Off-the-beaten-path Solutions for Decomposition-based Zero-forcing Precoding in xDSL Multi-user Downlinks

I.V. Semushin[1], **J.V. Tsyganova**[1]

[1]Ulyanovsk State University, Lev Tolstoy street 42, Ulyanovsk, Russia, 432017

Abstract. However broad the Decomposition-based Zero-forcing (DBZF) precoder acceptance may be, reducing the computational complexity of its implementation is an absolute necessity for the VDSL networking professionals. The paper digs deeper into this problem from the perspective of matrix inversion which is inherent in the very nature of the DBZF. Five strategies considered here differ in mode of action: three of them include matrix inversion, and two others drop implementing the procedure. While the baseline strategy itemized under No. 1 acts with the Gaussian LU-decomposition, strategy No. 2 deals with the Jordanian LU-decomposition thereby enabling mild reduction of the operation count. Strategy No. 3 works for more significant reduction as it operates with the elimination form of the inverse matrix. The most cost-cutting are strategies excluding the question of matrix inversion and replacing it by far more straightforward linear system solution, as it is in Strategy No. 4. An alternative strategy No. 5 uses the least squares-based square-root-type sequential system solution and it is the most accurate computational procedure when compared with other strategies.

Keywords: DSL data transmission, MIMO system, discrete multi-tone transmission, channel transfer matrix, crosstalk interference, decomposition-based ZF precoding, transmission overhead, computational complexity, matrix inversion avoidance.

1. Introduction

In the last two decades, communication technologies and transmission equipment are developing at an unprecedented pace to provide for both the residential and mobile access. Broadband wireline access networks offer promising and stable bandwidth to resident user premises. Their services include many modern network applications such as video-streaming, file sharing, telecommuting, online gaming, video-conferencing, and others. It became possible in the early 2000s with the invention of Very High-Speed Digital Subscriber Line (VDSL, approved by the International Telecommunication Union (ITU) in November 2001) and then VDSL2 (passed in February 2006).

Although passive optical networks (PON) is the most popular access network worldwide, countries with abundant copper line resources make good use of xDSL (such as VDSL and VDSL2/2+) by taking advantage of existing telephone lines. The combined technology called 'fiber-to-the-x' (FTTx), where 'x' may stand for 'N=node,' or 'C=curb,' or 'Cab=cabinet,' or 'B=building,' delivers both low deployment cost and better performance. In densely populated areas or cities, many customers are within 1.5 km of the central office (CO) or Local Exchange (LEx). In such cases, VDSL can be deployed directly from the CO. When fiber extends deeper into the network, public service carriers deploy VDSL from the optical network unit (ONU) in a configuration known as 'fiber-to-the-cabinet' (FTTCab) [1, p. 6]. In this case, PON connects the optical signal to the cabinet ONU, and then, telephone line or twisted pair (TP within a cable) will carry the signal to user premises using xDSL. That way, hybrid fiber-copper systems deploying VDSL in the last mile to the customer premises (CPs) make the optical network core closer to the customer, and so offer sufficient bandwidth retaining the edge over pure fiber networks as a more economical solution [2, p. 15], [3, p. 55].

However, a crucial problem in VDSL networks limiting both the data rate and reach of service is the phenomenon known as crosstalk. In VDSL applications, the coupling of unwanted signals from one or more TPs into another TP can take two forms: near-end crosstalk (NEXT) and far-end (FEXT) crosstalk. It is known that FEXT has more huge consequences on shorter VDSL lines: "it can dominate noise profiles" [1, p. 11]. Once the modems transmitting signals in the downstream direction are collocated at the CO, a technique of crosstalk precoding can be applied to each modem's signal before transmission. A comprehensive review of precoding techniques for digital communication systems is recently given in [4], however mostly for wireless communications. A near-optimal linear crosstalk precoder [5] helps to cope with this problem for wireline downstream VDSL. This precoder dates back to 2004 as can be seen from a broadcast draft paper version (cf. [5]). Thanks to a low complexity and no-need-for-additional-arithmetic at the receiver-side, it gained broader acceptance and further consideration as in [6, p. 34–35].

