

Review



An Overview on the Applications of Matrix Theory in Wireless Communications and Signal Processing

Xu Wang and Erchin Serpedin *

Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843, USA; xu.wang@tamu.edu

* Correspondence: eserpedin@tamu.edu; Tel.: +1-979-458-2287; Fax: +1-979-862-4630

Academic Editors: Costas Busch and Bruno Carpentieri Received: 26 June 2016; Accepted: 5 October 2016; Published: 14 October 2016

Abstract: This paper overviews the key applications enabled by matrix theory in two major fields of interest in electrical engineering, namely wireless communications and signal processing. The paper focuses on the fundamental role played by matrices in modeling and optimization of wireless communication systems, and in detection, extraction and processing of the information embedded in signals. Among the major applications in wireless communications, the role of matrix representations and decompositions in characterizing multiple-input multiple-output (MIMO) and orthogonal frequency division multiplexing (OFDM) communication systems is described. In addition, this paper points out the important contribution made by matrices in solving signal estimation and detection problems. Special attention is given to the implementation of matrices in sensor array signal processing and the design of adaptive filters. Furthermore, the crucial role played by matrices in representing and processing digital images is depicted by several illustrative applications. This paper concludes with some applications of matrix theory in the area of compressive sensing of signals and by outlining a few open research problems for future study.

Keywords: matrix; signal processing; wireless communications; image processing; compressive sensing

1. Introduction

Matrix theory is widely used in many disciplines of modern engineering, including wireless communications, signal processing, control, biomedical engineering, etc. From a mathematical perspective, matrices are used both as a representation as well as an information processing tool to solve efficiently and reliably practical engineering problems. Since matrices are prevalent in engineering and have enabled remarkable solutions to problems of paramount importance, this paper overviews some of the major applications of matrix theory in wireless communications and signal processing. Our hope is that this article will help in renewing the interest of engineering community in linear algebra, and in particular in the tools offered by matrix theory, and its applications in engineering. Rather than describing details for all the statements and developments, our goal is instead to provide the readers the basic facts and key features of how matrix techniques were applied in some major engineering applications that played a fundamental role during the last two decades in the fields of signal processing and wireless communications.

The outline of this paper is summarized as follows. Section 2 describes how matrix techniques are exploited in the modern communication systems. Specifically, we discuss the key role played by the concept of singular value decomposition (SVD) in modeling and characterization of the multiple-input multiple-output (MIMO) or multiple antennas based communication systems, and the important role played by unitary matrix transformations such as Discrete Fourier Transform (DFT) in the orthogonal frequency-division multiplexing (OFDM) signaling scheme, which currently has been adopted in many wireless communications standards. Section 3 focuses on several illustrative uses of matrices in signal

estimation and detection applications. In particular, the usefulness of matrix decompositions and matrix inversion lemma in enabling online (recursive) estimation of parameters and development of adaptive filters is illustrated. A special attention is also given to the applications of matrices in sensor array signal processing. A more modern estimation method based on large dimensional random matrix theory is also briefly discussed to exemplify the recent trends in the utilization of matrices in signal processing and wireless communications. Section 4 describes the close relationship between matrices and images. Two image compression methods, which rely on matrix transformation and separation operations, are described. In Section 5, a recently proposed data compression approach termed as compressive (compressed) sensing is briefly examined from the perspective of the role played by matrices. Finally, Section 6 concludes the paper by outlining the key role played by matrices in solving some important problems from the fields of signal processing and wireless communications. Several open research problems that require more advanced linear algebra concepts and results are pointed out as well.

Throughout this paper, we will denote scalars by lowercase italic letters (*a*), vectors by bold lowercase italic letters (**a**), and matrices by bold uppercase Roman letters (**A**).

2. Matrix Theory in Modern Wireless Communication Systems

Nowadays, wireless communications have emerged as the fastest growing segment of the telecommunications industry. For example, cellular communication systems have exhibited an explosive growth over the last several decades and currently present over 6 billion users all over the world. Existing and future applications of wireless communications include multimedia Internet-based cell phones, high-quality real-time streaming of video and data signals, and transmission, reception and processing of all sorts of data collected by sensors and devices. The requirements imposed by these communication systems are subject to many technical challenges. In this section, two modern signaling schemes: MIMO and OFDM, which are currently adopted in many wireless communications standards to improve the performance of current wireless communication systems, are discussed. The presentation will focus mainly on how the matrix theory concepts and results can be applied to implement and assess the performance of MIMO and OFDM-based communications systems.

2.1. SVD for Modeling MIMO Channels

The channel capacity, defined as the maximum data rate at which data can be sent with arbitrarily low probability of error, was reported for the Gaussian channel by Shannon in his seminal paper [1]. Currently, achieving higher and higher data rates with tolerable error rates and fixed transmitter power is one of the most demanding targets for the current wireless communication systems. For example, the evolution from LTE to 4 G LTE-Advanced communication standard requires an increase in the downlink data rate from 150 Mbps to 1 Gbps and an uplink data rate increase from 75 Mbps to 500 Mps [2].

Theoretically, there are many ways to improve the channel capacity, like increasing the signal power, reducing the noise power, exploitation of space/time/frequency diversity, multiplexing, etc. However, due to the power limitation and unexpected properties of the receiver noise, diversity and multiplexing have been two popular design strategies on which researchers focused extensively during the last decade. Among these techniques, using multiple transmit and multiple receive antennas has been advocated as one of the most efficient methods to achieve the spatial multiplexing gain.

MIMO signaling is defined as the communication system with multiple antennas at both the transmitter and receiver, and it can be modeled as a system with multiple inputs and multiple outputs. A MIMO communication system with N_t transmit and N_r receive antennas is shown in Figure 1. This system can be depicted by the following matrix representation:

$$\begin{bmatrix} y_1 \\ \vdots \\ y_{N_r} \end{bmatrix} = \begin{bmatrix} h_{11} & \cdots & h_{1N_t} \\ \vdots & \ddots & \vdots \\ h_{N_r1} & \cdots & h_{N_rN_t} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_{N_t} \end{bmatrix} + \begin{bmatrix} \nu_1 \\ \vdots \\ \nu_{N_r} \end{bmatrix},$$
(1)

or simply by $\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{v}$, where \mathbf{x} represents the $N_t \times 1$ dimensional vector of transmitted data symbols, \mathbf{y} denotes the $N_r \times 1$ dimensional vector of received data symbols, $\mathbf{H} = [h_{ij}]$ stands for the transmit (propagation channel) matrix with the entry h_{ij} denoting the gain from transmitter j to receiver i, and \mathbf{v} is assumed to represent the white Gaussian noise vector. From (1), one can infer that



Figure 1. Multiple-input multiple-output (MIMO) channel.

In other words, the output symbol y_i is a linear combination of the input symbols, which makes it difficult to recover the transmitted data symbols. This unfavorable phenomenon is referred to as the intersymbol interference (ISI) and results in signal aliasing. However, it is shown next that the MIMO channel can be decomposed into independent signal paths by the help of SVD. This feature enables to design efficient decoding schemes while achieving the spatial multiplexing gain to improve the channel capacity. The SVD representation of channel matrix **H** takes the form:

$$\mathbf{H} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^H, \tag{2}$$

where $\mathbf{U} \in \mathbb{C}^{N_r \times N_r}$ and $\mathbf{V} \in \mathbb{C}^{N_t \times N_t}$ are unitary matrices, $\mathbf{\Sigma} \in \mathbb{R}^{N_r \times N_t}$ is a diagonal matrix with r nonzero diagonal elements, where $r \leq \min(N_r, N_t)$. The parallel decomposition of the MIMO channel into a series of parallel single-input single-output (SISO) channels can be obtained by pre-multiplying the transmitted signal with \mathbf{V} and post-multiplying the received signal with \mathbf{U}^H , as shown in Figure 2.



Figure 2. Parallel decomposition.

In the new coordinate system, the relationship between the input and output of channel takes the form:

$$\begin{split} \tilde{\mathbf{y}} &= \mathbf{U}^{H}(\mathbf{H}\mathbf{x} + \boldsymbol{\nu}) \\ &= \mathbf{U}^{H}(\mathbf{H}\mathbf{V}\tilde{\mathbf{x}} + \boldsymbol{\nu}) \\ &= \mathbf{U}^{H}(\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{H}\mathbf{V}\tilde{\mathbf{x}} + \boldsymbol{\nu}) \\ &= \boldsymbol{\Sigma}\tilde{\mathbf{x}} + \tilde{\boldsymbol{\nu}}. \end{split}$$
(3)

Assume $\Sigma = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_r, 0, \dots, 0)$. Thus, the MIMO channel can be decomposed into r independent SISO channels. Specifically, the *i*th SISO channel has a gain σ_i and a noise factor \tilde{v}_i , and assumes the representation

$$\tilde{y}_i = \sigma_i \tilde{x}_i + \tilde{\nu}_i.$$

This implies that all the efficient signaling and decoding strategies well investigated in the SISO context could be transferred mutatis-mutandis to the MIMO framework with a minimal effort thanks to SVD.

