
Partition-Matrix Theory Applied to the Computation of Generalized-Inverses for MIMO Systems in Rayleigh Fading Channels

P. Cervantes¹, L. F. González¹, F. J. Ortiz² and
A. D. García²

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/48198>

1. Introduction

Partition-Matrix Theory and Generalized-Inverses are interesting topics explored in linear algebra and matrix computation. Partition-Matrix Theory is associated with the problem of properly partitioning a matrix into block matrices (i.e. an array of matrices), and is a matrix computation tool widely employed in several scientific-technological application areas. For instance, blockwise Toeplitz-based covariance matrices are used to model structural properties for space-time multivariate adaptive processing in radar applications [1], Jacobian response matrices are partitioned into several block-matrix instances in order to enhance medical images for Electrical-Impedance-Tomography [2], design of state-regulators and partial-observers for non-controllable/non-observable linear continuous systems contemplates matrix blocks for controllable/non-controllable and observable/non-observable eigenvalues [3]. The Generalized-Inverse is a common and natural problem found in a vast of applications. In control robotics, non-collocated partial linearization is applied to underactuated mechanical systems through inertia-decoupling regulators which employ a pseudoinverse as part of a modified input control law [4]. At sliding-mode control structures, a Right-Pseudoinverse is incorporated into a state-feedback control law in order to stabilize electromechanical non-linear systems [5]. Under the topic of system identification, definition of a Left-Pseudoinverse is present in auto-regressive moving-average models (ARMA) for matching dynamical properties of unknown systems [6]. An interesting approach arises whenever Partition-Matrix Theory and Generalized-Inverse are combined together yielding attractive solutions for solving the problem of block matrix inversion [7-10]. Nevertheless,

several assumptions and restrictions regarding numerical stability and structural properties are considered for these alternatives. For example, an attractive pivot-free block matrix inversion algorithm is proposed in [7], which unfortunately exhibits an overhead in matrix multiplications that are required in order to guarantee full-rank properties for particular blocks within it. For circumventing the expense in rank deficiency, [8] offers block-matrix completion strategies in order to find the Generalized-Inverse of any non-singular block matrix (irrespective of the singularity of their constituting sub-blocks). However, the existence of intermediate matrix inverses and pseudoinverses throughout this algorithm still rely on full-rank assumptions, as well as introducing more hardness to the problem. The proposals exposed in [9-10] avoid completion strategies and contemplate all possible scenarios for avoiding any rank deficiency among each matrix sub-block, yet demanding full-rank assumptions for each scenario. In this chapter, an iterative-recursive algorithm for computing a Left-Pseudoinverse (LPI) of a MIMO channel matrix is developed by combining Partition-Matrix Theory and Generalized-Inverse concepts. For this approach, no matrix-operations' overhead nor any particular block matrix full-rank assumptions are needed because of structural attributes of the MIMO channel matrix, which models dynamical properties of a Rayleigh fading channel (RFC) within wireless MIMO communication systems.

The content of this work is outlined as follows. Section 2 provides a description of the MIMO communication link, pointing out its principal physical effects and the mathematical model considered for RFC-based environments. Section 3 defines formally the problem of computing the Left-Pseudoinverse as the Generalized-Inverse for the MIMO channel matrix applying Partition-Matrix Theory concepts. Section 4 presents linear algebra and matrix computation concepts and tools needed for tracking a solution for the aforementioned problem. Section 5 analyzes important properties of the MIMO channel matrix derived from a Rayleigh fading channel scenario. Section 6 explains the proposed novel algorithm. Section 7 presents a brief analysis of VLSI (Very Large Scale of Integration) aspects towards implementation of arithmetic operations presented in this algorithm. Section 8 concludes the chapter. Due to the vast literature about MIMO systems, and to the best of the authors' knowledge, this chapter provides a nice and strategic list of references in order to easily correlate essential concepts between matrix theory and MIMO systems. For instance, [11-16] describe and analyze information and system aspects about MIMO communication systems, as well as studying MIMO channel matrix behavior under RFC-based environments; [17-18] contain all useful linear algebra and matrix computation theoretical concepts around the mathematical background immersed in MIMO systems; [19-21] provide practical guidelines and examples for MIMO channel matrix realizations comprising RFC scenarios; [22] treats the formulation and development of the algorithm presented in this chapter; [23-27] detail a splendid survey on architectural aspects for implementing several arithmetic operations.

2. MIMO systems

In the context of wireless communication systems, MIMO (Multiple-Input Multiple-Output) is an extension of the classical SISO (Single-Input Single-Output) communication paradigm,

where instead of having a communication link composed of a single transmitter-end and a receiver-end element (or antenna), wireless MIMO communication systems (or just MIMO systems) consist of an array of multiple elements at both the transmission and reception parts [11-16,19-21]. Generally speaking, the MIMO communication link contains n_T transmitter-end and n_R receiver-end antennas sending-and-receiving information through a wireless channel. Extensive studies on MIMO systems and commercial devices already employing them reveal that these communication systems offer promising results in terms of: a) spectral efficiency and channel capacity enhancements (many user-end applications supporting high-data rates at limited available bandwidth); b) improvements on Bit-Error-Rate (BER) performance; and c) practical feasibility already seen in several wireless communication standards. The conceptualization of this paradigm is illustrated in figure 1, where Tx is the transmitter-end, Rx the receiver-end, and Chx the channel.

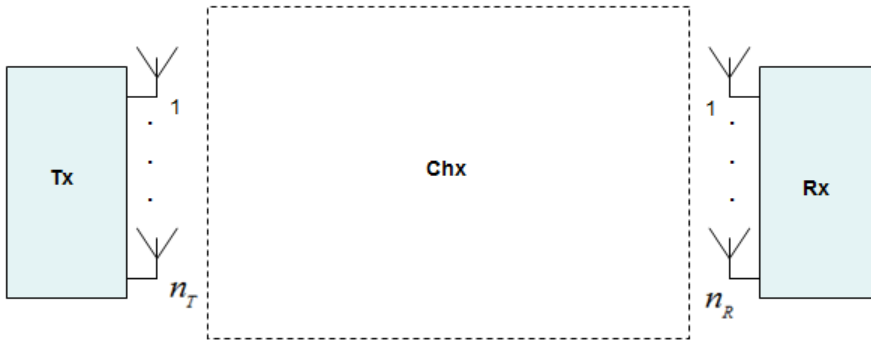


Figure 1. The MIMO system: conceptualization for the MIMO communication paradigm.

Notice that information sent from the transmission part (Tx label on figure 1) will suffer from several degradative and distortional effects inherent in the channel (Chx label on figure 1), forcing the reception part (Rx label on figure 1) to decode information properly. Information at Rx will suffer from degradations caused by time, frequency, and spatial characteristics of the MIMO communication link [11-12,14]. These issues are directly related to: i) the presence of physical obstacles obstructing the Line-of-Sight (LOS) between Tx and Rx (existence of non-LOS); ii) time delays between received and transmitted information signals due to Tx and Rx dynamical properties (time-selectivity of Chx); iii) frequency distortion and interference among signal carriers through Chx (frequency-selectivity of Chx); iv) correlation of information between receiver-end elements. Fading (or fading mutlipath) and noise are the most common destructive phenomena that significantly affect information at Rx [11-16]. Fading is a combination of time-frequency replicas of the trasnmited information as a con-

sequence of the MIMO system phenomena i)-iv) exposed before, whereas noise affects information at every receiver-end element under an additive or multiplicative way. As a consequence, degradation of signal information rests mainly upon magnitude attenuation and time-frequency shiftings. The simplest treatable MIMO communication link has a slow-flat quasi-static fading channel (proper of a non-LOS indoor environment). For this type of scenario, a well-known dynamical-stochastic model considers a Rayleigh fading channel (RFC) [13,15-16,19-21], which gives a quantitative clue of how information has been degraded by means of Chx. Moreover, this type of channels allows to: a) distinguish among each information block transmitted from the n_T elements at every Chx realization (i.e. the time during which the channel's properties remain unvariant); and b) implement easily symbol decoding tasks related to channel equalization (CE) techniques. Likewise, noise is commonly assumed to have additive effects over Rx. Once again, all of these assumptions provide a treatable information-decoding problem (referred as MIMO demodulation [12]), and the mathematical model that suits the aforementioned MIMO communication link characteristics will be represented by

$$y = Hx + \eta \quad (1)$$

where: $x \in \mathbb{Z}_{[j]}^{n_T \times 1} \subset \mathbb{C}^{n_T \times 1}$ is a complex-valued n_T - dimensional transmitted vector with entries drawn from a Gaussian-integer finite-lattice constellation (digital modulators, such as: q -QAM, QPSK); $y \in \mathbb{C}^{n_R \times 1}$ is a complex-valued n_R - dimensional received vector; $\eta \in \mathbb{C}^{n_R \times 1}$ is a n_R - dimensional independent-identically-distributed (idd) complex-circularly-symmetric (ccs) Additive White Gaussian Noise (AWGN) vector; and $H \in \mathbb{C}^{n_R \times n_T}$ is the $(n_R \times n_T)$ - dimensional MIMO channel matrix whose entries model: a) the RFC-based environment behavior according to a Gaussian probabilistic density function with zero-mean and 0.5-variance statistics; and b) the time-invariant transfer function (which measures the degradation of the signal information) between the i -th receiver-end and the j -th transmitter-end antennas [11-16,19-21]. Figure 2 gives a representation of (1). As shown therein, the MIMO communication link model stated in (1) can be also expressed as

$$\begin{bmatrix} y_1 \\ \vdots \\ y_{n_R} \end{bmatrix} = \begin{bmatrix} h_{11} & \dots & h_{1n_T} \\ \vdots & & \vdots \\ h_{n_R1} & \dots & h_{n_Rn_T} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_{n_T} \end{bmatrix} + \begin{bmatrix} \eta_1 \\ \vdots \\ \eta_{n_R} \end{bmatrix} \quad (2)$$

Notice from (1-2) that an important requisite for CE purposes within RFC scenarios is that H is provided somehow to the Rx. This MIMO system requirement is classically known as Channel State Information (CSI) [11-16]. In the sequel of this work, symbol-decoding efforts will consider the problem of finding x from y regarding CSI at the Rx part within a slow-flat quasi-static RFC-based environment as modeled in (1-2). In simpler words, Rx must find x from degraded information y through calculating an inversion over H . Moreover, $n_R \geq n_T$ is

commonly assumed for MIMO demodulation tasks [13-14] because it guarantees linear independency between row-entries of matrix H in (2), yielding a nonhomogeneous overdetermined system of linear equations.