This solution termed in work last cited as Decomposition-based Zero-forcing Precoder is producing the desired effect in crosstalk cancellation, but along with this, it poses a problem of computation complexity. The point is that the precoder computations include the operation of matrix inversion and so may have a meaningful effect on VDSL power consumption as stressed in [1, p. 20]. If VDSL modems are deployed from the ONU, which is typically located in a small curbside cabinet with no cooling or temperature control, VDSL power consumption must be very low. Besides, to fit in the ONU, VDSL line cards must also be small. For the preceding reasons, networking professionals refer to reducing the complexity of big matrices inversion as a 'perennial topic' [7, pp. 39, 55–70].

The paper digs deeper into the problem of complexity of different computational algorithms related to inverse matrices: both including and avoiding their inversion in the context of precoder design. It summarizes authors' research activities and results to report the possibility of CO/ONU precoder computations off-loading against the baseline solution and to communicate risks and alternatives requiring a decision to network carrier's management.

Section 1 recalls mind to the subject. Section 2 answers the question of how to obtain the sought-for solution for the problem formulated regarding inverse matrix. Section 3 presents the baseline solution using any one of three forms of LU-decomposition. Section 4 describes the Jordanian version of LU-decomposition aimed at computing A^{-1} as precoder P. Section 5 uses the baseline LU-decomposition to obtain the elimination form of A^{-1}. Section 6 represents an off-the-beaten-path proposal for precoding without direct computing A^{-1}. Section 7 introduces another non-mainstream alternative in the form of least squares-based square root sequential system solution to avoid A^{-1}. Section 8 makes tradeoff study from the perspective of matrix inversion in precoder design. Section 9 containes the obtained results of numerical experiments conducted in MATLAB. The last section concludes the paper.

Throughout we denote subtextually:

- the complex n vector space by $\mathbb{C}^n$ and the complex $m \times n$ matrix space by $\mathbb{C}^{m\times n}$,
- the real n vector space by $\mathbb{R}^n$ and the real $m \times n$ matrix space by $\mathbb{R}^{m\times n}$,
- n-vectors of the same dimension n by lowercase latin letters,
- $n \times n$ matrices by uppercase latin letters,
- scalars by small Greek letters.

2. Channel Model and DMT Transmission

Because of the below reasons VDSL systems are treated as Multiple Input Multiple Output (MIMO) systems.

(i) VDSL system is DMT (Discrete Multi-tone Transmission)-based. In this method, the allocated frequency band (or channel) is separated into many frequency subbands (or subchannels as they are specified at times). DMT uses the fast Fourier transform (FFT) algorithm for signal modulation (before transmission) and demodulation (at receiving side). The transmission process runs on each tone k, i. e., on the k-th carrier frequency f in the k-th subchannel, $k = 1, \dots, K$ (Fig. 1).

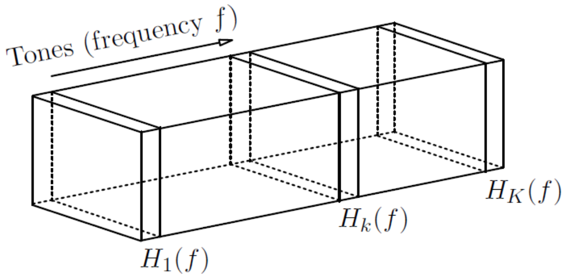

Figure 1. The channel transfer matrices $H_k(f)$ are complex-valued $N \times N$ matrices; $k = 1, \dots, K$, the number of subchannels (or tones).

(ii) Modulated data are passed to N users in parallel through N twisted pairs. As a general rule, N individual TPs are grouped in binder-groups of 4 to 10 cables, and 50 to 100 TPs are bundled together into a cable. So, N may be from two to ten hundred [8, p. 5], and thus the FEXT is rendered the most dangerous phenomenon (Fig. 2).

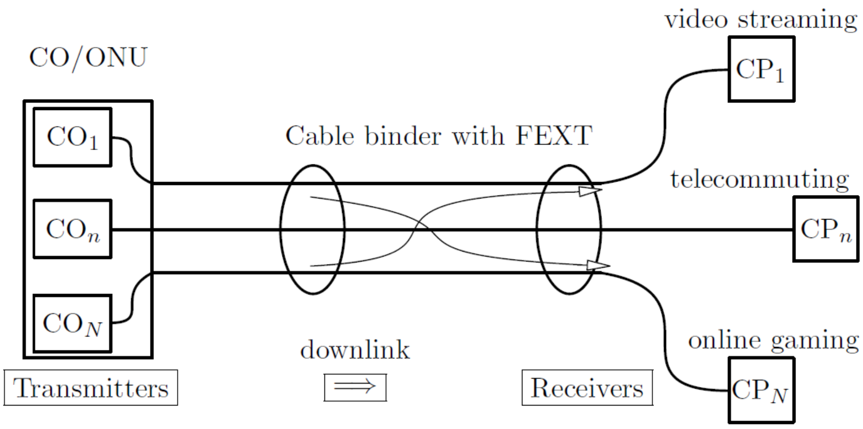

Figure 2. DSL downlink far-end crosstalk environment: $n = 1, \dots, N$, the number of lines to the end users.