Moreover, by assuming that the channel state information, or equivalently, the channel matrix **H** is known at the receiver but not at the transmitter, Telatar [3] showed that the MIMO channel capacity can be expressed as

$$C = \sum_{i=1}^{r} \log_2 \left(1 + \frac{\sigma_i^2 \rho}{N_t} \right), \tag{4}$$

where ρ is the transmitted signal power and the additive receiver noise is assumed to have unit power. This result is in perfect accordance with (3) since the MIMO capacity in (4) can be interpreted as the summation of independent SISO channel capacities. Under the assumption that the entries of **H** follow a zero-mean spatially white (ZMSW) model, the law of large numbers implies that $\lim_{N_t \to \infty} \frac{1}{N_t} \mathbf{H} \mathbf{H}^H = \mathbf{I}_{N_r}$, and the capacity can be expressed asymptotically [4] in the presence of a large number of transmit/receive antennas as

$$C = N_r \log_2(1+\rho).$$

This implies that as $N = \min(N_r, N_t)$ grows large, the channel capacity increases linearly with N [5]. This fact as well as some other results for MIMO capacity analysis [6,7] makes MIMO signaling superior to SISO signaling and it also explains the reason why currently MIMO techniques are so appealing in a broad range of applications such as wireless communications, radar, sonar, geo-physical (seismic) signal processing, wireless sensor networks, and antenna array processing. The proposed framework for determining the capacity of MIMO linear time-invariant channels could be further extended to determine the capacity of memoryless nonlinear MIMO channels subject to a power constraint [8]. Reference [8] re-formulates this problem within a functional framework and employs variational calculus techniques to determine the capacity-achieving input distribution of memoryless nonlinear MIMO channels. As a special case, the capacity-achieving input distribution for the linear MIMO channel is re-established in [8].

2.2. Matrix Representation of OFDM

Orthogonal frequency-division multiplexing (OFDM) is another core technology employed by modern wireless communication systems. It is the basic signaling technology in the recently proposed long-term evolution (LTE) wireless communication standard and has been already adopted as the basic signaling scheme in many communication standards such as Wireless Local Area Network (WLAN), Wireless Metropolitan Area Network (WMAN), Wireless Regional Area Network (WRAN), Wireless Personal Area Network (WPAN), Digital Audio Broadcasting (DAB), and Digital Video Broadcasting (DVB). OFDM represents a multicarrier system which divides the whole spectrum into several narrow frequency bands, and the data is transmitted in a parallel manner along all these narrow frequency bands. The basic signaling mechanism of OFDM [4] is depicted graphically in Figure 3.



Figure 3. Orthogonal frequency-division multiplexing (OFDM).

To simplify the exposition, we have represented all the operations between channel input vector **x** and output vector **y** as a system with transfer matrix **H**, such that $\mathbf{y} = \mathbf{H}\mathbf{x} + \nu$, where ν stands for the additive noise vector. OFDM implements an inverse discrete Fourier transform (IDFT) at the transmitter side and a discrete Fourier transform (DFT) at the receiver side. However, in practice, the fast Fourier transform (FFT) algorithm is widely used to compute DFT due to its fast implementation and computational efficiency. This is the reason why the processing blocks are marked in Figure 3 with FFT and IFFT instead of DFT and IDFT.

Given x[n], $n = 0, 1, \dots, N-1$, the *N*-point DFT of x[n] is defined as

$$X[k] = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi nk}{N}}, \ k = 0, 1, \cdots, N-1$$

The sequence x[n] can be recovered back by means of the IDFT as

$$x[n] = rac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X[k] e^{jrac{2\pi nk}{N}}, \ n = 0, 1, \cdots, N-1.$$

It is easy to show that the DFT operation can be represented as a matrix multiplication

$$\mathbf{X} = \mathbf{W}\mathbf{x}$$

where $\mathbf{X} = [X[0], X[1], \dots, X[N-1]]^T$, $\mathbf{x} = [x[0], x[1], \dots, x[N-1]]^T$, and the DFT matrix \mathbf{W} is a unitary matrix expressed as

$$\mathbf{W} = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & 1 & \cdots & 1\\ 1 & w & w^2 & \cdots & w^{N-1}\\ 1 & w^2 & w^4 & \cdots & w^{2(N-1)}\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ 1 & w^{N-1} & w^{2(N-1)} & \cdots & w^{(N-1)^2} \end{bmatrix}$$
(5)

with $w = e^{-j\frac{2\pi}{N}}$. Additionally, the IDFT operation can be expressed as

$$\mathbf{x} = \mathbf{W}^{-1}\mathbf{X} = \mathbf{W}^H\mathbf{X}$$

since $\mathbf{W}^{-1} = \mathbf{W}^{H}$ is due to the fact that \mathbf{W} is a unitary matrix.

Matrix Analysis of OFDM

Consider a discrete channel **H** with coefficients h[n], $n = 0, 1, \dots, M$. The cyclic prefix [4] is implemented to generate the new sequence $\tilde{x}[n]$, $n = -M, -M + 1, \dots, N - 1$, where $\tilde{x}[n] = x[n \mod N]$. Specifically,

$$\tilde{x}[-M], \cdots, \tilde{x}[-1], \tilde{x}[0], \cdots, \tilde{x}[N-1] = x[N-M], \cdots, x[N-1], x[0], \cdots, x[N-1].$$
 (6)

This yields the output $y[n] = \tilde{x}[n] * h[n] + v[n]$. Introducing the simplified notations $\tilde{x}[n]$ as \tilde{x}_n , h[n] as h_n , etc., it turns out that the relation between the input and output of the channel can be expressed as

$$\begin{bmatrix} y_{N-1} \\ y_{N-2} \\ \vdots \\ y_{0} \end{bmatrix} = \begin{bmatrix} h_{0} & h_{1} & \cdots & h_{M} & 0 & \cdots & 0 \\ 0 & h_{0} & \cdots & h_{M-1} & h_{M} & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & h_{0} & \cdots & h_{M-1} & h_{M} \end{bmatrix} \begin{bmatrix} \tilde{x}_{N-1} \\ \vdots \\ \tilde{x}_{0} \\ \tilde{x}_{-1} \\ \vdots \\ \tilde{x}_{-M} \end{bmatrix} + \begin{bmatrix} \nu_{N-1} \\ \nu_{N-2} \\ \vdots \\ \nu_{0} \end{bmatrix}.$$
(7)

Plugging (6) into (7) leads to

$$\begin{bmatrix} y_{N-1} \\ y_{N-2} \\ \vdots \\ y_0 \end{bmatrix} = \begin{bmatrix} h_0 & h_1 & \cdots & h_M & 0 & \cdots & 0 \\ 0 & h_0 & \cdots & h_{M-1} & h_M & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & h_0 & \cdots & h_{M-1} & h_M \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ h_2 & h_3 & \cdots & h_{M-2} & \cdots & h_0 & h_1 \\ h_1 & h_2 & \cdots & h_{M-1} & \cdots & 0 & h_0 \end{bmatrix} \begin{bmatrix} x_{N-1} \\ x_{N-2} \\ \vdots \\ x_0 \end{bmatrix} + \begin{bmatrix} v_{N-1} \\ v_{N-2} \\ \vdots \\ v_0 \end{bmatrix}, \quad (8)$$

which can be written compactly as

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \boldsymbol{\nu}.$$

It turns out that **H** is a circulant matrix with the generator vector $[h_0, 0, \dots, 0, h_M, \dots, h_1]^T$. According to the properties of circulant matrices [9], **H** admits the eigenvalue decomposition

$$\mathbf{H} = \mathbf{U} \mathbf{\Lambda} \mathbf{U}^{H}$$
,

where Λ is a diagonal matrix whose diagonal elements stand for the eigenvalues of **H** and **U** is a unitary matrix whose rows correspond to the columns of DFT matrix **W** in (5). In other words, $\mathbf{U} = \mathbf{W}^{H}$. Thus, one can infer the following relations:

$$Y = Wy$$

= W(Hx + ν)
= W(HW^HX + ν) (9)
= WUAU^HW^HX + W ν
= AX + $\tilde{\nu}$.

It turns out that by exploiting DFT and IDFT operations and inserting the cyclic prefix, OFDM transforms a frequency-selective channel into *N* orthogonal flat-fading subchannels without ISI, which helps to reduce the complexity of the receiver decoder. Due to its advantages in combining the energy of all the propagation paths of the channel and enabling a reduced complexity equalization, OFDM has been adopted in numerous communications standards (ADSL, WLAN, LTE, etc.) and has been also used in conjunction with the MIMO technique in what it is currently referred to as the MIMO-OFDM technology [10].

3. Matrix Applications in Signal Estimation Theory

Estimation theory is a branch of statistics that aims to estimating the values of parameters based on measured or empirical data. Applications of parameter estimation are prevalent in the area of wireless communications and signal processing. In this section, we briefly discuss how matrix theory can be applied to simplify signal estimation problems, reduce computational complexity, develop adaptive (recursive) filters and solve ill-conditioned estimation problems.

