P_V(\boldsymbol{0}) + \operatorname*{argmin}_{\boldsymbol{z} \in M} \left(\|P_M(\boldsymbol{x}) - \boldsymbol{z}\|^2 + \|P_{M^\perp}(\boldsymbol{x}) - P_V(\boldsymbol{0})\|^2\right) \tag{3.30}$$

$$= P_V(\boldsymbol{0}) + P_M(\boldsymbol{x}). \tag{3.31}$$

Proposition 3.21(b) implies that the computation of $P_V(\boldsymbol{x})$ is feasible if $P_V(\boldsymbol{0})$ and $P_M(\boldsymbol{x})$ are computable. There are two such situations. The first is the case that a linear variety in a Hilbert space $\mathcal{H}$ takes the form of $V := \boldsymbol{a} + \operatorname{span}(\{\boldsymbol{a}_i\}_{i=1}^n)$, $n \in \mathbb{N}^*$, for some $\boldsymbol{a} \in \mathcal{H}$ and a linearly independent set $\{\boldsymbol{a}_1, \cdots, \boldsymbol{a}_n\} \subset \mathcal{H}$. In this case, the discussion in Section 3.4 can directly be used to compute $P_M(\boldsymbol{x})$.

Exercise 13. Show that the problem of finding $P_V(\boldsymbol{0})$ for the first situation above ($V := \boldsymbol{a} + \operatorname{span}(\{\boldsymbol{a}_i\}_{i=1}^n)$) can be reduced to the solution of the normal equations presented in Section 3.4.

The second situation is that a linear variety in a Hilbert space $\mathcal{H}$ takes the form of $V := \{\boldsymbol{x} \in \mathcal{H} : \langle \boldsymbol{x}, \boldsymbol{a}_i \rangle = b_i,\ \forall i = 1, 2, \cdots, n\}$, $n \in \mathbb{N}^*$, for a linearly independent set $\{\boldsymbol{a}_1, \cdots, \boldsymbol{a}_n\} \subset \mathcal{H}$ and $b_1, b_2, \cdots, b_n \in \mathbb{R}$. To see that the V is a closed linear variety, let us consider the case of $b_1 = b_2 = \cdots = b_n = 0$. In this particular case, it is clear that V is the closed subspace $\operatorname{span}^\perp(\{\boldsymbol{a}_i\}_{i=1}^n)$ (see Proposition 2.51). Now let us go back to the general case. As nonemptiness of V is ensured by the linear independency of $\{\boldsymbol{a}_1, \cdots, \boldsymbol{a}_n\}$, pick up an arbitrary element $\boldsymbol{v} \in V$; i.e., $\langle \boldsymbol{v}, \boldsymbol{a}_i \rangle = b_i$, $\forall i = 1, 2, \cdots, n$. Hence V can be expressed as

$$V = \{\boldsymbol{x} \in \mathcal{H} : \langle \boldsymbol{x}, \boldsymbol{a}_i \rangle = \langle \boldsymbol{v}, \boldsymbol{a}_i \rangle,\ \forall i = 1, 2, \cdots, n\} \tag{3.32}$$

$$= \{\boldsymbol{x} \in \mathcal{H} : \langle \boldsymbol{x} - \boldsymbol{v}, \boldsymbol{a}_i \rangle = 0\ \forall i = 1, 2, \cdots, n\} \tag{3.33}$$

$$= \{\boldsymbol{y} + \boldsymbol{v} \in \mathcal{H} : \langle \boldsymbol{y}, \boldsymbol{a}_i \rangle = 0\ \forall i = 1, 2, \cdots, n\} \tag{3.34}$$

$$= \operatorname{span}^\perp(\{\boldsymbol{a}_i\}_{i=1}^n) + \boldsymbol{v}, \tag{3.35}$$

implying that V is a closed linear variety. The linear variety V is said to be of *codimension* n since its underlying subspace $\operatorname{span}^\perp(\{\boldsymbol{a}_i\}_{i=1}^n)$ has its orthogonal complement of dimension n. The projection $P_V(\boldsymbol{0})$ can be computed based on the following theorem.

Theorem 3.36. *In a Hilbert space $\mathcal{H}$, define $V := \{\boldsymbol{x} \in \mathcal{H} : \langle \boldsymbol{x}, \boldsymbol{a}_i \rangle = b_i,\ \forall i = 1, 2, \cdots, n\}$ for a linearly independent set $\{\boldsymbol{a}_1, \cdots, \boldsymbol{a}_n\} \subset \mathcal{H}$ and $b_1, b_2, \cdots, b_n \in \mathbb{R}$. Then, $P_V(\boldsymbol{0}) = \sum_{i=1}^n \beta_i \boldsymbol{a}_i$ with the unique vector $\boldsymbol{\beta} := [\beta_1, \beta_2, \cdots, \beta_n]^\mathsf{T} \in \mathbb{R}^n$ satisfying $\boldsymbol{G}^\mathsf{T}\boldsymbol{\beta} = \boldsymbol{b}$, where $\boldsymbol{b} := [b_1, b_2, \cdots, b_n]^\mathsf{T} \in \mathbb{R}^n$ and*

$$\boldsymbol{G} := \begin{bmatrix} \langle \boldsymbol{a}_1, \boldsymbol{a}_1 \rangle & \langle \boldsymbol{a}_1, \boldsymbol{a}_2 \rangle & \cdots & \langle \boldsymbol{a}_1, \boldsymbol{a}_n \rangle \\ \langle \boldsymbol{a}_2, \boldsymbol{a}_1 \rangle & \langle \boldsymbol{a}_2, \boldsymbol{a}_2 \rangle & \cdots & \langle \boldsymbol{a}_2, \boldsymbol{a}_n \rangle \\ \vdots & & \ddots & \vdots \\ \langle \boldsymbol{a}_n, \boldsymbol{a}_1 \rangle & \langle \boldsymbol{a}_n, \boldsymbol{a}_2 \rangle & \cdots & \langle \boldsymbol{a}_n, \boldsymbol{a}_n \rangle \end{bmatrix} \in \mathbb{R}^{n \times n}. \tag{3.37}$$

Exercise 14. Prove Theorem 3.36.

Definition 3.38. In a Hilbert space $\mathcal{H}$, a closed linear variety $H := \{\boldsymbol{x} \in \mathcal{H} : \langle \boldsymbol{x}, \boldsymbol{a} \rangle = b\}$ for some nonzero vector $\boldsymbol{a} \in \mathcal{H}$ and $b \in \mathbb{R}$ is specially called a *hyperplane*. $\boldsymbol{a}$ is called the *normal vector* of H.

Example 3.39. Consider the case in Theorem 3.36 that $\mathcal{H} := \mathbb{R}^N$ with the standard inner product (i.e., $\langle \boldsymbol{x}, \boldsymbol{y} \rangle := \boldsymbol{x}^\mathsf{T}\boldsymbol{y}$, $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{H}$). Let $\boldsymbol{A} := [\boldsymbol{a}_1 \boldsymbol{a}_2 \cdots \boldsymbol{a}_n] \in \mathbb{R}^{N \times n}$. Then, we have $\boldsymbol{\beta} = \left(\boldsymbol{A}^\mathsf{T}\boldsymbol{A}\right)^{-1}\boldsymbol{b}$, and thus $P_V(\boldsymbol{0}) = \boldsymbol{A}\boldsymbol{\beta} = \boldsymbol{A}\left(\boldsymbol{A}^\mathsf{T}\boldsymbol{A}\right)^{-1}\boldsymbol{b}$. Letting $M := \mathrm{span}^\perp(\{\boldsymbol{a}_i\}_{i=1}^n)$, $P_{M^\perp}(\boldsymbol{x}) = \boldsymbol{A}(\boldsymbol{A}^\mathsf{T}\boldsymbol{A})^{-1}\boldsymbol{A}^\mathsf{T}\boldsymbol{x}$ for any $\boldsymbol{x} \in \mathcal{H}$ (see Example 3.20). Hence, by Proposition 3.21(b), we can verify that

$$P_V(\boldsymbol{x}) = \boldsymbol{x} - \boldsymbol{A}(\boldsymbol{A}^\mathsf{T}\boldsymbol{A})^{-1}(\boldsymbol{A}^\mathsf{T}\boldsymbol{x} - \boldsymbol{b}). \tag{3.40}$$

Let us consider the case of $n = 1$; in this case $H := \{\boldsymbol{x} \in \mathcal{H} : \langle \boldsymbol{x}, \boldsymbol{a} \rangle = b\}$ ($\boldsymbol{a} \neq \boldsymbol{0}$) is a hyperplane (its underlying subspace has dimension $N - 1$). The projection is given by

$$P_H(\boldsymbol{x}) = \boldsymbol{x} - \frac{\langle \boldsymbol{a}, \boldsymbol{x} \rangle - b}{\|\boldsymbol{a}\|^2}\boldsymbol{a}. \tag{3.41}$$

Remark. How should we do when $\{\boldsymbol{a}_1, \cdots, \boldsymbol{a}_n\} \subset \mathcal{H}(:= \mathbb{R}^N)$ is linearly dependent? In such a case, $V := \{\boldsymbol{x} \in \mathcal{H} : \langle \boldsymbol{x}, \boldsymbol{a}_i \rangle = b_i,\ \forall i = 1, 2, \cdots, n\}$ could possibly be empty. As an alternative, we can define a linear variety as $V := \mathrm{argmin}_{\boldsymbol{x} \in \mathcal{H}} \left\|\boldsymbol{A}^\mathsf{T}\boldsymbol{x} - \boldsymbol{b}\right\|_n$, where $\|\cdot\|_n$ stands for the Euclidean norm in $\mathbb{R}^n$, and the projection is given by

$$P_V(\boldsymbol{x}) = \boldsymbol{x} - (\boldsymbol{A}^\mathsf{T})^\dagger(\boldsymbol{A}^\mathsf{T}\boldsymbol{x} - \boldsymbol{b}). \tag{3.42}$$

Here, $(\cdot)^\dagger$ is the *Moore-Penrose pseudoinverse* [**10, 16**]. Substituting $\boldsymbol{x} := \boldsymbol{0}$ into (3.42) yields $P_V(\boldsymbol{0}) = (\boldsymbol{A}^\mathsf{T})^\dagger\boldsymbol{b}$. This implies that $(\boldsymbol{A}^\mathsf{T})^\dagger\boldsymbol{b}$ gives the minimum norm solution to the following least squares problem: minimize $\left\|\boldsymbol{A}^\mathsf{T}\boldsymbol{x} - \boldsymbol{b}\right\|_n$ over $\mathcal{H}$. Suppose in particular that the inverse $(\boldsymbol{A}^\mathsf{T}\boldsymbol{A})^{-1}$ exists, which holds if and only if $\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_n$ are

linearly independent. Then, $(\boldsymbol{A}^\mathsf{T})^\dagger = \boldsymbol{A}(\boldsymbol{A}^\mathsf{T}\boldsymbol{A})^{-1}$, reproducing the result in (3.40), and $(\boldsymbol{A}^\mathsf{T})^\dagger\boldsymbol{b}$ gives the minimum norm solution to the system of linear equations $\boldsymbol{A}^\mathsf{T}\boldsymbol{x} = \boldsymbol{b}$, $\boldsymbol{x} \in \mathcal{H}$.

3.7. Methods of Projections onto Subspaces

Orthogonal projection bridges the worlds of *algebra* and *geometry*. More precisely, one can solve algebraic equations through a geometric approach. Figure 3-1 describes the behavior of the alternating projection method in the two-dimensional case. The lines M_1 and M_2 stand for linear subspaces in $\mathbb{R}^2$, and, starting from an initial point $\boldsymbol{x}_0 \in \mathbb{R}^2$, the method operates the projections P_{M_1} and P_{M_2} alternately. It is easily seen that the sequence $(\boldsymbol{x}_k)_{k \in \mathbb{N}} \subset \mathbb{R}^2$ converges to the intersecting point. J. von Neumann, one of the greatest mathematicians in the twenty century, has proven that this applies in a general Hilbert space [**17**] as shown below.[1]

Theorem 3.43 (von Neumann 1933)**.** *Let M_1 and M_2 be closed subspaces in a Hilbert space $\mathcal{H}$. Assume that $M_1 \cap M_2 \neq \emptyset$. Then, for any $\boldsymbol{x} \in \mathcal{H}$,*

$$\lim_{k \to \infty} (P_{M_2}P_{M_1})^k(\boldsymbol{x}) = P_{M_1 \cap M_2}(\boldsymbol{x}). \tag{3.44}$$

Theorem 3.43 can be rephrased as follows: for any $\boldsymbol{x}_0 \in \mathcal{H}$, the sequence $(\boldsymbol{x}_k)_{k \in \mathbb{N}}$ generated recursively as $\boldsymbol{x}_{k+1} := P_{M_2}P_{M_1}(\boldsymbol{x}_k)$, $n \in \mathbb{N}$, converges strongly to the projection $P_{M_1 \cap M_2}(\boldsymbol{x}_0)$. In the case of Euclidean spaces, algorithmic solutions based on projection have been proposed by S. Kaczmarz in 1937 [**19**] and G. Cimmino in 1938 [**20**] for solving systems of linear equations. Let $H_1, H_2, \cdots, H_n$ be hyperplanes in $\mathbb{R}^N$ such that $H := \bigcap_{i=1}^n H_i \neq \emptyset$. Kaczmarz's method, based on *cyclic projections* onto each hyperplane, is given as follows (see Fig. 3-1):[2]

$$\boldsymbol{x}_{k+1} := P_{H_{i_k}}(\boldsymbol{x}_k),\ i_k := k \pmod n + 1. \tag{3.45}$$

Proposition 3.46. *Given any $\boldsymbol{x}_0 \in \mathbb{R}^N$, the sequence $(\boldsymbol{x}_k)_{k \in \mathbb{N}}$ generated by* (3.45) *converges to the projection $P_H(\boldsymbol{x}_0)$.*

[1] The first alternating projection algorithm seems to have been developed by H. A. Schwarz around 1870 [**18**].

[2] The method in (3.45) was independently discovered in the field of image reconstruction from projections where it was called *Algebraic Reconstruction Technique (ART)* [**21**].

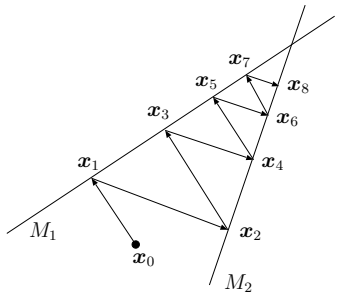


Fig. 3-1. Illustration of alternating projection method.

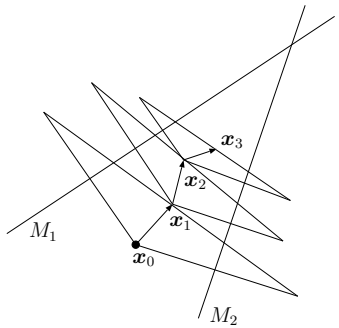


Fig. 3-2. Illustration of Cimmino's parallel projection method.

On the other hand, Cimmino's method, based on *parallel projections* onto each hyperplane, is given as follows (see Fig. 3-2):

$$\boldsymbol{x}_{k+1} := \sum_{i=1}^{n} \frac{1}{n} \left[2P_{H_i}(\boldsymbol{x}_k) - \boldsymbol{x}_k\right] = -\boldsymbol{x}_k + \frac{2}{n}\sum_{i=1}^{n} P_{H_i}(\boldsymbol{x}_k),\ k \in \mathbb{N}. \tag{3.47}$$

Here, the term $2P_{H_i}(\boldsymbol{x}_k) - \boldsymbol{x}_k$ is called the *reflection* of $\boldsymbol{x}_k$ with respect to H_i.

Proposition 3.48. *Given any $\boldsymbol{x}_0 \in \mathbb{R}^N$, the sequence $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ generated by* (3.47) *converges to a point in $H(:= \bigcap_{i=1}^n H_i)$.*

I. Halperin has proven the following result [**22**], which is a generalization of Theorem 3.43 and Proposition 3.46.

Proposition 3.49. *Let $M_1, M_2, \cdots, M_n$ be closed subspaces in a Hilbert space $\mathcal{H}$ such that $M := \bigcap_{i=1}^n M_i \neq \emptyset$. Then, for any $\boldsymbol{x} \in \mathcal{H}$,*

$$\lim_{k\to\infty} (P_{M_n} P_{M_{n-1}} \cdots P_{M_1})^k(\boldsymbol{x}) = P_M(\boldsymbol{x}). \tag{3.50}$$

Proposition 3.51. *Let $V_1, V_2, \cdots, V_n$ be closed linear varieties in a Hilbert space $\mathcal{H}$ such that $V := \bigcap_{i=1}^n V_i \neq \emptyset$. Then, for any $\boldsymbol{x} \in \mathcal{H}$,*

$$\lim_{k\to\infty} (P_{V_n} P_{V_{n-1}} \cdots P_{V_1})^k(\boldsymbol{x}) = P_V(\boldsymbol{x}). \tag{3.52}$$

Corresponding to Proposition 3.51, S. Reich has proven the following result on a parallel algorithm [**23**].[3]

Proposition 3.53. *Let $V_1, V_2, \cdots, V_n$ be closed linear varieties in a Hilbert space $\mathcal{H}$ such that $V := \bigcap_{i=1}^n V_i \neq \emptyset$. For any $\boldsymbol{x}_0 \in \mathcal{H}$, let $(\boldsymbol{x}_k)_{k\in\mathbb{N}} \subset \mathcal{H}$ be a sequence generated by*

$$\boldsymbol{x}_{k+1} := \sum_{i=1}^{n} w_i P_{V_i}(\boldsymbol{x}_k),\ k \in \mathbb{N}, \tag{3.54}$$

where $w_i > 0$ denote the weights satisfying $\sum_{i=1}^n w_i = 1$. Then, $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ converges to the projection $P_V(\boldsymbol{x}_0)$.

There are several nonlinear extensions of the projection methods presented above, which will be discussed in Lecture 4.

3.8. Rate of Convergence for Alternating Projections

Taking a fresh look at Fig. 3-1, we observe the following.

(a) If the "angle" between the two lines, M_1 and M_2, is $\pi/2$ [rad], the method needs to operate the consecutive projection $P_{M_2}P_{M_1}$ only once to reach the intersecting point.

(b) If in contrast the "angle" is close to 0 [rad], the method needs to operate the consecutive projection $P_{M_2}P_{M_1}$ many times to approach the intersecting point.

This observation naturally suggests that the rate of convergence for alternating projections onto subspaces depends on the "angle". This intuition can indeed be verified for a general Hilbert space. The question is: *how can we define the angle between two subspaces in a Hilbert space?*

[3]The case of $w_i = 1/n$, $\forall i = 1, 2, \cdots, n$, in (3.54) was proposed in 1972 for tomographic image reconstruction, and it is called the *Simultaneous Iterative Reconstruction Technique (SIRT)* [**24**] (a parallel counterpart of ART).

Definition 3.55. Let M_1 and M_2 be subspaces in a Hilbert space $\mathcal{H}$. We define the *angle* between M_1 and M_2 as $\alpha(M_1, M_2) \in [0, \pi/2]$ whose cosine $c(M_1, M_2) := \cos\alpha(M_1, M_2)$ is defined as follows:

$$c(M_1, M_2) := \sup\{|\langle \boldsymbol{x}, \boldsymbol{y}\rangle| : \boldsymbol{x} \in M_1 \cap (M_1 \cap M_2)^\perp, \|\boldsymbol{x}\| \le 1, \\ \boldsymbol{y} \in M_2 \cap (M_1 \cap M_2)^\perp, \|\boldsymbol{y}\| \le 1\}. \tag{3.56}$$

Exercise 15. Given any hyperplanes expressed as $H_1 := \{\boldsymbol{x} \in \mathcal{H} : \langle \boldsymbol{a}_1, \boldsymbol{x}\rangle = 0\}$ and $H_2 := \{\boldsymbol{x} \in \mathcal{H} : \langle \boldsymbol{a}_2, \boldsymbol{x}\rangle = 0\}$ for some nonzero vectors $\boldsymbol{a}_1, \boldsymbol{a}_2 \in \mathcal{H}$, show that $c(H_1, H_2) = \dfrac{|\langle \boldsymbol{a}_1, \boldsymbol{a}_2\rangle|}{\|\boldsymbol{a}_1\|\,\|\boldsymbol{a}_2\|}$.

Theorem 3.43 tells us that $\lim_{k\to\infty} \left\|(P_{M_2}P_{M_1})^k(\boldsymbol{x}) - P_{M_1\cap M_2}(\boldsymbol{x})\right\| = 0$. The following proposition shows how fast the real-number sequence $(\left\|(P_{M_2}P_{M_1})^k(\boldsymbol{x}) - P_{M_1\cap M_2}(\boldsymbol{x})\right\|)_{k\in\mathbb{N}}$ converges to zero.

Proposition 3.57. *Let M_1 and M_2 be subspaces in a Hilbert space $\mathcal{H}$ such that $M := M_1 \cap M_2 \neq \emptyset$, and $c := c(M_1, M_2)$. Then for any $\boldsymbol{x} \in \mathcal{H}$*

$$\left\|(P_{M_2}P_{M_1})^k(\boldsymbol{x}) - P_M(\boldsymbol{x})\right\| \le c^{2k-1}\left\|\boldsymbol{x} - P_M(\boldsymbol{x})\right\|,\ k \in \mathbb{N}. \tag{3.58}$$

In the general case of multiple subspaces, the following theorem holds.

Theorem 3.59. *Let $M_1, M_2, \cdots, M_n$ be subspaces in a Hilbert space $\mathcal{H}$ such that $M := \bigcap_{i=1}^n M_i \neq \emptyset$, and*

(3.60)

$$c := \left[1 - \prod_{i=1}^{n-1}(1 - c_i^2)\right]^{1/2} \text{ with } c_i := c\left(M_i, \bigcap_{j=i+1}^{n} M_j\right),\ i = 1, 2, \cdots, n-1.$$

Then for any $\boldsymbol{x} \in \mathcal{H}$

$$\left\|(P_{M_n}P_{M_{n-1}}\cdots P_{M_1})^k(\boldsymbol{x}) - P_M(\boldsymbol{x})\right\| \le c^k\left\|\boldsymbol{x}\right\|,\ k \in \mathbb{N}. \tag{3.61}$$

Corollary 3.62. *Let $V_1, V_2, \cdots, V_n$ be linear varieties in a Hilbert space $\mathcal{H}$ such that $V := \bigcap_{i=1}^n V_i \neq \emptyset$, and $M_i \subset \mathcal{H}$, $i = 1, 2, \cdots, n$, be the underlying subspace of each V_i. Also let $c \in [0, 1]$ be a constant defined as in* (3.60)*. Then for any $\boldsymbol{x} \in \mathcal{H}$*

$$\left\|(P_{V_n}P_{V_{n-1}}\cdots P_{V_1})^k(\boldsymbol{x}) - P_V(\boldsymbol{x})\right\| \le c^k\left\|\boldsymbol{x} - P_V(\boldsymbol{x})\right\|,\ k \in \mathbb{N}. \tag{3.63}$$

3.9. Projection Based Adaptive Filtering Algorithm: NLMS

We consider the Hilbert space $\mathcal{H} := \mathbb{R}^N$ equipped with the standard inner product. The LMS algorithm is then represented as follows:

$$\boldsymbol{h}_{k+1} = \boldsymbol{h}_k - 2\lambda(\langle \boldsymbol{u}_k, \boldsymbol{h}_k\rangle - d_k)\boldsymbol{u}_k,\ k \in \mathbb{N}. \tag{3.64}$$

As mentioned in Section 1.10, the step size parameter λ should be sufficiently small to stabilize the algorithm, and it results in slow convergence. The reason for this is that, when the $\|\boldsymbol{u}_k\|^2$ is large (i.e., when large inputs come), the amount of update $\|\boldsymbol{h}_{k+1} - \boldsymbol{h}_k\|$ becomes proportionally large. To avoid this, the following normalized algorithm has been proposed [**25, 26**]:

$$\boldsymbol{h}_{k+1} := \boldsymbol{h}_k - \lambda\frac{\langle \boldsymbol{u}_k, \boldsymbol{h}_k\rangle - d_k}{\|\boldsymbol{u}_k\|^2}\boldsymbol{u}_k,\ k \in \mathbb{N}, \tag{3.65}$$

where $\lambda \in [0, 2]$. This is called the *Normalized Least Mean Square (NLMS) algorithm*. At the first glance, NLMS looks no more than a variant of the LMS algorithm. However, the following discussion reveals its nice geometric property. Define a hyperplane at each $k \in \mathbb{N}$ as

$$H_k := \left\{\boldsymbol{x} \in \mathbb{R}^N : \langle \boldsymbol{u}_k, \boldsymbol{x}\rangle = d_k\right\},\ k \in \mathbb{N}. \tag{3.66}$$

Then the projection of an arbitrary $\boldsymbol{y} \in \mathbb{R}^N$ onto H_k is given by (see (3.41)):

$$P_{H_k}(\boldsymbol{y}) = \boldsymbol{y} - \frac{\langle \boldsymbol{u}_k, \boldsymbol{y}\rangle - d_k}{\|\boldsymbol{u}_k\|^2}\boldsymbol{u}_k, \tag{3.67}$$

which implies that (3.65) can be rewritten as follows:

$$\boldsymbol{h}_{k+1} = \boldsymbol{h}_k + \lambda\left(P_{H_k}(\boldsymbol{h}_k) - \boldsymbol{h}_k\right),\ k \in \mathbb{N}. \tag{3.68}$$

A geometric interpretation of NLMS is given in Fig. 3-3. In the case of $\lambda = 2$, $\boldsymbol{h}_{k+1} = 2P_{H_k}(\boldsymbol{h}_k) - \boldsymbol{h}_k$ is the reflection of $\boldsymbol{h}_k$ with respect to H_k. Let us consider the noiseless situation; i.e., $n_k = 0$, $k \in \mathbb{N}$. In this case, $d_k := \langle \boldsymbol{u}_k, \boldsymbol{h}^*\rangle + n_k = \langle \boldsymbol{u}_k, \boldsymbol{h}^*\rangle$ and the hyperplane becomes $H_k = \left\{\boldsymbol{x} \in \mathbb{R}^N : \langle \boldsymbol{u}_k, \boldsymbol{x}\rangle = \langle \boldsymbol{u}_k, \boldsymbol{h}^*\rangle\right\}$, implying $\boldsymbol{h}^* \in H_k$. Therefore, referring to Fig. 3-4, it is seen that Pythagorean theorem (see Lemma 2.37) ensures that $\|\boldsymbol{h}_{k+1} - \boldsymbol{h}^*\| \le \|\boldsymbol{h}_k - \boldsymbol{h}^*\|$ for any $\lambda \in [0, 2]$. In words, $(\boldsymbol{h}_k)_{k\in\mathbb{N}}$ monotonically approaches $\boldsymbol{h}^*$ at every iteration step. Also it is seen that $\boldsymbol{h}_{k+1}$ for $\lambda = 1$ is closest to $\boldsymbol{h}^*$ over $\lambda \in [0, 2]$. This implies that the use of $\lambda = 1$ provides the fastest convergence. (Note here that in the presence of noise we cannot guarantee $\boldsymbol{h}^* \in H_k$, thus one needs to use λ smaller than one, depending on SNR conditions.)