The k-th subchannel is modeled by the k-th complex valued channel transfer matrix $H_k \triangleq H_k(f) \in \mathbb{C}^{N \times N}$ on each tone k (*cf.* Fig. 1). The n-th diagonal elements of H_k correspond to the direct channel coefficients of the different TPs and describe the impact of the direct channel of user n on his transmit signal. The off-diagonal elements correspond to the crosstalk interference contributions and are the crosstalk coefficients.

Assuming that the modems are synchronized, and DMT modulation is employed, one can model transmission independently on each tone by the relation (Fig. 3, a)

$$y_k = H_k x_k + v_k \tag{1}$$

where v_k is the additive noise on tone k. It is comprised of thermal noise, alien crosstalk, radio frequency interference (RFI), etc. [5], and is frequently modeled as an additive white Gaussian noise (AWGN).

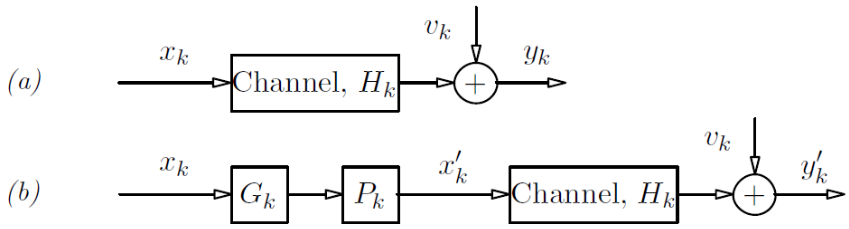

Figure 3. *(a)* Channel model (1). *(b)* Signal transmission modification using gain G_k and precoder P_k.

3. Signal Transmission Modification with Gain G and Precoder P

To control the transmit power spectral density (PSD) of n user on tone k, which is denoted $s_k^n \triangleq \mathbf{E}\left\{|x_k^n|^2\right\}$, the transmit signal $x_k \triangleq [x_k^1, \cdots x_k^N]^T \in \mathbb{C}^N$ is pre-multiplied by a diagonal matrix $G_k \triangleq \text{diag}\ [g_k^1, \cdots, g_k^N] \in \mathbb{R}^{N\times N}$. Then the resulting vector $(G_k x_k)$ is pre-disturbed by the precoder matrix $P_k \in \mathbb{C}^{N\times N}$ to obtain the channel input signal $x'_k = P_k(G_k x_k)$. Matrix P_k is to be specific for every tone k, while the tones number into thousands ($K = 2048$ as is in a typical case or may reach 4096). So, relation (1) is replaced by

$$y'_k = H_k x'_k + v_k = H_k P_k (G_k x_k) + v_k. \tag{2}$$

Question:

- How to design Precoder P_k in order to cancel crosstalk in y'_k, i. e., to ensure an element-wise relation between y'_k (the received vector) and x_k (the transmit vector)? In other words, we need:

$$y_k^{\prime,n} = \alpha_k^n x_k^n + v_k^n \ \text{ with a scalar } \alpha_k^n,\ n = 1, \ldots, N. \tag{3}$$

Formal solution:

❶ Premultiplying x_k by G_k yields $(G_k x_k) = \left[g_k^1 x_k^1, \cdots, g_k^N x_k^N\right]^T$.

❷ Denote $F_k \triangleq \text{diag}\ \left[\left(h_k^{1,1}\right)^{-1} \mid \cdots \mid \left(h_k^{N,N}\right)^{-1}\right]$; therefore $F_k^{-1} \triangleq \text{diag}\ \left[h_k^{1,1} \mid \cdots \mid h_k^{N,N}\right]$.