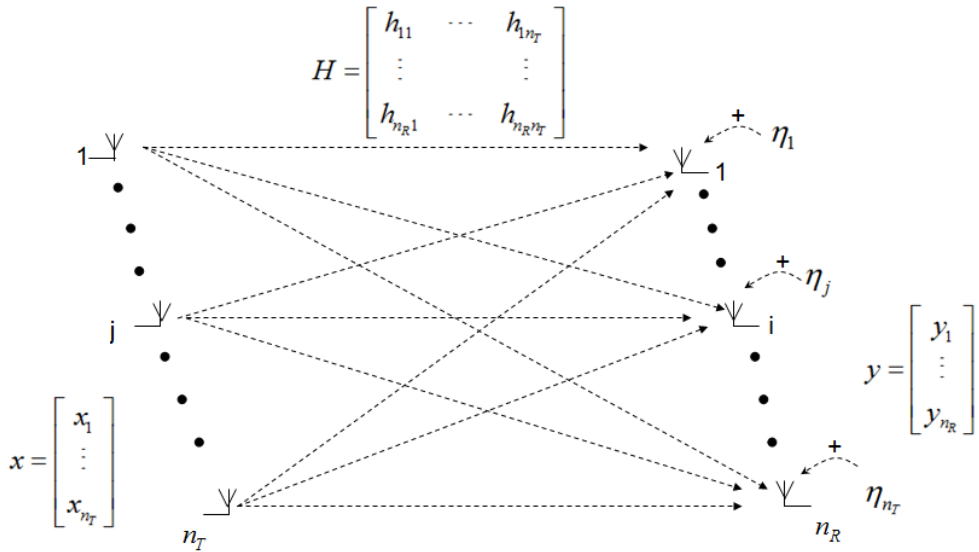


Figure 2. Representation for the MIMO communication link model according to $y = Hx + \eta$. Here, each dotted arrow represents an entry h_{ij} in H which determines channel degradation between the j -th transmitter and the i -th receiver elements. AWGN appears additively in each receiver-end antenna.

3. Problem definition

Recall for the moment the mathematical model provided in (1). Consider Φ^r and Φ^i to be the real and imaginary parts of a complex-valued matrix (vector) Φ , that is, $\Phi = \Phi^r + j\Phi^i$. Then, Equation (1) can be expanded as follows:

$$y^r + jy^i = (H^r x^r - H^i x^i + \eta^r) + j(H^i x^r + H^r x^i + \eta^i) \tag{3}$$

It can be noticed from Equation (3) that: $x^r, x^i \in \mathbb{Z}^{n_T \times 1}$; $y^r, y^i \in \mathbb{R}^{n_R \times 1}$; $\eta^r, \eta^i \in \mathbb{R}^{n_R \times 1}$; and $H^r, H^i \in \mathbb{R}^{n_R \times n_T}$. An alternative representation for the MIMO communication link model in (2) can be expressed as

$$\begin{bmatrix} y^r \\ y^i \end{bmatrix} = \begin{bmatrix} H^r & -H^i \\ H^i & H^r \end{bmatrix} \begin{bmatrix} x^r \\ x^i \end{bmatrix} + \begin{bmatrix} \eta^r \\ \eta^i \end{bmatrix} \tag{4}$$

where $\begin{bmatrix} y^r \\ y^i \end{bmatrix} \doteq Y \in \mathbb{R}^{2n_R \times 1}$, $\begin{bmatrix} H^r & -H^i \\ H^i & H^r \end{bmatrix} \doteq h \in \mathbb{R}^{2n_R \times 2n_T}$, $\begin{bmatrix} x^r \\ x^i \end{bmatrix} \doteq X \in \mathbb{Z}^{2n_T \times 1}$, and $\begin{bmatrix} \eta^r \\ \eta^i \end{bmatrix} \doteq N \in \mathbb{R}^{2n_R \times 1}$. CSI is still needed for MIMO demodulation purposes involving (4).

Moreover, if $N_r = 2n_R$ and $N_t = 2n_T$, then $N_r \geq N_t$. Obviously, while seeking for a solution of signal vector X from (4), the reception part Rx will provide also the solution for signal vector x , and thus MIMO demodulation tasks will be fulfilled. This problem can be defined formally into the following manner:

Definition 1. Given parameters $N_r = 2n_R \in \mathbb{Z}^+$ and $N_t = 2n_T \in \mathbb{Z}^+$, and a block-matrix $h \in \mathbb{R}^{N_r \times N_t}$, there exists an operator $\Gamma : (\mathbb{R}^{N_r \times 1} \times \mathbb{R}^{N_r \times N_t}) \mapsto \mathbb{R}^{N_t \times 1}$ which solves the matrix-block equation $Y=hX+N$ so that $\Gamma[Y,h] = X$. ■

From Definition 1, the following affirmations hold: i) CSI over his a necessary condition as an input argument for the operator Γ ; and ii) Γ can be naively defined as a Generalized-Inverse of the block-matrix h . In simpler terms, $X=h^\dagger Y$ is associated with $\Gamma[Y,h]$ and $h^\dagger \in \mathbb{R}^{N_t \times N_r}$ stands for the Generalized-Inverse of the block-matrix h , where $h^\dagger = (h^T h)^{-1} h^T$ [17-18]. Clearly, $[]^{-1}$ and $[]^T$ represent the inverse and transpose matrix operations over real-valued matrices. As a concluding remark, computing the Generalized-Inverse h^\dagger can be separated into two operations: 1) a block-matrix inversion $(h^T h)^{-1}$; 2) a typical matrix multiplication $(h^T h)^{-1} \cdot h^T$. For these tasks, Partition-Matrix Theory will be employed in order to find a novel algorithm for computing a Generalized-Inverse related to (4).

4. Mathematical background

4.1. Partition-matrix theory

Partition-Matrix Theory embraces structures related to block matrices (or partition matrices: an array of matrices) [17-18]. Furthermore, a block-matrix L with $(n + q) \times (m + p)$ dimension can be constructed (or partitioned) consistently according to matrix sub-blocks $A, B, C,$ and D of $n \times m, n \times p, q \times m,$ and $q \times p$ dimensions, respectively, yielding

1 In the context of MIMO systems, this matrix operation is commonly found in Babai estimators for symbol-decoding purposes at the Rx part [12,13]. For the reader's interest, refer to [11-16] for other MIMO demodulation techniques.

2 Notice that and .

$$L = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \tag{5}$$

An interesting operation to be performed for these structures given in (5) is the inversion, i.e. a blockwise inversion L^{-1} . For instance, let $L \in \mathbb{R}^{(n+m) \times (n+m)}$ be a full-rank real-valued block matrix (the subsequent treatment is also valid for complex-valued entities, i.e. $L \in \mathbb{C}^{(n+m) \times (n+m)}$). An alternative partition can be performed with $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{m \times n}$, and $D \in \mathbb{R}^{m \times m}$. Assume also A and D to be full-rank matrices. Then,

$$L^{-1} = \begin{bmatrix} (A - BD^{-1}C)^{-1} & -(A - BD^{-1}C)^{-1}BD^{-1} \\ -(D - CA^{-1}B)^{-1}CA^{-1} & (D - CA^{-1}B)^{-1} \end{bmatrix} \tag{6}$$

This strategy (to be proved in the next part) requires additionally and mandatorily full-rank over matrices $A - BD^{-1}C$ and $D - CA^{-1}B$. The simple case is defined for $L = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ (indistinctly for $\mathbb{R}^{2 \times 2}$ or $\mathbb{C}^{2 \times 2}$). Once again, assuming $\det(L) \neq 0$, $a \neq 0$, and $d \neq 0$ (related to full-rank restrictions within block-matrix L):

$$L^{-1} = \begin{bmatrix} (a - bd^{-1}c)^{-1} & -(a - bd^{-1}c)^{-1}bd^{-1} \\ -(d - ca^{-1}b)^{-1}ca^{-1} & (d - ca^{-1}b)^{-1} \end{bmatrix} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

where evidently $(ad - bc) \neq 0$, $\mathbb{R}\mathbb{C}^{(n+m) \times (n+m)}(a - bd^{-1}c) \neq 0$, and $(d - ca^{-1}b) \neq 0$.

4.2. Matrix Inversion Lemma

The Matrix Inversion Lemma is an indirect consequence of inverting non-singular block matrices [17-18], either real-valued or complex-valued, e.g., under certain restrictions³. Lemma 1 states this result.

Lemma 1. Let $\Psi \in \mathbb{R}\mathbb{C}^{r \times r}$, $\Sigma \in \mathbb{R}\mathbb{C}^{r \times s}$, $\Xi \in \mathbb{R}\mathbb{C}^{s \times s}$, and $\Xi \in \mathbb{R}\mathbb{C}^{s \times r}$ be real-valued or complex-valued matrices. Assume these matrices to be non-singular: Ψ , $(\Psi + \Sigma \Xi)$, and $(^{-1} + \Xi \Psi^{-1} \Sigma)$. Then,

$$(\Psi + \Sigma \Xi)^{-1} = \Psi^{-1} - \Psi^{-1} \Sigma (^{-1} + \Xi \Psi^{-1} \Sigma)^{-1} \Xi \Psi^{-1} \tag{7}$$

Proof. The validation of (7) must satisfy

i.
$$(\Psi + \Sigma \Xi) \cdot (\Psi^{-1} - \Psi^{-1} \Sigma (^{-1} + \Xi \Psi^{-1} \Sigma)^{-1} \Xi \Psi^{-1}) = I_r \tag{8}$$

³ Refer to [3,7-10,17,18] to review lemmata exposed for these issues and related results.