3.1. Cholesky Decomposition for Whitening the Noise

The Cholesky decomposition [11] of a Hermitian positive-semidefinite matrix **A** admits the representation:

$$\mathbf{A} = \mathbf{D}\mathbf{D}^{H}$$
,

where **D** is a lower triangular matrix with nonnegative diagonal entries. Every Hermitian positive-definite matrix has a unique Cholesky decomposition.

Cholesky decomposition is mainly used in whitening the noise for signal estimation and detection applications. Consider a simple vector parameter estimation problem expressed as [12]:

$$\mathbf{x} = \boldsymbol{\theta} + \boldsymbol{\nu},\tag{10}$$

where $\mathbf{x} \in \mathbb{R}^N$ stands for the observed data, $\nu \in \mathbb{R}^N$ denotes the additive noise with Gaussian distribution $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ ($\mathbf{\Sigma}$ is not diagonal) and $\boldsymbol{\theta}$ represents the unknown vector to be estimated. Knowledge of the probability density function (PDF) $p(\mathbf{x}; \boldsymbol{\theta})$ lies at the basis for several very well-known and used estimators such as the minimum variance unbiased estimator (MVUE) and the maximum likelihood estimator (MLE) [12]. In the present application, $p(\mathbf{x}; \boldsymbol{\theta})$ stands for a multivariate Gaussian PDF of the form:

$$p(\mathbf{x};\boldsymbol{\theta}) = (2\pi)^{-\frac{N}{2}} |\boldsymbol{\Sigma}|^{-\frac{1}{2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\theta})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\theta})},\tag{11}$$

where $|\cdot|$ denotes the matrix determinant.

Interestingly, if we assume the covariance matrix Σ to be full rank and consider its Cholesky decomposition, i.e., $\Sigma = DD^T$, and multiply with the Cholesky factor D^{-1} both sides of (10), we obtain

$$\mathbf{D}^{-1}\mathbf{x} = \mathbf{D}^{-1}\boldsymbol{\theta} + \mathbf{D}^{-1}\boldsymbol{\nu},$$

$$\tilde{\mathbf{x}} = \tilde{\boldsymbol{\theta}} + \tilde{\boldsymbol{\nu}},$$
 (12)

where the covariance matrix of \tilde{v} is now given by

$$E(\tilde{\boldsymbol{v}}\tilde{\boldsymbol{v}}^{T}) = E(\mathbf{D}^{-1}\boldsymbol{v}\boldsymbol{v}^{T}\mathbf{D}^{-T}) = \mathbf{D}^{-1}E(\boldsymbol{v}\boldsymbol{v}^{T})\mathbf{D}^{-T} = \mathbf{D}^{-1}\boldsymbol{\Sigma}\mathbf{D}^{-T} = \mathbf{D}^{-1}\mathbf{D}\mathbf{D}^{T}\mathbf{D}^{-T} = \mathbf{I}.$$

This transformation whitens the noise such that the new PDF takes the form:

$$p(\tilde{\mathbf{x}}; \tilde{\boldsymbol{\theta}}) = (2\pi)^{-\frac{N}{2}} e^{-\frac{1}{2} (\tilde{\mathbf{x}} - \tilde{\boldsymbol{\theta}})^T (\tilde{\mathbf{x}} - \tilde{\boldsymbol{\theta}})}$$

$$= (2\pi)^{-\frac{N}{2}} e^{-\frac{1}{2} \sum_{i=0}^{N-1} (\tilde{x}_i - \tilde{\theta}_i)^2}$$

$$= \prod_{i=0}^{N-1} p(\tilde{x}_i; \tilde{\theta}_i).$$
 (13)

Using representation (13), where the noise is white (not correlated anymore), the estimation task can be performed by processing independently each $\tilde{\theta}_i$ on a one-by-one basis, which, in general, for many applications, leads to a closed form solution. The estimate θ can be obtained by $\theta = \mathbf{D}\tilde{\theta}$. In contrast, in (11), we must process the entire vector \mathbf{x} due to the dependence caused by the colored noise. Thus, the whitening method significantly simplifies the estimation problem when the noise is colored.

3.2. SVD for Least-Squares Based Estimation

Section 2.1 pointed out the important role played by SVD in modeling, implementation and characterization of MIMO channels. In this new section, we will illustrate the benefits offered by SVD in solving least-squares (LS) estimation problems [13].

Consider a linear estimation problem defined as follows:

$$\mathbf{x} = \mathbf{A}\boldsymbol{\theta} + \boldsymbol{\nu}$$

where $\mathbf{x} \in \mathbb{R}^N$ is the observed data, $\boldsymbol{\nu} \in \mathbb{R}^N$ stands for the additive noise, $\mathbf{A} \in \mathbb{R}^{N \times P}$ is the given linear interaction matrix, and $\boldsymbol{\theta} \in \mathbb{R}^P$ is the unknown vector to be estimated.

The LS estimator is defined as

$$\hat{\theta}_{\rm LS} = \arg\min_{\theta} ||\mathbf{A}\theta - \mathbf{x}||_2^2, \tag{14}$$

where $|| \cdot ||_2$ stands for the Euclidean norm. If **A** is full column rank, i.e., rank(**A**) = *P* and *N* ≥ *P*, it admits the closed form solution

$$\hat{\boldsymbol{\theta}}_{\text{LS}} = (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \mathbf{x}.$$
(15)

However, in the case of N < P, matrix $\mathbf{A}^H \mathbf{A}$ is rank deficient and this solution is invalid. In this case, an infinite number of minimizers exist, and the set of these solutions is denoted by Θ . A unique solution for the LS problem can be enforced by selecting the minimum l_2 (Euclidean) norm solution from the set Θ . Thus, one should add this constraint to (14):

$$\boldsymbol{\theta}_{\text{LS}} = \arg\min ||\boldsymbol{\theta}_{\text{LS}}||_2^2$$

$$\boldsymbol{\hat{\theta}}_{\text{LS}} \in \boldsymbol{\Theta}.$$
 (16)

The new optimization problem defined by (14) and (16) can be solved by means of SVD. In particular, due to the SVD of **A**, i.e., $\mathbf{A} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^{H}$, and the fact that multiplying a vector with a unitary matrix does not change the Euclidean norm of the vector, it turns out that

$$||\mathbf{A}\boldsymbol{\theta} - \mathbf{x}||_{2}^{2} = ||\mathbf{U}^{H}(\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{H}\boldsymbol{\theta} - \mathbf{x})||_{2}^{2} = ||\boldsymbol{\Sigma}\mathbf{V}^{H}\boldsymbol{\theta} - \mathbf{U}^{H}\mathbf{x}||_{2}^{2} = ||\boldsymbol{\Sigma}\mathbf{y} - \mathbf{b}||_{2}^{2},$$
(17)

where $\mathbf{V}^{H}\boldsymbol{\theta} = \mathbf{y}$ and $\mathbf{U}^{H}\mathbf{x} = \mathbf{b}$. Since N < P, the singular matrix $\boldsymbol{\Sigma}$ can be represented as

Algorithms 2016, 9, 68

$$\Sigma = \left[egin{array}{cc} \Sigma_1 & 0 \ 0 & 0 \end{array}
ight].$$

Moreover, **b** and **y** can also be decomposed as $\mathbf{b} = [\mathbf{b}_1^T \mathbf{b}_2^T]^T$ and $\mathbf{y} = [\mathbf{y}_1^T \mathbf{y}_2^T]^T$, respectively. Hence, (17) resumes to

$$||\mathbf{A}\boldsymbol{\theta} - \mathbf{x}||_{2}^{2} = ||\boldsymbol{\Sigma}_{1}\mathbf{y}_{1} - \mathbf{b}_{1}||_{2}^{2} + ||\mathbf{b}_{2}||_{2}^{2}.$$
 (18)

Therefore, the minimum-norm LS-solution that minimizes $||\mathbf{A}\theta - \mathbf{x}||_2^2$ is given by $\mathbf{y}_1 = \mathbf{\Sigma}_1^{-1}\mathbf{b}_1$ and $\mathbf{y}_2 = \mathbf{0}$. Alternatively,

$$\mathbf{y} = \begin{bmatrix} \boldsymbol{\Sigma}_1^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix} = \boldsymbol{\Sigma}^{\dagger} \mathbf{b}_1$$

where the block matrix is denoted as Σ^{\dagger} . Finally, the LS-estimate of θ can be obtained by

$$\theta_{\rm LS} = \mathbf{V}\mathbf{y} = \mathbf{V}\mathbf{\Sigma}^{\dagger}\mathbf{b} = \mathbf{V}\mathbf{\Sigma}^{\dagger}\mathbf{U}^{H}\mathbf{x}.$$

3.3. Matrix Inverse Lemma for Derivation of the Recursive Least-Squares Filter

In many signal processing applications, the data is generated by sampling a continuous-time signal. In this way, more data becomes available as the time progresses. One can either wait for all the available data or process the data iteratively in time. The latter approach requires a recursive (online) approach to update the filter taps. Next, the key role played by the matrix inverse lemma (Woodbury matrix identity) in deriving the least-squares (RLS) filter will be presented.