❸ Denote $A_k \triangleq F_k H_k = \begin{bmatrix} 1 & \left(h_k^{1,1}\right)^{-1} h_k^{1,2} & \cdots & \left(h_k^{1,1}\right)^{-1} h_k^{1,N} \\ \left(h_k^{2,2}\right)^{-1} h_k^{2,1} & 1 & \cdots & \left(h_k^{2,2}\right)^{-1} h_k^{2,N} \\ \cdots & \cdots & \ddots & \cdots \\ \left(h_k^{N,N}\right)^{-1} h_k^{N,1} & \left(h_k^{N,N}\right)^{-1} h_k^{N,2} & \cdots & 1 \end{bmatrix}$.

❹ Denote $P_k \triangleq A_k^{-1}$, so that $P_k = H_k^{-1} F_k^{-1}$. This relation means the decomposition: $\boxed{H_k = F_k^{-1} A_k}$.

❺ With such P_k, form $x'_k = P_k(G_k x_k)$, as it is shown in the above Fig. 3, b to obtain at receiving side

$$y'_k = H_k x'_k + v_k = H_k P_k (G_k x_k) + v_k = H_k H_k^{-1} F_k^{-1} (G_k x_k) + v_k = F_k^{-1}(G_k x_k) + v_k$$

$$y'_k = \left[h_k^{1,1} g_k^1 x_k^1 \mid \cdots \mid h_k^{N,N} g_k^N x_k^N \right]^T + v_k;\ y_k^{\prime,n} = \alpha_k^n x_k^n + v_k^n,\ \ \alpha_k^n = h_k^{nn} g_k^n,\ n = 1, \ldots, N.$$

Steps from ❶ to ❺ in the above list are presented to substantiate the known answer to the above *question*:

$$\boxed{P_k \text{ should be defined by formula } P_k \triangleq A_k^{-1}}.$$

Thus, decomposition-based zero-forcing precoder P_k is the inverse of the *normalized* (i. e., unit diagonal) channel matrix A_k [6, p. 34–35]. Formally, A_k equals the channel matrix $H_k = \left[h_k^{n,m} \right] \in \mathbb{C}^{N \times N}$ whose row and column indices n and m for entries $h_k^{n,m}$ run the range $1, \ldots, N$, premultiplied by matrix F_k. Such decomposing matrix H_k into $F_k^{-1} \times A_k$ and precoding only with P_k leads to a high transmission overhead of the CO/ONU due to the increased computational complexity if only matrix P_k is to be precomputed in explicit form by inverting matrix A_k.

Below for the sake of simplicity, we omit index k and consider alternative execution strategies for obtaining the channel input signal x'_k satisfying $A_k x'_k = (G_k x_k)$.

Note: We use the same symbol Σ_i with subindex $_i$ to designate the i-th strategy and its *complexity understood as the total multiplication/division count in it.* Calculations for Section 3 to Section 7 are rigorously substantiated by summing finite series of positive numbers in [9].

4. Baseline solution Σ_1: Gaussian LU-decomposition followed by the forward and backward substitutions to designate A^{-1} as precoder P

LU-decomposition of $N \times N$ matrix A is well known and may be performed by a variety of ways [10, pp. 27–81, 117–120, 137]. Take for consideraton: (a) Gauss column sweep algorithm; (b) Crout's reduction algorithm; and (c) bordering algorithm, in each case matrix L being lower triangular, matrix U unit diagonal upper triangular, and non-trivial elements of L and U overwrite the given matrix A.

Step 1: $A = LU$. Irrespective of differences in the hierarchy of actions, all the algorithms have the same complexity: $\Sigma_1^{\text{Step 1}} = (N-1)N(N+1)/3$.

Step 2: (a) From system $LW = I$, we find W. At this point, $\Sigma_1^{\text{Step 2(a)}} = N(N+1)(N+2)/6$. (b) From system $UX = W$, we obtain $X \triangleq A^{-1}$, $P := X$. At this point, $\Sigma_1^{\text{Step 2(b)}} = (N-1)N^2/2$. Totalling: $\Sigma_1^{\text{Step 2}} = \Sigma_1^{\text{Step 2(a)}} + \Sigma_1^{\text{Step 2(b)}} = N(2N^2+1)/3$.

Step 3: $x' = Px$. At this point, $\Sigma_1^{\text{Step 3}} = N^2$.

Consequently, strategy Σ_1 has the following complexity:

$$\Sigma_1 = \Sigma_1^{\text{Step 1}} + \Sigma_1^{\text{Step 2}} + \Sigma_1^{\text{Step 3}} = N^2(N+1). \tag{4}$$

5. Alternative Σ_2: Jordanian LU-decomposition followed by the forward substitution only to designate A^{-1} as precoder P

Jordan's method provides for complete elimination procedure. Operation count shows:

Step 1: $A = LU$ with resulting in L and $-U^{-1}$: $\Sigma_2^{\text{Step 1}} = (N-1)N^2/2$.