$(\Psi^{-1} - \Psi^{-1}\Sigma(-^1 + \Xi\Psi^{-1}\Sigma)^{-1}\Xi\Psi^{-1}) \cdot (\Psi + \Sigma\Xi) = I_r$, where I_r represents the $r \times r$ identity matrix. Notice the existence of matrices Ψ^{-1} , $^{-1}$, $(\Psi + \Sigma\Xi)^{-1}$ and $(^{-1} + \Xi\Psi^{-1}\Sigma)^{-1}$. Manipulating i) shows:

$$\begin{aligned} & (\Psi + \Sigma\Xi) \cdot (\Psi^{-1} - \Psi^{-1}\Sigma(-^1 + \Xi\Psi^{-1}\Sigma)^{-1}\Xi\Psi^{-1}) \\ &= I_r - \Sigma(-^1 + \Xi\Psi^{-1}\Sigma)^{-1}\Xi\Psi^{-1} + \Sigma\Xi\Psi^{-1} - \Sigma\Xi\Psi^{-1}\Sigma(-^1 + \Xi\Psi^{-1}\Sigma)^{-1}\Xi\Psi^{-1} \\ &= I_r + \Sigma\Xi\Psi^{-1} - \Sigma(-^1 + \Xi\Psi^{-1}\Sigma)(^{-1} + \Xi\Psi^{-1}\Sigma)^{-1}\Xi\Psi^{-1} \\ &= I_r + \Sigma\Xi\Psi^{-1} - \Sigma\Xi\Psi^{-1} = I_r. \end{aligned}$$

Likewise for ii):

$$\begin{aligned} & (\Psi^{-1} - \Psi^{-1}\Sigma(-^1 + \Xi\Psi^{-1}\Sigma)^{-1}\Xi\Psi^{-1}) \cdot (\Psi + \Sigma\Xi) \\ &= I_r + \Psi^{-1}\Sigma\Xi - \Psi^{-1}\Sigma(-^1 + \Xi\Psi^{-1}\Sigma)^{-1}\Xi - \Psi^{-1}\Sigma(-^1 + \Xi\Psi^{-1}\Sigma)^{-1}\Xi\Psi^{-1}\Sigma\Xi \\ &= I_r + \Psi^{-1}\Sigma\Xi - \Psi^{-1}\Sigma(-^1 + \Xi\Psi^{-1}\Sigma)^{-1}(-^1 + \Xi\Psi^{-1}\Sigma)\Xi \\ &= I_r + \Psi^{-1}\Sigma\Xi - \Psi^{-1}\Sigma\Xi = I_r. \blacksquare \end{aligned}$$

Now it is pertinent to demonstrate (6) with the aid of Lemma 1. It must be verified that both $L L^{-1}$ and $L^{-1}L$ must be equal to the $(n+m) \times (n+m)$ identity block matrix $I_{(n+m)} = \begin{bmatrix} I_n & 0_{n \times m} \\ 0_{m \times n} & I_m \end{bmatrix}$, with consistent-dimensional identity and zero sub-blocks: $I_n, I_m, 0_{n \times m}, 0_{m \times n}$ respectively. We start by calculating

$$L L^{-1} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} (A - BD^{-1}C)^{-1} & -(A - BD^{-1}C)^{-1}BD^{-1} \\ -(D - CA^{-1}B)^{-1}CA^{-1} & (D - CA^{-1}B)^{-1} \end{bmatrix} \quad (9)$$

and

$$L^{-1}L = \begin{bmatrix} (A - BD^{-1}C)^{-1} & -(A - BD^{-1}C)^{-1}BD^{-1} \\ -(D - CA^{-1}B)^{-1}CA^{-1} & (D - CA^{-1}B)^{-1} \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix} \quad (10)$$

by applying (7) in Lemma 1 to both matrices $(A - BD^{-1}C)^{-1} \in \mathbb{R}^{n \times n}$ and $(D - CA^{-1}B)^{-1} \in \mathbb{R}^{m \times m}$, which are present in (8) and (9), and recalling full-rank conditions not only over those matrices but also for A and D , yields the relations

$$(A - BD^{-1}C)^{-1} = A^{-1} + A^{-1}B(D - CA^{-1}B)^{-1}CA^{-1} \quad (11)$$

$$(D - CA^{-1}B)^{-1} = D^{-1} + D^{-1}C(A - BD^{-1}C)^{-1}BD^{-1} \quad (12)$$

Using (10-11) in (8-9), the following results arise:

a. for operations involved in sub-blocks of $L L^{-1}$:

$$\begin{aligned} & A(A - BD^{-1}C)^{-1} - B(D - CA^{-1}B)^{-1}CA^{-1} \\ &= A[A^{-1} + A^{-1}B(D - CA^{-1}B)^{-1}CA^{-1}] - B(D - CA^{-1}B)^{-1}CA^{-1} \\ &= I_n + B(D - CA^{-1}B)^{-1}CA^{-1} - B(D - CA^{-1}B)^{-1}CA^{-1} = I_n \\ & - A(A - BD^{-1}C)^{-1}BD^{-1} + B(D - CA^{-1}B)^{-1} \\ &= -A[A^{-1} + A^{-1}B(D - CA^{-1}B)^{-1}CA^{-1}]BD^{-1} + B(D - CA^{-1}B)^{-1} \\ &= -BD^{-1} - B(D - CA^{-1}B)^{-1}CA^{-1}BD^{-1} + B(D - CA^{-1}B)^{-1} \\ &= -BD^{-1} - B(D - CA^{-1}B)^{-1}(-CA^{-1}B + D)D^{-1} = 0_{n \times m} \end{aligned}$$

$$\begin{aligned} & C(A - BD^{-1}C)^{-1} - D(D - CA^{-1}B)^{-1}CA^{-1} \\ &= C(A - BD^{-1}C)^{-1} - D[D^{-1} + D^{-1}C(A - BD^{-1}C)^{-1}BD^{-1}]CA^{-1} \\ &= C(A - BD^{-1}C)^{-1} - CA^{-1} - C(A - BD^{-1}C)^{-1}BD^{-1}CA^{-1} \\ &= C(A - BD^{-1}C)^{-1}[A - BD^{-1}C]A^{-1} - CA^{-1} = 0_{m \times n}; \\ & - C(A - BD^{-1}C)^{-1}BD^{-1} + D(D - CA^{-1}B)^{-1} \\ &= -C(A - BD^{-1}C)^{-1}BD^{-1} + D[D^{-1} + D^{-1}C(A - BD^{-1}C)^{-1}BD^{-1}] \\ &= -C(A - BD^{-1}C)^{-1}BD^{-1} + I_m + C(A - BD^{-1}C)^{-1}BD^{-1} = I_m; \end{aligned}$$

thus,

$$L L^{-1} = I_{(n+m)} \quad (13)$$

2. for operations involved in sub-blocks of $L^{-1}L$:

$$\begin{aligned} & (A - BD^{-1}C)^{-1}A - (A - BD^{-1}C)^{-1}BD^{-1}C \\ &= (A - BD^{-1}C)^{-1}[A - BD^{-1}C] = I_n; \\ & (A - BD^{-1}C)^{-1}B - (A - BD^{-1}C)^{-1}BD^{-1}D = 0_{n \times m}; \end{aligned}$$

$$-(D - CA^{-1}B)^{-1}CA^{-1}A + (D - CA^{-1}B)^{-1}C = 0_{m \times n};$$

$$-(D - CA^{-1}B)^{-1}CA^{-1}B + (D - CA^{-1}B)^{-1}D$$

$$= -(D - CA^{-1}B)^{-1}[-CA^{-1}B + D] = I_m;$$

thus, $L^{-1}L = I_{(n+m)}$.

4.3. Generalized-Inverse

The concept of Generalized-Inverse is an extension of a matrix inversion operations applied to non-singular rectangular matrices [17-18]. For notation purposes and without loss of generalization, $\rho(G)$ and G^T denote the rank of a rectangular matrix $G \in M^{m \times n}$, and $G^T = G^H$ is the transpose-conjugate of G (when $M = \mathbb{C} \rightarrow G \in \mathbb{C}^{m \times n}$) or $G^T = G^T$ is the transpose of G (when $M = \mathbb{R} \rightarrow G \in \mathbb{R}^{m \times n}$), respectively.

Definition 2. Let $G \in M^{m \times n}$ and $0 \leq \rho(G) \leq \min(m, n)$. Then, there exists a matrix $G^\dagger \in M^{n \times m}$ (identified as the Generalized-Inverse), such that it satisfies several conditions for the following cases:

case i: if $m > n$ and $0 \leq \rho(G) \leq \min(m, n) \Rightarrow \rho(G) = n$, then there exists a unique matrix $G^\dagger \doteq G^+ \in M^{n \times m}$ (identified as Left-Pseudoinverse: LPI) such that $G^+G = I_n$, satisfying: a) $GG^+G = G$, and b) $G^+GG^+ = G^+$. Therefore, the LPI matrix is proposed as $G^+ = (G^T G)^{-1} G^T$.

case ii: if $m = n$ and $\det(G) \neq 0 \Leftrightarrow \rho(G) = n$, then there exists a unique matrix $G^\dagger \doteq G^{-1} \in M^{n \times n}$ (identified as Inverse) such that $G^{-1}G = GG^{-1} = I_n$.

case iii: if $m < n$ and $0 \leq \rho(G) \leq \min(m, n) \Rightarrow \rho(G) = m$, then there exists a unique matrix $G^\dagger \doteq G^- \in M^{n \times m}$ (identified as Right-Pseudoinverse: RPI) such that $GG^- = I_m$, satisfying: a) $GG^-G = G$, and b) $G^-GG^- = G^-$. Therefore, the RPI matrix is proposed as $G^- = G^T (GG^T)^{-1}$. ■

Given the mathematical structure for G^\dagger provided in Definition 2, it can be easily validated that: 1) For a LPI matrix stipulated in case i, $GG^+G = G$ and $G^+GG^+ = G^+$ with $G^+ = (G^T G)^{-1} G^T$; 2) For a RPI matrix stipulated in case iii, $GG^-G = G$ and $G^-GG^- = G^-$ with $G^- = G^T (GG^T)^{-1}$; iii) For the Inverse in case ii, $G^{-1}G = (G^T G)^{-1} G^T = G^T (GG^T)^{-1} = G^{-1}$. For a uniqueness test for all cases, assume the existence of matrices $G_1^\dagger \in M^{n \times m}$ and $G_2^\dagger \in M^{n \times m}$ such that $G_1^\dagger G = I_n$ and $G_2^\dagger G = I_n$ (for case i), and $GG_1^\dagger = I_m$ and $GG_2^\dagger = I_m$ (for case iii). Notice immediately, $(G_1^\dagger - G_2^\dagger)G = 0_n$ (for case i) and $G(G_1^\dagger - G_2^\dagger) = 0_m$ (for case iii), which obligates $G_1^\dagger = G_2^\dagger$ for both cases, because of full-rank properties over G . Clearly, case ii is a particular consequence of cases i and iii.