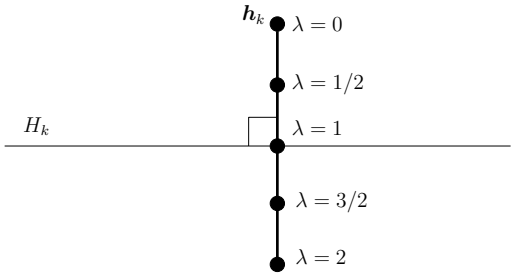


Fig. 3-3. Illustration of the NLMS algorithm. Each dot along the line orthogonal to the hyperplane H_k stands for $\boldsymbol{h}_{k+1}$ for each value of λ.

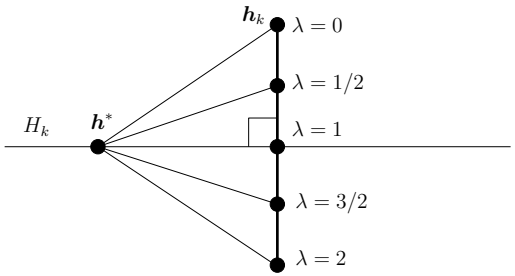
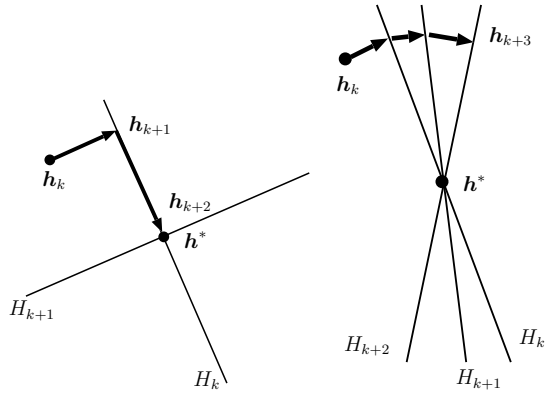


Fig. 3-4. Illustration of the NLMS algorithm in noiseless case.

The use of $\lambda = 1$ reduces NLMS to $\boldsymbol{h}_{k+1} := P_{H_k}(\boldsymbol{h}_k)$ which is similar to the Kaczmarz's alternating projection method (3.45). The difference is that the Kaczmarz's method utilizes each hyperplane infinitely many times whereas NLMS utilizes each hyperplane only once. Because of this difference, the analysis of NLMS becomes quite different from that of the Kaczmarz's method. Nevertheless the result about the rate of convergence for the Kaczmarz's method gives an insight into the behavior of NLMS. Consider the two situations: (i) the input signals are *uncorrelated* and (ii) the input signals are *strongly correlated*. The angle between two hyperplanes H_k and H_j $(k \neq j)$ is determined by $c(H_k, H_j) = \dfrac{\langle \boldsymbol{u}_k, \boldsymbol{u}_j \rangle}{\|\boldsymbol{u}_k\| \, \|\boldsymbol{u}_j\|}$. For the case (i) it is expected that $c(H_k, H_j)$



(a) uncorrelated input case (b) correlated input case

Fig. 3-5. Illustration of a few steps of NLMS in noiseless case.

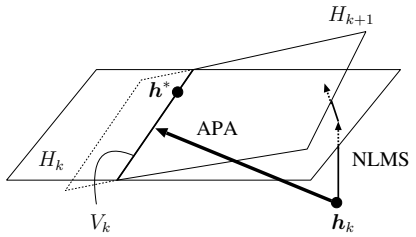


Fig. 3-6. Illustration of APA ($r = 2$) and NLMS for $\lambda = 1$ in noiseless case.

is nearly zero (meaning that the angle is nearly $\pi/2$ [rad]), whereas for the case (ii) it is expected that $c(H_k, H_j)$ is nearly unity (meaning that the angle is nearly 0 [rad]). Therefore, we can expect that the NLMS algorithm converges faster in the case (i) compared to the case (ii). This is clearly demonstrated in Fig. 3-5.

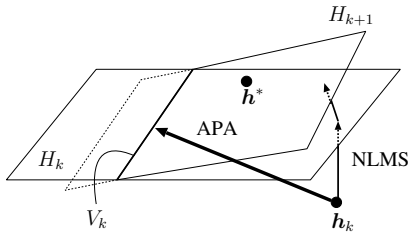


Fig. 3-7. Illustration of APA ($r = 2$) and NLMS for $\lambda = 1$ in the presence of noise.

3.10. Projection Based Adaptive Filtering Algorithm: APA

The slow convergence for strongly correlated input signals, such as *speech*, is indeed the major drawback of the NLMS algorithm. To alleviate the drawback, the *Affine Projection Algorithm (APA)* has been proposed. The idea is the following. Since each hyperplane H_k, $k \in \mathbb{N}$, contains $\boldsymbol{h}^*$ in noiseless case, their intersection $V_k := H_k \cap H_{k-1} \cap \cdots \cap H_{k-r+1}$ for some $r \in \mathbb{N}^*$ should also contain $\boldsymbol{h}^*$ with the nonemptiness of V_k assumed. Substituting V_k for H_k in (3.68), we obtain the following algorithm [**27**]:

$$\boldsymbol{h}_{k+1} := \boldsymbol{h}_k + \lambda \left(P_{V_k}(\boldsymbol{h}_k) - \boldsymbol{h}_k \right), \ k \in \mathbb{N}. \tag{3.69}$$

The linear variety V_k can be rewritten in a matrix form as $V_k = \{\boldsymbol{x} \in \mathbb{R}^N : \boldsymbol{U}_k^{\mathsf{T}} \boldsymbol{x} = \boldsymbol{d}_k\}$ with $\boldsymbol{U}_k := [\boldsymbol{u}_k \boldsymbol{u}_{k-1} \cdots \boldsymbol{u}_{k-r+1}] \in \mathbb{R}^{N \times r}$ and $\boldsymbol{d}_k := [d_k, d_{k-1}, \cdots, d_{k-r+1}]^{\mathsf{T}} \in \mathbb{R}^r$. The projection $P_{V_k}(\boldsymbol{h}_k)$ has the following closed form expression (see Example 3.39):

$$P_{V_k}(\boldsymbol{h}_k) = \boldsymbol{h}_k - \boldsymbol{U}_k (\boldsymbol{U}_k^{\mathsf{T}} \boldsymbol{U}_k)^{-1} (\boldsymbol{U}_k^{\mathsf{T}} \boldsymbol{h}_k - \boldsymbol{d}_k). \tag{3.70}$$

Obviously the NLMS algorithm is a particular case of APA for $r = 1$, hence APA is a generalization of NLMS. Figure 3-6 illustrates the behavior of APA ($r = 2$) and NLMS in the noiseless case. It is seen that APA gets closer to $\boldsymbol{h}^*$ than NLMS, suggesting that APA converges faster than NLMS for strongly correlated input signals. Unfortunately, this does *not* apply in the noisy case. Figure 3-7 illustrates the behavior of APA ($r = 2$) and NLMS in the presence of noise. Since it is not ensured that $\boldsymbol{h}^* \in V_k$ (nor $\boldsymbol{h}^* \in H_k$), we cannot say that APA gets closer to $\boldsymbol{h}^*$ than NLMS. This issue will further be discussed in Lecture

4. In [**28**], APA has been slightly generalized into the following:

$$\boldsymbol{h}_{k+1} := \boldsymbol{h}_k - \lambda (\boldsymbol{U}_k^{\mathsf{T}})^{\dagger} (\boldsymbol{U}_k^{\mathsf{T}} \boldsymbol{h}_k - \boldsymbol{d}_k), \tag{3.71}$$

which covers the case that the columns of $\boldsymbol{U}_k$ are linearly dependent. Referring to Remark 3.6, (3.71) can be expressed in the form of (3.69) with

$$V_k := \operatorname*{argmin}_{\boldsymbol{x} \in \mathbb{R}^N} \left\| \boldsymbol{U}_k^{\mathsf{T}} \boldsymbol{x} - \boldsymbol{d}_k \right\|_r, \tag{3.72}$$

where $\|\cdot\|_r$ stands for the Euclidean norm in $\mathbb{R}^r$.

LECTURE 4
Set Theoretic Adaptive Filtering

4.1. Outline of Lecture 4

4.2. Introduction

In the previous lectures, we have already presented the theory and method of orthogonal projections. In this lecture, we present its nonlinear extension. The existence and uniqueness of the orthogonal projection (Theorem 2.48) can be extended to the more general class of sets referred to as *closed convex*. The contents of the remainder of the lectures stem highly on *convex analysis* [**29**–**34**] which is a well established segment of *nonlinear functional analysis*.

The theme of this lecture is the *set theoretic adaptive filtering* [**35, 36**], which is motivated by the *set theoretic estimation* [**37**]. An ordinary approach to estimation problems is to optimize a cost function under possible constraints. In practical scenarios, the observed data are corrupted by ambient noise, and therefore the cost function defined

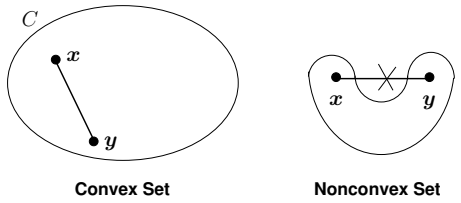


Fig. 4-1. Convex and nonconvex sets.

with the noisy data could be unrealistic or even unsolvable. Consequently, the reliability of the solutions, if exist, becomes questionable. The set theoretic estimation takes a significantly different approach based on the notion of *feasibility*. Several iterative algorithms, including the popular POCS method, to realize the set theoretic estimation are presented; the algorithms are actually generalizations of the projection algorithms presented in Section 3.7. The set theoretic frame for adaptive filtering provides a reason for the noise sensitivity of APA, and it brings the set theoretic adaptive filtering. Some known results about the convergence of a set theoretic adaptive filtering algorithm, as well as the iterative algorithms for the set theoretic estimation, are presented.

4.3. Convex Set and Convex Function

A set C is said to be *convex* if the line segment connecting any pair of points $\boldsymbol{x}, \boldsymbol{y} \in C$ is a subset of C (see Fig. 4-1). Its mathematical definition is given below.

Definition 4.1. A subset C of a vector space is said to be *convex* if $\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y} \in C$, $\forall \boldsymbol{x}, \boldsymbol{y} \in C$, $\forall \alpha \in (0,1)$.

If a set is closed (with a metric equipped) and convex, we say that it is *closed convex*. In the remainder of this lecture, we solely consider a Hilbert space $\mathcal{H}$ (rather than other vector spaces such as a Banach space) to avoid confusion, although it is not always necessary. Several examples of closed convex sets are given below.

Example 4.2.

(a) Comparing the definition of convex set and the necessary and sufficient condition for linear variety in Proposition 2.12, it is seen that the condition for convex set is weaker than that for linear variety; the difference is the range of α. Therefore all linear varieties (and obviously subspaces) are convex.

(b) The set $B[\mathbf{0}, \delta] := \{\boldsymbol{x} \in \mathcal{H} : \|\boldsymbol{x}\| \leq \delta\}$ for some $\delta > 0$ is called a *closed ball* (centered at $\mathbf{0}$). More generally, a closed ball centered at $\boldsymbol{x}_c \in \mathcal{H}$ with the radius $\delta > 0$ is defined as $B[\boldsymbol{x}_c, \delta] := \{\boldsymbol{x} \in \mathcal{H} : \|\boldsymbol{x} - \boldsymbol{x}_c\| \leq \delta\}$.[1]

(c) Given a nonzero vector $\boldsymbol{a} \in \mathcal{H}$ and $b \in \mathbb{R}$, the set $H^- := \{\boldsymbol{x} \in \mathcal{H} : \langle \boldsymbol{x}, \boldsymbol{a} \rangle \leq b\}$, or the set $H^+ := \{\boldsymbol{x} \in \mathcal{H} : \langle \boldsymbol{x}, \boldsymbol{a} \rangle \geq b\}$, is called a *closed halfspace*. The boundary of a *closed halfspace*, $H := \{\boldsymbol{x} \in \mathcal{H} : \langle \boldsymbol{x}, \boldsymbol{a} \rangle = b\}$, is called the *boundary hyperplane* of H^-, or H^+.

(d) Given a nonzero vector $\boldsymbol{a} \in \mathcal{H}$ and $b, c \in \mathbb{R}$ such that $b < c$, the set $S := \{\boldsymbol{x} \in \mathcal{H} : b \leq \langle \boldsymbol{x}, \boldsymbol{a} \rangle \leq c\}$, is called a *hyperslab*.

(e) Let $\mathcal{H}$ be of finite dimension and $\{\boldsymbol{u}_k\}_{i=1}^n$, $n \in \mathbb{N}^*$, be its orthonormal basis. Then, $C := \{\boldsymbol{x} \in \mathcal{H} : |\langle \boldsymbol{x}, \boldsymbol{u}_i \rangle| \leq b,\ \forall i = 1, 2, \cdots, n\}$ for some $b > 0$ is called a *hypercube*. A simplest example is $C := \left\{[x_1, x_2, \cdots, x_n]^\mathsf{T} \in \mathbb{R}^n : |x_i| \leq b,\ \forall i = 1, 2, \cdots, n\right\}$.

See [**14**] for other popular examples of convex sets such as *cone*, *polyhedra*, etc.

Proposition 4.3. *For an arbitrary collection of convex sets $\{C_i\}_{i \in \mathcal{I}}$, the intersection $\bigcap_{i \in \mathcal{I}} C_i$ is convex.*

Propositions 2.22 and 4.3 guarantee that *the intersection of an arbitrary collection of closed convex sets are closed convex.*

Definition 4.4. A function $f : \mathcal{H} \to \mathbb{R}$ is said to be *convex* if $\alpha f(\boldsymbol{x}) + (1-\alpha) f(\boldsymbol{y}) \geq f(\alpha \boldsymbol{x} + (1-\alpha)\boldsymbol{y})$, $\forall \boldsymbol{x}, \boldsymbol{y} \in C$, $\forall \alpha \in (0, 1)$.

Definition 4.5. Given a function $f : \mathcal{H} \to \mathbb{R}$ and $a \in \mathbb{R}$, the set $\mathrm{lev}_{\leq a} f := \{\boldsymbol{x} \in \mathcal{H} : f(\boldsymbol{x}) \leq a\}$, or $\mathrm{lev}_{< a} f := \{\boldsymbol{x} \in \mathcal{H} : f(\boldsymbol{x}) < a\}$, is said to be the (lower) level set, or the strict (lower) level set, of f at height a.

Proposition 4.6. *Let $f : \mathcal{H} \to \mathbb{R}$ be a continuous convex function. Then, for any $a \in \mathbb{R}$, the level set $\mathrm{lev}_{\leq a} f$ is closed convex.*[2]

[1] The surface of a closed ball is called a *hypersphere* in general, and it is nonconvex.

[2] The necessary and sufficient condition for $\mathrm{lev}_{\leq a} f$ to be closed for any $a \in \mathbb{R}$ is that f is *lower-semicontinuous* (which is more general than continuous).

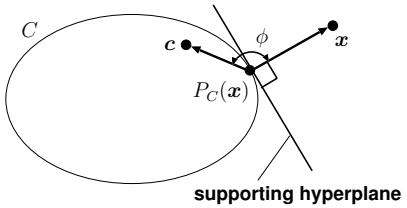


Fig. 4-2. Metric projection.

Remark. In fact, any closed convex set $C \subset \mathcal{H}$ can be characterized as the level set of a continuous convex function (at height 0). For instance, $C = \mathrm{lev}_{\leq 0}\, d_C$, where $d_C : \mathcal{H} \to [0, \infty), \boldsymbol{x} \mapsto \min_{\boldsymbol{y} \in C} \|\boldsymbol{x} - \boldsymbol{y}\|$ is a metric distance function; the existence of minimum is guaranteed by the convex projection theorem presented in Section 4.4.

4.4. Convex Projection Theorem

Theorem 4.7. *Let $\mathcal{H}$ be a Hilbert space, C a closed convex subset of $\mathcal{H}$, and $\boldsymbol{x} \in \mathcal{H}$ chosen arbitrarily. Then, there exists a unique point $\boldsymbol{c}_0 \in C$ such that $\|\boldsymbol{x} - \boldsymbol{c}_0\| \leq \|\boldsymbol{x} - \boldsymbol{c}\|$, $\forall \boldsymbol{c} \in C$. Moreover, $\boldsymbol{c}_0$ is the unique minimizer if and only if $\langle \boldsymbol{x} - \boldsymbol{c}_0, \boldsymbol{c} - \boldsymbol{c}_0 \rangle \leq 0$, $\forall \boldsymbol{c} \in C$. The $\boldsymbol{c}_0$ is called the convex projection, or the metric projection, of $\boldsymbol{x}$ onto C, and we denote it as $P_C(\boldsymbol{x}) := \mathrm{argmin}_{\boldsymbol{c} \in C} \|\boldsymbol{x} - \boldsymbol{c}\|$.*

Figure 4-2 illustrates the geometric property of metric projection; the characterization of projection $\langle \boldsymbol{x} - P_C(\boldsymbol{x}), \boldsymbol{c} - P_C(\boldsymbol{x}) \rangle \leq 0, \forall \boldsymbol{c} \in C$, means that $\phi \geq \pi/2$. Moreover, the characterization implies

$$C \subset H^-(\boldsymbol{x}) := \{\boldsymbol{y} \in \mathcal{H} : \langle \boldsymbol{x} - P_C(\boldsymbol{x}), \boldsymbol{y} - P_C(\boldsymbol{x}) \rangle \leq 0\}. \tag{4.8}$$

The boundary hyperplane of $H^-(\boldsymbol{x})$, which is tangent to C at $P_C(\boldsymbol{x})$, is called a *supporting hyperplane* of C at $P_C(\boldsymbol{x})$.

The orthogonal projection is a special example of metric projections; it is called so because of its geometric property. It should be remarked that existence and uniqueness of the projection (i.e., best approximation) is ensured for a general closed convex set. This does *not* apply when the set is nonconvex. This suggests that one should classify the world into *convex* and *nonconvex*, rather than *linear* and *nonlinear*. Several properties of metric projection are given below.

Proposition 4.9. *Let C be a closed convex set in a Hilbert space $\mathcal{H}$. Then the following statements hold.*

(a) *For any $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{H}$,*

$$\|P_C(\boldsymbol{x}) - P_C(\boldsymbol{y})\| \leq \|\boldsymbol{x} - \boldsymbol{y}\| . \tag{4.10}$$

(b) *For any $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{H}$,*

$$\|P_C(\boldsymbol{x}) - P_C(\boldsymbol{y})\|^2 \leq \langle \boldsymbol{x} - \boldsymbol{y}, P_C(\boldsymbol{x}) - P_C(\boldsymbol{y}) \rangle . \tag{4.11}$$

(c) *For any $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{H}$,*

$$\|\boldsymbol{x} - P_C(\boldsymbol{x})\|^2 \leq \|\boldsymbol{x} - \boldsymbol{y}\|^2 - \|P_C(\boldsymbol{x}) - P_C(\boldsymbol{y})\|^2 . \tag{4.12}$$

The properties in (4.10) – (4.12) will be discussed in detail in Lecture 5. Obviously, (4.11) implies (4.10) due to the Cauchy-Schwarz inequality.

Define the operator $T_C := I + \lambda(P_C - I)$, $\lambda \in [0, 2]$, where $I : \mathcal{H} \to \mathcal{H}, \boldsymbol{x} \mapsto \boldsymbol{x}$ is the identity operator (see Fig. 4-3). If $\lambda < 1$, $T_C(\boldsymbol{x})$ does not reach C, namely it *under-projects* $\boldsymbol{x}$ toward C. In contrast, if $\lambda > 1$, $T_C(\boldsymbol{x})$ lies farther away from $\boldsymbol{x}$ than $P_C(\boldsymbol{x})$, namely it *over-projects* $\boldsymbol{x}$ toward C. The operator T_C is thus called the *relaxed projector* for C with the *relaxation parameter* $\lambda \in [0, 2]$. In the case of $\lambda = 2$, $T_C(\boldsymbol{x})$ is said to be the *reflection* of $\boldsymbol{x}$ with respect to C.

Proposition 4.13. *Let C be a closed convex set in a Hilbert space $\mathcal{H}$. Then the following statements hold.*

(a) *For any $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{H}$,*

$$\|T_C(\boldsymbol{x}) - T_C(\boldsymbol{y})\| \leq \|\boldsymbol{x} - \boldsymbol{y}\| . \tag{4.14}$$

(b) *For any $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{H}$,*

$$\frac{2 - \lambda}{\lambda} \|\boldsymbol{x} - T_C(\boldsymbol{x})\|^2 \leq \|\boldsymbol{x} - \boldsymbol{y}\|^2 - \|T_C(\boldsymbol{x}) - T_C(\boldsymbol{y})\|^2 . \tag{4.15}$$

(c) *For any $\boldsymbol{x} \in \mathcal{H}$ and $\boldsymbol{y} \in C$,*

$$\lambda(2 - \lambda) \|\boldsymbol{x} - P_C(\boldsymbol{x})\|^2 \leq \|\boldsymbol{x} - \boldsymbol{y}\|^2 - \|T_C(\boldsymbol{x}) - \boldsymbol{y}\|^2 . \tag{4.16}$$

The relaxed projector T_C will play a main role in the theory of POCS presented in Section 4.7.

4.5. Calculus : Projections onto Convex Sets

This section presents closed-form formulae of the metric projections for several types of closed convex sets.

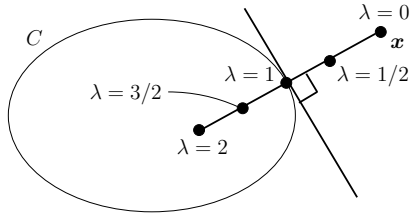


Fig. 4-3. Relaxed projection $T_C := I + \lambda(P_C - I)$, $\lambda \in [0, 2]$.

Example 4.17.

(a) See Lecture 3 for the projections onto certain types of closed linear varieties or closed subspaces.

(b) The projection of $\boldsymbol{y} \in \mathcal{H}$ onto a closed ball $B[\boldsymbol{x}_c, \delta] := \{\boldsymbol{x} \in \mathcal{H} : \|\boldsymbol{x} - \boldsymbol{x}_c\| \leq \delta\}$ is given by

$$P_{B[\boldsymbol{x}_c,\delta]}(\boldsymbol{y}) = \begin{cases} \boldsymbol{y} & \text{if } \|\boldsymbol{y} - \boldsymbol{x}_c\| \leq \delta \\ \boldsymbol{x}_c + \delta \dfrac{\boldsymbol{y} - \boldsymbol{x}_c}{\|\boldsymbol{y} - \boldsymbol{x}_c\|} & \text{otherwise.} \end{cases} \tag{4.18}$$

(c) The projection of $\boldsymbol{y} \in \mathcal{H}$ onto the closed halfspace $H^- := \{\boldsymbol{x} \in \mathcal{H} : \langle \boldsymbol{x}, \boldsymbol{a} \rangle \leq b\}$ is given by

$$P_{H^-}(\boldsymbol{y}) = \begin{cases} \boldsymbol{y} & \text{if } \langle \boldsymbol{y}, \boldsymbol{a} \rangle \leq b \\ \boldsymbol{y} - \dfrac{\langle \boldsymbol{a}, \boldsymbol{y} \rangle - b}{\|\boldsymbol{a}\|^2} \boldsymbol{a} & \text{otherwise.} \end{cases} \tag{4.19}$$

(d) The projection of $\boldsymbol{y} \in \mathcal{H}$ onto the hyperslab $S := \{\boldsymbol{x} \in \mathcal{H} : b \leq \langle \boldsymbol{x}, \boldsymbol{a} \rangle \leq c\}$ is given by

$$P_S(\boldsymbol{y}) = \begin{cases} \boldsymbol{y} - \dfrac{\langle \boldsymbol{a}, \boldsymbol{y} \rangle - b}{\|\boldsymbol{a}\|^2} \boldsymbol{a} & \text{if } \langle \boldsymbol{y}, \boldsymbol{a} \rangle < b \\ \boldsymbol{y} & \text{if } b \leq \langle \boldsymbol{y}, \boldsymbol{a} \rangle \leq c \\ \boldsymbol{y} - \dfrac{\langle \boldsymbol{a}, \boldsymbol{y} \rangle - c}{\|\boldsymbol{a}\|^2} \boldsymbol{a} & \text{if } c < \langle \boldsymbol{y}, \boldsymbol{a} \rangle . \end{cases} \tag{4.20}$$

(e) In the case of $\mathcal{H} := \mathbb{R}^N$, the projection of $[y_1, y_2, \cdots, y_N]^\mathsf{T} \in \mathbb{R}^N$ onto the hypercube $C := \{[x_1, x_2, \cdots, x_N]^\mathsf{T} \in \mathbb{R}^N : |x_i| \leq b, \ \forall i = 1, 2, \cdots, n\}$ is given by $P_C(\boldsymbol{y}) = [p_1, p_2, \cdots, p_N]^\mathsf{T}$ with

$$p_i := \begin{cases} y_i & \text{if } |y_i| \leq b \\ \dfrac{b y_i}{|y_i|} & \text{otherwise} \end{cases} \quad i = 1, 2, \cdots, N. \tag{4.21}$$

In Example 4.17(b)–(d), each projection is computable with $O(N)$ multiplications, where N stands for the number of multiplications required to evaluate the inner product. The same applies to the case of hyperplane. Therefore, hyperplane, closed halfspace, and hyperslab are often employed in signal processing applications as well as closed ball and hypercube.

4.6. Convex Feasibility Problem and Set Theoretic Estimation

Set theoretic estimation stems from a quite different concept from the usual optimization. The concept is finding a *feasible* solution — which is characterized as a vector consistent with all available information arising from the observed data and *a priori* knowledge — instead of finding an *optimal* solution, which is usually characterized as a minimizer of a certain cost function under possible constraints. The set of all feasible solutions is called a *feasible set* (or a solution set), and the problem of finding a feasible solution is called a *feasibility problem*. Of course, a feasibility problem can be formulated as the optimization problem of minimizing the distance to its associated feasible set. However, feasibility problems is distinguished from optimization problems, as its concept and formulation is significantly different.

In general, the feasibility set is characterized as the intersection of multiple sets as follows:

$$S := \bigcap_{i\in\mathcal{I}} S_i. \tag{4.22}$$

Each set S_i ($i \in \mathcal{I}$) accommodates each piece of information available; it is referred to as a *property set*. The formal definition of set theoretic estimation (the definition of S_i) is accompanied by the notion of *fuzzy proposition* and the interested reader may refer to [**37**]. In a large number of applications, the property sets S_i are closed convex (so S is) and in this case the problem is called a *convex feasibility problem* [**38, 39**]. The problem is said to be *consistent* if $S \neq \emptyset$.[3] In consistent cases, any point in S is called *set theoretic estimate*.

A simplest example is the case that each S_i is a hyperplane in $\mathbb{R}^N$. In this case, we have seen in Section 3.7 that the problem can be solved by using the projection method of Kaczmarz, Cimmino, or Reich. Another example is the case that each S_i is a closed subspace in a general Hilbert space $\mathcal{H}$. In this case, the Halperin's result (an extension of von

[3]If we assume existence of "true" object (or *estimandum*) $\boldsymbol{h} \in \mathcal{H}$, the problem is said to be *fair* if $\boldsymbol{h} \in S$, and *ideal* if $S = \{\boldsymbol{h}\}$.

