Performance of Matrix Decomposition Algorithms in 5G NR MIMO Precoding

Citation for published version:
Sibio, S, Newson, P, Ben Smida, S, Ding, Y, Offer, T, Ramsay, D, Wilkie, B & Goussetis, G 2022, 'Performance of Matrix Decomposition Algorithms in 5G NR MIMO Precoding', Paper presented at 2022 IEEE Vehicular Technology Society Asia Pacific Wireless Communications Symposium, Seoul, Korea, Democratic People's Republic of, 24/08/22 - 26/08/22.

Performance of Matrix Decomposition Algorithms in 5G NR MIMO Precoding

Simona Sibio [1], Paul Newson [2], Souheil Ben Smida [1], Yuan Ding [1], Tom Offer [2], David Ramsay [2], Bill Wilkie [2], George Goussetis [1]
[1] Institute of Sensors Signals and Systems, Heriot-Watt University , Edinburgh, UK {ss870@hw.ac.uk}
[2] Xilinx Inc, Edinburgh, UK

Abstract— **This paper presents a trade-off analysis associated with different matrix decomposition algorithms in the application of MIMO precoding. A scenario of 5G New Radio (NR) Multi-User MIMO (MU-MIMO) downlink involving one base station (BS) equipped with 64 antennas which serves multiple users simultaneously is studied. The channel model follows the 3GPP TDL-C with various levels of antenna correlation. Six different matrix decomposition algorithms are studied, namely QR Modified Gram-Schmidt (QR-MGS), QR householder, Cholesky, LDL, SVD Bidiagonalization, and SVD. An evaluation of the computational complexity of the algorithms is presented and the system performance compared. These analyses allow us to make an informed trade-off between performance and hardware complexity.**

Keywords— Computational cost, Matrix Inversion, MU-MIMO, Throughput analysis

I. INTRODUCTION

In order to support the anticipated growth in network traffic in cellular systems significant attention is given to the implementation of system architectures based on MU-MIMO [1]. In this scenario, spatial multiplexing is the key to serving multiple User Equipment (UE) and is commonly enabled through MIMO precoding, typically requiring the knowledge of the propagation channel [2]. Precoding can be implemented in different ways to remove or reduce channel interferences. Techniques commonly employed include: Zero-Forcing (ZF), Minimum Mean-Squared Error (MMSE), and Block Diagonalization (BD). These precoding methods allow the MU-MIMO system to improve capacity, but at the expense of increasing radio unit complexity which is strongly dependent on the number of antenna elements and the UEs [3]. Indeed, most of these techniques involve the complex matrix inversion, which is one of the most computationally demanding, quantified using million instructions per second (MIPS) [4]. Additionally, another factor to consider in the implementation is the latency introduced because the precoding calculation may have to be updated frequently in a dynamic channel environment [3].

Tackling above limitations, particular attention has been paid to the required computational cost versus the system performance for different precoding techniques in order to strike a favorable trade-off. In [5], three methods were reviewed, these being: ZF, BD, and the dirty-paper coding techniques. The ZF technique implements the channel inversion by computing the pseudo-inverse of the channel. The BD method is a generalized version of this and optimizes the power transfer to a group of antennas rather than a single antenna. In a multi-user scenario where the receivers have multiple antennas, the latter offers more optimal performance, however it is computationally expensive. The dirty-paper coding method is not considered in the analysis because it is a highly expensive technique and other alternative linear precoding schemes have already reached 98% of the optimal dirty-paper coding capacity for a small number of antennas in the base station (BS), around 20, in [6].

Focusing on the linear precoding techniques; each can be implemented using several algorithms and this has an impact on the two figures of merit chosen: computational cost, and system throughput.

The BD method requires singular value decomposition which can be implemented following different algorithms with different levels of accuracy and complexity as presented in [7]. Considering the ZF method, different algorithms are used to reduce the computational impact of the channel inversion by using a triangular matrix and back-substitution, instead of computing the inversion directly.