The matrix inverse lemma states that [12]:

$$(\mathbf{A} + \mathbf{x}\mathbf{y}^{H})^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{x}\mathbf{y}^{H}\mathbf{A}^{-1}}{1 + \mathbf{y}^{H}\mathbf{A}^{-1}\mathbf{x}'}$$
(19)

where matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$ is assumed invertible, and $\mathbf{x}, \mathbf{y} \in \mathbb{R}^{N}$.

Consider now the linear time-invariant (LTI) system shown in Figure 4. The problem is to adapt the *m*-tap FIR filter coefficients **h** so that its output y(n) follows the reference signal x(n). The received signal y(n) is expressed as:

$$y(n) = \sum_{i=0}^{m-1} h_n(i) a(n-i) = \mathbf{h}_n^H \mathbf{a}_n,$$
(20)

where $\mathbf{a}_n = [a(n), a(n-1), \dots, a(n-m+1)]^H$ is the vector containing the *m* most recent samples of input signal a(n). Our goal is to estimate the taps of filter **h** at each time slot *n* using a least-squares estimation approach. Specifically, the problem resumes to:

$$\hat{\mathbf{h}}_n = \arg\min_{\mathbf{h}_n} ||\mathbf{A}_n \mathbf{h}_n - \mathbf{x}_n||_2^2, \ n = 1, 2, \cdots, N,$$
(21)

where

$$\mathbf{A}_{n} = \begin{bmatrix} a(1) & a(0) & \cdots & a(-m+2) \\ a(2) & a(1) & \cdots & a(-m+1) \\ \vdots & \vdots & \ddots & \vdots \\ a(n-1) & a(n-2) & \cdots & a(n-m) \\ a(n) & a(n-1) & \cdots & a(n-m+1) \end{bmatrix} = \begin{bmatrix} \mathbf{a}_{1}^{H} \\ \mathbf{a}_{2}^{H} \\ \vdots \\ \mathbf{a}_{n-1}^{H} \\ \mathbf{a}_{n}^{H} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{n-1} \\ \mathbf{a}_{n}^{H} \end{bmatrix},$$

and

$$\mathbf{h}_{n} = \begin{bmatrix} h_{n}(0) \\ h_{n}(1) \\ \vdots \\ h_{n}(m-2) \\ h_{n}(m-1) \end{bmatrix} \text{ and } \mathbf{x}_{n} = \begin{bmatrix} x(1) \\ x(2) \\ \vdots \\ x(n-1) \\ x(n) \end{bmatrix} = \begin{bmatrix} \mathbf{x}_{n-1} \\ x(n) \end{bmatrix}.$$

As time evolves the number of rows of matrix \mathbf{A}_n keeps increasing, and to avoid repeatedly calculating from scratch the least-squares estimate of $\hat{\mathbf{h}}_n$ for each instant of time *n*, expressing $\hat{\mathbf{h}}_n$ in terms of $\hat{\mathbf{h}}_{n-1}$ helps to reduce the computational burden. Indeed, such an approach exhibits the huge benefit of reducing the computational complexity incurred by evaluating the taps of the filter at each time instant.



Figure 4. RLS filter.

As we described earlier in (14) and (15), the problem (21) admits the solution

$$\hat{\mathbf{h}}_n = (\mathbf{A}_n^H \mathbf{A}_n)^{-1} \mathbf{A}_n^H \mathbf{x}_n.$$

Setting $\mathbf{R}_n = \mathbf{A}_n^H \mathbf{A}_n$ and $\mathbf{p}_n = \mathbf{A}_n^H \mathbf{x}_n$ leads to

$$\hat{\mathbf{h}}_n = \mathbf{R}_n^{-1} \mathbf{p}_n. \tag{22}$$

Moreover, since

$$\mathbf{R}_n = \mathbf{A}_n^H \mathbf{A}_n = \sum_{i=1}^n \mathbf{a}_i \mathbf{a}_i^H = \sum_{i=1}^{n-1} \mathbf{a}_i \mathbf{a}_i^H + \mathbf{a}_n \mathbf{a}_n^H = \mathbf{R}_{n-1} + \mathbf{a}_n \mathbf{a}_n^H,$$

(22) can be expressed as

$$\hat{\mathbf{h}}_n = (\mathbf{R}_{n-1} + \mathbf{a}_n \mathbf{a}_n^H)^{-1} \mathbf{p}_n.$$
(23)

Applying the matrix inverse lemma (19) to the inverse of the matrix contained in Equation (23) yields:

$$\mathbf{R}_{n}^{-1} = (\mathbf{R}_{n-1}^{-1} + \mathbf{a}_{n}\mathbf{a}_{n}^{H})^{-1} = \mathbf{R}_{n-1}^{-1} - \frac{\mathbf{R}_{n-1}^{-1}\mathbf{a}_{n}\mathbf{a}_{n}^{H}\mathbf{R}_{n-1}^{-1}}{1 + \mathbf{a}_{n}^{H}\mathbf{R}_{n-1}^{-1}\mathbf{a}_{n}}.$$
(24)

Furthermore, defining $\mathbf{k}_n = \frac{\mathbf{R}_{n-1}^{-1}\mathbf{a}_n}{1+\mathbf{a}_n^H\mathbf{R}_{n-1}^{-1}\mathbf{a}_n}$ and $\boldsymbol{\Sigma}_n = \mathbf{R}_n^{-1}$, it follows that

$$\boldsymbol{\Sigma}_n = \boldsymbol{\Sigma}_{n-1} - \mathbf{k}_n \mathbf{a}_n^H \boldsymbol{\Sigma}_{n-1}, \tag{25}$$

and

$$\Sigma_{n} \mathbf{a}_{n} = \Sigma_{n-1} \mathbf{a}_{n} - \mathbf{k}_{n} \mathbf{a}_{n}^{H} \Sigma_{n-1} \mathbf{a}_{n}$$

= $\mathbf{k}_{n} (1 + \mathbf{a}_{n}^{H} \Sigma_{n-1} \mathbf{a}_{n}) - \mathbf{k}_{n} \mathbf{a}_{n}^{H} \Sigma_{n-1} \mathbf{a}_{n}$ (26)
= \mathbf{k}_{n} .

Plugging (25) and (26) into (22) results in

$$\begin{aligned} \hat{\mathbf{h}}_{n} &= \boldsymbol{\Sigma}_{n} \mathbf{p}_{n} \\ &= \boldsymbol{\Sigma}_{n} [\mathbf{p}_{n-1} + \mathbf{a}_{n} x(n)] \\ &= \boldsymbol{\Sigma}_{n} \mathbf{p}_{n-1} + \boldsymbol{\Sigma}_{n} \mathbf{a}_{n} x(n) \\ &= (\boldsymbol{\Sigma}_{n-1} - \mathbf{k}_{n} \mathbf{a}_{n}^{H} \boldsymbol{\Sigma}_{n-1}) \mathbf{p}_{n-1} + \mathbf{k}_{n} x(n) \\ &= \hat{\mathbf{h}}_{n-1} - \mathbf{k}_{n} \mathbf{a}_{n}^{H} \hat{\mathbf{h}}_{n-1} + \mathbf{k}_{n} x(n) \\ &= \hat{\mathbf{h}}_{n-1} + \mathbf{k}_{n} [x(n) - \mathbf{a}_{n}^{H} \hat{\mathbf{h}}_{n-1}] \\ &= \hat{\mathbf{h}}_{n-1} + \epsilon(n) \mathbf{k}_{n}, \end{aligned}$$
(27)

where $\epsilon(n) = x(n) - \mathbf{a}_n^H \hat{\mathbf{h}}_{n-1}$ and the second equality comes from the fact that

$$\mathbf{p}_n = \mathbf{A}_n^H \mathbf{x}_n = \mathbf{A}_{n-1}^H \mathbf{x}_{n-1} + \mathbf{a}_n x(n) = \mathbf{p}_{n-1} + \mathbf{a}_n x(n).$$

In a nutshell, the RLS algorithm can be summarized as follows:

- (1) Choose a large positive number α , and initialize $\Sigma_0 = \alpha \mathbf{I}$ and $\hat{\mathbf{h}}_0 = \mathbf{0}$.
- (2) For $n = 1, 2, \dots, N$, sequentially update

$$\mathbf{k}_{n} = \frac{\boldsymbol{\Sigma}_{n-1}\mathbf{a}_{n}}{1 + \mathbf{a}_{n}^{H}\boldsymbol{\Sigma}_{n-1}\mathbf{a}_{n}},$$

$$\boldsymbol{\epsilon}(n) = \boldsymbol{x}(n) - \mathbf{a}_{n}^{H}\hat{\mathbf{h}}_{n-1},$$

$$\hat{\mathbf{h}}_{n} = \hat{\mathbf{h}}_{n-1} + \boldsymbol{\epsilon}(n)\mathbf{k}_{n},$$

$$\boldsymbol{\Sigma}_{n} = \boldsymbol{\Sigma}_{n-1} - \mathbf{k}_{n}\mathbf{a}_{n}^{H}\boldsymbol{\Sigma}_{n-1}.$$

As the algorithm processes, the error made by approximating $\Sigma_0 = \alpha I$ and $\hat{h}_0 = 0$ becomes less and less significant [12].