Step 2: L^{-1} by the instrumentality of elementary matrices: $\Sigma_2^{\text{Step 2}} = (N-1)N(N+1)/6$.
Step 3: $P := A^{-1} = U^{-1}L^{-1}$: $\Sigma_2^{\text{Step 3}} = (N-1)N(N+1)/3$.
Step 4: $x' = Px$. At this point, $\Sigma_2^{\text{Step 4}} = N^2$.
Consequently, strategy Σ_2 has the following complexity:

$$\Sigma_2 = \Sigma_2^{\text{Step 1}} + \Sigma_2^{\text{Step 2}} + \Sigma_2^{\text{Step 3}} + \Sigma_2^{\text{Step 4}} = N(N+1)(2N-1)/2. \tag{5}$$

6. Alternative Σ_3: Gaussian LU-decomposition followed by finding the elimination form of A^{-1} to designate it as precoder P

This operational scheme differs significantly in character from that used in the above. Namely, L^{-1} and U^{-1} are found after the LU-decomposition by means of step-by-step elementary matrix multiplying [10, pp. 42–46]. The algorithm is comprised of the following five steps:
Step 1: $A = LU$, as in strategy Σ_1: $\Sigma_3^{\text{Step 1}} = (N-1)N(N+1)/3$.
Step 2: L^{-1} by the instrumentality of elementary matrices, as in strategy Σ_2: $\Sigma_3^{\text{Step 2}} = (N-1)N(N+1)/6$.
Step 3: U^{-1} by the instrumentality of elementary matrices: $\Sigma_3^{\text{Step 3}} = (N-2)(N-1)N/6$.
Step 4: $P := A^{-1} = U^{-1}L^{-1}$, as in strategy Σ_2: $\Sigma_3^{\text{Step 4}} = (N-1)N(N+1)/3$.
Step 5: $x' = Px$. At this point, $\Sigma_3^{\text{Step 5}} = N^2$.
Consequently, strategy Σ_3 has the following complexity:

$$\Sigma_3 = \Sigma_3^{\text{Step 1}} + \Sigma_3^{\text{Step 2}} + \Sigma_3^{\text{Step 3}} + \Sigma_3^{\text{Step 4}} + \Sigma_3^{\text{Step 5}} = N(N+1)(4N-1)/6. \tag{6}$$

7. Alternative Σ_4: Gaussian LU-decomposition followed by system solution to avoid finding and designating A^{-1} as precoder P

We now proceed to avoid matrix inversion in precoding.
Step 1: $A = LU$, as in strategy Σ_1: $\Sigma_4^{\text{Step 1}} = (N-1)N(N+1)/3$.
Step 2: (a) By solving $Lw = I$, we obtain w. At this point, $\Sigma_4^{\text{Step 2(a)}} = N(N+1)/2$. (b) By solving $Ux' = w$, we obtain x'. At this point, $\Sigma_4^{\text{Step 2(b)}} = N(N-1)/2$. Totalling: $\Sigma_4^{\text{Step 2}} = \Sigma_4^{\text{Step 2(a)}} + \Sigma_4^{\text{Step 2(b)}} = N^2$.
Consequently, strategy Σ_4 has the following complexity:

$$\Sigma_4 = \Sigma_4^{\text{Step 1}} + \Sigma_4^{\text{Step 2}} = N(N^2+3N-1)/3. \tag{7}$$

8. Alternative Σ_5: Least squares-based square root sequential system solution to avoid finding and designating A^{-1} as precoder P

As a test specimen, let us take Potter's square root least squares algorithm [10, pp. 250–251]. It avoids matrix inversion by means of row-by-row matrix processing while solving $Ax' = (Gx)$. Modelling on the LS-algorithm [10, p. 250]) and utilizing complex conjugate matrix transposition (where needed), we obtain:

I. *Initialization.* Initial values: LS-estimator x_0 and its covariance P_0. A'priori data is lacking, it means $x_0 = 0$ and $P_0 = \varepsilon^{-2}\,\text{diag}\,\left[1 \mid \cdots \mid 1\right]$ with as possible small ε, theoretically $\varepsilon \to 0$. Let S_0 be a matrix square root of P_0, i. e. $S_0 = P_0^{1/2} = \varepsilon^{-1}\,\text{diag}\,\left[1 \mid \cdots \mid 1\right]$. Set $\tilde{x} := x_0$ and $\tilde{S} := S_0$.