For large matrices in MIMO systems, the matrix inversion is traditionally implemented by applying QR factorization to the original matrix to generate an upper triangular matrix R [4]. As discussed in [7], there are several possible implementations of QR such as Modified Gram-Schmidt, Householder Transforms and Givens Rotations. While these methods are generic and work well for any type of matrix, it is possible to exploit the special structure of the input matrix in the algorithm related to the MIMO system to reduce the complexity of the linear precoding and make it more hardware-friendly. Since the product of the channel matrix and its transpose generates a Hermitian and positive-defined matrix, Cholesky and LDL decompositions can be efficiently applied; these have been described in [8]-[10]. A comparison between them in terms of the number of operations, relative errors, and word lengths in fixed-point implementations is presented in [11].

Another strategy to implement the channel inversion that has been investigated in the literature is using the Neumann series approximation, [3], [12], [13]. To exploit this technique, the input matrix must be strongly diagonally dominant [12]; this is not the case when the channels are highly correlated as is often the case in cellular systems. This approximation enjoys lower complexity than that of the QR Gram-Schmidt algorithm only when the ratio between the number of antennas equipped by the transmitter and by the receiver is more than 27 [13]. Considering 64 antennas in the base station and 8 or 16 users with one antenna each, this condition is not satisfied.

In our paper, we study the channel inversion performance in the 5G NR MU-MIMO downlink, considering different algorithms to reduce the required resources, with the aim to achieve a balance between the computational cost and system throughput performance. To the best of the authors' knowledge, there are no other reported studies jointly considering these two figures of merits of these precoding techniques using several algorithms in the 5G NR systems through 3GPP correlated channels.

II. System Description

The MU-MIMO system model and the Precoding described in the following section are in line with the corresponding settings described in [1].

We consider the downlink in a single-cell environment in which a BS equipped with a uniform planar array with M = 64 antennas serves K single-antenna users simultaneously, see the illustration in Fig. 1. The waveforms emitted by the BS are transmitted over the MIMO channel. These signals are subject to distortion; consequently, a corrupted version of the signals is captured by each user.

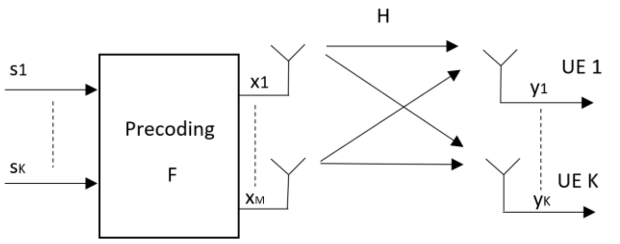


Fig. 1. System model of an MU-MIMO system with M-antenna BS and K single-antenna UEs.

The received signal is given by:

$$Y = \sqrt{p_t}\boldsymbol{H}\,X + N \qquad (1)$$

Where $\boldsymbol{H} \in \mathbb{C}^{K\times M}$ is channel matrix between BS and all users and its h_{km} element is the channel coefficient between the m-th BS antenna and the k-th user. The scalar p_t is used for transmit power normalization. The signals transmitted by the BS form the vector $X = (x_1\ x_2\ \ldots\ x_M)^T$ which is normalized to satisfy $E\{\|X^2\|\} = 1$ and contains a precoded version of the data symbol vector $S = (s_1\ s_2\ \ldots\ s_K)^T$, (2). Here, $(\bullet)^T$ denotes the transpose of the enclosed vector, and $E\{\|\bullet\|\}$ returns the expected value of the modulus.

The transmitted signal is expressed:

$$X = \boldsymbol{W}S \qquad (2)$$

Where $\boldsymbol{W}$ is an $M\times K$ pre-coding matrix including power allocation to data symbols which can be decomposed as

$$\boldsymbol{W} = \frac{1}{\|\boldsymbol{F}\|_F}\,\boldsymbol{F}, \qquad (3)$$

where $\boldsymbol{F}$ represents the output matrix of a particular linear precoding algorithm and the $\|F\|_F$ is a power normalization factor. Here $\|\bullet\|_F$ is Frobenius norm.