In some cases, a weighted least squares estimation error is used as the minimizer by multiplying the error terms with so-called "forgetting factors", which are intended to reduce the influence of old data [14]. The RLS algorithm has emerged as a powerful tool for adaptive filtering, prediction and system identification applications. Despite the fundamental role played by the RLS algorithm in the adaptive signal processing field, it encounters numerical problems when implemented in a finite precision environment, and especially in fixed-point arithmetic DSP processors [15]. The round-off error propagation, defined as the influence of subsequent computations when a single perturbation occurs at a random iteration, represents one of the major drawbacks of the RLS algorithm in a finite precision implementation and has been well studied in literature [16–18]. A popular method to improve the accuracy and stability of RLS algorithm is to implement a variable forgetting factor (VFF) [19–21]. Other modified RLS methods can be found in [22–25] and the references therein.

3.4. Matrix Theory in Sensor Array Signal Processing

Sensor array signal processing focuses mainly on signal enumeration and source location applications, and presents a huge importance in many domains such as radar, sonar and underwater surveillance. The classic problem in sensor array signal processing is to detect and locate the radiating sources given the temporal and spatial information collected from the sensors. The goal of this section is to introduce the basic framework of sensor array signal processing and illustrate the fundamental role of matrix theory in representing and addressing the detection and estimation problems associated with the radiating sources.

First consider the one-source-one-sensor scenario shown in Figure 5. It is proved in [26] that under the far-field and narrowband assumptions, the output of a sensor can be modeled as follows:

$$x(t) = g(\theta)s(t)e^{-j\mathbf{r}^{T}\mathbf{k}},$$

where $g(\theta)$ is a function of the direction of arrival (DOA) θ , and it represents the frequency response of the sensor. Variable s(t) represents the complex envelope of the source signal, and **r** and **k** stand for the radius vector and the wave vector, respectively. The magnitude of **k** is defined as $|\mathbf{k}| = \alpha = 2\pi/\lambda$, where λ is the wavelength. As shown in Figure 5, in a two-dimensional plane, we have $\mathbf{k} = \alpha(\cos \theta, \sin \theta)^T$ and $\mathbf{r} = (x, y)^T$. Thus, the output of the sensor can be simplified to

$$x(t) = g(\theta)e^{-j\alpha(x\cos\theta + y\sin\theta)}s(t) = a(\theta)s(t),$$
(28)

where $a(\theta) = g(\theta)e^{-j\alpha(x\cos\theta + y\sin\theta)}$.



Figure 5. One-sensor one-source array.

For the one-source-L-sensor case, the array output vector can be expressed as

$$\mathbf{x}(t) = \mathbf{a}(\theta)s(t),\tag{29}$$

where $\mathbf{x}(t) = [x_1(t), x_2(t), \dots, x_L(t)]^T$ denotes the output of each sensor and $\mathbf{a}(\theta) = [a_1(\theta), a_2(\theta), \dots, a_L(\theta)]^T$ stands for the steering vector and captures the sensor array geometry information. Two common array geometries, termed as the uniform linear array (ULA) and uniform circular array (UCA), are depicted in Figure 6. For ULA, $\mathbf{r}_l = [(l-1)d, 0]^T$, and for UCA, $\mathbf{r}_l = R[\cos(2\pi(l-1)/L), \sin(2\pi(l-1)/L)]^T$. Additionally, if we assume that all the sensors have the same frequency response, $g_1(\theta) = \dots = g_L(\theta) = g(\theta)$, the ULA steering vector takes the form:

$$\mathbf{a}_{\text{ULA}}(\theta) = g(\theta) [1, e^{-j\alpha d\cos\theta}, \cdots, e^{-j\alpha(L-1)d\cos\theta}]^T,$$
(30)

and the UCA admits the form:

$$\mathbf{a}_{\text{UCA}}(\theta) = g(\theta) [e^{-j\alpha R \cos \theta}, e^{-j\alpha R [\cos(2\pi/L)\cos\theta + \sin(2\pi/L)\sin\theta]}, \cdots,$$

$$e^{-j\alpha R [\cos(2\pi(L-1)/L)\cos\theta + \sin(2\pi(L-1)/L)\sin\theta]} T$$
(31)



Figure 6. Array geometries.

Furthermore, if *N* source signals with DOA θ_n , $n = 1, 2, \dots, N$, are superimposed on an *L*-dimensional array (*N*-source-*L*-sensor), the array output vector can be expressed as

$$\mathbf{x}(t) = \sum_{n=1}^{N} \mathbf{a}(\theta_n) s_n(t),$$

or compactly as

$$\mathbf{x}(t) = \mathbf{A}(\boldsymbol{\theta})\mathbf{s}(t),\tag{32}$$

where $\mathbf{x}(t) = [x_1(t), \dots, x_L(t)]^T \in \mathbb{R}^L$, $\mathbf{A}(\boldsymbol{\theta}) = [\mathbf{a}(\theta_1), \dots, \mathbf{a}(\theta_N)] \in \mathbb{R}^{L \times N}$ and $\mathbf{s}(t) = [s_1(t), \dots, s_N(t)]^T \in \mathbb{R}^N$. In the presence of an additive noise $\boldsymbol{\nu}(t)$, the sensor array equation can be expressed as:

$$\mathbf{x}(t) = \mathbf{A}(\boldsymbol{\theta})\mathbf{s}(t) + \boldsymbol{\nu}(t). \tag{33}$$

The central goal of array signal processing is to locate the source signals, or equivalently, to estimate the DOAs θ given the observed data $\mathbf{x}(t)$ at the sensors. From (33), we can see that the input and the output are just related by a steering matrix $\mathbf{A}(\theta)$ and the problem formulation is simplified using a steering matrix representation. Moreover, the estimation problem may also be addressed efficiently by exploring the special structure of the steering matrix. For instance, under the ULA geometry, we aim at estimating $\boldsymbol{\theta} = [\theta_1, \dots, \theta_N]^T$ subject to

$$\begin{bmatrix} x_{1}(t) \\ x_{2}(t) \\ \vdots \\ x_{L}(t) \end{bmatrix} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ e^{-j\alpha d\cos\theta_{1}} & e^{-j\alpha d\cos\theta_{2}} & \cdots & e^{-j\alpha d\cos\theta_{N}} \\ \vdots & \vdots & \ddots & \vdots \\ e^{-j\alpha(L-1)d\cos\theta_{1}} & e^{-j\alpha(L-1)d\cos\theta_{2}} & \cdots & e^{-j\alpha(L-1)d\cos\theta_{N}} \end{bmatrix} \begin{bmatrix} s_{1}(t) \\ s_{2}(t) \\ \vdots \\ s_{N}(t) \end{bmatrix} + \begin{bmatrix} \nu_{1}(t) \\ \nu_{2}(t) \\ \vdots \\ \nu_{L}(t) \end{bmatrix}.$$
(34)

It can be observed that the steering matrix $\mathbf{A}(\theta)$ for the ULA geometry is a Vandermonde matrix. Thus, the special properties of Vandermonde matrices can be exploited to develop efficient algorithms to estimate the DOA. For instance, the estimation of signal parameters by rotation invariant techniques (ESPRIT) [27] utilizes the so-called shifting property of Vandermonde matrices, which can be easily understood by considering two submatrices \mathbf{A}_1 and \mathbf{A}_2 of $\mathbf{A}(\theta)$, obtained by removing the first and last rows of the steering matrix $\mathbf{A}(\theta)$, respectively. Due to the structure of Vandermonde matrix $\mathbf{A}(\theta)$ in (34), the shifting property manifests through the fact that matrices \mathbf{A}_1 and \mathbf{A}_2 are related by means of shift matrix Φ via $\mathbf{A}_2 = \mathbf{A}_1 \Phi$, where Φ is a diagonal matrix with diagonal elements $[e^{-j\alpha d \cos \theta_1}, e^{-j\alpha d \cos \theta_2}, \dots, e^{-j\alpha d \cos \theta_N}]$. Therefore, the DOA estimation problem is simplified to that of finding Φ , which can be addressed by exploring the eigenvalue decomposition of the sensor array covariance matrix. The interested reader is referred to [14,26,28] for more details about ESPRIT and other matrix factorization based algorithms for sensor array estimation.