5. The MIMO channel matrix

The MIMO channel matrix is the mathematical representation for modeling the degradation phenomena presented in the RFC scenario presented in (2). The elements h_{ij} in $H \in \mathbb{C}^{n_R \times n_T}$ represent a time-invariant transfer function (possessing spectral information about magnitude and phase profiles) between a j -th transmitter and an i -th receiver antenna. Once again, dynamical properties of physical phenomena ⁴ such as path-loss, shadowing, multipath, Doppler spreading, coherence time, absorption, reflection, scattering, diffraction, basestation-user motion, antenna's physical properties-dimensions, information correlation, associated with a slow-flat quasi-static RFC scenario (proper of a non-LOS indoor wireless environments) are highlighted into a statistical model represented by matrix H . For H^\dagger purposes, CSI is a necessary feature required at the reception part in (2), as well as the $n_R \geq n_T$ condition. Table 1 provides several $n_R > n_T$ MIMO channel matrix realizations for RFC-based environments [19-21]. On table 1: a) $\text{MIMO}(n_R, n_T)$: refers to the MIMO communication link configuration, i.e. amount of receiver-end and transmitter-end elements; b) H_m : refers to a MIMO channel matrix realization; c) H_m^+ : refers to the corresponding LPI, computed as $H_m^+ = (H_m^H H_m)^{-1} H_m^H$; d) h : blockwise matrix version for H_m ; e) h^+ : refers to the corresponding LPI, computed as $h^+ = (h^T h)^{-1} h^T$. As an additional point of analysis, full-rank properties over H and h (and thus the existence of matrices H^+ , H^{-1} , h^+ , and h^{-1}) are validated and corroborated through a MATLAB simulation-driven model regarding frequency-selective and time-invariant properties for several RFC-based scenarios at different MIMO configurations. Experimental data were generated upon 10^6 MIMO channel matrix realizations. As illustrated in figure 3, a common pattern is found regarding the statistical evolution for full-rank properties of H and h with $n_R \geq n_T$ at several typical MIMO configurations, for instance, $\text{MIMO}(2, 2)$, $\text{MIMO}(4, 2)$, and $\text{MIMO}(4, 4)$. It is plotted therein $\text{REAL}(H, h)$ against $\text{IMAG}(H, h)$, where each axis label denote respectively the real and imaginary parts of: a) $\det(H)$ and $\det(h)$ when $n_R = n_T$, and b) $\det(H^H H)$ and $\det(h^T h)$ when. Blue crosses indicate the behavior of $\rho(H)$ related to $\det(H)$ and $\det(H^H H)$ ($\det(H)$ legend on top-left margin), while red crosses indicate the behavior of $\rho(h)$ related to $\det(h)$ and $\det(h^T h)$ ($\det(h)$ legend on top-left margin). The black-circled zone intersected with black-dotted lines locates the $0 + j0$ value. As depicted on figures (4)-(5), a closer glance at this statistical behavior reveals a prevalence on full-rank properties of H and h , meaning that non of the determinants $\det(H)$, $\det(h)$, $\det(H^H H)$ and $\det(h^T h)$ is equal to zero (behavior enclosed by the light-blue region and delimited by blue/red-dotted lines).

⁴ We suggest the reader consulting references [11-16] for a detail and clear explanation on these narrowband and wideband physical phenomena presented in wireless MIMO communication systems.

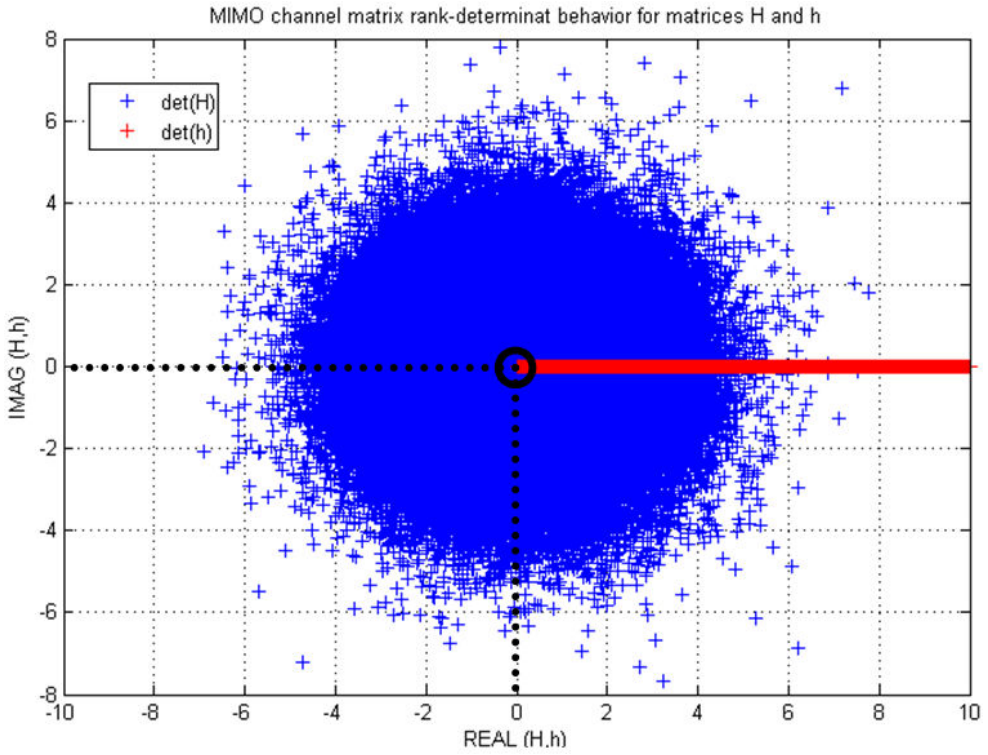


Figure 3. MIMO channel matrix rank-determinant behavior for several realizations for H and h . This statistical evolution is a common pattern found for several MIMO configurations involving slow-flat quasi-static RFC-based environments with $n_R \geq n_T$.

$\text{MIMO}(n_x, n_t)$	H_m	H_m^+
$\text{MIMO}(2,2)$	$\begin{bmatrix} 1.12 & 0.15 \\ 0.23 & 0.96 \end{bmatrix}$	$\begin{bmatrix} 0.922 & -0.144 \\ -0.221 & 1.076 \end{bmatrix}$
$\text{MIMO}(4,2)$	$\begin{bmatrix} 0.85 - j0.47 & -0.06 + j0.34 \\ -0.37 - j0.72 & 0.94 - j0.28 \\ -0.06 + j0.45 & 1.14 + j1.13 \\ -0.91 + j1.03 & 1.07 - j0.02 \end{bmatrix}$	$\begin{bmatrix} 0.256 + j0.115 & -0.06 + j0.26 \\ 0.051 - j0.082 & 0.217 + j0.116 \\ 0.088 - j0.15 & -0.206 - j0.258 \\ 0.234 - j0.275 & 0.15 - j0.018 \end{bmatrix}$
$\text{MIMO}(n_x, n_t)$	h	h^+
$\text{MIMO}(2,2)$	$\begin{bmatrix} 1.12 & 0.15 & 0 & 0 \\ 0.23 & 0.96 & 0 & 0 \\ 0 & 0 & 1.12 & 0.15 \\ 0 & 0 & 0.23 & 0.96 \end{bmatrix}$	$\begin{bmatrix} 0.922 & -0.144 & 0 & 0 \\ -0.221 & 1.076 & 0 & 0 \\ 0 & 0 & 0.922 & -0.144 \\ 0 & 0 & -0.221 & 1.076 \end{bmatrix}$
$\text{MIMO}(4,2)$	$\begin{bmatrix} 0.85 & -0.06 & 0.47 & -0.34 \\ -0.37 & 0.94 & 0.72 & 0.28 \\ -0.06 & 1.14 & -0.45 & -1.13 \\ -0.91 & 1.07 & -1.03 & 0.02 \\ -0.47 & 0.34 & 0.85 & -0.06 \\ -0.72 & -0.28 & -0.37 & 0.94 \\ 0.45 & 1.13 & -0.06 & 1.14 \\ 1.03 & -0.02 & -0.91 & 1.07 \end{bmatrix}$	$\begin{bmatrix} 0.256 & -0.06 & 0.088 & -0.206 \\ 0.051 & 0.217 & 0.234 & 0.15 \\ 0.115 & 0.26 & -0.15 & -0.258 \\ -0.082 & 0.116 & -0.275 & -0.018 \\ -0.115 & -0.26 & 0.15 & 0.258 \\ 0.082 & -0.116 & 0.275 & 0.018 \\ 0.256 & -0.06 & 0.088 & -0.206 \\ 0.051 & 0.217 & 0.234 & 0.15 \end{bmatrix}$

Table 1. MIMO channel matrix realizations for several MIMO communication link configurations at slow-flat quasi-static RFC scenarios.

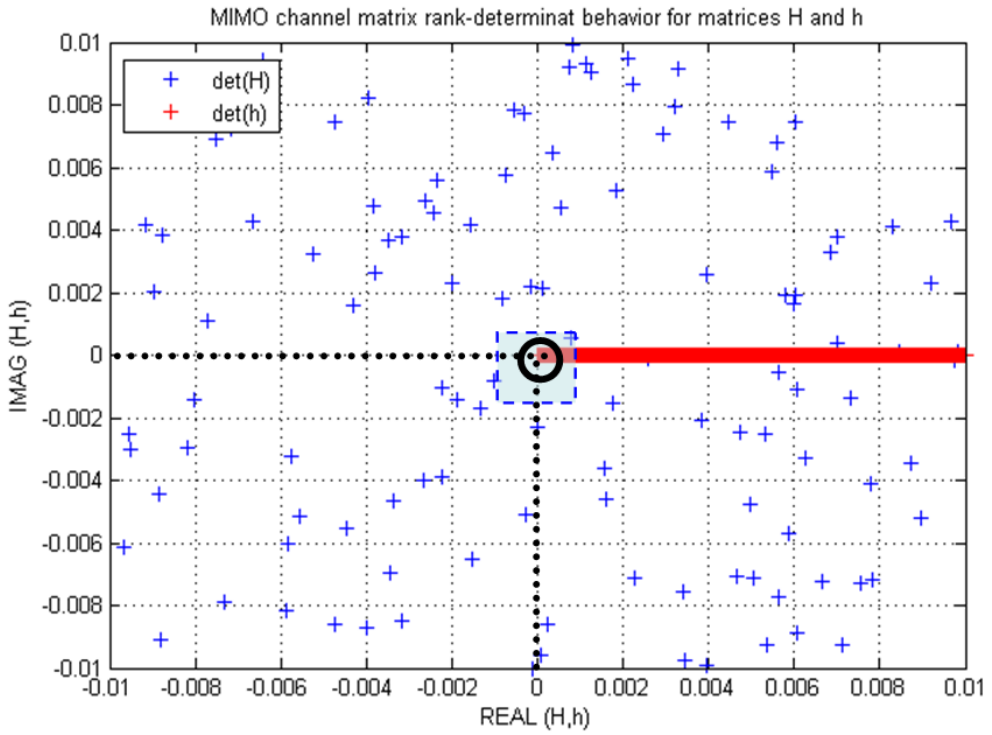


Figure 4. MIMO channel matrix rank-determinant behavior for several realizations for H . Full-rank properties for H and $H^H H$ prevail for RFC-based environments (light-blue region delimited by blue-dotted lines).