From these considerations, $\boldsymbol{F}$ is designed to improve the transmission, e.g., Maximum-Ratio Transmission Precoder, ZF Precoder, and Diagonalization of the channel [14].

A. Channel inversion

A less-complex technique for improving the transmission is Maximum-Ratio Transmission Precoding which maximizes the array gain of the transmission, but the interference to other users is still present [14]. To null the interference, the ZF technique implements the pseudo-inverse of the channel matrix, and the precoding matrix is defined by the following equation

$$\boldsymbol{F}_{ZF} = \boldsymbol{H}^H(\boldsymbol{H}\boldsymbol{H}^H)^{-1} \qquad (4)$$

Here, $(\bullet)^H$ is the Hermitian transpose operator. Because the pseudo-inverse of a square matrix matches the inverse matrix [7], this technique is able to cancel both inter-symbol and inter-user interference [14].

B. Diagonalization as generalized channel inversion

A variant of ZF is BD [15]. The key idea of generalized channel inversion is described using the following relation

$$\boldsymbol{H}_j\boldsymbol{F}_k = 0, \qquad \forall\, j \neq k \qquad (5)$$

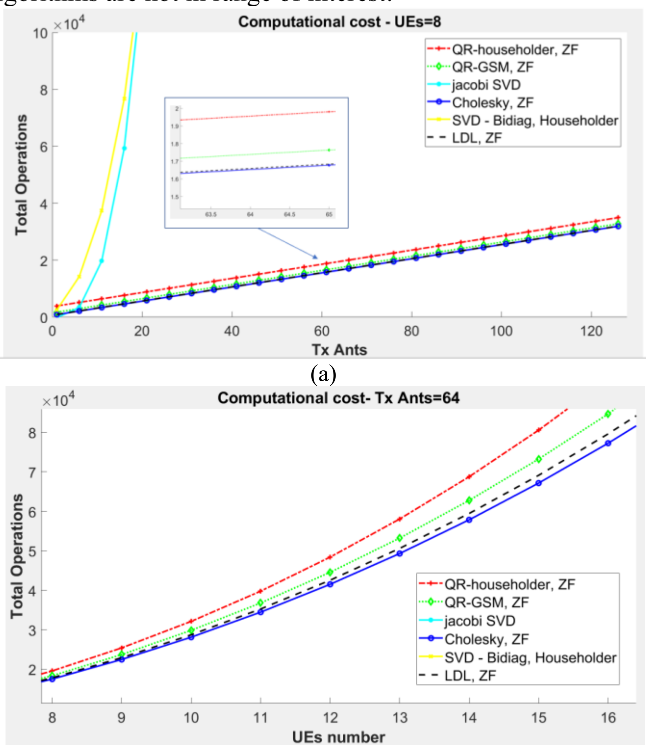
This condition eliminates all multi-user interferences because the F-matrix is composed by orthonormal vectors that multiplied by the H-matrix, generates no-null values only in the main diagonal.

III. Complexity of Channel Inversion

Matrix inversion is a traditional topic in the area of numerical analysis, and it is well elaborated in [7]. The choice of algorithm depends on the complexity and the hardware availability in any particular application.

In the Fig. 2a, the total number of operations needed, including the whole procedure to obtain the precoding matrix associated with each algorithm, is presented.

As expected, the two SVD-based algorithms are potentially prohibitively more complex compared to the other algorithms. Consequently, in Fig. 2b, calculating the computational cost in function of UEs number, SVD algorithms are not in range of interest.



(a)

(b)

Fig. 2 – (a) Flops to obtain the precoding matrix using different algorithms as a function of BS antennas.
(b) Flop as a function of the number of UEs.

Considering ZF using QR algorithms, Cholesky, and LDL, the computational cost has two different trendlines based on the chosen variable. Computational complexity increases linearly with the number of Tx antennas, Fig. 2 (a), and following a polynomial trend with the number of UEs, Fig. 2 (b). Evaluating both graphs, LDL and Cholesky require fewer operations to implement the channel inversion because they work on only half of the input matrix. In detail, the Tables I presents the exact analytical expressions for each channel inversion implementation.