3.5. Random Matrix Theory in Signal Estimation and Detection

Suppose we are given a parameter to be estimated with a "population size" of N and n observations of the population. Many estimation methods proposed in literature rely on the asymptotic behavior that the number of observations is much larger than the population size, i.e., $n/N \rightarrow \infty$. However, in modern signal processing, this assumption sometimes is not met in practice. For instance, in biological networks, the estimate generally consists of a large number of DNAs and the population size may also grow as the number of observations increases. Moreover, in radar applications, a large number of sensors are used to detect a fast moving object with observations available only during a short time period. Therefore, under such a set-up, the proposed estimators may perform poorly if the condition $n/N \rightarrow \infty$ is not satisfied. Recently, a novel estimation method based on the large dimensional random matrix theory has been proposed to cope with this situation. The random matrix theory originated from the work of Wigner [29] in 1955 and was later generalized to a large extent by Marcenko and Pastur [30] in 1967. Recently, a considerable amount of work based on random matrix theory has emerged in the literature of signal processing, wireless communications and information theory.

Consider a simple example from [31], where $\mathbf{x} \in \mathbb{C}^N$ is a complex random vector with zero mean and covariance matrix **R**. Given *n* observations of **x**, we will consider a classic estimator of **R** in the form of the sample covariance matrix $\mathbf{R}_n = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i \mathbf{x}_i^H$. Because of the law of large numbers, $||\mathbf{R}_n - \mathbf{R}||_2 \xrightarrow{\text{a.s.}} 0$ as $n \to \infty$, where $|| \cdot ||_2$ denotes the spectral norm of a matrix, i.e., the absolute value of the largest eigenvalue if the matrix is Hermitian, and $\xrightarrow{a.s.}$ stands for almost sure convergence. By taking the special case where $\mathbf{R} = \mathbf{I}_{N_{\ell}}$ it was shown in [31] that if $n \to \infty$ but n/N does not tend to zero, i.e., n/N = 1/2, the spectral norm of $\mathbf{R}_n - \mathbf{I}_N$ is at least greater than 1. The problem lies in the fact that the random eigenvalue distribution $F^{\mathbf{R}_n}$ of \mathbf{R}_n does not converge to a Dirac mass measure at 1 [31]. However, it is observed that in many scenarios, it converges to a limit distribution as $N, n \to \infty, N/n = c$. In the special case where $\mathbf{R} = \mathbf{I}_N$, the limit distribution is referred to as the Marcenko-Pastur distribution [30]. The derivation of the eigenvalue distribution for such large dimensional matrices brings the most fundamental tool of random matrix theory, namely the Stieltjes transform, and it is also a necessary prior step for parameter estimation and detection problems. Furthermore, the knowledge of the eigenvalue distribution for large dimensional matrices alone plays a paramount role in some wireless communication problems, such as computing or approximating the capacity of multi-antenna based channels. The interested reader can refer to [32–35] for more details about applications of random matrix theory in wireless communications.

In the following, we present a parameter estimation problem where the conventional sample covariance method fails for comparable population size and number of observations. In this case, an estimation method based on the spectral properties of large dimensional random matrices, referred to as the G-estimation method [36] can be employed to improve the performance. Rather than analyzing the complicated mathematical details of the random matrix theory and G-estimation method, our aim is instead to illustrate how the random matrix theory can be applied in a typical application. Consider a multi-source power estimation problem [31], where at time *t*, a number of *N* sensors receive signals from K sources which are equipped with n_1, n_2, \dots, n_K transmitters, respectively. The observed data $\mathbf{y} \in \mathbb{C}^N$ from sensors at time *t* can be expressed as

$$\mathbf{y}(t) = \sum_{k=1}^{K} \sqrt{P_i} \widetilde{\mathbf{U}}_i \mathbf{x}_i(t) + \sigma \boldsymbol{\nu}(t), \qquad (35)$$

where P_i denotes the transmit power of source i, $\mathbf{x}_i \in \mathbb{C}^{n_i}$ stands for the signal collected from source iand it is assumed to be complex Gaussian distributed with zero mean and identity covariance matrix, and σv represents the additive noise. For simplicity, the channel matrix $\widetilde{\mathbf{U}}_i \in \mathbb{C}^{N \times n_i}$ is selected such that $[\widetilde{\mathbf{U}}_1, \widetilde{\mathbf{U}}_2, \cdots, \widetilde{\mathbf{U}}_K] \in \mathbb{C}^{N \times (n_1 + n_2 + \cdots + n_K)}$ represent the $n_1 + n_2 + \cdots + n_K$ columns of the unitary matrix $\mathbf{U} \in \mathbb{C}^{N \times N}$.

If follows that

$$\mathbf{E}[\mathbf{y}(t)\mathbf{y}(t)^{H}] = \mathbf{U}\mathbf{P}\mathbf{U}^{H} + \sigma^{2}\mathbf{I}_{N}$$

where **P** is a diagonal matrix with diagonal elements

$$\underbrace{P_1,\cdots,P_1}_{n_1},\underbrace{P_2,\cdots,P_2}_{n_2},\cdots,\underbrace{P_K,\cdots,P_k}_{n_K},\underbrace{0,\cdots,0}_{N-n_1-\cdots-n_K}.$$

Suppose that *L* independent observations of $\mathbf{y}(t)$ are available, i.e., $\mathbf{y}(1)$, $\mathbf{y}(2)$, \cdots , $\mathbf{y}(L)$. Thus, it follows that as $L \to \infty$:

$$\frac{1}{L}\sum_{i=1}^{L} \mathbf{y}(i)\mathbf{y}(i)^{H} - \sigma^{2}\mathbf{I}_{N} - \mathbf{U}\mathbf{P}\mathbf{U}^{H} \parallel \xrightarrow{\text{a.s.}} 0,$$
(36)

and a consistent estimator can be derived. However, when N is also a large number and L and N are comparable, (36) does not hold anymore. Using the Stieltjes transform and its extensions, the G-estimation method is generally adopted to improve the estimation performance. More technical and mathematical details about the Stieltjes transform and its generalizations can be found in [31,35].

4. Matrices in Image Processing

In digital image processing, images are usually represented using matrices. For example, the gray-scale image "Lena.tiff" is illustrated in Figure 7a, while its matrix representation is shown in Figure 7b. A matrix element is called "pixel". Most digital images use integer numbers between 0 (to indicate black) and 255 (to indicate white) to represent the values of pixels, giving a total of $2^8 = 256$ gray levels. Color images, in turn, can be represented by three matrices. These matrices specify the amount of red (R), green (G) and blue (B) present in the image. This representation system is known as the RGB system.



(a) Lena



(b) Matrix representation of Lena

Figure 7. Example of gray-scale figure.

In the following sections, two image compression methods are discussed, namely the block transform coding and wavelet representation, both of which decompose the image matrix into several submatrices and use the sparsity of these submatrices to compress the image.

4.1. Block Transform Coding

The block transform coding [37] is a compression technique which divides an image into several non-overlapping subimages of equal size and then processes these subimages using a 2-D transform. It turns out that after implementing a block transformation, for most subimages, a large portion of the resulting coefficients present small magnitudes and thus can be discarded. Therefore, using the

block transform coding, one can develop a very simple storage mechanism since most of the block transformed entries assume small values and can be neglected.

Specifically, an $M \times N$ image is divided into MN/n^2 subimages each of which has a size $n \times n$. Consider a subimage with coefficients g(x, y), the 2-D transform can be expressed as follows:

$$T(u,v) = \sum_{x=0}^{n-1} \sum_{y=0}^{n-1} g(x,y)r(x,y,u,v).$$
(37)

The original image g(x, y) can be recovered by the inverse transform defined by

$$g(x,y) = \sum_{u=0}^{n-1} \sum_{v=0}^{n-1} T(u,v) s(x,y,u,v).$$
(38)

In these equations, r(x, y, u, v) and s(x, y, u, v) are referred to as forward and inverse transformation kernels, respectively. The kernels depend on what transformation (DFT, DCT, Haar Wavelet Transform, etc.) is implemented. In this paper, the JPEG image compression standard using the discrete cosine transform (DCT) is presented. In this case, we have

$$r(x, y, u, v) = s(x, y, u, v) = \alpha(u)\alpha(v)\cos\left[\frac{(2x+1)u\pi}{2n}\right]\cos\left[\frac{(2y+1)v\pi}{2n}\right]$$

where

$$\alpha(x) = \begin{cases} \sqrt{\frac{1}{n}}, & x = 0, \\ \sqrt{\frac{2}{n}}, & x = 1, 2, \cdots, n-1. \end{cases}$$

Equations (37) and (38) can be written in a compact form by using matrices. Consider (38) as an example,

$$\mathbf{G} = \sum_{u=0}^{n-1} \sum_{v=0}^{n-1} T(u, v) \mathbf{S}_{uv},$$
(39)

where **G** is the subimage matrix consisting of g(x, y) and

$$\mathbf{S}_{uv} = \begin{bmatrix} s(0,0,u,v) & s(0,1,u,v) & \cdots & s(0,n-1,u,v) \\ s(1,0,u,v) & s(1,1,u,v) & \cdots & s(1,n-1,u,v) \\ \vdots & \vdots & \ddots & \vdots \\ s(n-1,0,u,v) & s(n-1,1,u,v) & \cdots & s(n-1,n-1,u,v) \end{bmatrix}.$$

Similarly, (37) can be defined as

$$\mathbf{T} = \sum_{x=0}^{n-1} \sum_{y=1}^{n-1} g(x, y) \mathbf{R}_{xy}.$$
 (40)

4.2. Wavelet Representation

Besides the classical DCT image representation, another modern image representation method, that relies on the 2-D wavelet coding and lies at the basis of JPEG 2000 standard [37], is used extensively. Different from the transform coding, the wavelet coding processes the whole image and results in a modified image consisting of four subimages. For example, after processing Figure 7a using the Haar wavelet, the modified image in Figure 8a is obtained. The four submatrices represent the approximation, the horizontal details, the vertical details and the diagonal details, respectively. One can observe that the three detail matrices can be coarsely quantized since their coefficients roughly present similar values. Moreover, the approximation matrix in Figure 8a can also be decomposed into four sub-subimages as

shown in the upper left corner of Figure 8b. In this way, the image is compressed to a higher degree given the premise that the original image can be recovered with a tolerated quality.