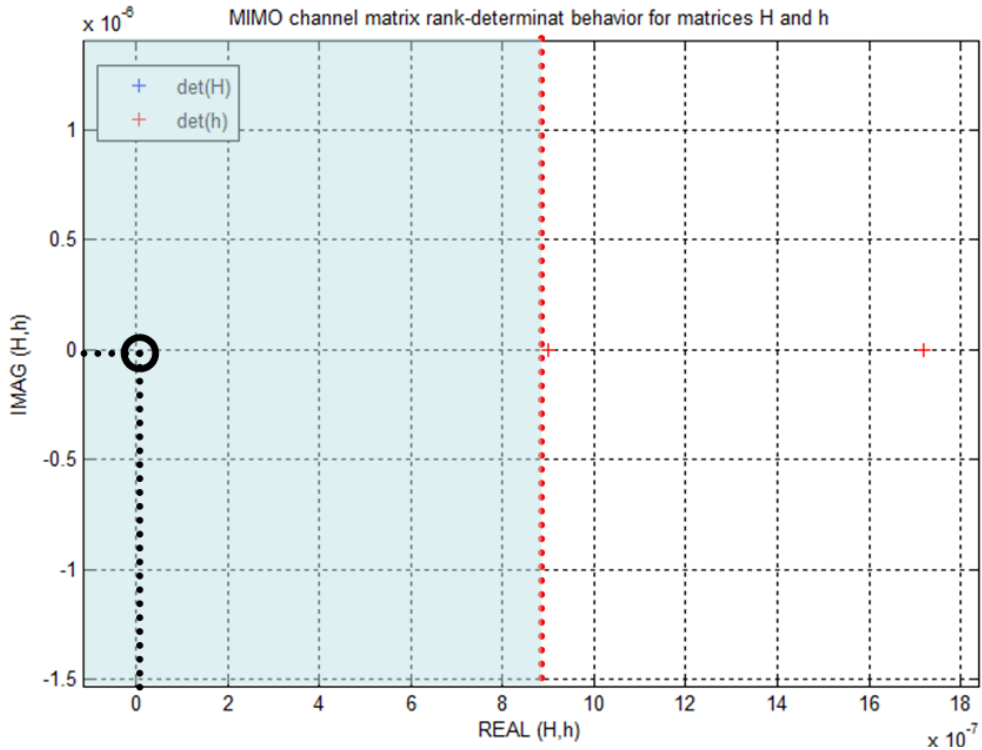


Figure 5. MIMO channel matrix rank-determinant behavior for several realizations for h . Full-rank properties for hand $h^T h$ prevail for RFC-based environments (light-blue region delimited by red-dotted line).

6. Proposed algorithm

The proposal for a novel algorithm for computing a LPI matrix $h^+ \in \mathbb{R}^{2n_T \times 2n_R}$ (with $n_R \geq n_T$) is based on the block-matrix structure of h as exhibited in (4). This idea is an extension of the approach presented in [22]. The existence for this Generalized-Inverse matrix is supported on the statistical properties of the slow-flat quasi-static RFC scenario which impact directly on the singularity of H at every MIMO channel matrix realization. Keeping in mind that other approaches attempting to solve the block-matrix inversion problem [7-10] requires several constraints and conditions, the subsequent proposal does not require any restriction at all mainly due to the aforementioned properties of H . From (4), it is suggested that $\begin{bmatrix} x^r \\ x^i \end{bmatrix}$ is

somehow related to $\begin{bmatrix} \Re\{H^+\} & -\Im\{H^+\} \\ \Im\{H^+\} & \Re\{H^+\} \end{bmatrix} \cdot Y$; hence, calculating h^+ will lead to this solution.

Let $A = H^T$ and $B = H^i$. It is known a priori that $\rho(A + jB) = n_T$. Then $h = \begin{bmatrix} A & -B \\ B & A \end{bmatrix}$ with $\rho(h) = 2n_T = N_i$. Define the matrix $\tilde{\Omega}$ as $\tilde{\Omega} \doteq h^T h \in \mathbb{R}^{N_i \times N_i}$, where $\tilde{\Omega} = \begin{bmatrix} M & -L \\ L & M \end{bmatrix}$ with $M = A^T A + B^T B \in \mathbb{R}^{n_r \times n_r}$, $L = A^T B - (A^T B)^T \in \mathbb{R}^{n_r \times n_r}$, and $\rho(\tilde{\Omega}) = N_i$ as a direct consequence from $2n_r \geq 2n_T \rightarrow N_r \geq N_i$. It can be seen that

$$h^+ = \tilde{\Omega}^{-1} h^T \in \mathbb{R}^{N_i \times N_i} \quad (15)$$

For simplicity, matrix operations involved in (12) require classic multiply-and-accumulate operations between row-entries of $\tilde{\Omega}^{-1} \in \mathbb{R}^{N_i \times N_i}$ and column-entries of $h^T \in \mathbb{R}^{N_i \times N_i}$. Notice immediately that the critical and essential task of computing h^+ relies on finding the block matrix inverse $\tilde{\Omega}^{-1}$. The strategy to be followed in order to solve $\tilde{\Omega}^{-1}$ in (12) will consist of the following steps: 1) the proposition of partitioning $\tilde{\Omega}$ without any restriction on rank-deficiency over inner matrix sub-blocks; 2) the definition of iterative multiply-and-accumulate operations within sub-blocks comprised in $\tilde{\Omega}$; 3) the recursive definition for compacting the overall blockwise matrix inversion. Keep in mind that matrix $\tilde{\Omega}$ can be also viewed as

$$\tilde{\Omega} = \begin{bmatrix} \tilde{\omega}_{1,1} & \dots & \tilde{\omega}_{1,N_i} \\ \vdots & \ddots & \vdots \\ \tilde{\omega}_{N_i,1} & \dots & \tilde{\omega}_{N_i,N_i} \end{bmatrix}. \text{ The symmetry presented in } \tilde{\Omega} = \begin{bmatrix} M & -L \\ L & M \end{bmatrix} \text{ will motivate the de-}$$

velopment for the pertinent LPI-based algorithm. From (12) and by the use of Lemma 1 it can be concluded that $\tilde{\Omega}^{-1} = \begin{bmatrix} Q & P \\ -P & Q \end{bmatrix}$, where $Q = (M + L M^{-1} L)^{-1} \in \mathbb{R}^{n_r \times n_r}$, $P = QX \in \mathbb{R}^{n_r \times n_r}$, and $X = L M^{-1} \in \mathbb{R}^{n_r \times n_r}$. Interesting enough, full-rank is identified at each matrix sub-block in the main diagonal of $\tilde{\Omega}$ (besides $\rho(Q) = n_r$). This structural behavior serves as the leitmotiv for the construction of an algorithm for computing the blockwise inverse $\tilde{\Omega}^{-1}$. Basically speaking and concerning step 1) of this strategy, the matrix partition procedure obeys the assignments (13-16) defined as:

$$W_k = \begin{bmatrix} \tilde{\omega}_{N_i-(2k+1),N_i-(2k+1)} & \tilde{\omega}_{N_i-(2k+1),N_i-2k} \\ \tilde{\omega}_{N_i-2k,N_i-(2k+1)} & \tilde{\omega}_{N_i-2k,N_i-2k} \end{bmatrix} \in \mathbb{R}^{2 \times 2} \quad (16)$$

$$X_k = \begin{bmatrix} \tilde{\omega}_{N_i-(2k+1),N_i-(2k-1)} & \dots & \tilde{\omega}_{N_i-(2k+1),N_i} \\ \tilde{\omega}_{N_i-2k,N_i-(2k-1)} & \dots & \tilde{\omega}_{N_i-2k,N_i} \end{bmatrix} \in \mathbb{R}^{2 \times 2k} \quad (17)$$

5 Notice that . Moreover, , where and .

$$Y_k = \begin{bmatrix} \tilde{\omega}_{N_i-(2k-1),N_i-(2k+1)} & \tilde{\omega}_{N_i-(2k-1),N_i-2k} \\ \vdots & \vdots \\ \tilde{\omega}_{N_i,N_i-(2k+1)} & \tilde{\omega}_{N_i,N_i-2k} \end{bmatrix} \in \mathbb{R}^{2k \times 2} \quad (18)$$

$$Z_0 = \begin{bmatrix} \tilde{\omega}_{N_i-1,N_i-1} & \tilde{\omega}_{N_i-1,N_i} \\ \tilde{\omega}_{N_i,N_i-1} & \tilde{\omega}_{N_i,N_i} \end{bmatrix} \in \mathbb{R}^{2 \times 2} \quad (19)$$

The matrix partition over $\tilde{\Omega}$ obeys the index $k = 1 : 1 : (N_i/2 - 1)$. Because of the even-rectangular dimensions of $\tilde{\Omega}$, matrix $\tilde{\Omega}$ owns exactly an amount of $N_i/2 = n_T$ sub-block matrices of 2×2 dimension along its main diagonal. Interesting enough, due to RFC-based environment characteristics studied in (1) and (4), it is found that:

$$\rho(W_k) = \rho(Z_0) = 2 \quad (20)$$

After performing these structural characteristics for $\tilde{\Omega}$, and with the use of (13-16), step 2) of the strategy consists of the following iterative operations also indexed by $k = 1 : 1 : (N_i/2 - 1)$, in the sense of performing:

$$\varphi_k = W_k - X_k Z_{k-1}^{-1} Y_k \quad (21)$$

$$\alpha_k = \varphi_k^{-1} X_k Z_{k-1}^{-1} \quad (22)$$

$$\theta_k = Z_{k-1}^{-1} + Z_{k-1}^{-1} Y_k \alpha_k \quad (23)$$

Here: $Z_{k-1}^{-1} \in \mathbb{R}^{2k \times 2k}$, $\varphi_k \in \mathbb{R}^{2 \times 2}$, $\alpha_k \in \mathbb{R}^{2 \times 2k}$, and $\theta_k \in \mathbb{R}^{2k \times 2k}$. Steps stated in (18-20) help to construct intermediate sub-blocks as

$$\tilde{\Omega}_k = \begin{bmatrix} W_k & X_k \\ Y_k & Z_{k-1} \end{bmatrix}_{\substack{2 \times 2 & 2 \times 2k \\ 2k \times 2 & 2k \times 2k}} \rightarrow \tilde{\Omega}_k^{-1} = \begin{bmatrix} \varphi_k^{-1} & -\alpha_k \\ -\theta_k Y_k W_k^{-1} & \theta_k \end{bmatrix}_{\substack{2 \times 2 & 2 \times 2k \\ 2k \times 2 & 2k \times 2k}} \quad (24)$$