II. *Processing the n-th matrix row $\left[A \mid (Gx)\right]_n$ of data $\left[A \mid (Gx)\right]$:* $a \triangleq a_n^T$ is the n-th row of A (viewed as a column) and $z \triangleq z_n$ is the n-th item of (Gx) in system $Ax' = (Gx)$. Cycle on $n = 1, \ldots, N$:

$$f = \tilde{S}^H a\,; \quad \alpha = f^H f\,; \quad K = \tilde{S} f/\alpha\,; \quad \hat{S} = \tilde{S} - K f^H\,; \quad \hat{x} = \tilde{x} + K(z - A\tilde{x})\,.$$

III. *Propagating the instantaneous solution estimator to the next matrix row for Item II repetition*:

$$\tilde{S} := \hat{S}\,, \quad \tilde{x} := \hat{x}\,.$$

On exit from Item III after having $n = N$ at Item II, one obtains $\tilde{x}$ as the desired solution x' for system $Ax' = (Gx)$.

Counting shows that strategy Σ_5 has the following complexity:

$$\Sigma_5 = N(3N^2 + 4N). \tag{8}$$

9. Complexity: Implementation tradeoff analysis in precoder design from the perspective of matrix inversion

For the five strategies considered, we have relations (4), (5), (6), (7), and (8) to characterize their complexity. To intercompare them, we introduce a complexity trend index associated with passing from Σ_i to Σ_j: $\Delta_{ij} \triangleq \Sigma_i - \Sigma_j$. We also propose the limit relative indices $\delta_{ij} \triangleq \lim_{N\to\infty} \Sigma_i/\Sigma_j$ obtained for $N \to \infty$ when moving from Σ_i to Σ_j, and sum up the exact results in Table 1.

Table 1. Processor off-loading Δ_{ij} at passing from strategy Σ_i to Σ_j.

$\Delta_{14} = \frac{N}{3}(2N^2 + 1)$	$\Delta_{24} = \frac{N}{6}(N-1)(4N+1)$	$\Delta_{34} = \frac{N}{6}(N-1)(2N-1)$
$\Delta_{12} = \frac{N}{2}(N+1)$	$\Delta_{23} = \frac{N}{3}(N-1)(N+1)$	$\Delta_{13} = \frac{N}{6}(N+1)(2N+1)$
$\Delta_{15} = -N^2(2N+3)$	$\Delta_{35} = -\frac{N}{6}(14N^2 + 21N + 1)$	$\Delta_{45} = -\frac{N}{3}(8N^2 + 9N + 1)$

The limit relative indices for processor off-loading prove to be as follows:
$\delta_{14} = 3$, $\delta_{24} = 3$, $\delta_{34} = 2 \Rightarrow$ processor load dropping.
$\delta_{12} = 1$, $\delta_{23} = 3/2$, $\delta_{13} = 3/2 \Rightarrow$ processor load keeping or dropping.
$\delta_{15} = 1/3$, $\delta_{35} = 2/9$, $\delta_{45} = 1/9 \Rightarrow$ processor load increasing.

10. Numerical Experiments

Keeping in mind the necessity of expanded analysis, we record a fact to be used later, namely that we rename Σ_3 as $\Sigma_{3(1)}$ and Σ_4 as $\Sigma_{4(1)}$. In parallel with them, we will test two more versions appropriately labeled as $\Sigma_{3(2)}$ and $\Sigma_{4(2)}$ which are different in that they make use of Jordanian LU-decomposition instead of Gaussian one. Thus, we test the following seven strategies for finding the desired solution x':

- Σ_1: Gaussian LU-decomposition followed by the forward and backward substitutions to designate A^{-1} as precoder P;
- Σ_2: Jordanian LU-decomposition followed by the forward substitution only to designate A^{-1} as precoder P
- $\Sigma_{3(1)}$: Gaussian LU-decomposition followed by finding the elimination form of A^{-1} to designate it as precoder P;
- $\Sigma_{3(2)}$: Jordanian LU-decomposition followed by finding the elimination form of A^{-1} to designate it as precoder P;
- $\Sigma_{4(1)}$: Gaussian LU-decomposition followed by system solution to avoid finding and designating A^{-1} as precoder P;
- $\Sigma_{4(2)}$: Jordanian LU-decomposition followed by system solution to avoid finding and designating A^{-1} as precoder P;

- Σ_5: Least squares-based square root sequential system solution to avoid finding and designating A^{-1} as precoder P.