TABLE I – ANALYTICAL EXPRESSION FOR EVALUATING THE COMPLEXITY IN CHANNEL INVERSION

Algorithm used for inversion	Analytical Expression for Complexity (UEs=8)
LDL	1240 nTx -273
Cholesky	1240 nTx - 336
SVD	1750 nTx^3 – 4125 nTx^2 +3710 nTx - 1518
SVD Bidiagonalization	750.01 nTx^3 + 1274.5 nTx^2+ 2572.2 nTx – 1997.7
QR	1240 nTX + 524
QRP	1240 nTx +2695

Algorithm used for inversion	Analytical Expression for Complexity (nTx=64)
LDL	3.50 nUE^3 + 248.9 nUE^2 – 582.6 nUE + 335.86
Cholesky	3 nUE^3 + 248.2 nUE^2 – 577 nUE + 329.24
SVD	16384 nUE + 2 10^6
SVD Bidiagonalization	16384 nUE + 2 10^6
QR	5 nUE^3 + 239.5 nUE^2 – 556.5 nUE + 312
QRP	6.99 nUE^3 + 235.58 nUE^2 – 546.8 nUE + 296.5

A. QR and QRP decompositions

Assuming A is an m-by-n matrix, it can be decomposed into

$$\boldsymbol{A} = \boldsymbol{QR} \text{ or } \boldsymbol{A} = \boldsymbol{QRP}^H, \quad (6)$$

where $\boldsymbol{Q}$ is an orthogonal matrix with dimensions m-by-m, $\boldsymbol{R}$ is an upper triangular matrix of dimensions m-by-n, and $\boldsymbol{P}$ is a permutation matrix of dimension n-by-n.

Adapting the algorithm into the ZF method, we get

$$\boldsymbol{F}_{ZF} = \boldsymbol{H}^H(\boldsymbol{HH}^H)^{-1} = \boldsymbol{H}^H (\boldsymbol{QR})^{-1} \quad (7)$$

It is more convenient to invert the $\boldsymbol{R}$-matrix and $\boldsymbol{Q}$-matrix because they are triangular and orthonormal matrices, respectively. The triangular matrix can be inverted using the back-substitution, and the inverse of an orthogonal matrix is its transpose.

B. Cholesky decomposition

If $\boldsymbol{A}$ is a Hermitian and positive definite n-by-n matrix, it can be decomposed into

$$\boldsymbol{A} = \boldsymbol{LL}^H, \quad (8)$$

where $\boldsymbol{L}$ is a lower triangular matrix of dimension n-by-n.

Adapting the algorithm into the ZF method, we have

$$(\boldsymbol{HH}^H)^{-1} = (\boldsymbol{LL}^H)^{-1}. \quad (9)$$

The channel information is now associated with the $\boldsymbol{L}$ matrix, so the inversion of $\boldsymbol{L}$ is the key. Being a triangular matrix, back-substitution can be applied to calculate the inverse of $\boldsymbol{L}$.

C. LDL decomposition

This LDL decomposition can be viewed as an alternative to the Cholesky decomposition. If $\boldsymbol{A}$ is a Hermitian and positive definite n-by-n matrix, it can be decomposed into

$$\boldsymbol{A} = \boldsymbol{LDL}^H \quad (10)$$

$\boldsymbol{L}$ and $\boldsymbol{L}^H$ are lower triangular and upper triangular matrices of size n-by-n, respectively, and $\boldsymbol{D}$ is a diagonal matrix to avoid the square root operations that are present in the Cholesky decomposition.

Thus, the $\boldsymbol{L}$ matrix can be inverted using back-substitution and the inverse of $\boldsymbol{D}$ matrix is already calculated in the algorithm.

Adapting the algorithm into the ZF method, we get

$$(\boldsymbol{HH}^H)^{-1} = (\boldsymbol{LDL}^H)^{-1}. \quad (11)$$

D. Singular Value Decomposition – SVD

If $\boldsymbol{A}$ is an m-by-n matrix, it can be decomposed into

$$\boldsymbol{A} = \boldsymbol{U\Sigma V}^H, \quad (12)$$

where $\boldsymbol{U}$ and $\boldsymbol{V}$ are orthogonal matrices, and $\boldsymbol{\Sigma}$ is a diagonal matrix. Different algorithms can be used to implement this decomposition, such as Jacobi.