Neumann' result for two subspaces) can be applied. From now on, we consider the case of general convex sets, and assume that the problem has no analytical solution. This situation frequently happens because the number of property sets increases as our theoretical and practical understanding of the physical system under study becomes better, which makes the problem more complicated. A popular approach in such a situation is algorithmic. Metric projection is a powerful tool, as like orthogonal projection in the case of subspaces. In Sections 4.7 and 4.8, we assume that every set S_i is "simple" in the sense that the projection onto S_i can be calculated explicitly. In Section 4.9, we discuss the case that some of S_is are *not* simple.

4.7. POCS — Successive Projection Methods

We present the fundamental theory of Projections onto Convex Sets (POCS) — which is also known as Successive Orthogonal Projections (SOP). As the name suggests, it operates the projection onto each individual closed convex set in a cyclic manner; it is an extension of Kaczmarz's method. The method is generically called *successive*, *serial*, or *sequential* projection algorithm (see Fig. 3-1). In contrast, we will present later a generalization of Cimmino's method and it is called *parallel* or *simultaneous* projection algorithm (see Fig. 3-2).

We formulate the convex feasibility problems in a Hilbert space $\mathcal{H}$ as follows:

$$\text{find } \boldsymbol{x}^* \in C := \bigcap_{i\in\mathcal{I}} C_i, \tag{4.23}$$

if such an $\boldsymbol{x}^*$ exists. Here, $C_i \subset \mathcal{H}$, $i \in \mathcal{I} := \{1, 2, \cdots, n\}$, is closed convex. Define the relaxed projectors as follows:

$$T_i := I + \lambda_i(P_{C_i} - I),\ i \in \mathcal{I}, \tag{4.24}$$

where $\lambda_i \in (0, 2)$. The following is a known fundamental result on the convergence of POCS in the consistent case.

Theorem 4.25. *Assume that C is nonempty. Then for any $\boldsymbol{x}_0 \in \mathcal{H}$ and any $\lambda_i \in (0, 2)$, $i \in \mathcal{I}$, the sequence $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ generated by*

$$\boldsymbol{x}_{k+1} := T_n T_{n-1} \cdots T_1(\boldsymbol{x}_k),\ k \in \mathbb{N}, \tag{4.26}$$

converges weakly to a point in C.

Theorem 4.25 was first proved by L. G. Gubin, B. T. Polyak, and E. V. Raik in 1967 [**40**]. Later, D. C. Youla provided an alternative proof in 1982 [**41**] based on Opial's lemma which is an important result in the fixed point theory of *nonexpansive mappings*. Lecture 5

will provide a proof based on a more general result in the fixed point theory. We emphasize that we *cannot* guarantee in general that the weak limit point of the sequence in Theorem 4.25 is $P_C(\boldsymbol{x}_0)$ unlike the case of subspaces or linear varieties (see Propositions 3.49 and 3.51). Nowadays, POCS is a quite popular technique in signal processing, but its early application to signal processing appeared in 1981 [**42**]. See, e.g., [**43, 44**] for a slightly more general form of POCS.

Remark. The special case that C_is are closed halfspaces in the Euclidean space $\mathbb{R}^N$ has been proved in 1954 by S. Agmon [**45**] and T. S. Motzkin and I. J. Schoenberg [**46**]. In the case of halfspaces, λ_i ($i \in \mathcal{I}$) can take a value in $(0, 2]$. In fact, their works are a generalization of Kaczmarz's method in two senses (i.e., hyperplanes to halfspaces, $\lambda_i = 1$ to $\lambda_i \in (0, 2]$). Finding a common point of halfspaces is equivalent to solving linear inequalities, thus in this case the problem is specially called a *linear feasibility problem*.

In the inconsistent case (i.e., $C = \emptyset$), the following result is known for the case of $\lambda_i = 1$, $\forall i \in \mathcal{I}$.

Theorem 4.27. *Assume that one of the sets $C_1, C_2, \cdots, C_n$ is bounded; i.e., there exists some $\mu < \infty$ such that $\|\boldsymbol{x}\| < \mu$, $\forall \boldsymbol{x} \in C$. Then, the sequence generated by*

$$\boldsymbol{x}_{k+1} := P_{C_n} P_{C_{n-1}} \cdots P_{C_1}(\boldsymbol{x}_k), \ k \in \mathbb{N}, \tag{4.28}$$

converges weakly to a point $\boldsymbol{x}^* \in C_n$. Moreover, letting $\boldsymbol{x}_1^* := P_{C_1}(\boldsymbol{x}^*)$ and $\boldsymbol{x}_{i+1}^* := P_{C_{i+1}}(\boldsymbol{x}_i^*)$, $i = 1, 2, \cdots, n-1$, then $\boldsymbol{x}_n^* = \boldsymbol{x}^*$.

4.8. Simultaneous Projection Methods

We shall present a parallel projection algorithm derived by G. Pierra in 1984 by formulating POCS in a *product space* [**47**]. Although the derivation is interesting, we simply state the results. Define convex combination coefficients $(w_i)_{i \in \mathcal{I}}$ as follows:

$$w_i > 0, \ \forall i \in \mathcal{I}, \ \text{and} \ \sum_{i \in \mathcal{I}} w_i = 1. \tag{4.29}$$

The results in the consistent case is the following.

Theorem 4.30. *Assume that C is nonempty. Then for any $\boldsymbol{x}_0 \in \mathcal{H}$, any $(w_i)_{i \in \mathcal{I}}$ satisfying* (4.29)*, and any $\lambda \in (0, 2)$, the sequence $(\boldsymbol{x}_k)_{k \in \mathbb{N}}$ generated by*

$$\boldsymbol{x}_{k+1} := \boldsymbol{x}_k + \lambda \left(\sum_{i \in \mathcal{I}} w_i P_{C_i}(\boldsymbol{x}_k) - \boldsymbol{x}_k \right), \ k \in \mathbb{N}, \tag{4.31}$$

converges weakly to a point in C.

Theorem 4.32. *Assume that C is nonempty. Then for any $\boldsymbol{x}_0 \in \mathcal{H}$, any $(w_i)_{i \in \mathcal{I}}$ satisfying* (4.29)*, and any $\lambda_k \in [\epsilon, 2 - \epsilon] \subset (0, 2)$, the sequence $(\boldsymbol{x}_k)_{k \in \mathbb{N}}$ generated by*

$$\boldsymbol{x}_{k+1} := \boldsymbol{x}_k + \lambda_k \left(\sum_{i \in \mathcal{I}} w_i P_{C_i}(\boldsymbol{x}_k) - \boldsymbol{x}_k \right), \ k \in \mathbb{N}, \tag{4.33}$$

converges weakly to a point in C.

Theorem 4.34. *Assume that C is nonempty. Then for any $\boldsymbol{x}_0 \in \mathcal{H}$ and any $(w_i)_{i \in \mathcal{I}}$ satisfying* (4.29)*, the sequence $(\boldsymbol{x}_k)_{k \in \mathbb{N}}$ generated by*

$$\boldsymbol{x}_{k+1} := \boldsymbol{x}_k + \lambda_k \left(\sum_{i \in \mathcal{I}} w_i P_{C_i}(\boldsymbol{x}_k) - \boldsymbol{x}_k \right), \ k \in \mathbb{N}, \tag{4.35}$$

converges weakly to a point in C, where $\lambda_k \in [\epsilon, L_k] \subset (0, L_k]$ with the extrapolation coefficient

$$L_k := \begin{cases} \dfrac{\sum_{i \in \mathcal{I}} w_i \left\| P_{C_i}(\boldsymbol{x}_k) - \boldsymbol{x}_k \right\|^2}{\left\| \sum_{i \in \mathcal{I}} w_i P_{C_i}(\boldsymbol{x}_k) - \boldsymbol{x}_k \right\|^2} & \text{if } \boldsymbol{x}_k \notin C \\ 1 & \text{otherwise.} \end{cases} \tag{4.36}$$

All the algorithms in Theorems 4.30, 4.32, and 4.34 take the following steps: (i) project the current estimate $\boldsymbol{x}_k$ onto individual closed convex sets C_i, and then (ii) combine them by taking a weighted average. In contrast to POCS, one can operate all the projections simultaneously, thus it suits for parallel computing. The only difference among the algorithms in Theorems 4.30, 4.32, and 4.34 is the relaxation parameter λ or λ_k. Theorem 4.32 is a generalization of Theorem 4.30 as it allows the relaxation parameter λ to vary from iteration to iteration. We *cannot* however say that Theorem 4.34 is a generalization of Theorem 4.30 because L_k may become less than 2, although the convexity of $\|\cdot\|^2$ ensures $L_k \geq 1$. In [**44**], it is discussed that, under a certain condition, λ_k is allowed to take a value in $[\epsilon, 2L_k] \subset (0, 2L_k]$.

Finally, we present a known result for the inconsistent case.

Theorem 4.37. *Assume that one of the sets $C_1, C_2, \cdots, C_n$ is bounded. Then for any $\boldsymbol{x}_0 \in \mathcal{H}$, any $(w_i)_{i \in \mathcal{I}}$ satisfying* (4.29)*, and any $\lambda_k \in [\epsilon, 2 - \epsilon] \subset (0, 2)$, the sequence $(\boldsymbol{x}_k)_{k \in \mathbb{N}}$ generated by*

$$\boldsymbol{x}_{k+1} := \boldsymbol{x}_k + \lambda_k \left(\sum_{i \in \mathcal{I}} w_i P_{C_i}(\boldsymbol{x}_k) - \boldsymbol{x}_k \right), \ k \in \mathbb{N}, \tag{4.38}$$

converges weakly to a point $\boldsymbol{x}^* \in \mathcal{H}$ *achieving weighted least squares; i.e.,* $\boldsymbol{x}^* \in \operatorname{argmin}_{\boldsymbol{x}\in\mathcal{H}} \sum_{i\in\mathcal{I}} w_i d^2(\boldsymbol{x}, C_i)$.

In short, Theorem 4.37 tells us that the simultaneous projection method generates a vector sequence convergent to a weighted least squares solution. Due to this property, it has been reported that SIRT (a simultaneous projection method) gives better behavior than ART (a successive projection method) in noisy tomographic reconstruction problems, because noisy data tend to make the hyperplanes nonintersecting.

4.9. Subgradient Projection

It has been assumed so far that the projection onto each C_i can be calculated explicitly. In this section, we consider the case that some of the sets are *not* simple.

Definition 4.39. Let $f : \mathcal{H} \to \mathbb{R}$ be continuous and convex. Then, for any $\boldsymbol{x} \in \mathcal{H}$, there always exists $\tilde{\boldsymbol{x}} \in \mathcal{H}$ satisfying

$$\langle \boldsymbol{y} - \boldsymbol{x}, \tilde{\boldsymbol{x}} \rangle + f(\boldsymbol{x}) \le f(\boldsymbol{y}),\ \forall \boldsymbol{y} \in \mathcal{H}. \tag{4.40}$$

The vector $\tilde{\boldsymbol{x}}$ is called a *subgradient* of f at $\boldsymbol{x}$. The set of all such vectors is called the *subdifferential* of f at $\boldsymbol{x}$, and it is denoted as $\partial f(\boldsymbol{x})$.

Remark. Subgradient is a generalization of gradient, since it can always be defined for any continuous convex functions. If in particular f is differentiable,[4] then the gradient $\boldsymbol{\nabla} f(\boldsymbol{x})$ is the unique subgradient; i.e., $\partial f(\boldsymbol{x}) = \{\boldsymbol{\nabla} f(\boldsymbol{x})\}$.

The following proposition can readily be verified.

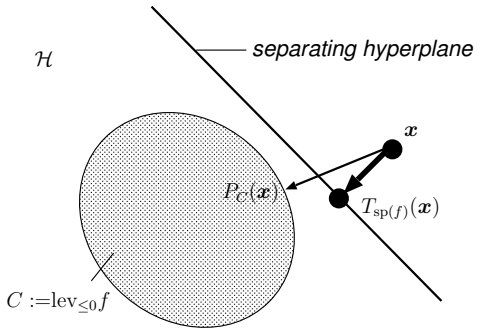
Proposition 4.41. *Let* $f : \mathcal{H} \to \mathbb{R}$ *be continuous and convex.*

(a) *Assume that* $f(\boldsymbol{x}) > \inf_{\boldsymbol{y}\in\mathcal{H}} f(\boldsymbol{y})$, $\forall \boldsymbol{x} \in \mathcal{H}$. *Then* $\boldsymbol{0} \notin \partial f(\boldsymbol{x})$ *for any* $\boldsymbol{x} \in \mathcal{H}$.

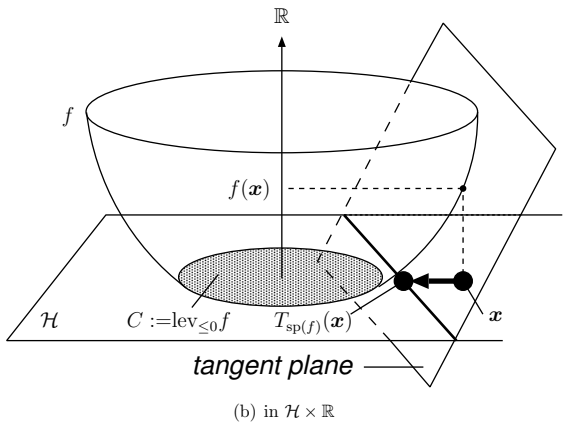
(b) *Assume that there exists* $\boldsymbol{x}$ *such that* $f(\boldsymbol{x}) = \inf_{\boldsymbol{y}\in\mathcal{H}} f(\boldsymbol{y})$. *Then* $\boldsymbol{0} \in \partial f(\boldsymbol{x})$ *if and only if* $f(\boldsymbol{x}) = \min_{\boldsymbol{y}\in\mathcal{H}} f(\boldsymbol{y})$.

Let $C \subset \mathcal{H}$ be a nonempty closed convex set, and $f : \mathcal{H} \to \mathbb{R}$ a continuous convex function such that $C = \operatorname{lev}_{\le 0} f$. When the projection onto C is not simple, a possible strategy is to employ *its outer approximation*, say $S(\supset C)$. Fortunately, the *separation theorem*, which is one of the fundamental results in convex analysis [**29**–**34**], guarantees

[4] To be precise, a continuous convex function f has a unique subgradient at $\boldsymbol{x} \in \mathcal{H}$ if it is *Gâteaux differentiable* at $\boldsymbol{x}$ [**30**]. The unique subgradient is identical to its Gâteaux differential at $\boldsymbol{x}$.



(a) in $\mathcal{H}$



(b) in $\mathcal{H} \times \mathbb{R}$

Fig. 4-4. Illustrations of subgradient projection. (a) $T_{\mathrm{sp}(f)}(\boldsymbol{x})$ is the metric projection of $\boldsymbol{x}$ onto a hyperplane separating $\boldsymbol{x}$ and C. (b) The separating hyperplane is the intersection, in the product space $\mathcal{H} \times \mathbb{R}$, of $\mathcal{H} \times \{0\}$ and the tangent plane at $(\boldsymbol{x}, f(\boldsymbol{x}))$.

the existence of a hyperplane separating any closed convex set from a

point in its complement (see Fig. 4-4.a); such a hyperplane is particularly called a *separating hyperplane*. Hence, a natural choice of the outer approximation of C would be the closed halfspace whose boundary is the separating hyperplane, since the projection can be computed easily. How can we construct such a closed halfspace systematically?

If $\boldsymbol{x} \in \mathrm{lev}_{\leq 0} f (= C)$, then the metric projection of $\boldsymbol{x}$ onto C is obviously given by $P_C(\boldsymbol{x}) = \boldsymbol{x}$. Assume now $\boldsymbol{x} \notin \mathrm{lev}_{\leq 0} f (= C)$, that is $f(\boldsymbol{x}) > 0$. In this case, Proposition 4.41 implies $\mathbf{0} \notin \partial f(\boldsymbol{x})$, thus for an arbitrary $f'(\boldsymbol{x}) \in \partial f(\boldsymbol{x})$ we can define a closed halfspace as follows:

$$H^-(\boldsymbol{x}) := \{\boldsymbol{y} \in \mathcal{H} : \langle \boldsymbol{y} - \boldsymbol{x}, f'(\boldsymbol{x}) \rangle + f(\boldsymbol{x}) \leq 0\}. \tag{4.42}$$

We can readily verify $\boldsymbol{x} \notin H^-(\boldsymbol{x})$. Moreover, we can show that $\mathrm{lev}_{\leq 0} f \subset H^-(\boldsymbol{x})$ as follows: letting $\boldsymbol{z} \in \mathrm{lev}_{\leq 0} f$, the definition of subgradient suggests that

$$\langle \boldsymbol{z} - \boldsymbol{x}, f'(\boldsymbol{x}) \rangle + f(\boldsymbol{x}) \leq f(\boldsymbol{z}) \leq 0, \tag{4.43}$$

which means $\boldsymbol{z} \in H^-(\boldsymbol{x})$. Therefore the boundary hyperplane of $H^-(\boldsymbol{x})$ separates $\boldsymbol{x}$ and $\mathrm{lev}_{\leq 0} f$ (see Fig. 4-4). As $\boldsymbol{x} \notin H^-(\boldsymbol{x})$, the projection of $\boldsymbol{x}$ onto $H^-(\boldsymbol{x})$ is given as follows (see Example 4.17.c):

$$P_{H^-(\boldsymbol{x})}(\boldsymbol{x}) := \boldsymbol{x} - \frac{f(\boldsymbol{x})}{\|f'(\boldsymbol{x})\|^2} f'(\boldsymbol{x}). \tag{4.44}$$

A subgradient projection is an operator that maps $\boldsymbol{x} \notin \mathrm{lev}_{\leq 0} f$ to $P_{H^-(\boldsymbol{x})}(\boldsymbol{x})$ and $\boldsymbol{x} \in \mathrm{lev}_{\leq 0} f$ to $\boldsymbol{x}$ itself.

Definition 4.45. Let $f : \mathcal{H} \to \mathbb{R}$ be a continuous convex function such that $\mathrm{lev}_{\leq 0} f \neq \emptyset$. We can then define

$$T_{\mathrm{sp}(f)} : \mathcal{H} \to \mathcal{H},\ \boldsymbol{x} \mapsto \begin{cases} \boldsymbol{x} - \dfrac{f(\boldsymbol{x})}{\|f'(\boldsymbol{x})\|^2} f'(\boldsymbol{x}) & \text{if } f(\boldsymbol{x}) > 0, \\ \boldsymbol{x} & \text{otherwise,} \end{cases} \tag{4.46}$$

where $f'(\boldsymbol{x}) \in \partial f(\boldsymbol{x})$. The mapping $T_{\mathrm{sp}(f)}$ is called a *subgradient projection* relative to f.

Remark. Given any closed convex set $C \subset \mathcal{H}$, a subgradient projection relative to the distance function d_C coincides with the metric projection onto C; i.e., $T_{\mathrm{sp}(d_C)} = P_C$ (see Remark in Section 4.3).

Theorem 4.47. *Let $f_i : \mathbb{R}^N \to \mathbb{R}$, $i \in \mathcal{I} := \{1, 2, \cdots, n\}$, be a continuous convex function such that $C := \bigcap_{i \in \mathcal{I}} \mathrm{lev}_{\leq 0} f_i \neq \emptyset$. Assume the uniform boundedness of the subgradients: i.e., for some $\tilde{\boldsymbol{x}} \in C$ there exists $K(\tilde{\boldsymbol{x}}) \in \mathbb{R}$ such that $\|f_i'(\boldsymbol{x})\| \leq K(\tilde{\boldsymbol{x}})$ for all the subgradients $f_i'(\boldsymbol{x}) \in \partial f_i(\boldsymbol{x})$, for any $i \in \mathcal{I}$ and any such $\boldsymbol{x} \in \mathbb{R}^N$ that satisfies*

$\|\boldsymbol{x} - \tilde{\boldsymbol{x}}\| \leq \|\boldsymbol{x}_0 - \tilde{\boldsymbol{x}}\|$, where $\boldsymbol{x}_0 \in \mathbb{R}^N$ is an arbitrary initial vector. Then, the sequence generated by

$$\boldsymbol{x}_{k+1} := T_n T_{n-1} \cdots T_1(\boldsymbol{x}_k),\ k \in \mathbb{N}, \tag{4.48}$$

converges to a point in C, where $T_i := I + \lambda_i (T_{\mathrm{sp}(f_i)} - I)$, $i \in \mathcal{I}$, for an arbitrary $\lambda_i \in (0, 2)$.

A remarkable advantage of the algorithm in (4.48) over POCS is that only the computation of a subgradient is required (instead of the computation of the projection which is obtained by solving a best approximation problem).

Theorem 4.49. *Let $f_i : \mathcal{H} \to \mathbb{R}$, $i \in \mathcal{I} := \{1, 2, \cdots, n\}$, be a continuous convex function such that $C := \bigcap_{i \in \mathcal{I}} \mathrm{lev}_{\leq 0} f_i \neq \emptyset$. Assume that the subdifferentials of $(f_i)_{i \in \mathcal{I}}$ are locally uniformly bounded (cf. [**44**]). Then, for any $\boldsymbol{x}_0 \in \mathcal{H}$ and any $(w_i)_{i \in \mathcal{I}}$ satisfying (4.29), the sequence $(\boldsymbol{x}_k)_{k \in \mathbb{N}}$ generated by*

$$\boldsymbol{x}_{k+1} := \boldsymbol{x}_k + \lambda_k \left(\sum_{i \in \mathcal{I}} w_i T_{\mathrm{sp}(f_i)}(\boldsymbol{x}_k) - \boldsymbol{x}_k \right),\ k \in \mathbb{N}, \tag{4.50}$$

converges weakly to a point in C, where $\lambda_k \in [\epsilon, (2 - \epsilon) L_k] \subset (0, 2L_k)$ with the extrapolation coefficient

$$L_k := \begin{cases} \dfrac{\sum_{i \in \mathcal{I}} w_i \left\| T_{\mathrm{sp}(f_i)}(\boldsymbol{x}_k) - \boldsymbol{x}_k \right\|^2}{\left\| \sum_{i \in \mathcal{I}} w_i T_{\mathrm{sp}(f_i)}(\boldsymbol{x}_k) - \boldsymbol{x}_k \right\|^2} & \text{if } \boldsymbol{x}_k \notin C \\ 1 & \text{otherwise.} \end{cases} \tag{4.51}$$

Remark. The key property that is common to the iterative algorithms in Theorems 4.47 and 4.49 is the following:

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\| \leq \|\boldsymbol{x}_k - \boldsymbol{x}^*\|,\ \forall \boldsymbol{x}^* \in C,\ \forall k \in \mathbb{N}. \tag{4.52}$$

The sequence $(\boldsymbol{x}_k)_{k \in \mathbb{N}}$ satisfying (4.52) is said to be *Fejér monotone with respect to C*.[5] It is clear that any Fejér monotone sequence is bounded. The Fejér monotonicity comes from the following property of subgradient projection:

$$\left\| T_{\mathrm{sp}(f)}(\boldsymbol{x}) - \boldsymbol{x}^* \right\| \leq \|\boldsymbol{x} - \boldsymbol{x}^*\|,\ \forall \boldsymbol{x} \in \mathcal{H},\ \forall \boldsymbol{x}^* \in \mathrm{lev}_{\leq 0} f. \tag{4.53}$$

The property in (4.53) is called *quasi-nonexpansivity*, which plays an important role in Lecture 5.

[5]The notion of Fejér monotonicity seems to be coined by T. S. Motzkin and I. J. Schoenberg [**46**].

4.10. Set Theoretic Frame for Adaptive Estimation

Recalling the discussion in Section 3.10, APA is based on the projection onto the following set:

$$V_k := \left\{ \boldsymbol{x} \in \mathbb{R}^N : \|\boldsymbol{e}_k(\boldsymbol{x})\|_r^2 = \delta_k \right\}, \tag{4.54}$$

where $\boldsymbol{e}_k(\boldsymbol{x}) := \boldsymbol{U}_k^{\mathsf{T}}\boldsymbol{x} - \boldsymbol{d}_k \in \mathbb{R}^r$, $\boldsymbol{x} \in \mathbb{R}^N$, $\boldsymbol{U}_k := [\boldsymbol{u}_k \boldsymbol{u}_{k-1} \cdots \boldsymbol{u}_{k-r+1}] \in \mathbb{R}^{N\times r}$, $\boldsymbol{d}_k := [d_k, d_{k-1}, \cdots, d_{k-r+1}]^{\mathsf{T}} \in \mathbb{R}^r$, and

$$\delta_k := \min_{\boldsymbol{y}\in\mathbb{R}^N} \|\boldsymbol{e}_k(\boldsymbol{y})\|_r^2, \ k \in \mathbb{N}. \tag{4.55}$$

As we know that $\boldsymbol{h}^* \notin V_k$ in the presence of noise, V_k should be "fattened" somehow to cover $\boldsymbol{h}^*$. The linear variety V_k is shaped like a "line" in $\mathbb{R}^N$, and we can "fatten" it with a constant $\rho \geq \delta_k \geq 0$ as follows:

$$C_k(\rho) := \left\{ \boldsymbol{x} \in \mathbb{R}^N : \|\boldsymbol{e}_k(\boldsymbol{x})\|_r^2 \leq \rho \right\}, \tag{4.56}$$

which is shaped like a "tube" whose "center core" is $V_k (= C_k(\delta_k))$. The set $C_k(\rho)$ quantitatively formulates the probability theoretic property of the noise process $(\boldsymbol{n}_k)_{k\in\mathbb{N}}$, and it is called a *stochastic property set*. Moreover, the convexity of $\|\cdot\|^2$ suggests $C_k(\rho) \neq \emptyset$ if and only if $\rho \geq \delta_k$. Since we typically have $r \ll N$, it tends to be satisfied that $\mathcal{R}(\boldsymbol{U}_k^{\mathsf{T}}) = \mathbb{R}^r$, implying $\delta_k = \min_{\boldsymbol{y}\in\mathbb{R}^N} \left\|\boldsymbol{U}_k^{\mathsf{T}}\boldsymbol{y} - \boldsymbol{d}_k\right\|_r^2 = 0$. Therefore, $C_k(\rho) \neq \emptyset$ for any $\rho \geq 0$ in practice.

It is clear that the parameter ρ governs the membership probability that $\boldsymbol{h}^* \in C_k(\rho)$. By $\boldsymbol{e}_k(\boldsymbol{h}^*) = -\boldsymbol{n}_k$, the membership probability is identical to the probability of $\xi := \sum_{i=1}^r n_{k-i+1}^2 = \|\boldsymbol{n}_k\|_r^2 \leq \rho$. Assume that $(n_k)_{k\in\mathbb{N}}$ is the noise process of i.i.d. (independent, identically distributed) Gaussian random variables $\mathcal{N}(0, \sigma^2)$; i.e., the normal distribution with the mean 0 and the variance σ^2. Then, the sum of its squares ξ is well known to follow the χ^2 statistic with r degrees of freedom whose probability density function is given by (see Fig. 4-5)[6]

$$f_r(\xi) = \begin{cases} \dfrac{1}{(\sigma\sqrt{2})^r \Gamma(r/2)} \xi^{r/2-1} e^{-\xi/2\sigma^2} & \text{if } \xi > 0 \\ 0 & \text{if } \xi \leq 0. \end{cases} \tag{4.57}$$

The membership probability is evaluated as follows:

$$\Pr\left(\boldsymbol{h}^* \in C_k(\rho)\right) = \Pr\left(\xi \leq \rho\right) = \int_0^\rho f_r(\xi) d\xi \in [0, 1], \tag{4.58}$$

where $\Pr(\cdot)$ stands for the probability that an *event* (e.g., $\boldsymbol{h}^* \in C_k(\rho)$) happens. As seen from Fig. 4-5, f_r is a strictly monotonically decreasing

[6] Γ represents the gamma function defined as $\Gamma(\alpha) := \int_0^\infty x^{\alpha-1}e^{-x}dx$, $\alpha > 0$.