The first four strategies suppose that having A^{-1} found, we designate the product $\tilde{x}' = P(Gx)$ with $P := A^{-1}$ as the desired solution x'. In the three last-named strategies, $\tilde{x}'$ is the result of the system $Ax' = (Gx)$ solving phase. Apart from method delivering the solution $\tilde{x}'$, it is essential to know the estimated accuracy $e \triangleq \tilde{x}' - x'$. Because the precise meaning x' is to be supposed unknown, all that remains is to verify the residual $r \triangleq A\tilde{x}' - (Gx)$. As is evident, $A^{-1}r = e$. Considering that matrices A for the downstream VDSL channels possess the property of row-wise diagonal dominance [5, p. 860], and so they are well conditioned, one may with good reason, estimate accuracy by $||r||_\infty = ||A\tilde{x}' - (Gx)||_\infty$ for all afore-mentioned strategies.

We have implemented seven m-functions in MATLAB. Using these implementations, we conducted computational experiments with the set of test matrices $H_k \in \mathbb{C}^{N\times N}$ on one arbitrarily selected tone k with $N = 20, 40, 80, \ldots, 800$. For each matrix H_k and given complex vector (Gx), we saved the computed solution $\tilde{x}'$ and the execution time (in the sec) and the accuracy of computations $||r||_\infty$. Tables 2 and 3 present the obtained results. One can see from there that the new proposed strategy Σ_5 has the minimal execution time and the best accuracy of computations for all test matrices. That gives rise to a suggestion that Σ_5 is a numerically efficient method for practical applications.

Table 2. Execution time (sec) according to strategy Σ_i.

N	Σ_1	Σ_2	$\Sigma_{3(1)}$	$\Sigma_{3(2)}$	$\Sigma_{4(1)}$	$\Sigma_{4(2)}$	Σ_5
20	0.0143	0.0170	0.0134	0.0143	0.0036	0.0070	0.0013
40	0.0506	0.0601	0.0654	0.0957	0.0120	0.0238	0.0033
100	0.2966	0.4501	0.3221	0.3720	0.0952	0.1755	0.0384
200	1.5384	2.0127	1.7208	1.6822	0.4936	0.7637	0.2323
400	6.5275	8.4242	7.1477	6.9706	2.6530	4.8078	5.0287
800	78.9794	97.7975	74.5325	73.4622	24.4492	36.8818	16.4508

Table 3. Accuracy of computations according to strategy Σ_i.

N	Σ_1	Σ_2	$\Sigma_{3(1)}$	$\Sigma_{3(2)}$	$\Sigma_{4(1)}$	$\Sigma_{4(2)}$	Σ_5
20	2.2225e-16	2.2225e-16	2.2225e-16	2.2225e-16	3.3309e-16	3.3306e-16	6.6613e-16
40	1.5142e-15	1.5142e-15	1.5143e-15	1.5143e-15	1.3528e-15	1.3228e-15	7.8773e-16
100	1.1770e-12	4.2381e-12	2.7423e-12	4.6210e-12	1.1143e-12	3.0986e-12	1.0900e-12
200	1.1803e-10	2.8348e-10	9.2966e-10	2.7933e-10	4.0387e-11	2.7968e-10	4.5409e-12
400	1.3830e-10	1.5888e-09	5.5337e-09	1.8709e-09	2.3257e-10	1.6365e-09	5.1123e-12
800	4.3237e-08	9.0371e-09	5.6475e-07	8.6912e-09	2.8337e-08	8.6935e-09	3.7705e-11

11. Conclusions

Three strategies involving explicit matrix inversion have been considered. They differ in mode of action: Σ_1 acts with Gaussian LU-decomposition, Σ_2 deals with Jordanian LU-decomposition, and Σ_3 operates with the elimination form of the inverse matrix. They are numbered in order of descending complexity: $\Sigma_1 > \Sigma_2 > \Sigma_3$. The smallest effect of descending complexity holds for passing $\Sigma_1 \to \Sigma_2$. It equals $\Delta_{12} = (1/2)N(N+1)$, and so it is of no concern in the limit (at N increased indefinitely). The passing $\Sigma_2 \to \Sigma_3$ works for $\Delta_{23} = N(N-1)(N+1)/3$. Thanks to it, the overall limit complexity is 50 percent lower than the value characteristic of Σ_2. With $\Delta_{13} = \Delta_{12} + \Delta_{23}$, it is the same result as passing $\Sigma_1 \to \Sigma_3$ gives at $N \to \infty$. The passing $\Sigma_1 \to \Sigma_4$ has the maximum effect: $\Delta_{14} = \Delta_{13} + \Delta_{34}$. At N increasing, it is of our main interest seeing that its winning factor tends to 3. Such sizable gain can be understood: it comes out from matrix inversion avoidance when solving linear systems.