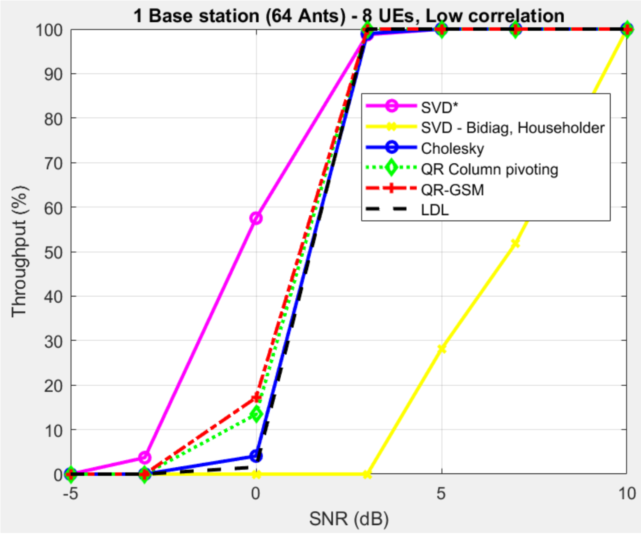
Applying this algorithm, the precoding matrix is the $\boldsymbol{V}$ matrix and, being an orthogonal matrix, its inverse is the transpose of the matrix.

E. SVD – Bidiagonalization

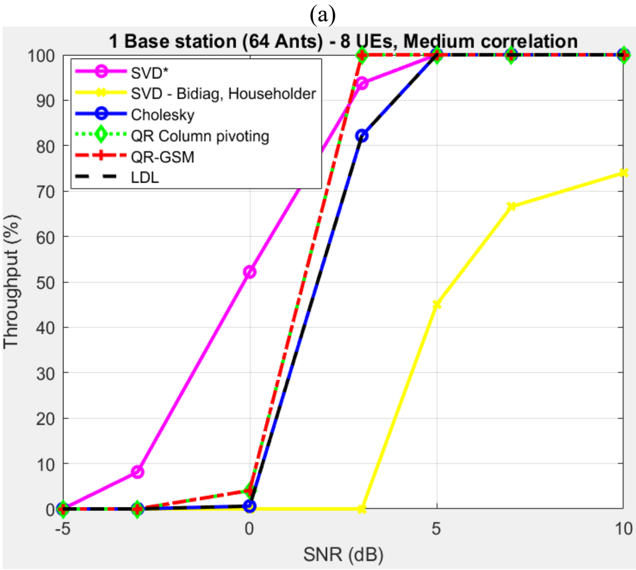
The bidiagonalization is a simplified version of the SVD where the construction of matrices is done following Householder algorithm. If $\boldsymbol{A}$ is an m-by-n matrix, it can be decomposed into,

$$\boldsymbol{A} = \boldsymbol{UBV}^H, \quad (13)$$

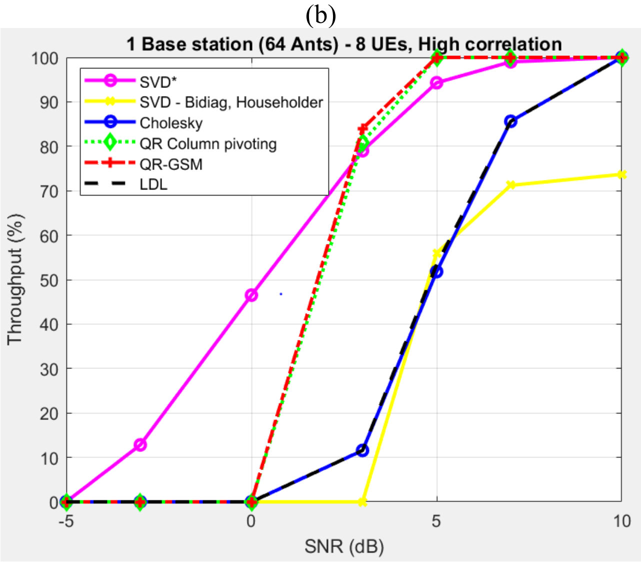
where $\boldsymbol{U}$ and $\boldsymbol{V}$ are orthogonal matrices, and $\boldsymbol{B}$ is a diagonal matrix. As in the SVD algorithm, the inversion is avoided.