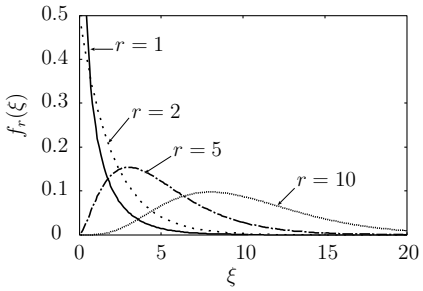


Fig. 4-5. χ^2 density function ($\sigma^2 = 1$).

function over $\xi \geq 0$ for $r = 1, 2$. For $r \geq 3$, f_r has its unique maximum at $\xi = (r-2)\sigma^2$. The mean and the variance of ξ are given by $m_\xi = r\sigma^2$ and $\sigma_\xi^2 = 2r\sigma^4$, respectively.[7]

Looking at the curve of $r = 5$ in Fig. 4-5, it is seen that $\int_0^\rho f_r(\xi)d\xi$ remains to be nearly zero if we slide slightly the value of ρ from zero in the positive direction. This implies that the membership probability that $\boldsymbol{h}^* \in C_k(\rho)$ is nearly zero for a small value of $\rho \geq 0$. In other words, the probability that $\boldsymbol{h}^*$ stays in the vicinity of $V_k (= C_k(\delta_k))$ is nearly zero. This observation applies to all $r \geq 3$, which explains the noise sensitivity of APA for $r \geq 3$. For $r = 1$, on the other hand, it is seen that $\int_0^\rho f_r(\xi)d\xi$ becomes large if the value of ρ is increased slightly from zero. This implies that $\boldsymbol{h}^*$ stays in the vicinity of $H_k (= C_k(0)$ for $r = 1$) with high probability (see (3.66)), which agrees with the noise robustness of NLMS.

4.11. Set Theoretic Adaptive Filtering Algorithm

How can we design the parameter ρ to design the stochastic property set $C_k(\rho)$ in (4.56)? How should we construct an efficient adaptive filtering algorithm? We discuss these topics in this section. Regarding the choice of ρ, there are two perspectives:

(a) the membership probability that $\boldsymbol{h}^* \in C_k(\rho)$ can be enhanced by increasing the value of ρ, contributing to the stability of algorithm;

[7] The case that the noise process $(n_k)_{k\in\mathbb{N}}$ is Gaussian but not necessarily i.i.d. was discussed in [**35**]. The case that $(n_k)_{k\in\mathbb{N}}$ is non-Gaussian was discussed in [**44**] with the well-known *central limit theorem*.

(b) increasing the value of ρ too much results in losing the information (In a extreme case, $C_k(\infty) = \mathbb{R}^N$ contains no information).

Therefore, the value of ρ should be chosen adequately. Under the assumption that $(n_k)_{k\in\mathbb{N}}$ is i.i.d. Gaussian random variables $\mathcal{N}(0, \sigma^2)$, the following have been proposed in [**35**].

Example 4.59.

(a) $\rho_1 := (r + \sqrt{2r})\sigma^2$ (mean + standard deviation)
(b) $\rho_2 := r\sigma^2$ (mean)
(c) $\rho_3 := \max\{0, (r-2)\sigma^2\}$ (peak, i.e., the value of ξ giving $f_r(\xi)$ its unique maximum)

It holds that $0 \le \rho_3 \le \rho_2 \le \rho_1$. Another possible choice is $\rho_4(\alpha) := \rho_3 + \alpha\sqrt{2r}\sigma^2$, $\alpha > 0$ (peak + standard deviation $\times\alpha$).

We now explain how to construct an efficient algorithm. In the following, we assume $C_k(\rho) \neq \emptyset$ ($\Leftrightarrow \rho \ge \delta_k$). It is readily verified that $C_k(\rho)$ is closed convex. Since $\boldsymbol{h}^*$ can be characterized as a common point of the closed convex sets $(C_k(\rho))_{k\in\mathcal{J}}$ for $\mathcal{J} := \{i \in \mathbb{N} : \boldsymbol{h}^* \in C_i(\rho)\}$, we can reformulate the adaptive filtering problem as a sort of convex feasibility problem. However, there are essential differences. First, data arrive sequentially, hence each set $C_k(\rho)$ accommodating the information carried by each datum becomes available one by one. Second, the number of sets $C_k(\rho)$ increases as time goes by, but finite memory storage implies that old data need to be discarded for storing newer data. This suggests that each set $C_k(\rho)$ can be exploited only a finite number of times, whereas the convergence theorems of the existing algorithms for the convex feasibility problems have been proved under the assumption that each set is exploited infinitely many times. This is important in practice because of the nature of the adaptive filtering problem as, e.g., in rapidly changing environments.

Since the metric projection onto $C_k(\rho)$ is computationally expensive, the subgradient projection is employed. It has been reported that the simultaneous projection algorithm as in Theorem 4.49 converges faster, than successive projection algorithms such as POCS, thanks to the extrapolation coefficient. Hence we present a simultaneous subgradient projection algorithm below.

Let $\mathcal{I}_k := \{\iota_1^{(k)}, \iota_2^{(k)}, \cdots, \iota_q^{(k)}\} \subset \{0, 1, 2, \cdots, k\}$ for some $q \in \mathbb{N}^*$, which indicates the sets to be processed at each iteration k and is called *control sequence*. In the case of parallel computing, q corresponds to the number of parallel processors to be engaged. A simple example is $\mathcal{I}_k = \{k, k-1, \cdots, k-q+1\}$ indicating the use of the q newest data. In addition, let $(w_\iota^{(k)})_{\iota\in\mathcal{I}_k}$, $k \in \mathbb{N}$, be the set of weights assigned to

$(C_\iota(\rho))_{\iota\in\mathcal{I}_k}$, satisfying

$$w_\iota^{(k)} > 0,\ \forall \iota \in \mathcal{I}_k,\ \text{and}\ \sum_{\iota\in\mathcal{I}_k} w_\iota^{(k)} = 1. \tag{4.60}$$

The stochastic property set in (4.56) can be expressed as

$$C_k(\rho) := \mathrm{lev}_{\le 0} g_k = \left\{\boldsymbol{x} \in \mathbb{R}^N : g_k(\boldsymbol{x}) \le 0\right\}, \tag{4.61}$$

where

$$g_k : \mathbb{R}^N \to \mathbb{R},\ \boldsymbol{x} \mapsto \|\boldsymbol{e}_k(\boldsymbol{x})\|_r^2 - \rho = \left\|\boldsymbol{U}_k^{\mathsf{T}}\boldsymbol{x} - \boldsymbol{d}_k\right\|_r^2 - \rho. \tag{4.62}$$

Since g_k is differentiable with its gradient given by

$$\boldsymbol{\nabla} g_k(\boldsymbol{x}) = 2\boldsymbol{U}_k(\boldsymbol{U}_k^{\mathsf{T}}\boldsymbol{x} - \boldsymbol{d}_k), \tag{4.63}$$

the subdifferential of g_k is a singleton: $\partial g_k(\boldsymbol{x}) = \{\boldsymbol{\nabla} g_k(\boldsymbol{x})\}$, $\forall \boldsymbol{x} \in \mathbb{R}^N$. The subgradient projection relative to g_ι is given as follows:

$$T_{\mathrm{sp}(g_\iota)} : \mathcal{H} \to \mathcal{H},\ \boldsymbol{x} \mapsto \begin{cases} \boldsymbol{x} - \dfrac{g_\iota(\boldsymbol{x})}{\|\boldsymbol{\nabla} g_\iota(\boldsymbol{x})\|^2}\boldsymbol{\nabla} g_\iota(\boldsymbol{x}) & \text{if } g_\iota(\boldsymbol{x}) > 0, \\ \boldsymbol{x} & \text{otherwise.} \end{cases} \tag{4.64}$$

In particular, if $\|\boldsymbol{e}_\iota(\boldsymbol{h}_k)\|_r^2 > \rho$, $\iota \in \mathcal{I}_k$, then $T_{\mathrm{sp}(g_\iota)}$ maps the current estimate $\boldsymbol{h}_k$ as follows:

$$T_{\mathrm{sp}(g_\iota)}(\boldsymbol{h}_k) = \boldsymbol{h}_k - \frac{\|\boldsymbol{e}_\iota(\boldsymbol{h}_k)\|_r^2 - \rho}{2\left\|\boldsymbol{U}_\iota(\boldsymbol{U}_\iota^{\mathsf{T}}\boldsymbol{h}_k - \boldsymbol{d}_\iota)\right\|^2}\boldsymbol{U}_\iota(\boldsymbol{U}_\iota^{\mathsf{T}}\boldsymbol{h}_k - \boldsymbol{d}_\iota), \tag{4.65}$$

which is the metric projection onto the closed halfspace

$$H_\iota^-(\boldsymbol{h}_k) := \left\{\boldsymbol{x} \in \mathbb{R}^N : \langle \boldsymbol{x} - \boldsymbol{h}_k, \boldsymbol{\nabla} g_\iota(\boldsymbol{h}_k)\rangle + g_\iota(\boldsymbol{h}_k) \le 0\right\} \supset C_\iota(\rho). \tag{4.66}$$

Theorem 4.67. *For an arbitrary $\boldsymbol{h}_0 \in \mathbb{R}^N$, the sequence $(\boldsymbol{h}_k)_{k\in\mathbb{N}}$ generated by*

$$\boldsymbol{h}_{k+1} := \boldsymbol{h}_k + \mu_k\left(\sum_{\iota\in\mathcal{I}_k} w_\iota^{(k)} T_{\mathrm{sp}(g_\iota)}(\boldsymbol{h}_k) - \boldsymbol{h}_k\right), \tag{4.68}$$

where $\mu_k \in (0, 2\mathcal{M}_k)$ with the extrapolation coefficient

$$\mathcal{M}_k := \begin{cases} \dfrac{\sum_{\iota\in\mathcal{I}_k} w_\iota^{(k)} \left\|T_{\mathrm{sp}(g_\iota)}(\boldsymbol{h}_k) - \boldsymbol{h}_k\right\|^2}{\left\|\sum_{\iota\in\mathcal{I}_k} w_\iota^{(k)} T_{\mathrm{sp}(g_\iota)}(\boldsymbol{h}_k) - \boldsymbol{h}_k\right\|^2} & \text{if } \boldsymbol{h}_k \notin \bigcap_{\iota\in\mathcal{I}_k} C_\iota(\rho) \\ 1 & \text{otherwise} \end{cases} \tag{4.69}$$

satisfies the following.

(a) *(Monotone approximation) Define* $\mathcal{J}_k := \{\iota \in \mathcal{I}_k : \boldsymbol{h}_k \notin C_\iota(\rho)\}$. *Suppose that* $C_0^{(k)} := \bigcap_{\iota\in\mathcal{J}_k} H_\iota^-(\boldsymbol{h}_k) \neq \emptyset$ *and* $\boldsymbol{h}_k \notin C_0^{(k)}$. *Then,*

$$\left\|\boldsymbol{h}_{k+1} - \hat{\boldsymbol{h}}_k^*\right\| < \left\|\boldsymbol{h}_k - \hat{\boldsymbol{h}}_k^*\right\|, \ \forall \hat{\boldsymbol{h}}_k^* \in C_0^{(k)}. \tag{4.70}$$

(b) *(Fejér monotonicity) Suppose that there exists* $\kappa_0 \in \mathbb{N}$ *such that* $C_0 := \bigcap_{k\geq\kappa_0} C_0^{(k)} \neq \emptyset$. *Then,*

$$\left\|\boldsymbol{h}_{k+1} - \hat{\boldsymbol{h}}^*\right\| \leq \left\|\boldsymbol{h}_k - \hat{\boldsymbol{h}}^*\right\|, \ \forall \hat{\boldsymbol{h}}^* \in C_0. \tag{4.71}$$

(c) *(Convergence) In addition to the condition in* Theorem 4.67.b, *assume that*
   (i) *there exist* $\epsilon_1, \epsilon_2 \in (0,2)$ *such that* $\mu_k \in [\epsilon_1\mathcal{M}_k, (2-\epsilon_2)\mathcal{M}_k]$, $\forall k \geq \kappa_0$, *and*
   (ii) C_0 *has an interior point.*
   Let $(w_\iota^{(k)})_{\iota\in\mathcal{I}_k}$, $k \in \mathbb{N}$, *be the weights satisfying* $\inf_{k\geq\kappa_0} \min_{\iota\in\mathcal{J}_k} w_\iota^{(k)} > 0$. *Then,* $(\boldsymbol{h}_k)_{k\in\mathbb{N}}$ *converges to a point* $\hat{\boldsymbol{h}} \in \liminf_{k\to\infty} C_0^{(k)}$, *where* $\liminf_{k\to\infty} C_0^{(k)} := \bigcup_{k=0}^{\infty}\bigcap_{n\geq k} C_0^{(n)}$.

The method in (4.68) is called the *adaptive parallel subgradient projection (APSP) algorithm*. The advantages of the APSP algorithm include the following.

(a) It converges faster than NLMS or APA even for colored inputs thanks to the simultaneous use of multiple pieces of information by means of parallel projection with the extrapolation coefficient (which enlarges the step size).
(b) It enjoys stable convergence/tracking even in noisy environments thanks to the use of reasonably designed stochastic property sets. The stability is implied by Theorem 4.67.a and Theorem 4.67.b.
(c) It enjoys low computational complexity and suits for parallel computing. If q concurrent processors are engaged, the computational complexity imposed on each processor at each iteration is $O(N)$. We emphasize that an algorithm with $O(N)$ complexity is strongly desired for real-time implementation of adaptive filters with large filter length.

Remark. Theorem 4.67 suggests that the sequence $(\boldsymbol{h}_k)_{k\in\mathbb{N}}$ converges under certain conditions no matter how we choose the weights $w_\iota^{(k)}$. The simplest example is the uniform weights: $w_\iota^{(k)} := 1/q$, $\forall \iota \in \mathcal{I}_k$, $k \in \mathbb{N}$. Note however that the weights govern the direction of update and thus affect the *rate of convergence*. The optimal weight designing

problem is difficult in general to solve with low computational costs. In [**36**], a practical weight designing technique with $O(N)$ complexity has been proposed. The technique inductively utilizes a simple closed-form formula to compute the projection onto the intersection of two closed halfspaces that are defined by a triplet of vectors. The resulting weights have been shown to be *optimal in a pairwise manner in the sense of certain worst-case (min-max) optimization*. Another technique that realizes exponentially decaying weights has been proposed for hyperplanes in [**48**] and for linear varieties in [**49**]. See [**48, 50–57**] and Lecture 6 for further developments and applications of the set-theoretic adaptive filtering algorithms.

LECTURE 5
Fixed Point Theory of Nonexpansive Mapping

5.1. Outline of Lecture 5

5.2. Introduction

In practical scenarios where signal processing is required, enough information is hardly available to identify the *ideal* solution without any ambiguity. The reasons for that include (i) the presence of ambient noise, distortion, etc., occurring in measurement process and (ii) possible loss of acquired information. In addition, the limitation of time and computational resources spent for signal processing makes it further unrealistic to accomplish the perfect identification of the ideal solution.

A realistic approach is thus to define *a set of solution candidates* from each piece of available information, and find a common point of the sets of candidates. This approach is undoubtedly *the set theoretic estimation* or *set theoretic adaptive filtering* that we learned in Lecture

4. What is desired for iterative algorithms? Let $S \subset \mathcal{H}$ be the set of all common points (i.e., the intersection of the sets of candidates) and $T : \mathcal{H} \to \mathcal{H}$ the mapping that shifts the current estimate $\boldsymbol{x}_k$ to its following estimate $\boldsymbol{x}_{k+1}$, $k \in \mathbb{N}$; i.e., $\boldsymbol{x}_{k+1} = T(\boldsymbol{x}_k)$. The desired properties should be as follows.

(a) If $\boldsymbol{x}_k$ is outside of S, then T should push it closer to S. Mathematically speaking: if $\boldsymbol{x}_k \notin S$, then it is desired to be satisfied that $\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\| < \|\boldsymbol{x}_k - \boldsymbol{x}^*\|$, $\forall \boldsymbol{x}^* \in S$. Note that the inequality is *strict* unlike the case of Fejér monotonicity.
(b) If $\boldsymbol{x}_k$ is inside of S, then T should keep it staying there. Mathematically speaking: if $\boldsymbol{x}_k \in S$, then it is desired to be satisfied that $T(\boldsymbol{x}_k) = \boldsymbol{x}_k$; in this case $\boldsymbol{x}_k$ is called a *fixed point* of T.

In fact, *metric projection* and *subgradient projection* realize the properties (a) and (b) if S is closed convex. So, why should we learn more than that? The reasons are the following. There are many problems that cannot be solved solely with metric projection and/or subgradient projection. There exist other mappings, satisfying the properties (a) and (b), which can be used to solve such challenging problems. The topic of this lecture, *the fixed point theory*, considers the family of mappings that satisfy certain properties such as the ones (a) and (b) mentioned above, and it greatly helps our understanding of the convergence mechanism of iterative algorithms. In a nutshell, *the fixed point theory is sufficiently simple and very powerful.* This is important more than anything for us engineers.

If T satisfies the properties (a) and (b) above, S is obviously the set of all fixed points, which is called the fixed point set of T. Importantly, in this case, the fixed point theory ensures that S is closed convex. In other words, *unless S is closed convex, we can never realize the desired properties* (a) *and* (b). The point is *how to construct T whose fixed point set is identical to S*. The applicability of metric projection is governed by the shape of S (see Section 4.5 for the examples of "simple" closed convex sets). In the case that S is not simple, subgradient projection is a reasonable alternative, as we have already seen in Lecture 4. To enhance the accuracy of estimation, we may incorporate more and more information about the estimandum, thereby focusing the intersection in which we believe our target stays.

We start with two simple algorithms for convex optimization, the projected gradient method and the projected subgradient method, followed by an adaptive extension of the projected subgradient method. Then we proceed to the fixed point theory of *nonexpansive mapping*.

We conclude this lecture by providing links between the fixed point theory and some of the iterative methods that we have presented.

5.3. Projected Gradient Method and Projected Subgradient Method

Consider the following convexly constrained optimization:

$$\min_{\boldsymbol{x}\in K}\varphi(\boldsymbol{x}) \tag{5.1}$$

where $K\subset\mathcal{H}$ is a closed convex set and $\varphi:\mathcal{H}\to\mathbb{R}$ a continuous convex function. For differentiable convex functions, A. A. Goldstein has invented the *projected gradient method* in 1964 [**58, 59**]:

$$\boldsymbol{x}_{k+1}:=P_K\left[\boldsymbol{x}_k-\lambda\boldsymbol{\nabla}\varphi(\boldsymbol{x}_k)\right],\ k\in\mathbb{N},\ \text{for some } \boldsymbol{x}_0\in\mathcal{H}, \tag{5.2}$$

where $\lambda>0$ is the step size and $\boldsymbol{\nabla}\varphi(\boldsymbol{x}_k)$ is the gradient of φ at $\boldsymbol{x}_k$. If $K:=\mathcal{H}$ (i.e., there is no constraint), then the projected gradient method is reduced to the standard gradient method (the steepest descent method), thus the projected gradient method is a generalization of the gradient method.

In 1969, B. T. Polyak has shown that one can employ a *subgradient*, rather than the gradient, for nondifferentiable convex functions under certain conditions [**60**]. Specifically, he has invented the *Projected Subgradient Method (PSM)*:

$$\boldsymbol{x}_{k+1}:=\begin{cases}P_K\left(\boldsymbol{x}_k-\lambda_k\dfrac{\varphi(\boldsymbol{x}_k)}{\|\varphi'(\boldsymbol{x}_k)\|^2}\varphi'(\boldsymbol{x}_k)\right) & \text{if } \varphi'(\boldsymbol{x}_k)\neq\boldsymbol{0}\\ \boldsymbol{x}_k & \text{otherwise}\end{cases} \tag{5.3}$$

$k\in\mathbb{N}$, where $\boldsymbol{x}_0\in K$, $\lambda_k\in(0,2)$, and $\varphi'(\boldsymbol{x}_k)\in\partial\varphi(\boldsymbol{x}_k)$. The convergence of the projected gradient method and PSM will be discussed in Section 5.10.

5.4. Adaptive Projected Subgradient Method

We repeat the NLMS update equation:

$$\boldsymbol{h}_{k+1}:=\boldsymbol{h}_k-\lambda\frac{\langle\boldsymbol{u}_k,\boldsymbol{h}_k\rangle-d_k}{\|\boldsymbol{u}_k\|^2}\boldsymbol{u}_k,\ k\in\mathbb{N}. \tag{5.4}$$

Comparing (5.3) and (5.4), we find NLMS having a similar structure to PSM. Indeed, Fig. 3-4 suggests that NLMS attempts to minimize the metric distance to the hyperplane H_k. Unlike the case of PSM, however, *the cost function d_{H_k} seems to change from iteration to iteration.* How can we do with this time-varying cost function?

In [**61, 62**], I. Yamada has formulated the problem as follows. Let $\varphi_k:\mathcal{H}\to[0,\infty)$, $k\in\mathbb{N}$, be a continuous convex function and $K\subset\mathcal{H}$ a nonempty closed convex set. Then, the problem is formulated as minimizing *the sequence of cost functions* $(\varphi_k)_{k\in\mathbb{N}}$ over K asymptotically. We present the *Adaptive Projected Subgradient Method (APSM)* for the asymptotic minimization problem and its convergence analysis below.

Theorem 5.5. *Given an arbitrary initial vector $\boldsymbol{h}_0\in K$, APSM generates the sequence $(\boldsymbol{h}_k)_{k\in\mathbb{N}}$ as follows:*

$$\boldsymbol{h}_{k+1}:=\begin{cases}P_K\left(\boldsymbol{h}_k-\lambda_k\dfrac{\varphi_k(\boldsymbol{h}_k)}{\|\varphi_k'(\boldsymbol{h}_k)\|^2}\varphi_k'(\boldsymbol{h}_k)\right) & \text{if } \varphi_k'(\boldsymbol{h}_k)\neq\boldsymbol{0}\\ \boldsymbol{h}_k & \text{otherwise}\end{cases} \tag{5.6}$$

where $\varphi_k'(\boldsymbol{h}_k)\in\partial\varphi_k(\boldsymbol{h}_k)$ and $\lambda_k\in[0,2]$. Assume the existence of minimizer of φ_k over K; i.e., there exists $\boldsymbol{x}^*\in K$ such that $\varphi_k(\boldsymbol{x}^*)=\varphi_k^*:=\inf_{\boldsymbol{y}\in K}\varphi_k(\boldsymbol{y})$. Then, the following statements hold.

(a) *(Monotone approximation) Suppose that $\varphi_k(\boldsymbol{h}_k)>\varphi_k^*$, or equivalently*

$$\boldsymbol{h}_k\notin\Omega_k:=\operatorname*{argmin}_{\boldsymbol{x}\in K}\varphi_k(\boldsymbol{x})\neq\emptyset. \tag{5.7}$$

Then, for any $\lambda_k\in\left(0,2\left(1-\dfrac{\varphi_k^*}{\varphi_k(\boldsymbol{h}_k)}\right)\right)$

$$\left\|\boldsymbol{h}_{k+1}-\hat{\boldsymbol{h}}_k^*\right\|\leq\left\|\boldsymbol{h}_k-\hat{\boldsymbol{h}}_k^*\right\|,\ \forall\hat{\boldsymbol{h}}_k^*\in\Omega_k. \tag{5.8}$$

Note that Ω_k is the set of minimizers of φ_k over K. If in particular the minimum is zero (i.e., $\varphi_k^*=0$), then (5.8) holds for any $\lambda_k\in(0,2)$.

(b) *(Boundedness, asymptotic optimality) Assume the existence of $\kappa_0\in\mathbb{N}$ such that there exists $\boldsymbol{x}^*\in K$ satisfying $\varphi_k(\boldsymbol{x}^*)=0$, $\forall k\geq\kappa_0$, or in other words*

$$\varphi_k^*=0,\ \forall k\geq\kappa_0,\ \text{and } \Omega:=\bigcap_{k\geq\kappa_0}\Omega_k\neq\emptyset. \tag{5.9}$$

Then, the sequence $(\boldsymbol{h}_k)_{k\in\mathbb{N}}$ is bounded. Assume in addition that the sequence of subgradients $(\varphi_k'(\boldsymbol{h}_k))_{k\in\mathbb{N}}$ is bounded. Then, for any $\lambda_k\in[\epsilon_1,2-\epsilon_2]\subset(0,2)$, $k\in\mathbb{N}$,

$$\lim_{k\to\infty}\varphi_k(\boldsymbol{h}_k)=0. \tag{5.10}$$

(c) *(Strong convergence, asymptotic optimality of the limit point) Assume the existence of $\kappa_0\in\mathbb{N}$ such that (5.9) holds and the set Ω has a relative interior with respect to a hyperplane $\Pi\subset\mathcal{H}$;*

i.e., there exist $\widetilde{\boldsymbol{h}} \in \Pi \cap \Omega$ *and* $\varepsilon_{\mathrm{r.i.}} > 0$ *such that* $B_\Pi(\widetilde{\boldsymbol{h}}, \varepsilon_{\mathrm{r.i.}}) := \{\boldsymbol{x} \in \Pi : \left\|\boldsymbol{x} - \widetilde{\boldsymbol{h}}\right\| < \varepsilon_{\mathrm{r.i.}}\} = B(\widetilde{\boldsymbol{h}}, \varepsilon_{\mathrm{r.i.}}) \cap \Pi \subset \Omega$.[1] *Then, for any* $\lambda_k \in [\epsilon_1, 2 - \epsilon_2] \subset (0, 2)$, $k \in \mathbb{N}$, *the sequence* $(\boldsymbol{h}_k)_{k \in \mathbb{N}}$ *converges strongly to a point* $\hat{\boldsymbol{h}} \in K$*; i.e.,*

$$\lim_{k \to \infty} \left\|\boldsymbol{h}_k - \hat{\boldsymbol{h}}\right\| = 0. \tag{5.11}$$

Assume in addition (i) the boundedness of the sequence of subgradients $(\varphi_k'(\boldsymbol{h}_k))_{k \in \mathbb{N}}$ *and (ii) the existence of a bounded sequence of subgradients* $(\varphi_k'(\hat{\boldsymbol{h}}))_{k \in \mathbb{N}}$*, where* $\varphi_k'(\hat{\boldsymbol{h}}) \in \partial\varphi_k(\hat{\boldsymbol{h}})$*. Then,*

$$\lim_{k \to \infty} \varphi_k(\hat{\boldsymbol{h}}) = 0. \tag{5.12}$$

(d) *(Characterization of the limit point* $\hat{\boldsymbol{h}}$*) Assume that all the conditions in* Theorem 5.5.c *are satisfied. Assume in addition that (i) the set* Ω *has an interior point* $\widetilde{\boldsymbol{h}}$ *(which is a slightly stronger condition than the existence of a relative interior) and (ii) for any* $\epsilon > 0$ *and any* $r > 0$*, there exists* $\delta > 0$ *such that*

$$\inf_{\substack{d(\boldsymbol{h}_k, \mathrm{lev}_{\le 0}\varphi_k) \ge \varepsilon, \\ \left\|\widetilde{\boldsymbol{h}} - \boldsymbol{h}_k\right\| \le r, \\ k \ge \kappa_0}} \varphi_k(\boldsymbol{h}_k) \ge \delta. \tag{5.13}$$

Then, for any $\lambda_k \in [\epsilon_1, 2 - \epsilon_2] \subset (0, 2)$, $k \in \mathbb{N}$, *the limit point* $\hat{\boldsymbol{h}} := \lim_{k \to \infty} \boldsymbol{h}_k \in K$ *is characterized as*

$$\hat{\boldsymbol{h}} \in \overline{\lim_{k \to \infty} \inf\ \Omega_k}, \tag{5.14}$$

where $\liminf_{k \to \infty} \Omega_k := \bigcup_{k=0}^{\infty} \bigcap_{n \ge k} \Omega_n$.