(a) First scale Haar wavelet transform



Figure 8. Haar Wavelet transform.

5. Compressive Sensing

The celebrated Shannon/Nyquist sampling theorem states that in order to reconstruct a signal without losing information, the sample rate must be at least two times larger than the signal bandwidth. So far, most signal acquisition techniques rely on this principle. As mentioned in Section 4.1, transform coding plays a key role in data acquisition systems. However, this compression framework suffers from several drawbacks [38]. For instance, consider a discrete signal $\mathbf{x} \in \mathbb{R}^N$ with entries x[n], $n = 0, 1, \dots, N - 1$. An image or a higher-dimensional data structure can be also vectorized into a long one-dimensional vector. It is known that any signal in \mathbb{R}^N can be represented in terms of a basis of N vectors $\{\psi_i\}_{i=0}^{N-1}$. In this case,

$$\mathbf{x} = \sum_{i=0}^{N-1} s_i \boldsymbol{\psi}_i = \mathbf{\Psi} \mathbf{s},$$

where **s** stands for the weighting coefficients and Ψ is the basis matrix with columns $[\psi_0, \dots, \psi_{N-1}]$. The signal **x** is called *K*-sparse if only *K* coefficients of **s** are non-zero and the remaining N - K coefficients are roughly zero (or can be truncated to zero). Thus, in order to process the transform coding, the locations of the *K* non-zero coefficients must be coded, which results in an overhead. Moreover, the N - K zero coefficients still need to be computed even though they will be discarded.

Compressive sensing addresses these issues by directly processing the compressed signal without acquiring the original *N*-sample signal provided that the signal is *K*-sparse [39]. This novel sampling method dramatically undersamples the signal at a greatly reduced rate. Next we will present the mathematical model behind this new technique.

Consider the following representation [38]:

$$\mathbf{y} = \mathbf{\Phi}\mathbf{x} = \mathbf{\Phi}\mathbf{\Psi}\mathbf{s} = \mathbf{\Theta}\mathbf{s},\tag{41}$$

where $\mathbf{x} \in \mathbb{R}^N$ stands for the original signal, $\mathbf{y} \in \mathbb{R}^M$ ($M \ll N$) denotes the compressed signal, and $\mathbf{\Phi} \in \mathbb{R}^{M \times N}$ represents the linear measurement process. Mathematically, the problem can be formulated as follows. First, find a matrix $\mathbf{\Phi}$ such that $\mathbf{y} = \mathbf{\Phi}\mathbf{x}$ preserves most of the information embedded in \mathbf{x} . Second, find an algorithm to recover \mathbf{x} from only M samples of \mathbf{y} .

5.1. Good Sensing Matrices

The most critical issue is to determine whether Φ is good for compressive sensing and reconstruction. The reconstruction seems to be impossible at first glance since the problem is ill-conditioned (M < N). However, it is now well-known that one can reconstruct the sparse signal from a limited number of measurements if the sensing matrix satisfies the restricted isometry property (RIP) [39–41]. A matrix $\Phi \in \mathbb{R}^{M \times N}$ is said to satisfy the RIP of order K < M with isometry constant $\delta_K \in (0, 1)$ if

$$|(1 - \delta_K)||\mathbf{v}||_2^2 \le ||\mathbf{\Phi}\mathbf{v}||_2^2 \le (1 + \delta_K)||\mathbf{v}||_2^2$$

holds for any *K*-sparse vector **v**.

Since verifying the RIP property of a matrix requires an exhaustive search of all $\binom{N}{K}$ possibilities, most algorithms developed so far are based on randomization. Typically, the matrix elements are produced by independent and identically distributed (i.i.d.) random variables with certain distribution. References [42] and [43] proved that if the elements of matrix $\mathbf{\Phi}$ are i.i.d. Gaussian random variables with zero means and variance 1/N, and $M \ge cK \log(N/K)$ where *c* is a small constant, then $\mathbf{\Phi}$ satisfies RIP with a very high probability. Therefore, the signal can be recovered from only $M \ge cK \log(N/K) \ll N$ variables. Other available selections for robust sensing matrices are discussed in [42,44].

5.2. Compressive Signal Reconstruction

The signal reconstruction problem requires the recovery of signal \mathbf{x} , or equivalent of the weighted coefficients \mathbf{s} , given the sensing matrix $\mathbf{\Phi}$, basis matrix $\mathbf{\Psi}$ and compressed signal \mathbf{y} . The ultimate goal of compressive sensing is to find the possible sparsest representation of the signal. Therefore, the ideal reconstruction problem can be expressed via the l_0 norm as follows:

$$\min_{\mathbf{s},\mathbf{t},\mathbf{t}} \frac{||\mathbf{s}||_0}{|\mathbf{s},\mathbf{t},\mathbf{\Theta}\mathbf{s}| = \mathbf{y}.$$

$$(42)$$

Unfortunately, such a formulation of the problem is both numerically unstable and computationally complex (NP-complete) [45]. For simplicity, problem (44) can be reduced to minimize the l_2 norm of **s**

$$\min ||\mathbf{s}||_2$$
s.t. $\Theta \mathbf{s} = \mathbf{y}$, (43)

which leads to the closed form solution $\hat{s} = \theta^T (\theta \theta^T)^{-1} y$. However, this method will almost never yield a *K*-sparse solution. Even worse, usually a full representation solution (no zero elements) will be generated by using the l_2 norm.

Surprisingly, the optimization based on the l_1 norm is the most promising recovery technique used so far for sparse signals. In this case, the recovery problem is formulated as follows:

$$\min_{\mathbf{s}, \mathbf{t}, \mathbf{\Theta}} \| \mathbf{s} \|_{1}$$

$$s.t. \mathbf{\Theta} \mathbf{s} = \mathbf{y}.$$

$$(44)$$

In some cases, the l_1 minimization yields the same result as the ideal l_0 minimization [46]. Moreover, the l_1 minimization can be efficiently solved via linear programming which presents a computational complexity of $O(N^3)$. For the last decade, a series of papers [39,43,47] explained why l_1 minimization can be approximated as an l_0 minimization and why it can recover a sparse signal. Until now, most of the works dedicated to compressive sensing, see, e.g., the works of Dandes, Romberg, Tao and Donoho [39–43] assume the l_1 norm formulation.

6. Conclusions

This paper overviewed how matrix theory can be applied in wireless communications and signal processing applications. In particular, this paper illustrated how matrix transformations, matrix decompositions and random matrix theory could be exploited in the implementation, characterization and optimization of MIMO communication systems, OFDM signaling schemes and signal estimation applications. This paper also reviewed the significant role played by matrices in representing and processing digital images and signals acquired by an array of sensors. Furthermore, a recent research activity termed as compressive sensing was briefly discussed herein paper. It will require continued efforts from researchers in wireless communications and signal processing to ultimately exploit the full potential applications of matrix theory. Furthermore, the complex phenomena in nature and engineering applications require higher-dimensional matrices for their modeling, characterization and understanding. In this regard, tensors have been used in mathematics, physics and engineering since they are a generalization of scalars, vectors and two-dimensional arrays to represent more complex quantities and properties in higher dimensions. Therefore, a promising research direction is to investigate new applications of tensors in wireless communications and signal processing, and to extend the arsenal of mathematical tools available for storing, extracting and processing information from tensors. Finally, we would like to mention that linear algebra plays a very important role in developing powerful algorithms in bioinformatics and systems biology applications. Numerous algorithms for the inference of gene and proteomic regulatory networks as well as for the inference of transcription factor networks have been proposed in literature (see, e.g., [48,49]) by exploiting linear algebra tools. However, reviewing these computational biology applications as well as numerous other machine learning applications where linear algebra plays a key role is beyond the scope of this paper.