The dimensions of each real-valued sub-block in (21) are indicated consistently⁶. For step 3) of the strategy, a recursion step $Z_k^{-1}(Z_{k-1}^{-1})$ is provided in terms of the assignment $Z_k^{-1} = \tilde{\Omega}_k^{-1} \in \mathbb{R}^{2(k+1) \times 2(k+1)}$. Clearly, only inversions of W_k , Z_0 , and φ_k (which are 2×2 matrices,

⁶ Matrix structure given in (21) is directly derived from applying Equation (6), and by the use of Lemma 1 as . See that this expansion is preferable instead of , which is undesirable due to an unnecessary matrix operation overhead related to computing , e.g. inverting , which comes preferably from the recursion.

yielding correspondingly W_k^{-1} , Z_0^{-1} , and φ_k^{-1} are required to be performed throughout this iterative-recursive process, unlike the operation linked to Z_{k-1}^{-1} , which comes from a previous updating step associated with the recursion belonging to Z_k^{-1} . Although $\rho(\tilde{\Omega}) = N_t$ assures the existence of $\tilde{\Omega}^{-1}$, full-rank requirements outlined in (17) and non-zero determinants for (18) are strongly needed for this iterative-recursive algorithm to work accordingly. Also, full-rank is expected for every recursive outcome related to $Z_k^{-1}(Z_{k-1}^{-1})$. Again, thank to the characteristics of the slow-flat quasi-static RFC-based environment in which these operations are involved among every MIMO channel matrix realization, conditions in (17) and full-rank of (18) are always satisfied. These issues are corroborated with the aid of the same MATLAB-based simulation framework used to validate full-rank properties over H and h . The statistical evolution for the determinants for W_k , Z_0 , and φ_k , and the behavior of singularity within the $Z_k^{-1}(Z_{k-1}^{-1})$ recursion are respectively illustrated in figures (6)-(8). MIMO(2, 2), MIMO(4, 2), and MIMO(4, 4) were the MIMO communication link configurations considered for these tests. These simulation-driven outcomes provide supportive evidence for the proper functionality of the proposed iterative-recursive algorithm for computing $\tilde{\Omega}^{-1}$ involving matrix sub-block inversions. On each figure, the statistical evolution for the determinants associated with Z_0 , W_k , φ_k , and $Z_k^{-1}(Z_{k-1}^{-1})$ are respectively indicated by labels $\det(Z_0)$, $\det(W_k)$, $\det(\varphi_k)$, and $\det(iZ_k, iZ_{k-1})$, while the light-blue zone at bottom delimited by a red-dotted line exhibits the gap which marks the avoidance in rank-deficiency over the involved matrices. The zero-determinant value is marked with a black circle.

The next point of analysis for the behavior of the h^+ LPI-based iterative-recursive algorithm is complexity, which in essence will consist of a demand in matrix partitions (amount of matrix sub-blocks: PART) and arithmetic operations (amount of additions-subtractions: ADD-SUB; multiplications: MULT; and divisions: DIV). Let PART-mtx and ARITH-ops be the nomenclature for complexity cost related to matrix partitions and arithmetic operations, respectively. Without loss of generalization, define $C[*]$ as the complexity in terms of the costs PART-mtx and ARITH-ops belonging to operations involved in $*$. Henceforth, $C[h^+] = C[\tilde{\Omega}^{-1}] + C[\tilde{\Omega}^{-1} \cdot h^T]$ denotes the cost of computing h^+ as the sum of the costs of inverting $\tilde{\Omega}$ and multiplying $\tilde{\Omega}^{-1}$ by h^T . It is evident that: a) $C[\tilde{\Omega}^{-1} \cdot h^T]$ implies PART=0 and ARITH-ops itemized into $MULT=8n_R n_T^2$, $ADD-SUB=4n_R n_T(2n_T - 1)$, and $DIV=0$; b) $C[\tilde{\Omega}^{-1}] = C[h^T \cdot h] + C[(h^T h)^{-1}]$. Clearly, $C[h^T \cdot h]$ demands no partitions at all, but with a ARITH-ops cost of $MULT=8n_R n_T^2$, and $ADD-SUB=4(2n_R - 1)n_T^2$. However, the principal complexity relies critically on $C[(h^T h)^{-1}]$, which is the backbone for h^+ , as presented in [22]. Table 2 summarizes these complexity results. For this treatment, $C[(h^T h)^{-1}]$ consists of $3n_T - 2$ partitions, $MULT = \sum_{k=1}^{n_T-1} C_k^I + 6$, $ADD-SUB = \sum_{k=1}^{n_T-1} C_k^{II} + 1$, and $DIV = \sum_{k=1}^{n_T-1} C_k^{III} + 1$. The ARITH-ops cost depends on C_k^I , C_k^{II} , and C_k^{III} ; the constant factors for each one of these items are proper of the complexity presented in $C[Z_0^{-1}]$. The remain of the complexities, i.e. C_k^I , C_k^{II} , and C_k^{III} , are

calculated according to the iterative steps defined in (18-20) and (21), particularly expressed in terms of

$$C[\varphi_k^{-1}] + C[-\alpha_k] + C[-\theta_k Y_k W_k^{-1}] + C[\theta_k] \tag{25}$$

(22)

It can be checked out that: a) no PART-mtx cost is required; b) the ARITH-ops cost employs (22) for each item, yielding: $C_k^I = 40k^2 + 24k + 12$ (for MULT), $C_k^{II} = 40k^2 + 2$ (for ADD_SUB), and $C_k^{III} = 2$ (for DIV).

An illustrative application example is given next. It considers a MIMO channel matrix realization obeying statistical behavior according to (1) and a MIMO(4, 4) configuration:

$$H = \begin{bmatrix} -0.3059 + j0.7543 & -0.8107 + j0.2082 & 0.2314 - j0.4892 & -0.416 - j1.0189 \\ -1.1777 + j0.0419 & 0.8421 - j0.9448 & 0.1235 + j0.6067 & 1.5437 + j0.4039 \\ 0.0886 - j0.0676 & 0.8409 + j0.5051 & -0.132 + j0.8867 & -0.0964 - j0.2828 \\ 0.2034 - j0.5886 & -0.0266 + j1.148 & 0.5132 - j1.1269 & 0.0806 + j0.4879 \end{bmatrix} \in \mathbb{C}^{4 \times 4}$$

with $\rho(H) = 4$. As a consequence,

$$\tilde{\Omega} = \begin{bmatrix} 2.4516 & -1.2671 & 0.1362 & -2.7028 & 0 & -1.9448 & 0.6022 & -0.2002 \\ -1.2671 & 4.5832 & -1.7292 & 1.3776 & 1.9448 & 0 & -1.229 & -2.4168 \\ 0.1362 & -1.7292 & 3.0132 & 0.0913 & -0.6022 & 1.229 & 0 & 0.862 \\ -2.7028 & 1.3776 & 0.0913 & 4.0913 & 0.2002 & 2.4168 & -0.862 & 0 \\ 0 & 1.9448 & -0.6022 & 0.2002 & 2.4516 & -1.2671 & 0.1362 & -2.7028 \\ -1.9448 & 0 & 1.229 & 2.4168 & -1.2671 & 4.5832 & -1.7292 & 1.3776 \\ 0.6022 & -1.229 & 0 & -0.862 & 0.1362 & -1.7292 & 3.0132 & 0.0913 \\ -0.2002 & -2.4168 & 0.862 & 0 & -2.7028 & 1.3776 & 0.0913 & 4.0913 \end{bmatrix}$$

$\in \mathbb{R}^{8 \times 8}$ with

$$\rho(\tilde{\Omega}) = 8 \tag{26}$$

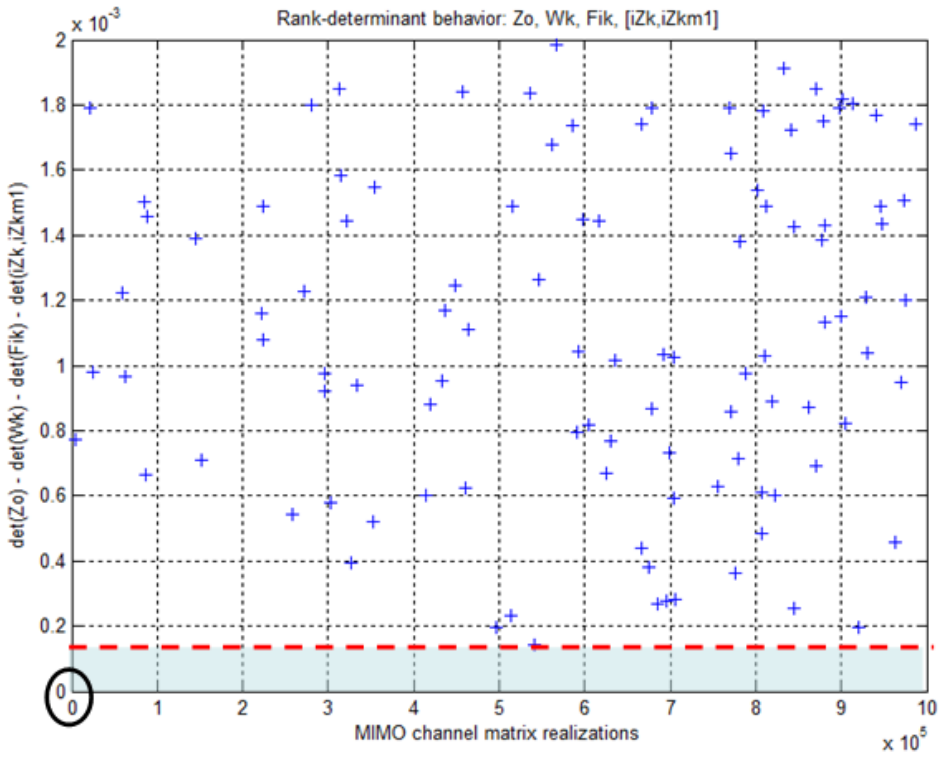


Figure 6. Statistical evolution of the rank-determinant behaviour concerning $Z_0, W_{k \times k}$, and $Z_k^{-1}(Z_{k-1}^{-1})$ for a MIMO(2, 2) configuration.