The fifth strategy Σ_5 may seem a bit unusual or even 'exotic.' Theoretically, as N tends to infinity, it imposes a thrice as much computational load on the processor. At the same time, MATLAB computations show that it is the fastest and the most accurate computational procedure when compared with other strategies. In that context, this new proposed strategy Σ_5 can be useful for practical applications.

12. References

[1] Jacobsen, K. S. Providing the Right Solution for VDSL / K. S. Jacobsen // Texas Instruments White Paper. —1999. —Version 1.05. Access mode: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.560.6051&rep=rep1&type=pdf.

[2] Aspects of VDSL Evolution : Technical Report : TR-040 / Digital Subscriber Line Forum : 2001. —June. —Access mode: https://www.broadband-forum.org/technical/download/TR-040.pdf. —(Formerly WT-047v8).

[3] Ma, Yi., Jia, Zh. Evolution and Trends of Broadband Access Technologies and Fiber-Wireless Systems (Chapter 2) / Yi. Ma, Zh. Jia // Fiber-Wireless Convergence in Next-Generation Communication Networks – Systems, Architectures, and Management / Ed. by M. Tornatore, G.-K. Chang, G. Ellinas. —Switzerland : Springer International Publishing, 2017. —ISBN: 9781301525089.

[4] Kuo, C. C., Tsai, S. H., Tadjpour, L., Chang, Y. H. Precoding Techniques for Digital Communication Systems / C. C. Kuo, S. H. Tsai, L. Tadjpour, Y. H. Chang // —[S. l.] : Springer US, 2008. —ISBN: 9780387717692. —Access mode: https://books.google.ru/books?id=cKd0GR9DB2YC.

[5] Cendrillon, R., Ginis, G., Bogaert, V. D. E., Moonen, M. A Near-Optimal Linear Crosstalk Precoder for Downstream VDSL / R. Cendrillon, G. Ginis, E. van den Bogaert, M. Moonen // IEEE Transactions on Communications. —2007. —Vol. 55, no. 5. —P. 860–863. —A broadcast draft version is dated 22nd December 2004 at https://espace.library.uq.edu.au/view/UQ:9258.

[6] Düngen, M. Crosstalk Mitigation Techniques for Digital Subscriber Line Systems (Doktor-Ingenieurin (Dr.-Ing.) genehmigte Dissertation) : Ph. D. thesis / M. Düngen ; Technischen Universität Hamburg-Harburg (TUHH). —Germany, 2016. —ISBN: 978-3-7369-9215-3, 978-3-7369-8215-4. Access mode: https://tubdok.tub.tuhh.de/handle/11420/1296.

[7] Semushin, I. V. Adaptive State Prediction Methods for Big Data Modeling and Design / I. V. Semushin // "The 3rd Professor Day Academic Conference," Huawei Russian Research Center, Moscow, 20–21 December 2016. —Access mode: https://www.dropbox.com/s/k2kq7r56dn0rhcd/semushinTalk-20Dec2016.pdf?dl=0&m=.

[8] Holte, N. Transmission properties of pair cables, Digital Kommunikasjon / N. Holte. —Norway : NTNU, 2002. —Access mode: https://www.ceid.upatras.gr/webpages/faculty/alexiou/ahts/other_pdf/4_Twisted_p.pdf.

[9] Semushin, I. V. Big Matrix Inverses Revisited: from Formalities to Implementation / I. V. Semushin // Automation of Control Processes. —2017. —No. 4(50). —P. 36–41. —[in Russian].

[10] Semushin, I. V. Computational Methods of Linear Algebra and Estimation / I. V. Semushin, —Ulyanovsk : UlSTU Publishers, 2011. —ISBN 978-5-9795-0902-0 (in Russian). Access mode: http://venec.ulstu.ru/lib/disk/2013/119.pdf.