5.5. Examples of APSM

To apply APSM to real-world problems, we only need to design the sequence of cost functions $(\varphi_k)_{k \in \mathbb{N}}$. We present some examples below.

Example 5.15.

(a) (NLMS/APA) Define φ_k, $k \in \mathbb{N}$, as the metric distance to the linear variety V_k defined in (3.72):

$$\varphi_k(\boldsymbol{x}) := d_{V_k}(\boldsymbol{x}) := \min_{\boldsymbol{y} \in V_k} \|\boldsymbol{x} - \boldsymbol{y}\|, \ k \in \mathbb{N}. \tag{5.16}$$

[1] The norm $\|\cdot\|$ can be arbitrary due to the norm equivalence for finite-dimensional vector spaces.

Then, we have

$$\partial\varphi_k(\boldsymbol{x}) \ni \varphi_k'(\boldsymbol{x}) = \begin{cases} \dfrac{\boldsymbol{x} - P_{V_k}(\boldsymbol{x})}{d_{V_k}(\boldsymbol{x})} & \text{if } \boldsymbol{x} \notin V_k \\ \boldsymbol{0} & \text{otherwise.} \end{cases} \tag{5.17}$$

Substituting $\boldsymbol{x} := \boldsymbol{h}_k$ in (5.16) and (5.17), plugging the resultant $\varphi_k(\boldsymbol{h}_k)$ and $\varphi_k'(\boldsymbol{h}_k)$ into (5.6), and letting $\lambda_k := \lambda$ and $K := \mathcal{H}$ (i.e., $P_K = I$), we reproduce APA given in (3.69). In particular, if $r = 1$, V_k is reduced to the hyperplane H_k in (3.66), thus NLMS is reproduced. We remark that for the φ_k in (5.16) the assumption of the existence of (relative) interior is hardly satisfied, hence the convergence is not guaranteed in general. A simple demonstration that shows NLMS not converging is provided in [**62**].

(b) Let $(C_\iota^{(k)})_{\iota \in \mathcal{I}_k}$, $\mathcal{I}_k \subset \mathbb{N}$, be a finite number of nonempty closed convex sets to be processed at each iteration $k \in \mathbb{N}$. Define φ_k, $k \in \mathbb{N}$, as a weighted squared distance as follows:

$$\varphi_k(\boldsymbol{x}) := \sum_{\iota \in \mathcal{I}_k} w_\iota^{(k)} d^2_{C_\iota^{(k)}}(\boldsymbol{x}), \ k \in \mathbb{N}, \tag{5.18}$$

where $w_\iota^{(k)} > 0$ satisfies $\sum_{\iota \in \mathcal{I}_k} w_\iota^{(k)} = 1$, $k \in \mathbb{N}$. In this case, φ_k is differentiable over $\mathcal{H}$, thus $\partial\varphi_k(\boldsymbol{x}) = \{\boldsymbol{\nabla}\varphi_k(\boldsymbol{x})\}$, $\forall \boldsymbol{x} \in \mathcal{H}$, with the derivative

$$\boldsymbol{\nabla}\varphi_k(\boldsymbol{x}) = 2 \sum_{\iota \in \mathcal{I}_k} w_\iota^{(k)} \left(\boldsymbol{x} - P_{C_\iota^{(k)}}(\boldsymbol{x})\right). \tag{5.19}$$

Then, we can deduce the following algorithm:

$$\boldsymbol{h}_{k+1} := P_K \left[\boldsymbol{h}_k + \mu_k \left(\sum_{\iota \in \mathcal{I}_k} w_\iota^{(k)} P_{C_\iota^{(k)}}(\boldsymbol{h}_k) - \boldsymbol{h}_k\right)\right], \tag{5.20}$$

where $\mu_k \in [0, \mathcal{M}_k^{(1)}]$ with the extrapolation coefficient

$$\mathcal{M}_k^{(1)} := \begin{cases} \dfrac{\sum_{\iota \in \mathcal{I}_k} w_\iota^{(k)} \left\|P_{C_\iota^{(k)}}(\boldsymbol{h}_k) - \boldsymbol{h}_k\right\|^2}{\left\|\sum_{\iota \in \mathcal{I}_k} w_\iota^{(k)} P_{C_\iota^{(k)}}(\boldsymbol{h}_k) - \boldsymbol{h}_k\right\|^2} & \text{if } \boldsymbol{h}_k \notin \bigcap_{\iota \in \mathcal{I}_k} C_\iota^{(k)} \\ 1 & \text{otherwise.} \end{cases} \tag{5.21}$$

(c) Let $\mathcal{I}_k \subset \mathbb{N}$, $(C_\iota^{(k)})_{\iota\in\mathcal{I}_k}$, and $(w_\iota^{(k)})_{\iota\in\mathcal{I}_k}$ be given as in Example 5.15.b. Define φ_k, $k \in \mathbb{N}$, as a weighted distance as follows:

(5.22)
$$\varphi_k(\boldsymbol{x}) := \begin{cases} \displaystyle\sum_{\iota\in\mathcal{I}_k} \frac{w_\iota^{(k)} d_{C_\iota^{(k)}}(\boldsymbol{h}_k)}{\nu_k} d_{C_\iota^{(k)}}(\boldsymbol{x}) & \text{if } \nu_k := \textstyle\sum_{\iota\in\mathcal{I}_k} w_\iota^{(k)} d_{C_\iota^{(k)}}(\boldsymbol{h}_k) \neq 0 \\ 0 & \text{otherwise.} \end{cases}$$

In this case, φ_k is nonsmooth. We can show that applying APSM to the φ_k in (5.22) yields the same algorithm as in (5.20) with the range of step size extended as $\mu_k \in [0, 2\mathcal{M}_k^{(1)}]$. The APSP algorithm in Section 4.11 is obtained by letting $K := \mathcal{H}$ and $C_\iota^{(k)} := H_\iota^-(\boldsymbol{h}_k)$. Theorem 4.67 is a direct consequence of Theorem 5.5 with the φ_k defined as in (5.22). It can be verified that the boundedness assumptions and the assumption (d)-(ii) in Theorem 5.5 are automatically satisfied in this specific case.

5.6. Fixed Point and Classification of Mappings

APSM (Theorem 5.5) is motivated by the *fixed point theory of nonexpansive mapping* ; most of the results in the following can be founded in [**62–64**] and the references therein. Studying the theory helps our understanding of APSM as well as various iterative methods such as POCS, the simultaneous projection methods, the projected gradient method, PSM, etc.

Definition 5.23. Given a mapping $T : \mathcal{H} \to \mathcal{H}$, a point $\boldsymbol{x}$ such that $T(\boldsymbol{x}) = \boldsymbol{x}$ is called a *fixed point of* T. The set of all fixed points is called the *fixed point set of* T and denoted by

(5.24)
$$\mathrm{Fix}\,(T) := \{\boldsymbol{x} \in \mathcal{H} : T(\boldsymbol{x}) = \boldsymbol{x}\}.$$

The following classification of mappings is employed.

Definition 5.25.

(a) A mapping $T : \mathcal{H} \to \mathcal{H}$ is said to be *Lipschitz continuous* over $\mathcal{H}$ if there exists $\nu > 0$ such that[2]

(5.26)
$$\|T(\boldsymbol{x}) - T(\boldsymbol{y})\| \leq \nu \|\boldsymbol{x} - \boldsymbol{y}\|\,,\ \forall \boldsymbol{x}, \boldsymbol{y} \in \mathcal{H}.$$

The minimum ν satisfying (5.26) is called the *Lipschitz constant* of T. A Lipschitz continuous mapping with its Lipschitz constant ν is referred to shortly as ν-*Lipschitzian*. In particular, T is said to be

[2]The definition of Lipschitz continuity can be given in a general complete metric space by replacing the norm of the difference between two points by their distance.

(i) *(strictly) contractive* if (5.26) holds for $\nu < 1$;
(ii) *nonexpansive* if (5.26) holds for $\nu = 1$.

Contractive mapping is widely referred to as *contraction mapping*, and it is a subclass of nonexpansive mapping.

(b) Suppose that a mapping $T : \mathcal{H} \to \mathcal{H}$ has a fixed point; i.e., $\mathrm{Fix}\,(T) \neq \emptyset$. Then, T is said to be *quasi-nonexpansive* (or *Fejér*) if

(5.27)
$$\|T(\boldsymbol{x}) - \boldsymbol{z}\| \leq \|\boldsymbol{x} - \boldsymbol{z}\|\,,\ \forall \boldsymbol{x} \in \mathcal{H}, \forall \boldsymbol{z} \in \mathrm{Fix}\,(T)\,.$$

The Lipschitz continuity is a sufficient condition for a function to be continuous; note for any Lipschitz continuous mapping T that $\|\boldsymbol{x} - \boldsymbol{y}\| \to 0$ implies $\|T(\boldsymbol{x}) - T(\boldsymbol{y})\| \to 0$. The following proposition supports the applicability of quasi-nonexpansive mapping.

Proposition 5.28 ([**62, 65**])**.** *Let $T : \mathcal{H} \to \mathcal{H}$ be a quasi-nonexpansive mapping. Then,* $\mathrm{Fix}\,(T)$ *has the following characterization:*

(5.29)
$$\mathrm{Fix}\,(T) = \bigcap_{\boldsymbol{y}\in\mathcal{H}} H^-(\boldsymbol{y})$$

with

(5.30)
$$H^-(\boldsymbol{y}) := \left\{\boldsymbol{x} \in \mathcal{H} : \left\langle \boldsymbol{y} - T(\boldsymbol{y}), \boldsymbol{x} - \frac{\boldsymbol{y} + T(\boldsymbol{y})}{2} \right\rangle \leq 0\right\}.$$

Proposition 5.28 implies that *the fixed point set of any quasi-nonexpansive mapping is closed convex*, because the intersection of arbitrary collection of closed convex sets is closed convex (see Propositions 2.22 and 4.3). More precise classification of nonexpansive and quasi-nonexpansive mappings is given below.

Definition 5.31.

(a) A mapping $T : \mathcal{H} \to \mathcal{H}$ is said to be *averaged* (or specifically α-*averaged*) if there exists $\alpha \in (0, 1)$ and a (quasi-)nonexpansive mapping $T_{\mathrm{N}} : \mathcal{H} \to \mathcal{H}$ such that

(5.32)
$$T = (1 - \alpha)I + \alpha T_{\mathrm{N}}.$$

It holds that $\mathrm{Fix}\,(T) = \mathrm{Fix}\,(T_{\mathrm{N}})$ since $T(\boldsymbol{x}) = \boldsymbol{x} \Leftrightarrow T_{\mathrm{N}}(\boldsymbol{x}) = \boldsymbol{x}$.

(b) A mapping $T : \mathcal{H} \to \mathcal{H}$ such that $\mathrm{Fix}\,(T) \neq \emptyset$ is said to be *attracting* if

(5.33)
$$\|T(\boldsymbol{x}) - \boldsymbol{z}\| < \|\boldsymbol{x} - \boldsymbol{z}\|\,,\ \forall \boldsymbol{x} \in \mathcal{H} \setminus \mathrm{Fix}\,(T)\,, \forall \boldsymbol{z} \in \mathrm{Fix}\,(T)\,.$$

In particular, an attracting mapping T is said to be *strongly attracting* (or specifically η-*attracting*) if there exists an $\eta > 0$

such that

$$\|\boldsymbol{x}-\boldsymbol{z}\|^2 - \|T(\boldsymbol{x})-\boldsymbol{z}\|^2 \geq \eta \|\boldsymbol{x}-T(\boldsymbol{x})\|^2, \quad \forall \boldsymbol{x}\in\mathcal{H}, \boldsymbol{z}\in \mathrm{Fix}\,(T). \tag{5.34}$$

Exercise 16. Show that T defined in (5.32) is automatically (quasi-)nonexpansive if T_{N} is (quasi-)nonexpansive.

The class of strongly attracting mappings (or averaged mappings) is of significant importance as seen later. The relation between strongly attracting mapping and averaged mapping is given below.

Proposition 5.35. *For $\alpha\in(0,1)$ and $T:\mathcal{H}\to\mathcal{H}$, the following two statements are equivalent* [**62**].

(a) *T is α-averaged with $\mathrm{Fix}\,(T)\neq\emptyset$.*
(b) *T is $(1-\alpha)/\alpha$-attracting.*

Definition 5.36. A mapping $T:\mathcal{H}\to\mathcal{H}$ is said to be *firmly (quasi-)nonexpansive* if it is 1/2-averaged (quasi-)nonexpansive.[3]

Proposition 5.37. *Given a mapping $T:\mathcal{H}\to\mathcal{H}$, the following three statements are equivalent.*

(a) *T is firmly (quasi-)nonexpansive.*
(b) *$2T-I$ is (quasi-)nonexpansive.*
(c) *$I-T$ is firmly (quasi-)nonexpansive.*

By Proposition 5.35, a mapping T is 1-attracting if and only if it is firmly (quasi-)nonexpansive with $\mathrm{Fix}\,(T)\neq\emptyset$. Some remarks are given below (see Fig. 5-1).

Remark.

(a) A nonexpansive mapping T is quasi-nonexpansive provided that $\mathrm{Fix}\,(T)\neq\emptyset$.
(b) An attracting mapping is quasi-nonexpansive *but not necessarily nonexpansive*.
(c) A metric projection mapping $P_C:\mathcal{H}\to C$ for a nonempty closed convex set $C\subset\mathcal{H}$ is firmly nonexpansive with $\mathrm{Fix}\,(T)=C$ (see Proposition 4.9).
(d) A subgradient projection mapping $T_{\mathrm{sp}(f)}$ relative to a continuous convex function $f:\mathcal{H}\to\mathbb{R}$ with $\mathrm{lev}_{\leq 0}f\neq\emptyset$ is firmly quasi-nonexpansive with $\mathrm{Fix}\left(T_{\mathrm{sp}(f)}\right)=\mathrm{lev}_{\leq 0}f$. However, it is *not nonexpansive*.

[3] An equivalent definition of firmly nonexpansive mapping is as follows: a mapping $T:\mathcal{H}\to\mathcal{H}$ is said to be firmly nonexpansive if $\|T(\boldsymbol{x})-T(\boldsymbol{y})\|^2 \leq \langle \boldsymbol{x}-\boldsymbol{y}, T(\boldsymbol{x})-T(\boldsymbol{y})\rangle$, $\forall \boldsymbol{x},\boldsymbol{y}\in\mathcal{H}$.

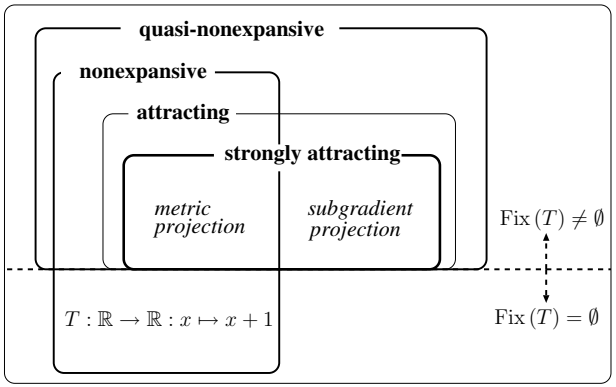


Fig. 5-1. A classification of nonlinear mappings. The dashed line classifies mappings according to whether having a fixed point.

Definition 5.38. A mapping $T:\mathcal{H}\to\mathcal{H}$ is said to be *η-inverse strongly monotone* (or *firmly monotone*) over $\mathcal{H}$ if there exists $\eta>0$ such that

$$\eta\|T(\boldsymbol{x})-T(\boldsymbol{y})\|^2 \leq \langle \boldsymbol{x}-\boldsymbol{y}, T(\boldsymbol{x})-T(\boldsymbol{y})\rangle, \ \forall \boldsymbol{x},\boldsymbol{y}\in\mathcal{H}. \tag{5.39}$$

Proposition 5.40.

(a) *Let $\phi:\mathcal{H}\to\mathbb{R}\cup\{\infty\}$ be a differentiable convex function with derivative $\boldsymbol{\nabla}\phi:\mathcal{H}\to\mathcal{H}$. Then, the following three statements are equivalent* [**66, 67**].
  (i) *$\boldsymbol{\nabla}\phi$ is ν-Lipschitzian over $\mathcal{H}$.*
  (ii) *$\boldsymbol{\nabla}\phi$ is $1/\nu$-inverse strongly monotone over $\mathcal{H}$.*
  (iii) *$I-\frac{2}{\nu}\boldsymbol{\nabla}\phi$ is nonexpansive over $\mathcal{H}$.*
(b) *Given $\alpha\in(0,1)$, $T:\mathcal{H}\to\mathcal{H}$ is α-averaged nonexpansive if and only if its complement $I-T$ is $\frac{1}{2\alpha}$-inverse strongly monotone* [**68**].

5.7. Fixed Point Theorems

The following theorem is one of the simplest results in the fixed point theory.[4]

[4] Theorem 5.41 holds for a general complete metric space.

Theorem 5.41 (Banach-Picard Fixed Point Theorem)**.** *Let $T : \mathcal{H} \to \mathcal{H}$ be contractive (i.e., ν-Lipschitzian for $\nu < 1$). Then*

(a) *T has a unique fixed point $\boldsymbol{x}^* \in \mathcal{H}$.*
(b) *For any $\boldsymbol{x} \in \mathcal{H}$, $\lim_{k\to\infty} T^k(\boldsymbol{x}) = \boldsymbol{x}^*$.*
(c) *For any $\boldsymbol{x} \in \mathcal{H}$, $\left\|T^k(\boldsymbol{x}) - \boldsymbol{x}^*\right\| \leq \dfrac{\nu^k}{1-\nu}\left\|\boldsymbol{x} - \boldsymbol{x}^*\right\|$, $k \in \mathbb{N}$.*

Remark. Theorem 5.41 guarantees (a) the existence of the unique fixed point of contraction mapping, and provides (b) an iterative method to compute the fixed point and (c) its rate of convergence. Note that the condition $\|T(\boldsymbol{x}) - T(\boldsymbol{y})\| < \|\boldsymbol{x} - \boldsymbol{y}\|$, $\forall \boldsymbol{x}, \boldsymbol{y} \in \mathcal{H}$, which is a necessary condition to be contractive, is *not sufficient* to guarantee the existence of a fixed point.

To present an extension of Theorem 5.41 to a more general class of mappings, we prove the following.

Proposition 5.42. *A contraction mapping $T : \mathcal{H} \to \mathcal{H}$ is averaged nonexpansive. To be specific, if T is ν-Lipschitzian for $\nu < 1$, then it is α-averaged nonexpansive for any $\alpha \in [\frac{1+\nu}{2}, 1)$.*

Proof: In general, T is α-averaged nonexpansive for $\alpha \in (0,1)$ if and only if $T_{\mathrm{N}} := \frac{1}{\alpha}T - \frac{1-\alpha}{\alpha}I$ is nonexpansive. By the Cauchy-Schwarz inequality and the definition of ν-Lipschitzian, we can verify the following inequality for any $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{H}$:

$$\|T_{\mathrm{N}}(\boldsymbol{x}) - T_{\mathrm{N}}(\boldsymbol{y})\| \leq \frac{(1-\alpha+\nu)^2}{\alpha^2}\|\boldsymbol{x} - \boldsymbol{y}\|. \tag{5.43}$$

Noting that $1 - \alpha + \nu > 0$ and $\alpha > 0$, we can verify that T_{N} is nonexpansive if and only if $\alpha \geq \frac{1+\nu}{2}$, hence T is α-averaged nonexpansive for any $\alpha \in [\frac{1+\nu}{2}, 1)$. □

We mention that a contraction mapping T is also averaged (or equivalently strongly attracting) quasi-nonexpansive as $\mathrm{Fix}\,(T) \neq \emptyset$. In contrast to contraction mapping, existence of a fixed point is *not* guaranteed in general for nonexpansive mapping (see Fig. 5-1). However, under the assumption of the existence, Theorem 5.41 can be extended as follows.

Theorem 5.44 ([**69**])**.** *Let $T : \mathcal{H} \to \mathcal{H}$ be nonexpansive with $\mathrm{Fix}\,(T) \neq \emptyset$. Also let $(\alpha_k)_{k\in\mathbb{N}} \subset [0,1]$ be a real-number sequence such that $\sum_{k=0}^{\infty} \alpha_k(1-\alpha_k) = \infty$. Then, for any initial point $\boldsymbol{x}_0 \in \mathcal{H}$, the sequence $(\boldsymbol{x}_k)_{k\in\mathbb{N}} \subset \mathcal{H}$ generated by*

$$\boldsymbol{x}_{k+1} := (1-\alpha_k)\boldsymbol{x}_k + \alpha_k T(\boldsymbol{x}_k),\ k \in \mathbb{N}, \tag{5.45}$$

converges weakly to a point in $\mathrm{Fix}\,(T)$.

Corollary 5.46. *Let $T : \mathcal{H} \to \mathcal{H}$ be averaged nonexpansive with $\mathrm{Fix}\,(T) \neq \emptyset$. Then, for any initial point $\boldsymbol{x}_0 \in \mathcal{H}$, the sequence $(\boldsymbol{x}_k)_{k\in\mathbb{N}} \subset \mathcal{H}$ generated by*

$$\boldsymbol{x}_{k+1} := T(\boldsymbol{x}_k),\ k \in \mathbb{N}, \tag{5.47}$$

converges weakly to a point in $\mathrm{Fix}\,(T)$.

Proof: Since T is averaged nonexpansive, there exists a nonexpansive mapping $T_{\mathrm{N}} : \mathcal{H} \to \mathcal{H}$ and $\alpha \in (0,1)$ such that $T = (1-\alpha)i + \alpha T_{\mathrm{N}}$. Since $\sum_{k=1}^{\infty}(1-\alpha)\alpha = \infty$, Theorem 5.44 implies the weak convergence of $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ to a point in $\mathrm{Fix}\,(T_{\mathrm{N}}) = \mathrm{Fix}\,(T)$, which completes the proof. □

The formula in (5.45) is commonly referred to as *Mann iteration* or *Krasnosel'skiĭ-Mann iteration* [**70, 71**]. There exist several types of theorems on the Mann iteration. Another version which is for quasi-nonexpansive mappings is based on the following definition.

Definition 5.48. A mapping $T : \mathcal{H} \to \mathcal{H}$ is said to be *demiclosed at $\boldsymbol{y} \in \mathcal{H}$* if

(a) weak convergence of $(\boldsymbol{x}_k)_{k\in\mathbb{N}} \subset \mathcal{H}$ to $\boldsymbol{x} \in \mathcal{H}$ and
(b) strong convergence of $(T(\boldsymbol{x}_k))_{k\in\mathbb{N}} \subset \mathcal{H}$ to $\boldsymbol{y}$

implies $T(\boldsymbol{x}) = \boldsymbol{y}$.

To show that a mapping is demiclosed, the following propositions are useful.

Proposition 5.49. *Let $T : \mathcal{H} \to \mathcal{H}$ be nonexpansive. Then $I - T$ is demiclosed at every point in $\mathcal{H}$.*

Proposition 5.50. *Given a continuous convex function $f : \mathcal{H} \to \mathbb{R}$, suppose that $\mathrm{lev}_{\leq 0} f \neq \emptyset$ and the set-valued mapping $\partial f : \mathcal{H} \to 2^{\mathcal{H}}$ is bounded in the sense that it maps bounded sets to bounded sets; $2^{\mathcal{H}}$ stands for the collection of all subsets of $\mathcal{H}$. Then $I - T_{\mathrm{sp}(f)}$ is demiclosed at $\mathbf{0} \in \mathcal{H}$.*

Theorem 5.51 ([**72**])**.** *Let $T : \mathcal{H} \to \mathcal{H}$ be quasi-nonexpansive with $I - T$ demiclosed at $\mathbf{0} \in \mathcal{H}$. Also let $(\alpha_k)_{k\in\mathbb{N}} \subset [\epsilon_1, 1-\epsilon_2]$ for some $\epsilon_1, \epsilon_2 > 0$. Then, for any initial point $\boldsymbol{x}_0 \in \mathcal{H}$, the sequence $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ generated by* (5.45) *converges weakly to a point in $\mathrm{Fix}\,(T)$.*

5.8. Algebraic Properties of Quasi-Nonexpansive Mapping

The following proposition is quite useful to incorporate multiple pieces of information in applications.

Proposition 5.52. *Let $T_1 : \mathcal{H} \to \mathcal{H}$ and $T_2 : \mathcal{H} \to \mathcal{H}$ be quasi-nonexpansive mappings such that $C := \mathrm{Fix}\,(T_1) \cap \mathrm{Fix}\,(T_2) \neq \emptyset$. Then a quasi-nonexpansive mapping T such that $\mathrm{Fix}\,(T) = C$ can be constructed as follows.*

(a) *The mapping*

$$T_{\mathrm{a}} := wT_1 + (1-w)T_2,\ w \in (0,1), \tag{5.53}$$

is quasi-nonexpansive with $\mathrm{Fix}\,(T_{\mathrm{a}}) = C$.

(b) *If T_2 is attracting, then*

$$T_{\mathrm{b}} := T_2T_1 \tag{5.54}$$

is quasi-nonexpansive with $\mathrm{Fix}\,(T_{\mathrm{b}}) = C$. In this case, T_{a} defined in (5.53) is attracting quasi-nonexpansive.

(c) *If T_1 is η_1-attracting and T_2 is η_2-attracting for some $\eta_1, \eta_2 > 0$, then*

(i) *T_{a} defined in (5.53) is $\left(\dfrac{(\eta_1+1)(\eta_2+1)}{(1-w)\eta_2 + w\eta_1 + 1} - 1\right)$-attracting;*

(ii) *T_{b} defined in (5.54) is $\left(\dfrac{\eta_1\eta_2}{\eta_1+\eta_2}\right)$-attracting.*

(d) *If T_1 is α_1-averaged and T_2 is α_2-averaged for some $\alpha_1, \alpha_2 \in (0,1)$, then*[5]

(i) *T_{a} defined in (5.53) is $((1-w)\alpha_1 + w\alpha_2)$-averaged;*

(ii) *T_{b} defined in (5.54) is $\dfrac{\alpha_1 + \alpha_2 - 2\alpha_1\alpha_2}{1-\alpha_1\alpha_2}$-averaged.*

(e) *If T_1 and T_2 are nonexpansive, T_{a} and T_{b} defined in (5.53) and (5.54), respectively, are nonexpansive with $\mathrm{Fix}\,(T_{\mathrm{a}}) = \mathrm{Fix}\,(T_{\mathrm{b}}) = C$.*[6]

Remark. Suppose for instance that we are given two (not necessarily attracting) quasi-nonexpansive mappings $T_1 : \mathcal{H} \to \mathcal{H}$ and $T_2 : \mathcal{H} \to \mathcal{H}$ such that $C := \mathrm{Fix}\,(T_1) \cap \mathrm{Fix}\,(T_2) \neq \emptyset$. Then, an averaged quasi-nonexpansive mapping T such that $\mathrm{Fix}\,(T) = C$ can be constructed as

$$T := (1-\alpha)I + \alpha T_{\mathrm{a}},\ \alpha \in (0,1), \tag{5.55}$$

with T_{a} defined as in (5.53). More specific examples will be given in the following section.