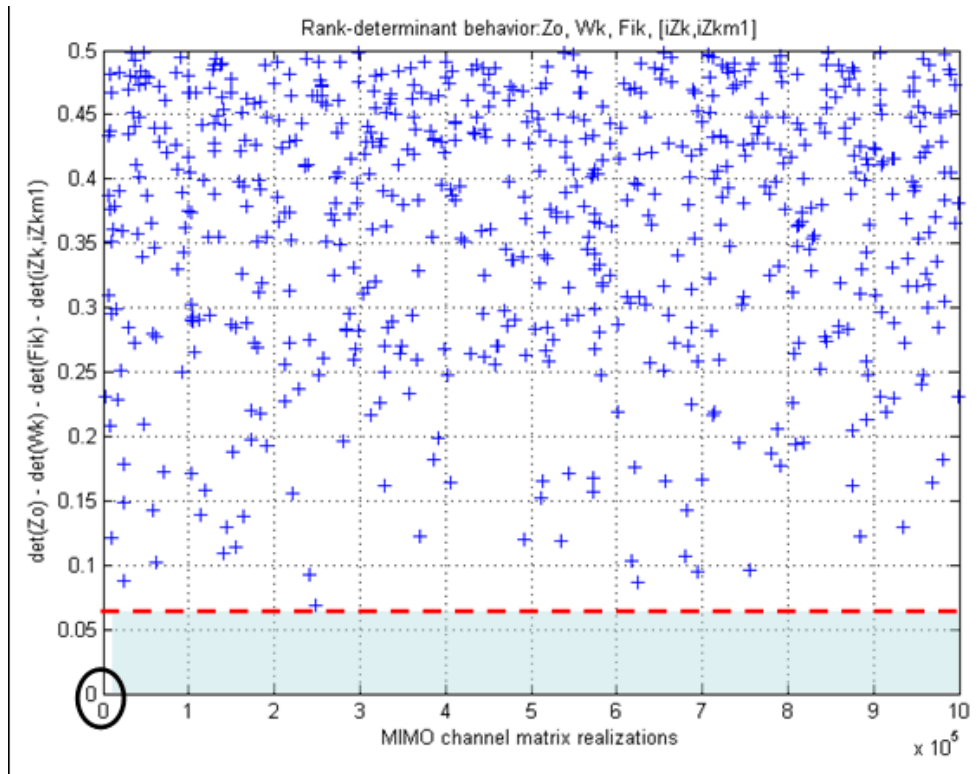


Figure 7. Statistical evolution of the rank-determinant behaviour concerning Z_0, W_k, F_{ik} , and $Z_k^{-1}(Z_{k-1}^{-1})$ for a MIMO(4, 2) configuration.

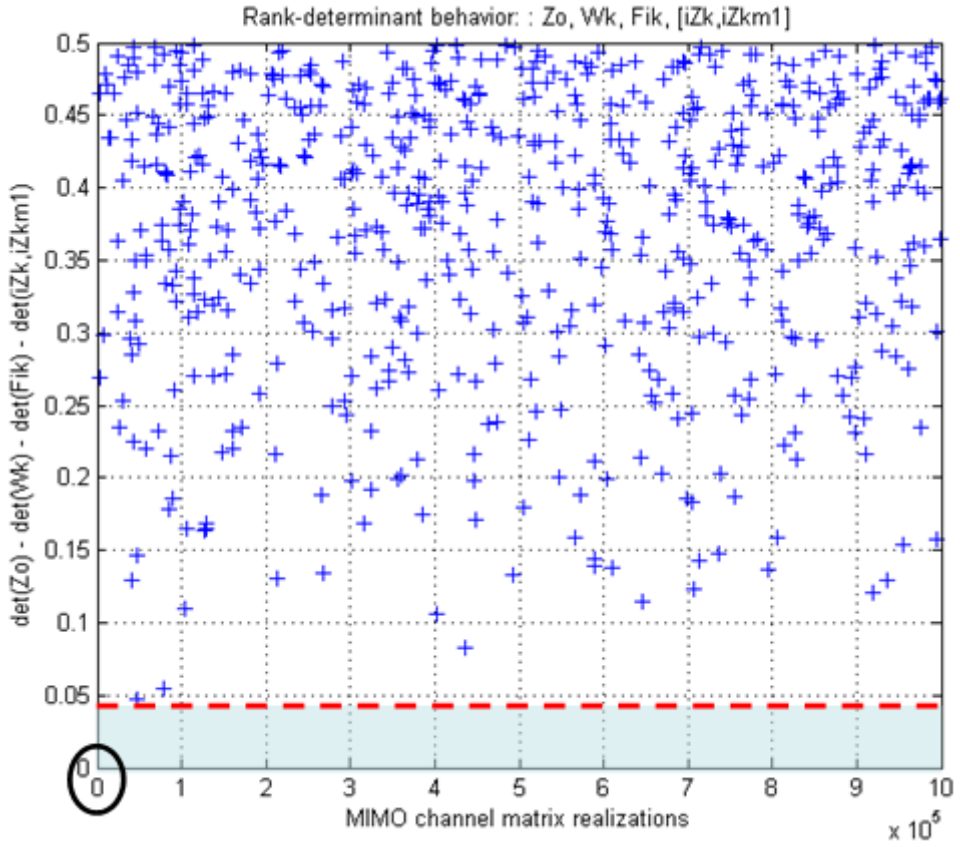


Figure 8. Statistical evolution of the rank-determinant behaviour concerning Z_0, W_k, F_{ik} , and $Z_k^{-1}(Z_{k-1}^{-1})$ for a MIMO(4, 4) configuration.

$C[h^+]$				
$C[\tilde{\Omega}^{-1} \cdot h^T]$	$C[\tilde{\Omega}^{-1}]$			
	$C[h^T \cdot h]$	$C[(h^T h)^{-1}]$		
0	0	$3n_T - 2$	PART-mtx cost [in PART operations]	
$8n_R n_T^2$	$8n_R n_T^2$	$\sum_{k=1}^{n_T-1} C_k^I + 6$	MULT	ARITH-ops cost [in amount of itemized operations]
$4n_R n_T (2n_T - 1)$	$4(2n_R - 1)n_T^2$	$\sum_{k=1}^{n_T-1} C_k^{II} + 1$	ADD-SUBB	
0	0	$\sum_{k=1}^{n_T-1} C_k^{III} + 1$	DIV	

	Based on (22), and for		
	$C_k^I :$	$C_k^{II} :$	$C_k^{III} :$
$C[\phi_k^{-1}]$	$8k^2 + 8k + 6$	$8k^2 + 4k + 1$	1
$C[-\alpha_k]$	$8k^2 + 8k$	$8k^2$	0
$C[-\theta_k Y_k W_k^{-1}]$	$8k^2 + 8k + 6$	$8k^2 + 1$	1
$C[\theta_k]$	$16k^2$	$16k^2 - 4k$	0

Table 2. Complexity cost results of the LPI-based iterative-recursive algorithm for h^+ .

Applying partition criteria (13-16) and given $k = 1 : 1 : 3$, the following matrix sub-blocks are generated:

$$W_1 = \begin{bmatrix} 2.4516 & -1.2671 \\ -1.2671 & 4.5832 \end{bmatrix},$$

$$X_1 = \begin{bmatrix} 0.1362 & -2.7028 \\ -1.7292 & 1.3776 \end{bmatrix}, Y_1 = \begin{bmatrix} 0.1362 & -1.7292 \\ -2.7028 & 1.3776 \end{bmatrix}, Z_0 = \begin{bmatrix} 3.0132 & 0.0913 \\ 0.0913 & 4.0913 \end{bmatrix},$$

$$W_2 = \begin{bmatrix} 3.0132 & 0.0913 \\ 0.0913 & 4.0913 \end{bmatrix}$$

$$X_2 = \begin{bmatrix} -0.6022 & 1.2290 & 0 & 0.862 \\ 0.2002 & 2.4168 & -0.862 & 0 \end{bmatrix}, Y_2 = \begin{bmatrix} -0.6022 & 0.2002 \\ 1.229 & 2.4168 \\ 0 & -0.862 \\ 0.862 & 0 \end{bmatrix},$$

$$W_3 = \begin{bmatrix} 2.4516 & -1.2671 \\ -1.2671 & 4.5832 \end{bmatrix}'$$

$$X_3 = \begin{bmatrix} 0.1362 & -2.7028 & 0 & -1.9448 & 0.6022 & -0.2002 \\ -1.7292 & 1.3776 & 1.9448 & 0 & -1.229 & -2.4168 \end{bmatrix}'$$

$$\text{and } Y_3 = \begin{bmatrix} 0.1362 & -1.7292 \\ -2.7028 & 1.3776 \\ 0 & 1.9448 \\ -1.9448 & 0 \\ 0.6022 & -1.229 \\ -0.2002 & -2.4168 \end{bmatrix}. \text{ Suggested by (18-20), iterative operations (23-25) are comput-}$$

ed as:

$$\varphi_1 = W_1 - X_1 Z_0^{-1} Y_1, \quad \alpha_1 = \varphi_1^{-1} X_1 Z_0^{-1}, \quad \theta_1 = Z_0^{-1} + Z_0^{-1} Y_1 \alpha_1 \tag{27}$$

$$\varphi_2 = W_2 - X_2 Z_1^{-1} Y_2, \quad \alpha_2 = \varphi_2^{-1} X_2 Z_1^{-1}, \quad \theta_2 = Z_1^{-1} + Z_1^{-1} Y_2 \alpha_2 \tag{28}$$

$$\varphi_3 = W_3 - X_3 Z_2^{-1} Y_3, \quad \alpha_3 = \varphi_3^{-1} X_3 Z_2^{-1}, \quad \theta_3 = Z_2^{-1} + Z_2^{-1} Y_3 \alpha_3 \tag{29}$$