(a)

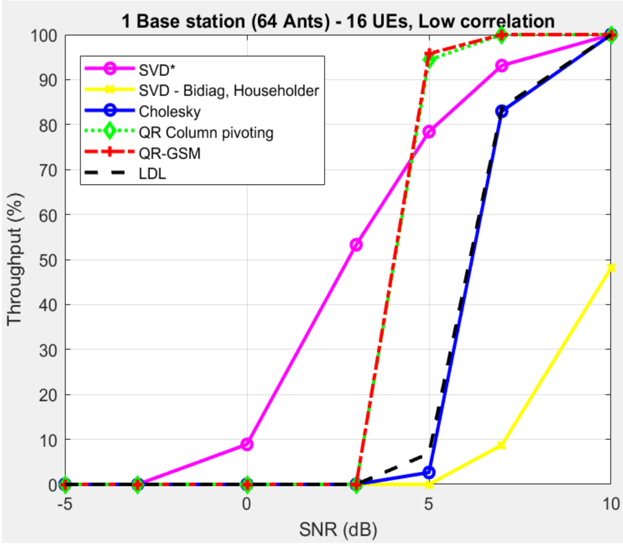


(b)

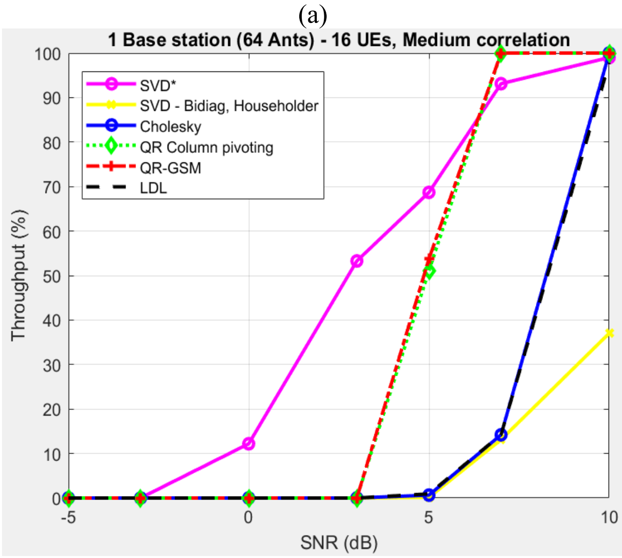


(c)

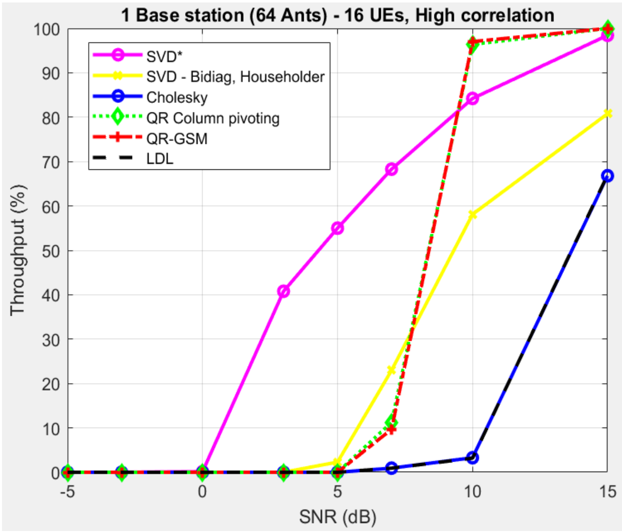
Fig. 3 – System throughput (Mbps) with 8 users as a function of SNRs under (a) low correlation channel (ρ=0), (b) medium correlation channel (ρ=0.7), and (c) high correlation channel (ρ=0.9).