Acknowledgments: This work was supported by NSF Award CCF-1318338.

Author Contributions: Xu Wang contributed to draft writing. Erchin Serpedin proposed the framework of the paper and helped to improve the manuscript. All authors read and approved the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

- 1. Shannon, C.E. A mathematical theory of communication. Bell Syst. Tech. J. 1948, 27, 379–423.
- 2. Introduction to LTE-Advanced: Application Note. *Agilent Technol.* **2011**. Available online: http://cp. literature.agilent.com/litweb/pdf/5990-6706EN.pdf (accessed on 6 January 2016).
- 3. Telatar, E. Capacity of multi-antenna Gaussian channels. Eur. Trans. Telecommun. 1999, 10, 585–596.
- 4. Goldsmith, A. Wireless Communications; Cambridge University Press: Cambridge, UK, 2005.
- 5. Foschini, G.J. Layered space-time architecture for wireless communication in fading environments when using multi-element antennas. *Bell Labs Tech. J.* **1996**, *1*, 41–59.
- 6. Foschini, G.J.; Gans, M.J. On limits of wireless communications in fading environment when using multiple antennas. *Wirel. Pers. Commun.* **1998**, *6*, 311–335.
- 7. Goldsmith, A.; Jafar, S.A.; Jindal, N.; Vishwanath, S. Capacity limits of MIMO channels. *IEEE J. Sel. Areas Commun.* **2003**, *21*, 684–702.
- 8. Wang, X.; Serpedin, E.; Qaraqe, K. A variational approach for assessing the capacity of a memoryless nonlinear MIMO channel. *IEEE Commun. Lett.* **2014**, *18*, 1315–1318.
- 9. Gray, R.M. Toeplitz and Circulant Matrices: A Review; Now Publishers Inc.: Norwell, MA, USA, 2006.
- Bolcskei, H.; Zurich, E. MIMO-OFDM wireless systems: Basics, perspectives, and challenges. *IEEE Trans. Wirel. Commun.* 2006, 13, 31–37.
- 11. Golub, G.H.; VanLoan, C.F. *Matrix Computations*; The Johns Hopkins University Press: Baltimore, MA, USA, 1996.
- 12. Kay, S. Fundamentals of Statistical Signal Processing: Estimation Theory; Prentice Hall: Upper Saddle River, NJ, USA, 1993.

- 13. Reilly, J.P. *Matrix Computations in Signal Processing: Lecture Notes;* Electrical and Computer Engineering Department, McMaster University: Hamilton, ON, Canada, 24 September 2006.
- 14. Haykin, S. Array Signal Processing; Prentice Hall: Upper Saddle River, NJ, USA, 1985.
- 15. Paleologu, C.; Ciochina, S.; Enescu, A. A family of recursive least-squares adaptive algorithms suitable for fixed-point implementation. *Int. J. Adv. Telecommun.* **2009**, *2*, 88–97.
- 16. Liavas, A.; Regalia, P. On the numerical stability and accuracy of the conventional recursive least squares algorithm. *IEEE Trans. Signal Process.* **1999**, 47, 88–96.
- 17. Ljung, S.; Ljung, L. Error propagation properties of recursive least-squares adaptation algorithms. *Automatica* **1985**, *21*, 157–167.
- 18. Verhaegen, M. Round-off error propagation in four generally-applicable, recursive, least-squares estimation schemes. *Automatica* **1989**, 25, 437–444.
- 19. Bhotto, Z.; Antoniou, A. New improved recursive least-squares adaptive-filtering algorithms. *IEEE Trans. Circuits Syst. I Reg. Pap.* **2013**, *60*, 1548–1558.
- 20. Leung, S.; So, C. Gradient-based variable forgetting factor RLS algorithm in time-varying environments. *IEEE Trans. Signal Process.* **2005**, *53*, 3141–3150.
- 21. Paleologu, C.; Benesty, J.; Ciochina, S. A robust variable forgetting factor recursive least-squares algorithm for system identification. *IEEE Signal Process. Lett.* **2008**, *15*, 597–600.
- 22. Ardalan, S. Floating-point roundoff error analysis of the exponentially windowed RLS algorithm for time-varying systems. *IEEE Trans. Circuits Syst.* **1986**, *33*, 1192–1208.
- 23. Bottomley, G.; Alexander, S. A novel approach for stabilizing recursive least squares filters. *IEEE Signal Process. Lett.* **1991**, *39*, 1770–1779.
- 24. Chansakar, M.; Desai, U. A robust recursive least squares algorithm. *IEEE Signal Process. Lett.* **1997**, 45, 1726–1735.
- 25. Haykin, S. Adpative Filtering Theory; Prentice Hall: Upper Saddle River, NJ, USA, 2002.
- 26. Krim, H.; Viberg, M. Two decades of array signal processing research. *IEEE Signal Process. Mag.* **1996**, 13, 67–94.
- 27. Roy, R.; Kailath, T. ESPRIT-estimation of signal parameters via rotational invariance techniques. *IEEE Trans. Acoust. Speech Signal Process.* **1989**, *37*, 984–995.
- 28. Stoica, P.; Moses, R.L. Introduction to Spectral Analysis; Prentice Hall: Upper Saddle River, NJ, USA, 1997.
- 29. Wigner, E. Characteristic vectors of bordered matrices with infinite dimensions. *Ann. Math.* **1955**, *62*, 548–564.
- Marcenko, V.A.; Pastur, L.A. Distributions of eigenvalues for some sets of random matrices. *Math. USSR-Sbornik* 1967, 1, 457–483.
- 31. Couillet, R.; Debbah, M. Signal processing in large systems: A new paradigm. *IEEE Signal Process. Mag.* **2003**, 30, 24–39.
- 32. Couillet, R.; Debbah, M.; Siverstein, J.W. A deterministic equivalent for the analysis of correlated MIMO multiple access channels. *IEEE Trans. Inf. Theory* **2011**, *57*, 3493–3514.
- 33. Dupuy, F.; Loubaton, P. On the capacity achieving covariance matrix for frequency selective MIMO channels using the asymptotic approach. *IEEE Trans. Inf. Theory* **2011**, *57*, 5737–5753.
- 34. Couillet, R.; Debbah, M. *Random Matrix Methods for Wireless Communications*; Cambridge University Press: Cambridge, UK, 2011.
- 35. Tulino, M.A.; Verdu, S. *Random Matrix Theory and Wireless Communications*; Now Publishers: Norwell, MA, USA, 2004.
- 36. Girko, V.L. Theory of Random Determinants; Kluwer Academic Publisher: Dordrecht, The Netherlands, 1990.
- 37. Gonzalez, R.C.; Woods, R.E. Digital Image Processing; Prentice Hall: Upper Saddle River, NJ, USA, 2007.
- 38. Baraniuk, R.G. Compressive Sensing. *IEEE Signal Process. Mag.* 2007, 24, 119–121.
- 39. Dandes, E.; Romberg, J.; Tao, T. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory* **2006**, 52, 489–509.
- 40. Dandes, E.; Romberg, J. Quantitative robust uncertainty principles and optimally sparse decompositions. *Found. Comput. Math.* **2006**, *6*, 227–254.
- 41. Candes, E.; Tao, T. Decoding by linear programming. *IEEE Trans. Inf. Theory* 2005, 51, 4203–4215.
- 42. Candes, E.; Tao, T. Near optimal signal recovery from random projections: Universal encoding strategies? *IEEE Trans. Inf. Theory* **2006**, *51*, 5406–5425.

- 43. Donoho, D.L. For most large underdetermined systems of equations, the minimal *l*₁-norm near-solution approximates the sparsest near-solution. *Commun. Pure Appl. Math.* **2004**, *59*, 797–829.
- 44. Baraniuk, R.G.; Davenport, M.A.; DeVore, R.A.; Wakin, M.B. A simple proof of the restricted isometry property for random matrices. *Constr. Approx.* **2008**, *28*, 253–263.
- 45. Natarajan, B.K. Sparse approximate solutions to linear systems. SIAM J. Comput. 1995, 24, 227–234.
- 46. Holtz, H.V. Compressive sensing: A paradigm shift in signal processing. 2008, arXiv:0812.3137.
- 47. Donoho, D.L.; Huo, X. Uncertainty principles and ideal atomic decomposition. *IEEE Trans. Inf. Theory* **2001**, 47, 2845–2862.
- 48. Wang, X.; Alshawaqfe, M.; Dang, X.; Wajid, B.; Noor, A.; Qaraqe, M.; Serpedin, E. An overview of NCA-based algorithms for transcriptional regulatory network inference. *Microarrays* **2015**, *4*, 596–617.
- 49. Noor, A.; Serpedin, E.; Nounou, M.; Nounou, H. An overview of the statistical methods for inferring gene regulatory networks and protein-protein interaction networks. *Adv. Bioinform.* **2013**, 2013, 953814.



© 2016 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (http://creativecommons.org/licenses/by/4.0/).