From (21), the matrix assignments related to recursion $Z_k^{-1}(Z_{k-1}^{-1})$ produces the following intermediate blockwise matrix results:

$$Z_1^{-1}(Z_0^{-1}) = \tilde{\Omega}_1^{-1} = \begin{bmatrix} \varphi_1^{-1} & -\alpha_1 \\ -\theta_1 Y_1 W_1^{-1} & \theta_1 \end{bmatrix} = \begin{bmatrix} 1.5765 & 0.1235 & -0.0307 & 1.0005 \\ 0.1235 & 0.3332 & 0.1867 & -0.0348 \\ -0.0307 & 0.1867 & 0.4432 & -0.093 \\ 1.0005 & -0.0348 & -0.093 & 0.9191 \end{bmatrix}'$$

$$Z_2^{-1}(Z_1^{-1}) = \tilde{\Omega}_2^{-1} = \begin{bmatrix} \varphi_2^{-1} & -\alpha_2 \\ -\theta_2 Y_2 W_2^{-1} & \theta_2 \end{bmatrix}$$

$$= \begin{bmatrix} 0.4098 & 0.0879 & -0.0829 & -0.1839 & -0.0743 & -0.0775 \\ 0.0879 & 0.4355 & -0.2847 & -0.3182 & -0.0422 & -0.0985 \\ -0.0829 & -0.2847 & 1.7642 & 0.3393 & 0.0012 & 1.0686 \\ -0.1839 & -0.3182 & 0.3393 & 0.6023 & 0.2376 & 0.0548 \\ -0.0743 & -0.0422 & 0.0012 & 0.2376 & 0.4583 & -0.0738 \\ -0.0775 & -0.0985 & 1.0686 & 0.0548 & -0.0738 & 0.9499 \end{bmatrix}'$$

$$Z_3^{-1}(Z_2^{-1}) = \tilde{\Omega}_3^{-1} = \begin{bmatrix} \varphi_3^{-1} & -\alpha_3 \\ -\theta_3 Y_3 W_3^{-1} & \theta_3 \end{bmatrix}$$

$$= \begin{bmatrix} 1.9798 & 0.3808 & -0.1114 & 1.0224 & 0 & 0.3605 & 0.2524 & 0.2183 \\ 0.3808 & 0.6759 & 0.2619 & 0.0856 & -0.3605 & 0 & 0.2368 & 0.1193 \\ -0.1114 & 0.2619 & 0.5493 & -0.0218 & -0.2524 & -0.2368 & 0 & -0.0535 \\ 1.02224 & 0.0856 & -0.0218 & 0.9839 & -0.2183 & -0.1193 & 0.0535 & 0 \\ 0 & -0.3605 & -0.2524 & -0.2183 & 1.9798 & 0.3808 & -0.1114 & 1.0224 \\ 0.3605 & 0 & -0.2368 & -0.1193 & 0.3808 & 0.6759 & 0.2619 & 0.0856 \\ 0.2524 & 0.2368 & 0 & 0.0535 & -0.1114 & 0.2619 & 0.5493 & -0.0218 \\ 0.2183 & 0.1193 & -0.0535 & 0 & -0.1114 & 0.0856 & -0.0218 & 0.9839 \end{bmatrix} \cdot \text{This}$$

last recursive outcome from $Z_k^{-1}(Z_{k-1}^{-1})$ corresponds to $\widetilde{\Omega}^{-1}$, and is further used for calculating $h^+ = \widetilde{\Omega}^{-1}h^T \in \mathbb{R}^{8 \times 8}$. Moreover, notice that full-rank properties are always presented in matrices $Z_0, W_1, W_2, W_3, \varphi_1, \varphi_2, \varphi_3, Z_1^{-1}, Z_2^{-1}$, and Z_3^{-1} .

7. VLSI implementation aspects

The arithmetic operations presented in the algorithm for computing h^+ can be implemented under a modular-iterative fashion towards a VLSI (Very Large Scale of Integration) design. The partition strategy comprised in (13-16) provides modularity, while (18-20) is naturally associated with iterativeness; recursion is just used for constructing matrix-blocks in (21). Several well-studied aspects aid to implement a further VLSI architecture [23-27] given the nature of the mathematical structure of the algorithm. For instance, systolic arrays [25-27] are a suitable choice for efficient, parallel-processing architectures concerning matrix multiplications-additions. Bidimensional processing arrays are typical architectural outcomes, whose design consist basically in interconnecting processing elements (PE) among different array layers. The configuration of each PE comes from projection or linear mapping techniques [25-27] derived from multiplications and additions presented in (18-20). Also, systolic arrays tend to concurrently perform arithmetic operations dealing with the matrix concatenated multiplications $X_k Z_{k-1}^{-1} Y_k, \varphi_k^{-1} X_k Z_{k-1}^{-1}, Z_{k-1}^{-1} Y_k \alpha_k$, and $\theta_k Y_k W_k^{-1}$ presented in (18-20). Consecutive additions inside every PE can be favourably implemented via Carry-Save-Adder (CSA) architectures [23-24], while multiplications may recur to Booth multipliers [23-24] in order to reduce latencies caused by adding accumulated partial products. Divisions presented in W_k^{-1}, Z_0^{-1} , and φ_k^{-1} can be built through regular shift-and-subtract modules or classic serial-parallel subtractors [23-24]; in fact, CORDIC (Coordinate Rotate Digital Computer) processors [23] are also employed and configured in order to solve numerical divisions. The aforementioned architectural aspects offer an attractive and alternative framework for consolidating an ultimate VLSI design for implementing the h^+ algorithm without compromising the overall system data throughput (intrinsically related to operation frequencies) for it.

8. Conclusions

This chapter presented the development of a novel iterative-recursive algorithm for computing a Left-Pseudoinverse (LPI) as a Generalized-Inverse for a MIMO channel matrix within a Rayleigh fading channel (RFC). The formulation of this algorithm consisted in the following step: i) first, structural properties for the MIMO channel matrix acquired permanent full-rank due to statistical properties of the RFC scenario; ii) second, Partition-Matrix Theory was applied allowing the generation of a block-matrix version of the MIMO channel matrix; iii) third, iterative addition-multiplication operations were applied at these matrix sub-blocks in order to construct blockwise sub-matrix inverses, and recursively reusing them for obtaining the LPI. For accomplishing this purpose, required mathematical background and MIMO systems concepts were provided for consolidating a solid scientific framework to understand the context of the problem this algorithm was attempting to solve. Proper functionality for this approach was validated through simulation-driven experiments, as well as providing an example of this operation. As an additional remark, some VLSI aspects and architectures were outlined for basically implementing arithmetic operations within the proposed LPI-based algorithm.

Acknowledgement

This work was supported by CONACYT (National Council of Science and Technology) under the supervision, revision, and sponsorship of ITESM University (Instituto Tecnológico y de Estudios Superiores de Monterrey).

Author details

P. Cervantes¹, L. F. González¹, F. J. Ortiz² and A. D. García²

Instituto Tecnológico y de Estudios Superiores de Monterrey, Campus Guadalajara, ITESM University,, Mexico

Instituto Tecnológico y de Estudios Superiores de Monterrey, Campus Estado de México, ITESM University,, Mexico

References

- [1] Abramovich YI, Johnson BA, and Spencer NK(2008). Two-Dimensional Multivariate Parametric Models for Radar Applications-Part I: Maximum-Entropy Extensions for Toeplitz-Block Matrices. IEEE Transactions on Signal Processing, November 2008: 5509-5526., 56(11)

- [2] Bera TK, et al(2011). Improving the image reconstruction in Electrical Impedance Tomography (EIT) with block matrix-based Multiple Regularization (BMMR): A practical phantom study. World Congress on Information and Communication Technologies (WICT). , 2011, 1346-1351.
- [3] Kailath, T. (1980). Linear Systems. Prentice-Hall. 682 p.
- [4] Spong MW(1998). Underactuated Mechanical Systems. Control Problems in Robotics and Automation, Lecture Notes in Control and Information Sciences, Springer-Verlag: (230), 135-150.
- [5] Utkin, V., Guldner, J., & Shi, J. X. (1992). Sliding Mode Control in Electro-Mechanical Systems. CRC Press. April 1999: 338 p.
- [6] Juang J-N(1993). Applied System Identification. Prentice Hall. 400 p.
- [7] Watt SM(2006). Pivot-Free Block Matrix Inversion. Proceedings of the Eighth International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), IEEE Computer Society: 5 p.
- [8] Tian, Y., & Tanake, Y. (2009). The inverse of any two-by-two nonsingular partitioned matrix and three matrix inverse completion problems. Journal Computers & Mathematics with Applications, April 2009: 12 p., 57(8)
- [9] Choi, Y. (2009). New Form of Block Matrix Inversion. International Conference on Advanced Intelligent Mechatronics. July , 2009, 1952-1957.
- [10] Choi, Y., & Cheong, J. (2009). New Expressions of 2X2 Block Matrix Inversion and Their Application. IEEE Transactions on Automatic Control, November 2009: 2648-2653., 54(11)
- [11] Fontán FP, and Espiñera PM(2008). Modeling the Wireless Propagation Channel. Wiley. 268 p.
- [12] El -Hajjar, M., & Hanzo, L. (2010). Multifunctional MIMO Systems: A Combined Diversity and Multiplexing Design Perspective. IEEE Wireless Communications. April , 2010, 73-79.
- [13] Biglieri, E., et al. (2007). MIMO Wireless Communications. Cambridge University Press: United Kingdom. 344 p.
- [14] Jankiraman, M. (2004). Space-Time Codes and MIMO Systems. Artech House: United States. 327 p.
- [15] Biglieri, E., Proakis, J., & Shamai, S. (1998). Fading Channels: Information-Theoretic and Communications Aspects. IEEE Transactions on Information Theory, October 1998: 2619-2692., 44(6)
- [16] Almers, P., Bonek, E., Burr, A., et al. (2007). Survey of Channel and Radio Propagation Models for Wireless MIMO Systems. EURASIP Journal on Wireless Communications and Networking, January 2007: 19 p., 2011(1)

- [17] Golub GH, and Van Loan CF(1996). Matrix Computations. The Johns Hopkins University Press. 694 p.
- [18] Serre, D. (2001). Matrices: Theory and Applications. Springer Verlag. 202 p.
- [19] R&S®. Rohde & Schwarz GmbH & Co. KG. WLAN 802.11n: From SISO to MIMO. Application Note: 1MA179_9E. Available: www.rohde-schwarz.com:p.
- [20] Agilent, ©., & Technologies, Inc. (2008). Agilent MIMO Wireless LAN PHY Layer [RF] : Operation & Measurement: Application Note: 1509. Available: www.agilent.com:p.
- [21] Paul, T., & Ogunfunmi, T. (2008). Wireless LAN Comes of Age : Understanding the IEEE 802.11n Amendment. IEEE Circuits and Systems Magazine. First Quarter , 2008, 28-54.
- [22] Cervantes, P., González, V. M., & Mejía, P. A. (2009). Left-Pseudoinverse MIMO Channel Matrix Computation. 19th International Conference on Electronics, Communications, and Computers (CONIELECOMP 2009). July , 2009, 134-138.
- [23] Milos, E., & Tomas, L. (2004). Digital Arithmetic. Morgan Kauffmann Publishers. 709 p.
- [24] Parhi KK(1999). VLSI Digital Signal Processing Systems: Design and Implementation. John Wiley & Sons. 784 p.
- [25] Song SW(1994). Systolic Algorithms: Concepts, Synthesis, and Evolution. Institute of Mathematics, University of Sao Paulo, Brazil. Available: <http://www.ime.usp.br/~song/papers/cimpa.pdf>. DOIp.(10)
- [26] Kung SY(1985). VLSI Array Processors. IEEE ASSP Magazine. July , 1985, 4-22.
- [27] Jagadish, H. V., Rao, S. K., & Kailath, T. (1987). Array Architectures for Iterative Algorithms. Proceedings of the IEEE, September 1987: 1304-1321., 75(9)