[5] Proposition 5.52.d holds for general averaged (quasi-)nonexpansive mappings *without any assumption about fixed points* [**73**].

[6] For any nonexpansive mappings T_1 and T_2 *without any assumption about fixed points*, T_{a} and T_{b} are nonexpansive.

5.9. Useful Mappings

The following *design-tool mappings* are useful to construct fixed point iterations for practical applications.

Example 5.56.

(a) (Metric projection) Given a nonempty closed convex set $C \subset \mathcal{H}$, $P_C : \mathcal{H} \to C$ is firmly nonexpansive with $\mathrm{Fix}\,(T) = C$; equivalently P_C is 1/2-averaged, and also 1-attracting (see Proposition 5.35). Therefore, $2P_C - I$ is nonexpansive, and $T := (1-\alpha)I + \alpha(2P_C - I) = I + 2\alpha(P_C - I)$, $\alpha \in (0,1)$, is α-averaged nonexpansive.

(b) (Subgradient projection) Given a continuous convex function $f : \mathcal{H} \to \mathbb{R}$ such that $\mathrm{lev}_{\leq 0} f \neq \emptyset$, $T_{\mathrm{sp}(f)}$ is firmly quasi-nonexpansive with $\mathrm{Fix}\left(T_{\mathrm{sp}(f)}\right) = \mathrm{lev}_{\leq 0} f$. Therefore, $2T_{\mathrm{sp}(f)} - I$ is quasi-nonexpansive, and $T := I + 2\alpha(T_{\mathrm{sp}(f)} - I)$, $\alpha \in (0,1)$, is α-averaged quasi-nonexpansive.

(c) (Steepest descent) Let $f : \mathcal{H} \to \mathbb{R}$ be a differentiable convex function with its derivative $\boldsymbol{\nabla} f : \mathcal{H} \to \mathcal{H}$ ν-Lipschitzian, $\nu > 0$, over $\mathcal{H}$. Then $\lambda \boldsymbol{\nabla} f$, $\lambda > 0$, is $\lambda\nu$-Lipschitzian, thus $\frac{1}{\lambda\nu}$-inverse strongly monotone by Proposition 5.40.a. Therefore, by Proposition 5.40.b, the complement $I - \lambda\boldsymbol{\nabla} f$ is $\frac{\lambda\nu}{2}$-averaged nonexpansive, provided that $\frac{\lambda\nu}{2} < 1$.[7] Namely, for any $\alpha \in (0,1)$, $I - \frac{2\alpha}{\nu}\boldsymbol{\nabla} f$ is α-averaged nonexpansive. Assuming that there exists $\boldsymbol{x} \in \mathcal{H}$ such that $f(\boldsymbol{x}) = \inf_{\boldsymbol{y} \in \mathcal{H}} f(\boldsymbol{y})$, $\mathrm{Fix}\left(I - \frac{2\alpha}{\nu}\boldsymbol{\nabla} f\right) = \mathrm{argmin}_{\boldsymbol{x} \in \mathcal{H}} f(\boldsymbol{x})$; see Proposition 4.41.

For instance, consider the convex feasibility problem with nonempty closed convex sets $(C_i)_{i \in \mathcal{I}} \subset \mathcal{H}$, $\mathcal{I} := \{1, 2, \cdots, n\}$, such that $C := \bigcap_{i \in \mathcal{I}} C_i \neq \emptyset$. Define the proximity function

$$p : \mathcal{H} \to [0, \infty),\ \boldsymbol{x} \mapsto \frac{1}{2}\sum_{i \in \mathcal{I}} w_i d^2_{C_i}(\boldsymbol{x}), \tag{5.57}$$

where $w_i > 0$ satisfies $\sum_{i \in \mathcal{I}} w_i = 1$. Then its derivative

$$\boldsymbol{\nabla} p : \mathcal{H} \to \mathcal{H},\ \boldsymbol{x} \mapsto \sum_{i \in \mathcal{I}} w_i(\boldsymbol{x} - P_{C_i}(\boldsymbol{x})) \tag{5.58}$$

is known to be 1-Lipschitzian. Therefore, for any $\alpha \in (0,1)$,

$$T := I - 2\alpha\boldsymbol{\nabla} p \tag{5.59}$$

is α-averaged nonexpansive with $\mathrm{Fix}\,(T) = C$.

[7] This can also be verified by Proposition 5.40.a with the following observation: $I - \lambda\boldsymbol{\nabla} f = (1 - \frac{\lambda\nu}{2})I + \frac{\lambda\nu}{2}\left(I - \frac{2}{\nu}\boldsymbol{\nabla} f\right)$.

5.10. Iterative Methods from Fixed Point Theoretic Perspective

Example 5.60 (Methods Based on Mann Iteration)**.**

(a) (Projected gradient method) Let $K \subset$ be a nonempty closed convex set and $\varphi : \mathcal{H} \to \mathbb{R}$ a differentiable convex function with its derivative $\boldsymbol{\nabla}\varphi : \mathcal{H} \to \mathcal{H}$ ν-Lipschitzian, $\nu > 0$, over $\mathcal{H}$. Then, by the discussion in Example 5.56.c, it can readily verified $I - \lambda\boldsymbol{\nabla}\varphi$, $\lambda \in (0, \frac{2}{\nu})$, is $\frac{\lambda\nu}{2}$-averaged nonexpansive. Since P_K is 1/2-averaged nonexpansive, the composition $P_K(I - \lambda\boldsymbol{\nabla}\varphi)$ is $\frac{2}{4-\lambda\nu}$-averaged nonexpansive (see Proposition 5.52.d). Assume that the problem in (5.1) has a solution; i.e., $\operatorname{argmin}_{\boldsymbol{x}\in K} \varphi \neq \emptyset$. In this case, for any $\lambda \in (0, \frac{2}{\nu})$ [**74**]

$$\operatorname{Fix}\left(P_K(I - \lambda\boldsymbol{\nabla}\varphi)\right) = \operatorname*{argmin}_{\boldsymbol{x}\in K} \varphi. \tag{5.61}$$

Therefore Corollary 5.46 implies that, for any $\boldsymbol{x}_0 \in \mathcal{H}$ and any $\lambda \in (0, \frac{2}{\nu})$, the sequence $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ generated by (5.2) converges weakly to a solution $\boldsymbol{x}^*$ that minimizes φ over K.

(b) (POCS) Let $(C_i)_{i\in\mathcal{I}} \subset \mathcal{H}$, $\mathcal{I} := \{1, 2, \cdots, n\}$, be nonempty closed convex sets such that $C := \bigcap_{i\in\mathcal{I}} C_i \neq \emptyset$. Then, for each $i \in \mathcal{I}$, T_i defined in (4.24) can be expressed as

$$T_i := I + \lambda_i(P_{C_i} - I) = \left(1 - \frac{\lambda_i}{2}\right) I + \frac{\lambda_i}{2}\left(2P_{C_i} - I\right). \tag{5.62}$$

Here the firm nonexpansivity of P_{C_i} suggests nonexpansivity of $2P_{C_i} - I$ (see Proposition 5.37), implying that T_i is $\frac{\lambda_i}{2}$-averaged nonexpansive for any $\lambda_i \in (0, 2)$. It is readily verified that $\operatorname{Fix}(T_i) = \operatorname{Fix}(2P_{C_i} - I) = \operatorname{Fix}(P_{C_i}) = C_i$. Therefore the mapping $T := T_nT_{n-1}\cdots T_1$ is averaged nonexpansive with $\operatorname{Fix}(T) = C$ (see Proposition 5.52.d). Corollary 5.46 thus reproduces the result in Theorem 4.25.

(c) (Parallel projection method [**37, 75**]) Let $K \subset \mathcal{H}$ and $(C_i)_{i\in\mathcal{I}} \subset \mathcal{H}$, $\mathcal{I} := \{1, 2, \cdots, n\}$, be nonempty closed convex sets. We consider the following *hardly-constrained inconsistent convex feasibility problem*: find a point in K that least violates the "feasibility" in the sense of being closest to all C_is. To be specific, the problem is to find a point that minimizes the proximity function p defined in (5.57) over K. Assume that the problem has a solution; i.e., $\operatorname{argmin}_{\boldsymbol{x}\in K} p(\boldsymbol{x}) \neq \emptyset$. Recall the discussion in Example 5.60.a. Noting that the gradient $\boldsymbol{\nabla}p$ is 1-Lipschitzian, it can easily be seen that, for any $\boldsymbol{x}_0 \in \mathcal{H}$ and

any $\lambda \in (0, 2)$, the sequence $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ generated by

$$\boldsymbol{x}_{k+1} := P_K\left(\boldsymbol{x}_k + \lambda\left(\sum_{i\in\mathcal{I}} w_i P_{C_i}(\boldsymbol{x}_k) - \boldsymbol{x}_k\right)\right) \tag{5.63}$$

converges weakly to a solution $\boldsymbol{x}^* \in \operatorname{Fix}\left(P_K(I - \lambda\boldsymbol{\nabla}p)\right) = \operatorname{argmin}_{\boldsymbol{x}\in K} p(\boldsymbol{x})$. In particular, letting $K := \mathcal{H}$ and assuming $C := \bigcap_{i\in\mathcal{I}} C_i \neq \emptyset$, it follows that $\operatorname{argmin}_{\boldsymbol{x}\in K} p(\boldsymbol{x}) = C$, thus reproducing the result in Theorem 4.30. Theorem 4.32, which is the case that the step size λ is replaced by $\lambda_k \in [\epsilon, 2 - \epsilon] \subset (0, 2)$, can be addressed as follows. Let $K = \mathcal{H}$, and observe

$$I - \lambda_k\boldsymbol{\nabla}p = \left(1 - \frac{\lambda_k}{2}\right) I + \frac{\lambda_k}{2}\left(I - 2\boldsymbol{\nabla}p\right). \tag{5.64}$$

Because $I - 2\boldsymbol{\nabla}p$ is nonexpansive (see Proposition 5.40.a) and

$$\sum_{k=1}^{\infty}\left(1 - \frac{\lambda_k}{2}\right)\frac{\lambda_k}{2} \geq \sum_{k=1}^{\infty}\left(\frac{\epsilon}{2}\right)^2 = \infty, \tag{5.65}$$

Theorem 5.44 can be applied to reproduce Theorem 4.32.

Now we present the convergence theorem of PSM proved by B. T. Polyak [**60**] below.

Theorem 5.66. *Let $\varphi : \mathcal{H} \to [0, \infty)$ be a continuous convex function and $K \subset \mathcal{H}$ a nonempty closed convex set. Assume that*

(a) $\Omega := \{\boldsymbol{x} \in K : \varphi(\boldsymbol{x}) = 0\} \neq \emptyset$*;*

(b) *for an arbitrarily fixed $\boldsymbol{x}_0 \in K$, there exist a constant $c > 0$ and $\boldsymbol{x}^* \in \Omega$ such that $\|\varphi'(\boldsymbol{x})\| \leq c$ for any $\boldsymbol{x} \in K$ satisfying $\|\boldsymbol{x} - \boldsymbol{x}^*\| \leq \|\boldsymbol{x}_0 - \boldsymbol{x}^*\|$.*

Then, for any $\lambda_k \in [\epsilon_1, 2 - \epsilon_2] \subset (0, 2)$, $\forall k \in \mathbb{N}$, the sequence $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ generated by (5.3) *converges weakly to a point $\boldsymbol{x}^* \in \Omega$ (i.e., $\varphi(\boldsymbol{x}^*) = 0$) and it satisfies $\lim_{k\to\infty} \varphi(\boldsymbol{x}_k) = 0$.*

Remark (Fixed point characterization of PSM)**.** As $\operatorname{lev}_{\leq 0}\varphi \neq \emptyset$, (5.3) can be rewritten as follows:

$$\boldsymbol{x}_{k+1} = P_K\left[\boldsymbol{x}_k + \lambda_k\left(T_{\operatorname{sp}(\varphi)}(\boldsymbol{x}_k) - \boldsymbol{x}_k\right)\right] \tag{5.67}$$

$$= P_K\left[\left(1 - \frac{\lambda_k}{2}\right) I + \frac{\lambda_k}{2}\left(2T_{\operatorname{sp}(\varphi)} - I\right)\right](\boldsymbol{x}_k). \tag{5.68}$$

Because $T_{\operatorname{sp}(\varphi)}$ is firmly quasi-nonexpansive with $\operatorname{Fix}\left(T_{\operatorname{sp}(\varphi)}\right) = \operatorname{lev}_{\leq 0}\varphi$, $2T_{\operatorname{sp}(\varphi)} - I$ is quasi-nonexpansive, thus

$$\hat{T}_k := \left(1 - \frac{\lambda_k}{2}\right) I + \frac{\lambda_k}{2}\left(2T_{\operatorname{sp}(\varphi)} - I\right),\ k \in \mathbb{N}, \tag{5.69}$$

is $\frac{\lambda_k}{2}$-averaged (i.e., $\frac{2-\lambda_k}{\lambda_k}$-attracting) quasi-nonexpansive with $\operatorname{Fix}\left(\hat{T}_k\right) = \operatorname{lev}_{\leq 0}\varphi$. Since P_K is 1-attracting nonexpansive with $\operatorname{Fix}(P_K) = K$, $P_K\hat{T}_k$ is a composition of strongly attracting mappings (see Proposition 5.52.c). Therefore, the assumption (a) in Theorem 5.66 can be interpreted as assuming $\operatorname{Fix}(P_K) \cap \operatorname{Fix}\left(\hat{T}_k\right) = K \cap \operatorname{lev}_{\leq 0}\varphi \neq \emptyset$, which actually coincides with Ω. Hence, $P_K\hat{T}_k$ is $\frac{2-\lambda_k}{2}$-attracting quasi-nonexpansive with $\operatorname{Fix}\left(P_K\hat{T}_k\right) = K \cap \operatorname{lev}_{\leq 0}\varphi = \Omega$. This suggests that the sequence $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ generated by PSM has the following monotone approximation property:

$$\|\boldsymbol{x}_k - \boldsymbol{x}^*\|^2 - \|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\|^2 \geq \frac{2-\lambda_k}{2}\|\boldsymbol{x}_k - \boldsymbol{x}_{k+1}\|^2,\ \forall \boldsymbol{x}^* \in \Omega. \tag{5.70}$$

Remark (Fixed point characterization of APSM)**.** Assume $\operatorname{lev}_{\leq 0}\varphi_k \neq \emptyset$, $k \in \mathbb{N}$. Then, the APSM recursion in (5.6) can be expressed as follows:

$$\boldsymbol{h}_{k+1} := P_K\widehat{T}_k(\boldsymbol{h}_k). \tag{5.71}$$

where

$$\widehat{T}_k := \left(1 - \frac{\lambda_k}{2}\right)I + \frac{\lambda_k}{2}\left(2T_{\operatorname{sp}(\varphi_k)} - I\right),\ k \in \mathbb{N}. \tag{5.72}$$

Since $\operatorname{Fix}\left(\widehat{T}_k\right) = \operatorname{Fix}\left(T_{\operatorname{sp}(\varphi_k)}\right) = \operatorname{lev}_{\leq 0}\varphi_k$ and $\operatorname{Fix}(P_K) = K$, we have

$$\operatorname{Fix}\left(\widehat{T}_k\right) \cap \operatorname{Fix}(P_K) = \operatorname{lev}_{\leq 0}\varphi_k \cap K. \tag{5.73}$$

Moreover, it is readily verified that $\operatorname{lev}_{\leq 0}\varphi_k \cap K \neq \emptyset$ if and only if $\Omega_k \neq \emptyset$ and $\varphi_k^* = 0$ (see (5.7) and (5.9)). Assuming $\operatorname{lev}_{\leq 0}\varphi_k \cap K \neq \emptyset$, we have $\Omega_k = \operatorname{lev}_{\leq 0}\varphi_k \cap K$. Since $P_K\widehat{T}_k$ is $\frac{2-\lambda_k}{2}$-attracting quasi-nonexpansive with $\operatorname{Fix}\left(P_K\widehat{T}_k\right) = K \cap \operatorname{lev}_{\leq 0}\varphi_k = \Omega_k$, the sequence $(\boldsymbol{h}_k)_{k\in\mathbb{N}}$ generated by APSM satisfies

$$\left\|\boldsymbol{h}_k - \hat{\boldsymbol{h}}_k^*\right\|^2 - \left\|\boldsymbol{h}_{k+1} - \hat{\boldsymbol{h}}_k^*\right\|^2 \geq \frac{2-\lambda_k}{2}\|\boldsymbol{h}_k - \boldsymbol{h}_{k+1}\|^2,\ \forall \hat{\boldsymbol{h}}_k^* \in \Omega_k. \tag{5.74}$$

If in addition $\Omega := \bigcap_{k\geq\kappa_0}\Omega_k \neq \emptyset$ for some $\kappa_0 \in \mathbb{N}$ (see (5.9)), then for $k \geq \kappa_0$

$$\left\|\boldsymbol{h}_k - \hat{\boldsymbol{h}}^*\right\|^2 - \left\|\boldsymbol{h}_{k+1} - \hat{\boldsymbol{h}}^*\right\|^2 \geq \frac{2-\lambda_k}{2}\|\boldsymbol{h}_k - \boldsymbol{h}_{k+1}\|^2,\ \forall \hat{\boldsymbol{h}}^* \in \Omega, \tag{5.75}$$

which implies that $(\boldsymbol{h}_k)_{k\geq\kappa_0}$ is Fejér monotone with respect to Ω. The property in (5.75) was used to prove the strong convergence of $(\boldsymbol{h}_k)_{k\in\mathbb{N}}$ in Theorem 5.5.c.

LECTURE 6
Topics in Adaptive Filtering

6.1. Outline of Lecture 6

6.2. Introduction

In this lecture, we provide several topics in adaptive filtering. First we discuss about the advances of APSM, and then introduce sparse adaptive filters. The variable-metric APSM is presented as a unified framework encompassing *the proportionate adaptive filtering algorithms* for the sparse adaptive filters. Adaptive learning with kernels is presented as a nonlinear extension of linear adaptive filters; some basics of *reproducing kernels* are provided. We finally give brief discussions about two other topics: distributed adaptive filtering and multi-domain adaptive filtering.

6.3. Advances of APSM

As seen in Lecture 5, APSM minimizes a sequence of continuous convex cost functions over a given closed convex set. *How should we do however if multiple convex constraints are imposed on our estimator?* To be specific, suppose that we have multiple closed convex sets $K_i \in \mathcal{H}$, $i = 1, 2, \cdots, m$, to which the estimator $\boldsymbol{h}_k \in \mathcal{H}$ is required to belong. If the constraints are consistent, i.e. $K := \bigcap_{i=1}^{m} K_i \neq \emptyset$, then the composition $T := T_m T_{m-1} \cdots T_1$ of the mappings T_is defined as in (5.62) is strongly attracting nonexpansive with $\mathrm{Fix}\,(T) = K$. Therefore in such a case it is desired to minimize a sequence of cost functions over $\mathrm{Fix}\,(T)$ that is the set of all points satisfying every constraint. In [**76**], APSM in (5.6) has been extended to the following form:

$$\boldsymbol{h}_{k+1} := \begin{cases} T\left(\boldsymbol{h}_k - \lambda_k \dfrac{\varphi_k(\boldsymbol{h}_k)}{\|\varphi_k'(\boldsymbol{h}_k)\|^2}\varphi_k'(\boldsymbol{h}_k)\right) & \text{if } \varphi_k'(\boldsymbol{h}_k) \neq \boldsymbol{0} \\ T(\boldsymbol{h}_k) & \text{otherwise,} \end{cases} \tag{6.1}$$

where $T : \mathcal{H} \to \mathcal{H}$ is strongly attracting nonexpansive. It has been proven that Theorem 5.5 can be extended to (6.1) essentially with the replacement of K by $\mathrm{Fix}\,(T)$. We repeat that the metric projection P_K is strongly attracting (specifically 1-attracting) nonexpansive with $\mathrm{Fix}\,(P_K) = K$. The above strategy is efficient when P_{K_i} for each i can be calculated explicitly but P_K *cannot*.

How about the case that each projection P_{K_i} cannot be calculated explicitly? In such a case, the use of subgradient projection would be an alternative. Since subgradient projection is strongly attracting quasi-nonexpansive but *not nonexpansive*, the results in [**76**] are not applicable directly. It has been proven that the results in [**76**] can further be extended to the case that T is strongly attracting quasi-nonexpansive [**77**].

Finally we give a remark on the case that the constraint set K of APSM is a linear variety; this arises in a variety of applications such as adaptive beamforming, blind multiple access interference suppression in wireless communication systems, etc [**78**–**80**]. One may think that Theorem 5.5 does not apply to this case, because the set Ω is a subset of K hence does not have an interior point. However, in this case, we can regard the underlying subspace of K as a Hilbert space; this has been discussed in [**81**].

6.4. Sparse Adaptive Filters

In the first decade of the twenty first century, a significant amount of attention has been paid to developing adaptive filtering algorithms exploiting *sparseness* of the estimandum. Here, the estimandum $\boldsymbol{h}^*$ is said to be *sparse* when it has only a few coefficients different significantly from zero (in other words it has many coefficients equal to, either approximately or exactly, zero). There are basically two streams. One is based on the *proportionate adaptive filtering* developed originally by

D. L. Duttweiler in the year of 2000 [**82**]. Thereafter, a variety of its improved versions have been proposed [**83**–**88**]. The other one is motivated by *compressed sensing* [**89**–**93**]. A connection between the proportionate adaptive filtering and compressed sensing has been discussed in [**94**]. Let us present the idea of those two steams one by one below.

The original work of the proportionate adaptive filtering [**82**] is based on the following idea. Consider the situation that we have no a priori knowledge about the estimandum except that it is *sparse*. In such a situation, a natural choice of the initial point would be the null vector (i.e., $\boldsymbol{h}_0 = \boldsymbol{0}$). The sparseness of the estimandum suggests that some coefficients of the filter should be corrected to a larger extent than the other ones. This implies that it makes sense to assign an individual step size to each coefficient of the filter in such a way that the step size is proportional to the magnitude of the corresponding coefficient of the estimandum. It has been reported that this idea results in faster convergence with a slight increase of computational complexity. The algorithm is given as follows:

$$\boldsymbol{h}_{k+1} := \boldsymbol{h}_k - \lambda_k e_k(\boldsymbol{h}_k)\boldsymbol{G}_k\boldsymbol{u}_k,\ k \in \mathbb{N}, \tag{6.2}$$

where $\boldsymbol{G}_k \in \mathbb{R}^N$ is a diagonal matrix whose diagonal entries are proportional roughly to the magnitude of each coefficient of $\boldsymbol{h}_k$. To realize the proportionality, $\boldsymbol{h}^*$ is approximated by its instantaneous estimate $\boldsymbol{h}_k$. It should be mentioned that a certain heuristic approach was used to avoid each diagonal entry of $\boldsymbol{G}_k$ from becoming zero. Its normalized version has been presented in [**83**]:

$$\boldsymbol{h}_{k+1} := \boldsymbol{h}_k - \lambda_k \frac{e_k(\boldsymbol{h}_k)}{\boldsymbol{u}_k^{\mathsf{T}}\boldsymbol{G}_k\boldsymbol{u}_k}\boldsymbol{G}_k\boldsymbol{u}_k,\ k \in \mathbb{N}, \tag{6.3}$$

which is called the *Proportionate Normalized Least Mean Square (PNLMS) algorithm*. It has been pointed out in [**52**] that PNLMS can be interpreted as an iterative projection method onto the same hyperplane H_k as NLMS (see (3.66)) with respect to the *time-variable metric* induced by the following inner product:

$$\langle \boldsymbol{x}, \boldsymbol{y} \rangle_{\boldsymbol{G}_k^{-1}} := \boldsymbol{x}^{\mathsf{T}}\boldsymbol{G}_k^{-1}\boldsymbol{y},\ \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^N. \tag{6.4}$$

More details about this topic will be provided in Section 6.5.

The second stream is based on the minimization of cost functions penalized by the ℓ_1-*norm* (or *a weighted* ℓ_1-*norm*), inspired by the fact that the ℓ_1-norm promotes the sparsity of the solution unlike the ℓ_2-norm. These methods basically involve another operation to promote the sparsity combined with the conventional adaptive filtering

algorithms such as NLMS, APA, APSP, etc. Such additional operation includes the *soft-thresholding* (originally proposed for *denoising* by D. L. Donoho in 1995 [**95**]) and *the metric projection onto the weighted* ℓ_1*-ball*. The soft-thresholding operator is given as follows:

$$T_{\mathrm{st}} : \mathbb{R}^N \to \mathbb{R}^N, \boldsymbol{x} \mapsto \sum_{i=1}^{N} \mathrm{sgn}(\langle \boldsymbol{x}, \boldsymbol{e}_i \rangle) \max\{0, |\langle \boldsymbol{x}, \boldsymbol{e}_i \rangle| - \omega\}\boldsymbol{e}_i, \tag{6.5}$$

where $\omega > 0$, the inner product is the standard one, $\boldsymbol{e}_i \in \mathbb{R}^N$ denotes the unit vector having only one nonzero element at the ith position, and $\mathrm{sgn} : \mathbb{R} \to \{-1, 0, 1\}$ stands for the *signum function* that maps a positive- and negative- valued number to 1 and -1, respectively, and zero to zero itself. The metric projection onto the weighted ℓ_1-ball requires $O(N \log_2(N))$ complexity, and its low complexity version based on the subgradient projection has been presented in [**96**].

6.5. Variable-Metric APSM

A metric can be induced by an inner product $\langle \cdot, \cdot \rangle_{\boldsymbol{Q}}$ (see (6.4)). Note here that $\boldsymbol{Q} \in \mathbb{R}^{N \times N}$ must be positive definite in order that $\langle \boldsymbol{x}, \boldsymbol{y} \rangle_{\boldsymbol{Q}} := \boldsymbol{x}^{\mathsf{T}}\boldsymbol{Q}\boldsymbol{y}$, $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^N$, defines an inner product; see Lecture 2. In this section, we refer to such $\boldsymbol{Q}$ as a *metric* for convenience. We explicitly express the metric employed in defining the metric projection by the superscript $(\cdot)^{(\boldsymbol{Q})}$ such as $P_C^{(\boldsymbol{Q})}$. Likewise, we express the metric employed in defining a subgradient projection by the superscript such as $T_{\mathrm{sp}(f)}^{(\boldsymbol{Q})}$. We define the *variable-metric APSM* as follows:

$$\boldsymbol{h}_{k+1} := P_K^{(\boldsymbol{Q}_k)}\left[\boldsymbol{h}_k + \lambda_k(T_{\mathrm{sp}(\varphi_k)}^{(\boldsymbol{Q}_k)}(\boldsymbol{h}_k) - \boldsymbol{h}_k)\right],\ k \in \mathbb{N}, \tag{6.6}$$

where $\lambda_k \in [0, 2]$, $\forall k \in \mathbb{N}$, and $\boldsymbol{Q}_k \in \mathbb{R}^{N \times N}$ is a positive definite matrix. Here, as in the setup of APSM in Section 5.4, $K \subset \mathcal{H}$ is a closed convex set and $\varphi_k : \mathcal{H} \to [0, \infty)$, $k \in \mathbb{N}$, is a continuous convex function which we assume to satisfy $K \cap \mathrm{lev}_{\leq 0}\varphi_k \neq \emptyset$. We repeat that $K \cap \mathrm{lev}_{\leq 0}\varphi_k \neq \emptyset$ if and only if $\varphi_k^* = 0$ and $\Omega_k \neq \emptyset$, and in this case (see Section 5.10)

$$\mathrm{Fix}\left(P_K^{(\boldsymbol{Q}_k)}\left[I + \lambda_k(T_{\mathrm{sp}(\varphi_k)}^{(\boldsymbol{Q}_k)} - I)\right]\right) = K \cap \mathrm{lev}_{\leq 0}\varphi_k. \tag{6.7}$$

For any positive definite matrix $\boldsymbol{A} \in \mathbb{R}^{N \times N}$, we denote by $\sigma_{\boldsymbol{A}}^{\min}$ and $\sigma_{\boldsymbol{A}}^{\max}$ the minimum and maximum eigenvalues of $\boldsymbol{A}$, respectively.

Assumption 6.8.

(a) There exist $\delta_{\min}, \delta_{\max} \in (0, \infty)$ such that $\delta_{\min} < \sigma_{\boldsymbol{Q}_k}^{\min} \leq \sigma_{\boldsymbol{Q}_k}^{\max} < \delta_{\max}$, $\forall k \in \mathbb{N}$.

(b) Let $\lambda_k \in [\varepsilon_1, 2-\varepsilon_2] \subset (0,2)$, $k \geq K_0$, for $\varepsilon_1, \varepsilon_2 > 0$. Under Assumption 6.8.a, there exist a positive definite matrix $\boldsymbol{Q} \in \mathbb{R}^{N\times N}$, $K_1 \geq K_0$, $\tau > 0$, and a closed convex subset $\Gamma \subseteq \Omega$ such that the *fluctuation matrix* $\boldsymbol{E}_k := \boldsymbol{Q}_k - \boldsymbol{Q}$ satisfies

$$\frac{\|\boldsymbol{h}_{k+1} + \boldsymbol{h}_k - 2\boldsymbol{z}^*\|_{\boldsymbol{Q}} \|\boldsymbol{E}_k\|_{\boldsymbol{Q}}}{\|\boldsymbol{h}_{k+1} - \boldsymbol{h}_k\|_{\boldsymbol{Q}}} < \frac{\varepsilon_1\varepsilon_2\sigma^{\min}_{\boldsymbol{Q}}\delta^2_{\min}}{(2-\varepsilon_2)^2\sigma^{\max}_{\boldsymbol{Q}}\delta_{\max}} - \tau,$$

$$(\forall k \geq K_1 \text{ such that } \boldsymbol{h}_k \notin \Omega_k), \forall \boldsymbol{z}^* \in \Gamma. \tag{6.9}$$

Here, for any matrix $\boldsymbol{A} \in \mathbb{R}^{N\times N}$,

$$\|\boldsymbol{A}\|_2 := \sup_{\boldsymbol{x}\in\mathbb{R}^N} \frac{\|\boldsymbol{A}\boldsymbol{x}\|_2}{\|\boldsymbol{x}\|_2} = \sqrt{\sigma^{\max}_{\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}}} \tag{6.10}$$

denotes the spectral norm of $\boldsymbol{A}$ [**8**].

□

Intuitively, Assumption 6.8 requires *small fluctuations of the metric $\boldsymbol{Q}_k$ around some fixed one $\boldsymbol{Q}$*. It has been proven in [**97**] that, if Γ has an interior point under Assumption 6.8, we can extend Theorem 5.5 to the variable-metric scheme in (6.6). The key in the proof is that Assumption 6.8 ensures the following for any $\boldsymbol{z}^* \in \Gamma$:

$$\|\boldsymbol{h}_k - \boldsymbol{z}^*\|^2_{\boldsymbol{Q}} - \|\boldsymbol{h}_{k+1} - \boldsymbol{z}^*\|^2_{\boldsymbol{Q}} \geq \frac{\tau}{\sigma^{\max}_{\boldsymbol{Q}}} \|\boldsymbol{h}_k - \boldsymbol{h}_{k+1}\|^2_{\boldsymbol{Q}}, \ \forall k \geq K_1. \tag{6.11}$$

Letting $\boldsymbol{Q}_k := \boldsymbol{G}_k^{-1}$ and $\varphi_k(\boldsymbol{x}) := d^{(\boldsymbol{G}_k^{-1})}_{H_k}(\boldsymbol{x}) := \min_{\boldsymbol{y}\in H_k} \|\boldsymbol{x} - \boldsymbol{y}\|_{\boldsymbol{G}_k^{-1}}$ in (6.6), we can reproduce PNLMS in (6.3).

Interestingly, the classical RLS algorithm can also be seen as a variable-metric projection method, as shown below. If we manipulate (1.22), we obtain

$$\boldsymbol{h}_{k+1} := \boldsymbol{h}_k - \widehat{\lambda}_k \frac{e_k(\boldsymbol{h}_k)}{\boldsymbol{u}_k^{\mathrm{T}}\boldsymbol{R}_k^{-1}\boldsymbol{u}_k}\boldsymbol{R}_k^{-1}\boldsymbol{u}_k, \tag{6.12}$$

where

$$\widehat{\lambda}_k := \lambda_k \boldsymbol{u}_k^{\mathrm{T}}\boldsymbol{R}_k^{-1}\boldsymbol{u}_k = \frac{\boldsymbol{u}_k^{\mathrm{T}}\boldsymbol{R}_k^{-1}\boldsymbol{u}_k}{\boldsymbol{u}_k^{\mathrm{T}}\boldsymbol{R}_k^{-1}\boldsymbol{u}_k + \gamma} \in (0,1). \tag{6.13}$$

The algorithm (6.12) can be reproduced by letting $\boldsymbol{Q}_k := \boldsymbol{R}_k$, $\varphi_k(\boldsymbol{x}) := d^{(\boldsymbol{R}_k)}_{H_k}(\boldsymbol{x}) := \min_{\boldsymbol{y}\in H_k} \|\boldsymbol{x} - \boldsymbol{y}\|_{\boldsymbol{R}_k}$, and $\lambda_k := \frac{\boldsymbol{u}_k^{\mathrm{T}}\boldsymbol{R}_k^{-1}\boldsymbol{u}_k}{\boldsymbol{u}_k^{\mathrm{T}}\boldsymbol{R}_k^{-1}\boldsymbol{u}_k+\gamma}$ in (6.6).

In addition to the above two algorithms, it has been shown in [**97**] that the variable-metric APSM includes as its special examples the following four algorithms: (i) the LMS-Newton adaptive filter (LNAF) [**98**–**100**], (ii) the quasi-Newton adaptive filter (QNAF) [**101, 102**], (iii) the transform-domain adaptive filtering algorithm [**103, 104**], and

(iv) the Krylov-proportionate adaptive filtering algorithm [**105, 106**]. The LNAF and QNAF algorithms are based on a similar spirit to RLS. The transform-domain algorithm decorrelates colored inputs to attain faster convergence. The Krylov-proportionate adaptive filtering algorithm is based on the spirit of *sparsifying the estimandum that is not necessarily sparse*, and it is an extension of the proportionate adaptive filtering to nonsparse estimandum.

6.6. Nonlinear Adaptive Filters Based on Kernels

Nonlinear adaptive filters based on kernels have potentials to outperform linear ones. This section provides its rough idea together with some basics of *reproducing kernels*.

6.6.1. Short Historical-Introduction to Reproducing Kernel Hilbert Space

Positive definite kernels [**107**] have been proven a powerful tool in a wide range of applications when a system of interest involves "nonlinearity" [**108, 109**]. The kernels are widely referred to also as *Mercer kernels* (named after J. Mercer), *reproducing kernels* [**110**] etc.; unless otherwise stated, we mean by kernels the positive definite kernels. The key findings of particular importance is the so-called *reproducing property* [**110**–**114**] together with the discovery of the existence of a Hilbert space associated with each kernel, credited to E. H. Moore [**112**–**114**] and N. Aronszajn [**110**].[1] The space, of which the elements and the inner product are both characterized by a kernel, is specially called *reproducing kernel Hilbert space (RKHS)*.

A remarkable feature of RKHS is that, although it may become of infinite dimension, inner products can always be computed by simple evaluations of the kernel function; this is known as *kernel trick*. Another one is the so-called *representer theorem* [**115, 116**] allowing us to operate solely in a finite-dimensional subspace spanned by vectors that are parametrized by patterns (data samples). This is of great importance for engineering applications in which the computation time is strictly limited, thus having motivated a considerable amount of researches. Typical examples exploiting the RKHS theory include the popular support vector machine, the Gaussian process regression, the kernel principal component analysis, and the kernel Fisher discriminant analysis, among others.

[1] The widely used term "metric projection" was firstly used by N. Aronszajn (and K. T. Smith) in 1954 [**15**, p. 87].

6.6.2. Kernels – Definition, Examples, Properties

Given nonempty set $\mathcal{X}$, a mapping[2] $\kappa(\cdot,\cdot) : \mathcal{X}^2 (:= \mathcal{X}\times\mathcal{X}) \to \mathbb{R}$ is called *a positive definite kernel* if, for any $m \in \mathbb{N}^*$ and any $(x_i, x_j) \in \mathcal{X}^2$, the $m \times m$ Gram matrix $\boldsymbol{K}$ with its (i,j) element $K_{i,j} := \kappa(x_i, x_j)$ is positive semidefinite; i.e.,

$$\boldsymbol{a}^{\mathsf{T}}\boldsymbol{K}\boldsymbol{a} \geq 0,\ \forall \boldsymbol{a} \in \mathbb{R}^m, \tag{6.14}$$

where $(\cdot)^{\mathsf{T}}$ denotes *transpose*.

Example 6.15. (Positive definite kernels) We present celebrated examples of positive definite kernels when $\mathcal{X} := \mathbb{R}^N$ for some $N \in \mathbb{N}^*$.

(a) Linear kernel:

$$\kappa(\boldsymbol{x}_1, \boldsymbol{x}_2) := \boldsymbol{x}_1^{\mathsf{T}}\boldsymbol{x}_2,\ \forall \boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{X}. \tag{6.16}$$

(b) Polynomial kernel:

$$\kappa(\boldsymbol{x}_1, \boldsymbol{x}_2) := (\alpha + \boldsymbol{x}_1^{\mathsf{T}}\boldsymbol{x}_2)^p,\ \forall \boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{X}, \tag{6.17}$$

where $\alpha \geq 0$ and $p \in \mathbb{N}^*$.

(c) Gaussian (or radial basis function) kernel:

$$\kappa(\boldsymbol{x}_1, \boldsymbol{x}_2) := \exp\left(-\frac{(\boldsymbol{x}_1 - \boldsymbol{x}_2)^{\mathsf{T}}(\boldsymbol{x}_1 - \boldsymbol{x}_2)}{2\sigma^2}\right),\ \forall \boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{X}, \tag{6.18}$$

where $\sigma > 0$.

(d) Laplacian kernel:

$$\kappa(\boldsymbol{x}_1, \boldsymbol{x}_2) := \exp\left(-\frac{\sqrt{(\boldsymbol{x}_1 - \boldsymbol{x}_2)^{\mathsf{T}}(\boldsymbol{x}_1 - \boldsymbol{x}_2)}}{\sigma}\right),\ \forall \boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{X}, \tag{6.19}$$

where $\sigma > 0$.

Lemma 6.20 (Properties of positive definite kernels)**.**

(a) *Nonnegativity:* $\kappa(x,x) \geq 0\ \forall x \in \mathcal{X}$.
(b) *Symmetry:* $\kappa(x_1, x_2) = \kappa(x_2, x_1),\ \forall (x_1, x_2) \in \mathcal{X}^2$.
(c) *Cauchy-Schwarz Inequality:*
$|\kappa(x_1, x_2)|^2 \leq \kappa(x_1, x_1)\kappa(x_2, x_2)$.

One may think that kernels have similar properties to inner products. From this point of view, kernels are considered as *similarity measures* to quantify how close two vectors are to each other. A remarkable difference is however that *linearity does no longer hold*. It should be mentioned that $\mathcal{X}$ is not necessarily a vector space, thus the existence of a null vector in $\mathcal{X}$ is not guaranteed [**10**]. Moreover, even

[2]Most part of the paper can easily be extended to complex-valued kernels $\kappa(\cdot,\cdot) : \mathcal{X}^2 \to \mathbb{C}$; cf. [**108**].

if there exists a null vector $\theta \in \mathcal{X}$, $\kappa(x,x) = 0 \not\Leftrightarrow x = \theta$; for instance, $\kappa(\theta,\theta) = 1$ if κ is a Gaussian (or Laplacian) kernel for a Euclidean space $\mathcal{X}$.

6.6.3. Reproducing Kernel Hilbert Space

We shall shortly describe a recipe for constructing a Hilbert space with only the ingredients $\kappa(\cdot,\cdot)$ and $\mathcal{X}$ [**110, 112–114**]. The first step is to define a mapping ψ that associates $x \in \mathcal{X}$ with $\kappa(\cdot, x)$, which can now be considered as *a function with a single argument* (because the second argument is already specified as x). In other words, ψ is a mapping from $\mathcal{X}$ to *the space of functions* $\mathbb{R}^{\mathcal{X}} := \{\boldsymbol{f} : \mathcal{X} \to \mathbb{R}\}$; i.e.,

$$\psi : \mathcal{X} \to \mathbb{R}^{\mathcal{X}},\ x \mapsto \kappa(\cdot, x). \tag{6.21}$$

In the context of learning, $\mathcal{X}$ stands for *the input space* from which the patterns (data samples) are taken. Every possible pattern $x \in \mathcal{X}$ is mapped to its corresponding function $\psi(x)(\cdot) = \kappa(\cdot, x)$. The goal of this section is to construct a Hilbert space that contains all such functions; the resulting space is widely called *the feature space* associated with ψ.

The second step is to construct a pre-Hilbert space (i.e., a vector space equipped with an inner product). Taking the linear span of the image of ψ, we can construct a vector space $\mathrm{span}\,\{\psi(x) : x \in \mathcal{X}\}$. Given any pair of vectors in the space

$$\boldsymbol{f}(\cdot) := \sum_{i=1}^{m} \alpha_i \kappa(\cdot, x_i),\ m \in \mathbb{N}^*,\ \alpha_i \in \mathbb{R},\ x_i \in \mathcal{X}$$
$$\boldsymbol{g}(\cdot) := \sum_{j=1}^{n} \beta_j \kappa(\cdot, y_j),\ n \in \mathbb{N}^*,\ \beta_j \in \mathbb{R},\ y_j \in \mathcal{X},$$

we can define an inner product as follows:

$$\langle \boldsymbol{f}, \boldsymbol{g} \rangle := \sum_{i=1}^{m}\sum_{j=1}^{n} \alpha_i \beta_j \kappa(x_i, y_j). \tag{6.22}$$

The operator $\langle\cdot,\cdot\rangle$ satisfies the conditions of inner product, and it is well-defined although the expansion coefficients of f and g could be non-unique [**108**].

The final step is to turn the space into a Hilbert space. Defining the induced norm as $\|\boldsymbol{f}\| := \sqrt{\langle \boldsymbol{f}, \boldsymbol{f}\rangle}$ for any $\boldsymbol{f}$ in the space, we can *complete* the space by adding all its limit points. The resultant space $\mathcal{H}$ is called *reproducing kernel Hilbert space (RKHS)* due to the following

properties.[3] Given any $\boldsymbol{f} \in \mathcal{H}$, it is readily verified that

$$\boldsymbol{f}(x) = \langle \boldsymbol{f}, \kappa(\cdot, x) \rangle\,,\ \forall x \in \mathcal{X}, \tag{6.23}$$

which is called *the reproducing property.*[4] In particular, for any x_1, $x_2 \in \mathcal{X}$,

$$\langle \psi(x_1), \psi(x_2) \rangle = \langle \kappa(\cdot, x_1), \kappa(\cdot, x_2) \rangle = \kappa(x_1, x_2). \tag{6.24}$$

This implies that inner products of elements in the high (possibly infinite) dimensional space $\mathcal{H}$ can be obtained through simple computation, which is called *kernel trick*. The following fact is of particular importance for kernel learning.

Theorem 6.25 (Representer Theorem [**115, 116**])**.** *An empirical loss function is generally a function of a collection of triplets* $\{(x_i, y_i, f(x_i))\}_{i=1}^{q} \subset \mathcal{X} \times \mathbb{R} \times \mathbb{R}$*, where* $f(x_i)$ *is an estimate/hypothesis of the* i*th output* y_i *based on the* i*th input* x_i *(*$f \in \mathcal{H}$*). It is well-known that learning based on minimization of a loss function often causes over fitting.*[5] *A common strategy to overcome the over fitting problem is to minimize a loss function penalized with a regularization term,*[6] *which can usually be written as* $\Omega(\|f\|)$ *with a strictly monotonically increasing function* $\Omega : [0, \infty) \to \mathbb{R}$*. Namely, a widely used cost function to be minimized is the regularized risk functional*

$$\varphi(\boldsymbol{f}) := \ell\left((x_1, y_1, f(x_1)), \cdots, (x_q, y_q, f(x_q))\right) + \Omega(\|\boldsymbol{f}\|),$$

where $\ell : (\mathcal{X} \times \mathbb{R} \times \mathbb{R})^q \to \mathbb{R} \cup \{\infty\}$ *is an arbitrary loss function.*

It is clear that $\boldsymbol{f}$ *can be decomposed as* $\boldsymbol{f} = \boldsymbol{f}_M + \boldsymbol{f}_{M^\perp}$*, where* $\boldsymbol{f}_M \in M := span(\kappa(\cdot, x_1), \cdots, \kappa(\cdot, x_q))$ *and* $\boldsymbol{f}_{M^\perp} \in M^\perp$ *(*$M^\perp$ *denotes the orthogonal complement of* M*;* $\mathcal{H} = M \oplus M^\perp$*). Since* $\boldsymbol{f}_{M^\perp}$ *is orthogonal to all* $\kappa(\cdot, x_i)$*s,* $\boldsymbol{f}_{M^\perp}(x_i) = \langle \boldsymbol{f}_{M^\perp}, \kappa(\cdot, x_i) \rangle = 0$*, thus* $\boldsymbol{f}(x_i) = \boldsymbol{f}_M(x_i)$*,* $\forall i = 1, 2, \cdots, q$*. This means that* $\boldsymbol{f}_{M^\perp}$ *does not affect the loss function. Moreover, as* Ω *is strictly monotonically increasing,* $\boldsymbol{f}_{M^\perp}$ *should be chosen in such a way that* $\|\boldsymbol{f}\|$ *is minimized. Noticing that* $\boldsymbol{f}_M = P_M(\boldsymbol{f})$ *and* $\boldsymbol{f}_{M^\perp} = P_{M^\perp}(\boldsymbol{f})$*, the Pythagorean theorem tells us* $\|\boldsymbol{f}\|^2 = \|\boldsymbol{f}_M\|^2 + \|\boldsymbol{f}_{M^\perp}\|^2$*, which is minimized by letting* $\boldsymbol{f}_{M^\perp} = 0$*. This indicates the important consequence: the minimizer* $\boldsymbol{f}^*$

[3] It is interesting to see that Volterra and Wiener series can be represented implicitly as elements of a RKHS by using polynomial kernels (see, e.g., [**117**]).

[4] In [**110**], RKHS is characterized by the properties (6.23) and $\kappa(\cdot, x) \in \mathcal{H}$, $\forall x \in \mathcal{X}$.

[5] The learner tends to fit a training data set well but not to be generalized to a test data set.

[6] The underlying philosophy is to constrain f to be chosen from a subclass of $\mathcal{H}$, based on the *Vapnik-Chervonenkis theory* [**118**].

of $\varphi(\boldsymbol{f})$ *satisfies* $\boldsymbol{f}^* \in span(\kappa(\cdot, x_1), \cdots, \kappa(\cdot, x_q))$*, which is known as the representer theorem in the machine learning community.*

6.6.4. Online Learning with Kernels

The success of the kernel methods in batch settings has motivated the study of *online learning with kernels* [**53, 54, 119–123**]. A *linear* adaptive filter $\boldsymbol{h}_k \in \mathbb{R}^N$ is expressed as a linear combination of past input vectors $(\boldsymbol{u}_i)_{i \le k-1}$ and an initial estimate $\boldsymbol{h}_0$; i.e.,

$$\boldsymbol{h}_k = \sum_{i=0}^{k-1} \alpha_i^{(k)} \boldsymbol{u}_i + \boldsymbol{h}_0,\ k \in \mathbb{N}, \tag{6.26}$$

where $\alpha_i^{(k)} \in \mathbb{R}$ is updated by an adaptive algorithm. The filter $\boldsymbol{h}_k \in \mathbb{R}^N$ processes a new input vector $\boldsymbol{u}_k$ *linearly* as

$$\langle \boldsymbol{h}_k, \boldsymbol{u}_k \rangle = \sum_{i=0}^{k-1} \alpha_i \langle \boldsymbol{u}_i, \boldsymbol{u}_k \rangle + \langle \boldsymbol{h}_0, \boldsymbol{u}_k \rangle\,,\ k \in \mathbb{N}, \tag{6.27}$$

where $\langle \cdot, \cdot \rangle$ denotes the standard inner product. A *nonlinear* adaptive filter based on kernels takes the following form:

$$\boldsymbol{f}_k(\cdot) := \sum_{i=0}^{k-1} \alpha_i^{(k)} \kappa(\cdot, \boldsymbol{u}_i) + \boldsymbol{f}_0(\cdot),\ k \in \mathbb{N}, \tag{6.28}$$

where $\alpha_i^{(k)} \in \mathbb{R}$ is updated by an adaptive algorithm. The filter $\boldsymbol{f}_k \in \mathcal{H}$, where $\mathcal{H}$ denotes the RKHS associated with the kernel $\kappa(\cdot, \cdot)$, processes a new input vector $\boldsymbol{u}_k$ *nonlinearly* as

$$\boldsymbol{f}_k(\boldsymbol{u}_k) = \langle \boldsymbol{f}_k, \kappa(\cdot, \boldsymbol{u}_k) \rangle := \sum_{i=0}^{k-1} \alpha_i^{(k)} \kappa(\boldsymbol{u}_k, \boldsymbol{u}_i) + \boldsymbol{f}_0(\boldsymbol{u}_k),\ k \in \mathbb{N}. \tag{6.29}$$

Compare the linear and nonlinear processing in (6.27) and (6.29) from the computational viewpoint. In (6.27), the left hand side can be directly evaluated by N multiplications since the N components of $\boldsymbol{h}_k \in \mathbb{R}^N$ are available. The computational costs and the memory requirements for the processing are therefore constant; note that the memory requirements to update the filter coefficients depend on algorithms. On the other hand, $\boldsymbol{f}_k$ is a function and one needs to evaluate the right hand side of (6.29). Namely, one needs to (i) store the coefficients $(\alpha_i^{(k)})_{i=0}^{k-1} \subset \mathbb{R}$ and input vectors $(\boldsymbol{u}_i)_{i=0}^{k-1} \subset \mathbb{R}^N$ and (ii) compute $\kappa(\boldsymbol{u}_k, \boldsymbol{u}_i)$ multiplied by $\alpha_i^{(k)}$ for every $i \in \{0, 1, \cdots, k-1\}$ as well as $\boldsymbol{f}_0(\boldsymbol{u}_k)$. Both the computational costs and the memory requirements

for the processing may thus grow linearly as time goes by. This obviously conflicts with the limitations of memory and computational resource/time. Several *sparsification* techniques have been proposed and investigated for dynamically updating the *dictionary*, a subset of input vectors (or basis vectors), in such a way that only dominant ones remain among all the input vectors $(\boldsymbol{u}_i)_{i=0}^{k-1} \subset \mathbb{R}^N$. A simplest sparsification strategy exploited in [**119**] is to use a fixed number, say q, of the newest data $(\boldsymbol{u}_i)_{i=k-q}^{k-1}$. More sophisticated strategies have been proposed in [**53, 54, 120–123**]. In the following, we do not consider sparsification for simplicity.

As $\mathcal{H}$ is a (possibly infinite dimensional) Hilbert space, the adaptive filtering algorithms developed for linear filters can naturally be extended to nonlinear filters. For instance, the NLMS algorithm in $\mathcal{H}$ is given by

$$\boldsymbol{f}_{k+1} := \boldsymbol{f}_k + \lambda \left(P_{H_k}(\boldsymbol{f}_k) - \boldsymbol{f}_k\right), \ k \in \mathbb{N}, \tag{6.30}$$

where

$$H_k := \{\boldsymbol{f} \in \mathcal{H} : \langle \boldsymbol{f}, \kappa(\cdot, \boldsymbol{u}_k)\rangle = \boldsymbol{f}(\boldsymbol{u}_k) = d_k\}. \tag{6.31}$$

Since the normal vector of H_k is $\kappa(\cdot, \boldsymbol{u}_k)$, each coefficient $\alpha_i^{(k)}$ corresponding to each $\kappa(\cdot, \boldsymbol{u}_i)$, $i \in \mathbb{N}$, is updated only once at time i by the algorithm (6.35). In [**53, 54**], APSM (which is formulated in a general Hilbert space) has been applied to the adaptive learning in RKHS.

There is another possibility to construct a nonlinear adaptive filtering algorithm based on NLMS. From (6.29), the coefficients $(\alpha_i^{(k)})_{i=0}^k$ should be updated in such a way that

$$\boldsymbol{\alpha}_{k+1}^{\mathsf{T}} \boldsymbol{\kappa}_k = d_k - \boldsymbol{f}_0(\boldsymbol{u}_k), \tag{6.32}$$

where

$$\boldsymbol{\alpha}_{k+1} := [\alpha_0^{(k+1)}, \alpha_1^{(k+1)}, \cdots, \alpha_k^{(k+1)}]^{\mathsf{T}} \in \mathbb{R}^{k+1} \tag{6.33}$$

$$\boldsymbol{\kappa}_k := [\kappa(\boldsymbol{u}_k, \boldsymbol{u}_0), \kappa(\boldsymbol{u}_k, \boldsymbol{u}_1), \cdots, \kappa(\boldsymbol{u}_k, \boldsymbol{u}_k)]^{\mathsf{T}} \in \mathbb{R}^{k+1}. \tag{6.34}$$

Thus, the coefficient vector $\boldsymbol{\alpha}_k$ can be updated as follows:

$$\boldsymbol{\alpha}_{k+1} := \widetilde{\boldsymbol{\alpha}}_k + \lambda \left(P_{\widetilde{H}_k}(\widetilde{\boldsymbol{\alpha}}_k) - \widetilde{\boldsymbol{\alpha}}_k\right), \ k \in \mathbb{N}, \tag{6.35}$$

where $\widetilde{\boldsymbol{\alpha}}_k := [\boldsymbol{\alpha}_k^{\mathsf{T}}, 0]^{\mathsf{T}} \in \mathbb{R}^{k+1}$ and

$$\widetilde{H}_k := \{\boldsymbol{\alpha} \in \mathbb{R}^{k+1} : \boldsymbol{\alpha}^{\mathsf{T}} \boldsymbol{\kappa}_k = d_k - \boldsymbol{f}_0(\boldsymbol{u}_k)\}. \tag{6.36}$$

We finally mention that the *kernelized* adaptive filters looks more complex than the linear ones, but it is simpler in the sense that the filter can be expressed with a smaller number of data samples. Linear and nonlinear adaptive learning with projections is reviewed in [**124**].

6.7. Adaptive Learning over Networks

We consider the situation where multiple sensors are employed to collect data, and the sensors are linked partially and are able to communicate with each other according to some prescribed mode of cooperation. The modes of cooperation are classified into two categories: the *incremental* mode and the *diffusion* mode. In the incremental mode, the sensors (i.e., the nodes in a network) are activated in a cyclic pattern; each node processes its local information with the information transferred from the previous node, and transfers the processed data to the next node [**125, 126**]. This mode consumes a small amount of power and suits for a small network. In the diffusion mode, on the other hand, each node processes its local information with the information transferred from its neighboring nodes in parallel, and transfers the processed data to a subset of its neighbors [**127–130**]. This mode is suitable for a large network and it can easily deal with changing topologies, node failures, etc. In [**56**], APSM has been extended to the adaptive learning in diffusion networks. *Probabilistic diffusion* is one of the interesting topics in this research area, changing the network topologies randomly for attaining significant gain with the lowest possible communication costs [**131**].

6.8. Multi-Domain Adaptive Filtering

Consider the following scenarios.

(a) The amount of data observable at a measuring equipment (such as a sensor) is strictly limited due to practical reasons. In such a case, one may need to gather and process a priori and measurable information in all the possible domains (e.g. time, frequency, space by means of multiple sensors, etc.) to compensate for the lack of information.

(b) There are many requirements from a variety of aspects such as high-performance, low power-consumption, harmless to human bodies, desirable specification in frequency domain, etc.

In such scenarios as above, each piece of information is associated with a closed convex set in *each individual domain* and, if we consider feasibility solely in a specific domain (let us call it *basic domain*), the closed convex sets in other domains should be pulled into the basic domain. Metric projection is a useful tool in the set-theoretic adaptive filtering and, for computing the projections onto the sets efficiently, the 'shapes' of the sets should be significantly simple. Each set usually has a simple 'shape' in the individual domain, but once pulled into the basic

domain, there is no guarantee that its shape remains simple. If we stick to the conventional one-domain feasibility approach, the projection methods lose its computational efficiency. The idea of *feasibility splitting* —dealing with feasibility in each individual domain— is quite useful to preserve the computational efficiency. Its original notion has been presented in [**132**–**135**], and it has been successfully extended to adaptive scenarios with the framework of APSM in [**57, 136**].

BIBLIOGRAPHY

1. A. N. Kolmogorov, "Sur l'interpolation et extrapolation des suites stationaires," *Comptes Rendus de l'Académie des Sciences*, vol. 208, pp. 2043–2045, 1939.
2. A. N. Kolmogorov, "Interpolation and extrapolation," *Bulletin de l'Académie des Sciences de U.S.S.R.*, vol. Series Mathematics 5, pp. 3–14, 1941.
3. N. Wiener, *Extrapolation, Interpolation, and Smoothing of Stationary Times Series: With Engineering Applications*, MIT Press, Cambridge, MA, 1949 (The first print: Classified National Defense Research Report (February 1942).
4. B. Widrow and M. E. Hoff Jr., "Adaptive switching circuits," in *IRE WESCON Conv. Rec.*, 1960, vol. 4, pp. 96–104.
5. A. H. Sayed, *Fundamentals of adaptive filtering*, Wiley, New Jersey, 2003.
6. R. L. Plackett, "Some theorems in least-squares," *Biometrika*, vol. 37, pp. 149, 1950.
7. S. Haykin, *Adaptive Filter Theory*, Prentice Hall, New Jersey, 4th edition, 2002.
8. R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, 1985.
9. B. Hassibi, A. H. Sayed, and T. Kailath, "H^∞ optimality of the LMS algorithm," *IEEE Trans. Signal Processing*, vol. 44, no. 2, pp. 267–280, 1996.
10. D. G. Luenberger, *Optimization by Vector Space Methods*, New York: Wiley, 1969.
11. H. Stark and Y. Yang, *Vector Space Projections—A Numerical Approach to Signal and Image Processing, Neural Nets, and Optics*, John Wiley & Sons, New York, 1998.

12. I. Yamada, *Kougaku no Tameno Kansu Kaiseki (Functional Analysis for Engineering)*, Suurikougaku-Sha / Saiensu-Sha, Tokyo, May 2009, in Japanese.
13. E. Kreyszig, *Introductory Functional Analysis with Applications*, John Wiley & Sons, New York, 1989.
14. S. Boyd and L.Vandenberghe, *Convex optimization*, Cambridge University Press, Cambridge, 2004.
15. F. Deutsch, *Best Approximation in Inner Product Spaces*, Springer, 2001.
16. M. Z. Nashed, *Generalized Inverse and Applications*, Academic Press, New York, 1976.
17. J. von Neumann, *Functional Operators –Vol. II. The Geometry of Orthogonal Spaces*, Annals of Math. Studies #22. Princeton University Press, NJ, 1950, (This is a reprint of mimeographed lecture notes first distributed in 1933).
18. H. A. Schwarz, *Grenzübergang durch alternirendes Varfahren*, 1870, reprinted in *Gesammelte Mathematische Abhandlungen.* Berlin: Springer-Verlag, vol. 2, 1890, pp. 133–143.
19. S. Kaczmarz, "Angenäherte auflösung von systemen linearer gleichungen," *Bulletin de l'Académie des Sciences de Pologne*, vol. A35, pp. 355–357, 1937.
20. G. Cimmino, "Calcolo approssimato per le soluzioni dei sistemi di equazioni lineari," *La Ricerca Scientifica (Roma)*, vol. 1, pp. 326–333, 1938.
21. R. Gordon, R. Bender, and G. T. Herman, "Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and X-ray photography," *J. Theoret. Biol.*, vol. 29, pp. 471–481, Dec. 1970.
22. I. Halperin, "The product of projection operators," *Acta Scientiarum Mathematicarum (Szeged)*, vol. 23, no. 1, pp. 96–99, 1962.
23. S. Reich, "A limit theorem for projections," *Linear Multilinear Algebra*, vol. 13, no. 3, pp. 281–290, 1983.
24. P. Gilbert, "Iterative methods for the three-dimensional reconstruction of an object from projections," *J. Theoret. Biol.*, vol. 36, pp. 105–117, 1972.
25. J. Nagumo and J. Noda, "A learning method for system identification," *IEEE Transactions on Automatic Control*, vol. 12, no. 3, pp. 282–287, June 1967.
26. A. E. Albert and L. S. Gardner, Jr., *Stochastic approximation and nonlinear regression*, Cambridge MA: MIT Press, 1967.
27. T. Hinamoto and S. Maekawa, "Extended theory of learning identification," *Trans. IEE Japan*, vol. 95, no. 10, pp. 227–234, 1975,

in Japanese.
28. K. Ozeki and T. Umeda, "An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties," *IEICE Trans.*, vol. 67-A, no. 5, pp. 126–132, 1984, in Japanese.
29. R. T. Rockafellar, *Convex Analysis*, Princeton University Press, NJ, 1970.
30. I. Ekeland and R. Temam, *Convex Analysis and Variational Problems*, North-Holland, Amsterdam, 1976.
31. J. B. Hiriart-Urruty and C. Lemaréchal, *Fundamentals of convex analysis*, Springer, Berlin, 2001.
32. C. Zălinescu, *Convex Analysis in General Vector Spaces*, World Scientific, NJ, 2002.
33. D. P. Bertsekas, A. Nedic, and A. E. Ozdaglar, *Convex Analysis and Optimization*, Athena Scientific, MA, 2003.
34. J. M. Borwein and A. S. Lewis, *Convex Analysis and Nonlinear Optimization*, Springer, NY, 2nd edition, 2006.
35. I. Yamada, K. Slavakis, and K. Yamada, "An efficient robust adaptive filtering algorithm based on parallel subgradient projection techniques," *IEEE Trans. Signal Processing*, vol. 50, no. 5, pp. 1091–1101, May 2002.
36. M. Yukawa and I. Yamada, "Pairwise optimal weight realization —Acceleration technique for set-theoretic adaptive parallel subgradient projection algorithm," *IEEE Trans. Signal Processing*, vol. 54, no. 12, pp. 4557–4571, Dec. 2006.
37. P. L. Combettes, "The foundations of set theoretic estimation," *Proc. of IEEE*, vol. 81, no. 2, pp. 182–208, 1993.
38. H. H. Bauschke and J. M. Borwein, "On projection algorithms for solving convex feasibility problems," *SIAM Review*, vol. 38, no. 3, pp. 367–426, 1996.
39. H. H. Bauschke, P. L. Combettes, and S. G. Kruk, "Extrapolation algorithm for affine-convex feasibility problems," *Numerical Algorithms*, vol. 41, pp. 239–274, 2006.
40. L. G. Gubin, B. T. Polyak, and E. V. Raik, "The method of projections for finding the common point of convex sets," *USSR Comput. Math. Phys.*, vol. 7, pp. 1–24, 1967.
41. D. C. Youla and H. Webb, "Image restoration by the method of convex projections: Part 1-theory," *IEEE Transactions on Medical Imaging*, vol. MI-1, pp. 81–94, Oct. 1982.
42. A. Lent and H. Tuy, "An iterative method for the extrapolation of band-limited functions," *J. Math. Anal. Applicat.*, vol. 83, pp. 554–565, 1981.

43. Y. Censor and S. A. Zenios, *Parallel Optimization: Theory, Algorithm, and Optimization*, Oxford University Press, 1997.
44. P. L. Combettes, "Convex set theoretic image recovery by extrapolated iterations of parallel subgradient projections," *IEEE Trans. Image Processing*, vol. 6, no. 4, pp. 493–506, 1997.
45. S. Agmon, "The relaxation method for linear inequalities," *Canadian J. Mat.*, vol. 6, no. 3, pp. 382–392, 1954.
46. T. S. Motzkin and I. J. Schoenberg, "The relaxation method for linear inequalities," *Canadian J. Mat.*, vol. 6, no. 3, pp. 393–404, 1954.
47. G. Pierra, "Decomposition through formalization in a product space," *Math. Programming*, vol. 28, pp. 96–115, 1984.
48. R. L. G. Cavalcante and I. Yamada, "Multiaccess interference suppression in orthogonal space-time block coded MIMO systems by adaptive projected subgradient method," *IEEE Trans. Signal Processing*, vol. 56, no. 3, pp. 1028–1042, Mar. 2008.
49. M. Yamagishi, M. Yukawa, and I. Yamada, "Sparse system identification by exponentially weighted adaptive parallel projection and generalized soft-thresholding," in *Proc. APSIPA Annual Summit and Conference*, 2010, to be presented.
50. M. Yukawa, R. L. G. Cavalcante, and I. Yamada, "Efficient blind MAI suppression in DS/CDMA systems by embedded constraint parallel projection techniques," *IEICE Trans. Fundamentals*, vol. E88-A, no. 8, pp. 2062–2071, Aug. 2005.
51. M. Yukawa, N. Murakoshi, and I. Yamada, "Efficient fast stereo acoustic echo cancellation based on pairwise optimal weight realization technique," *EURASIP J. Appl. Signal Processing*, vol. 2006, Article ID 84797, 15 pages, 2006.
52. M. Yukawa, K. Slavakis, and I. Yamada, "Adaptive parallel quadratic-metric projection algorithms," *IEEE Trans. Audio, Speech and Language Processing*, vol. 15, no. 5, pp. 1665–1680, July 2007.
53. K. Slavakis, S. Theodoridis, and I. Yamada, "Online kernel-based classification using adaptive projection algorithms," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 2781–2796, July 2008.
54. K. Slavakis, S. Theodoridis, and I. Yamada, "Adaptive constrained learning in reproducing kernel Hilbert spaces: the robust beamforming case," *IEEE Trans. Signal Process.*, vol. 57, no. 12, pp. 4744–4764, Dec. 2009.
55. R. L. G. Cavalcante and I. Yamada, "A flexible peak-to-average power ratio reduction scheme for OFDM systems by the adaptive projected subgradient method," *IEEE Trans. Signal Processing*,

vol. 57, no. 4, pp. 1456–1468, Apr. 2009.
56. R. L. G. Cavalcante, I. Yamada, and B. Mulgrew, "An adaptive projected subgradient approach to learning in diffusion networks," *IEEE Trans. Signal Processing*, vol. 57, no. 7, pp. 2762–2774, Jul. 2009.
57. M. Yukawa, K. Slavakis, and I. Yamada, "Multi-domain adaptive learning based on feasibility splitting and adaptive projected subgradient method," *IEICE Trans. Fundamentals*, vol. E93-A, no. 2, pp. 456–466, Feb. 2010.
58. A. A. Goldstein, "Convex programming in Hilbert space," *Bull. Amer. Math. Soc. 70*, pp. 709–710, 1964.
59. E. S. Levitin and B. T. Polyak, "Constrained minimization method," *USSR Comput. Math. Physics 6*, pp. 1–50, 1966.
60. B. T. Polyak, "Minimization of unsmooth functionals," *USSR Comput. Math. Physics 9*, pp. 14–29, 1969.
61. I. Yamada, "Adaptive projected subgradient method: A unified view for projection based adaptive algorithms," *The Journal of IEICE*, vol. 86, no. 8, pp. 654–658, Aug. 2003, in Japanese.
62. I. Yamada and N. Ogura, "Adaptive projected subgradient method for asymptotic minimization of sequence of nonnegative convex functions," *Numer. Funct. Anal. Optim.*, vol. 25, no. 7&8, pp. 593–617, 2004.
63. I. Yamada and N. Ogura, "Hybrid steepest descent method for variational inequality problem over the fixed point set of certain quasi-nonexpansive mappings," *Numer. Funct. Anal. Optim.*, vol. 25, no. 7&8, pp. 619–655, 2004.
64. I. Yamada, M. Yukawa, and M. Yamagishi, *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, chapter Minimizing Moreau envelope of nonsmooth convex function over the fixed point set of certain quasi-nonexpansive mappings, the Series Springer Optimization and Its Applications. Springer, 2010, to be published.
65. H. H. Bauschke and P. L. Combettes, "A weak-to-strong convergence principle for Fejér monotone methods in Hilbert spaces," *Mathematics of Operations Research*, vol. 26, no. 2, pp. 248–264, May 2001.
66. J. C. Dunn, "Convexity, monotonicity, and gradient processes," *J. Math. Anal. Appl.*, vol. 53, pp. 145–158, 1976.
67. J.-B. Baillon and G. Haddad, "Quelques propriétés des opérateurs angle-bornés et n-cycliquement monotones," *Israel J. Math.*, vol. 22, pp. 137–150, 1977.

68. C. Byrne, "A unified treatment of some iterative algorithms in signal processing and image reconstruction," *Inverse Problems*, vol. 20, pp. 103–120, 2004.
69. C. W. Groetsch, *Inverse Problems in the Mathematical Sciences*, Wiesbaden-Vieweg, 1993.
70. W. R. Mann, "Mean value methods in iteration," *Proc. Amer. Math. Soc.*, vol. 4, pp. 506–510, 1953.
71. M. A. Krasnosel'skiĭ, "Two remarks on the method of successive approximations," *Uspekhi Mat. Nauk*, vol. 10, no. 1(63), pp. 123–127, 1955, (in Russian).
72. W. G. Dotson, Jr., "On the Mann iterative process," *Trans. Amer. Math. Soc.*, vol. 149, May 1970.
73. N. Ogura and I. Yamada, "Non-strictly convex minimization over the fixed point set of an asymptotically shrinking nonexpansive mapping," *Numer. Funct. Anal. Optim.*, vol. 23, pp. 113–137, 2002.
74. I. Yamada, N. Ogura, and N. Shirakawa, "A numerically robust hybrid steepest descent method for the convexly constrained generalized inverse problems," *Contemporary Mathematics*, vol. 313, pp. 269–305, 2002.
75. P. L. Combettes and P. Bondon, "Hard-constrained inconsistent signal feasibility problems," *IEEE Trans. Signal Processing*, vol. 47, no. 9, pp. 2460–2468, Sep. 1999.
76. K. Slavakis, I. Yamada, and N. Ogura, "Adaptive projected subgradient method over the fixed point set of strongly attracting non-expansive mappings," *Numer. Funct. Anal. Optim.*, vol. 27, no. 7&8, pp. 905–930, 2006.
77. K. Slavakis and I. Yamada, "Asymptotic minimization of sequences of loss functions constrained by families of quasi-nonexpansive mappings and its application to online learning," 2010, submitted for publication.
78. O. L. III Frost, "An algorithm for linearly constrained adaptive array processing," *Proc. of IEEE*, vol. 60, no. 8, pp. 926–935, Aug. 1972.
79. L. J. Griffiths and C. W. Jim, "An alternative approach to linearly constrained adaptive beamforming," *IEEE Trans. Antennas and Propagation*, pp. 27–34, Jan. 1982.
80. M. Honig, U. Madhow, and S. Verdu, "Blind adaptive multiuser detection," *IEEE Trans. Inform. Theory*, vol. 41, no. 4, pp. 944–960, July 1995.

81. M. Yukawa and I. Yamada, "A note on adaptive projected subgradient method under linear constraints," in *Proc. IEICE Signal Processing Symposium*, Nov. 2010.
82. D. L. Duttweiler, "Proportionate normalized least-mean-squares adaptation in echo cancelers," *IEEE Trans. Speech Audio Processing*, vol. 8, no. 5, pp. 508–518, Sept. 2000.
83. S. L. Gay, "An efficient fast converging adaptive filter for network echo cancellation," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, 1998, pp. 394–398.
84. J. Benesty, T. Gänsler, D. R. Morgan, M. M. Sondhi, and S. L. Gay, *Advances in Network and Acoustic Echo Cancellation*, Springer-Verlag, Berlin, 2001.
85. J. Benesty and S. L. Gay, "An improved PNLMS algorithm," in *Proc. IEEE ICASSP*, 2002, pp. 1881–1884.
86. O. Hoshuyama, R. A. Goubran, and A. Sugiyama, "A generalized proportionate variable step-size algorithm for fast changing acoustic environment," in *Proc. IEEE ICASSP*, 2004, pp. IV–161–IV–164.
87. H. Deng and M. Doroslovački, "Proportionate adaptive algorithms for network echo cancellation," *IEEE Trans. Signal Processing*, vol. 54, no. 5, pp. 1794–1803, May 2006.
88. K. T. Wagner and M. Doroslovački, "Proportionate-type NLMS algorithms based on maximization of the joint conditional PDF for the weight deviation vector," in *Proc. IEEE ICASSP*, 2010, pp. 3738–3741.
89. Y. Gu, J. Jin, and S. Mei, "ℓ_0 norm constraint LMS algorithm for sparse system identification," *IEEE Signal Processing Lett.*, vol. 16, no. 9, pp. 774–777, Sep. 2009.
90. Y. Chen, Y. Gu, and A. O. Hero, "Sparse LMS for system identification," in *Proc. IEEE ICASSP*, 2009, pp. 3125–3128.
91. D. Angelosante, J. A. Bazerque, and G. B. Giannakis, "Online adaptive estimation of sparse signals: where RLS meets the ℓ_1-norm," *IEEE Trans. Signal Processing*, vol. 58, no. 7, pp. 3436–3447, 2010.
92. Y. Murakami, M. Yamagishi, M. Yukawa, and I. Yamada, "A sparse adaptive filtering using time-varying soft-thresholding techniques," in *Proc. IEEE ICASSP*, 2010, pp. 3734–3737.
93. K. Slavakis, Y. Kopsinis, and S. Theodoridis, "Adaptive algorithm for sparse system identification using projections onto weighted ℓ_1 balls," in *Proc. IEEE ICASSP*, 2010, pp. 3742–3745.

94. J. Benesty, C. Paleologu, and S. Ciochină, "Proportionate adaptive filters from a basis pursuit perspective," *IEEE Signal Processing Lett.*, vol. 17, no. 12, pp. 985–988, Dec. 2010.
95. D. L. Donoho, "De-noising by soft-thresholding," *IEEE Trans. Inform. Theory*, vol. 41, no. 3, pp. 613–627, May 1995.
96. K. Slavakis, S. Theodoridis, and I. Yamada, "Low complexity projection-based adaptive algorithm for sparse system identification and signal reconstruction," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, 2010, to appear.
97. M. Yukawa and I. Yamada, "A unified view of adaptive variable-metric projection algorithms," *EURASIP J. Advances in Signal Processing*, vol. 2009, Article ID 589260, 13 pages, 2009.
98. B. Widrow and S. D. Stearns, *Adaptive Signal Processing*, Prentice Hall, Englewood Cliffs: NJ, 1985.
99. P. S. R. Diniz, M. L. R. de Campos, and A. Antoniou, "Analysis of LMS-Newton adaptive filtering algorithms with variable convergence factor," *IEEE Trans. Signal Processing*, vol. 43, no. 3, pp. 617–627, Mar. 1995.
100. B. Farhang-Boroujeny, *Adaptive Filters: Theory and Applications*, Wiley, Chichester: UK, 1998.
101. D. F. Marshall and W. K. Jenkins, "A fast quasi-Newton adaptive filtering algorithm," *IEEE Trans. Signal Processing*, vol. 40, no. 7, pp. 1652–1662, Jul. 1992.
102. M. L. R. de Campos and A. Antoniou, "A new quasi-Newton adaptive filtering algorithm," *IEEE Trans. Circuits and Systems II*, vol. 44, no. 11, pp. 924–934, Nov. 1997.
103. S. S. Narayan, A. M. Peterson, and M. J. Narasimha, "Transform domain LMS algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, no. 3, pp. 609–615, Jun. 1983.
104. D. F. Marshall, W. K. Jenkins, and J. J. Murphy, "The use of orthogonal transforms for improving performance of adaptive filters," *IEEE Trans. Circuits and Systems*, vol. 36, no. 4, pp. 474–484, Apr. 1989.
105. M. Yukawa, "Krylov-proportionate adaptive filtering techniques not limited to sparse systems," *IEEE Trans. Signal Processing*, vol. 57, no. 3, pp. 927–943, Mar. 2009.
106. M. Yukawa and W. Utschick, "A fast stochastic gradient algorithm: Maximal use of sparsification benefits under computational constraints," *IEICE Trans. Fundamentals*, vol. E93-A, no. 2, pp. 467–475, Feb. 2010.

107. J. Mercer, "Functions of positive and negative type and their connection with the theory of integral equations," *Philos. Trans. Roy. Soc. London*, vol. A 209, pp. 415–446, 1909.
108. B. Schölkopf and A. J. Smola, *Learning with Kernels*, MIT Press, Cambridge, MA, 2001.
109. S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, Academic, New York, 4th edition, 2008.
110. N. Aronszajn, "Theory of reproducing kernels," *Trans. Amer. Math. Soc.*, vol. 68, no. 3, pp. 337–404, May 1950.
111. S. Bergman, "Über die entwicklung der harmonischen funktionen der ebene und des raumes nach orthogonalfunktionen," *Math. Ann.*, vol. 86, no. 3–4, pp. 238–271, 1922.
112. E. H. Moore, "On properly positive Hermitian matrices," *Bull. Amer. Math. Soc.*, vol. 23, 1916.
113. E. H. Moore, *General Analysis*, Part I. Memoirs of the American Philosophical Society, 1935.
114. E. H. Moore, *General Analysis*, Part II. Memoirs of the American Philosophical Society, 1939.
115. G. Kimeldorf and G. Wahba, "Some results on Tchebycheffian spline functions," *J. Math. Anal. Appl.*, vol. 33, pp. 82–95, 1971.
116. B. Schölkopf, R. Herbrich, and A. J. Smola, "A generalized representer theorem," in *Proc. Annual Conf. Comput. Learn. Theory and European Conf. Comput.Learn. Theory*. July 2001, vol. 2111, pp. 416–426, Springer.
117. M. Franz and B. Schölkopf, "A unifying view of Wiener and Volterra theory and polynomial kernel regression," *Neural Computation*, vol. 18, no. 12, pp. 3097–3118, 2006.
118. V. Vapnik, *The Nature of Statistical Learning Theory*, Springer, 2nd edition, 1999.
119. J. Kivinen, A. J. Smola, and R. C. Williamson, "Online learning with kernels," *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2165–2176, Aug. 2004.
120. Y. Engel, S. Mannor, and R. Meir, "The kernel recursive least-squares algorithm," *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2275–2285, Aug. 2004.
121. A. V. Malipatil, Y.-F. Huang, S. Andra, and K. Bennett, "Kernelized set-membership approach to nonlinear adaptive filtering," in *Proc. IEEE ICASSP*, 2005, pp. 149–152.
122. W. Liu and J. Príncipe, "Kernel affine projection algorithms," *EURASIP J. Adv. Signal Process.*, vol. 2008, pp. 1–12, 2008, Article ID 784292.

123. C. Richard, J. Bermudez, and P. Honeine, "Online prediction of time series data with kernels," *IEEE Trans. Signal Process.*, vol. 57, no. 3, pp. 1058–1067, Mar. 2009.
124. S. Theodoridis, K. Slavakis, and I. Yamada, "Adaptive learning in a world of projections: a unifying framework for linear and nonlinear classification and regression tasks," *IEEE Signal Processing Magazine*, Feb. 2011, to appear.
125. D. Blatt and A. O. Hero III, "Energy-based sensor network source localization via projection onto convex sets," *IEEE Trans. Signal Process.*, vol. 54, no. 9, pp. 3614–3619, Sep. 2006.
126. C. G. Lopes and A. H. Sayed, "Incremental adaptive strategies over distributed networks," *IEEE Trans. Signal Process.*, vol. 55, no. 8, pp. 4064–4077, Aug. 2007.
127. R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.
128. F. S. Cattivelli, C. G. Lopes, and A. H. Sayed, "Diffusion recursive least-squares for distributed estimation over adaptive networks," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1865–1877, May 2008.
129. C. G. Lopes and A. H. Sayed, "Diffusion least-mean squares over adaptive networks: Formulation and performance analysis," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3122–3136, July 2008.
130. A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Trans. Automatic Control*, vol. 54, no. 1, pp. 48–61, Jan. 2009.
131. C. G. Lopes and A. H. Sayed, "Diffusion adaptive networks with changing topologies," in *Proc. IEEE ICASSP*, 2008, pp. 3285–3288.
132. Y. Censor and T. Elfving, "A multiprojection algorithm using Bregman projections in a product space," *Numerical Algorithms*, vol. 8, no. 2, pp. 221–239, 1994.
133. C. Byrne, *Inherently parallel algorithms in feasibility and optimization and their applications*, chapter Bregman-Legendre multidistance projection algorithms for convex feasibility and optimization, pp. 87–100, Elsevier, Amsterdam, 2001.
134. C. Byrne, "Iterative oblique projection onto convex sets and the split feasibility problem," *Inverse Problems*, vol. 18, pp. 441–453, 2002.

135. Y. Censor, T. Elfving, N. Kopf, and T. Bortfeld, "The multiple-sets split feasibility problem and its applications for inverse problems," *Inverse Problems*, vol. 21, pp. 2071–2084, 2005.
136. M. Yukawa, K. Slavakis, and I. Yamada, "Multi-domain adaptive filtering by feasibility splitting," in *Proc. IEEE ICASSP*, 2010, pp. 3814–3817.