(a)



(b)



(c)

Fig. 4 – System Throughput (Mbps) with 16 users as a function of SNR values through (a) low correlated channel (ρ=0), (b) medium correlated channel (ρ=0.7), and (c) high correlated channel (ρ=0.9).

IV. SIMULATED SCENARIOS

For given channel conditions, summarized in Table II, the throughput performance is determined by the chosen algorithms to generate the precoding matrix. The simulation results are obtained in the MATLAB environment using the 5G toolbox [16] including the Downlink data chain as shown in Fig.5, and are presented in Figs. 3 and 4. The codewords are generated from the data bits in the Downlink Shared Channel (DL-SCH), and then modulated, converted into layers, precoded, and mapped into resources in the Physical Downlink Shared Channel (PDSCH).

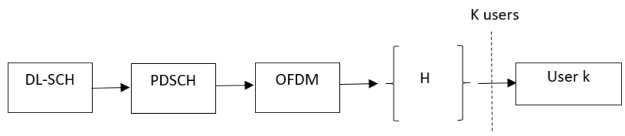


Fig. 5 – Simulated 5G NR System.

TABLE II. SIMULATED CHANNEL CHARACTERISTICS

Main Channel Characteristics	
Delay profile	TDL-C
Doppler frequency shift	100 Hz
Delay Spread	30 ns
Transmit antennas	Planar Array of 64 elements
Correlation coefficient	0 – 0.7 – 0.9

Performance is analyzed in floating point accuracy considering different levels of channel correlation levels and 8 and 16 UEs.

In both scenarios, the QR algorithms and SVD are the most stable with any level of correlation, while the LDL and Cholesky algorithms are more affected by the correlation coefficient. The SVD algorithm, considering the scenario with eight UEs, is the most convenient at low SNR values, while, reaching 5 dB, SVD performance is equal to the QR algorithms. Increasing the number of users, the throughput trends of the system using QR algorithms and SVD are much closer. In both systems, considering 8 and 16 users, Cholesky and LDL trends provide lower performance than SVD and QR algorithms. Despite, in the first scenario, the results of Cholesky and LDL are still comparable with the others, in the second case, they reach a good throughput only at higher SNR values.

V. CONCLUSION

To meet the requirements of the hardware implementation and system-level performance, different channel inversion implementations were analyzed. The evaluation is based on the complexity of the precoding in terms of required operations and the percentage of successful transmissions in the system, considering 8 and 16 UEs. In both scenarios at several levels of correlation, QR-GSM provides the best trade-off between the computational cost and throughput performance; however, LDL and Cholesky could be a valid alternative in the 8-users system or at higher SNR values.

REFERENCES

[1] F. Rusek, D. Persson, B. K. Lau, E. G. Larsson, , T. L. Marzetta, O. Edfors, and F. Tufvesson, "Scaling up MIMO: Opportunities and challenges with very large arrays," in *IEEE Signal Processing. Magazine*, vol. 30, no. 1, pp. 40–60, Jan. 2013.

[2] L. Sanguinetti, E. Björnson, and J. Hoydis, "Toward massive MIMO 2.0: Understanding spatial correlation, interference suppression, and pilot contamination," in *IEEE Transactions on Communications*, vol. *68*, no.1, pp. 232-257, Jan. 2020.

[3] H. Prabhu, O. Edfors, J. Rodrigues, L. Liu, and F. Rusek, "Hardware efficient approximative matrix inversion for linear pre-coding in massive MIMO," *2014 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1700-1703, June 2014.

[4] J. Eilert, D. Wu, and D. Liu, "Efficient complex matrix inversion for MIMO software defined radio," *2007 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 2610-2613, May 2007.

[5] Q. H. Spencer, C. B. Peel, A. L. Swindlehurst, and M. Haardt, "An introduction to the multi-user MIMO downlink," in *IEEE communications Magazine*, vol. 42, no. 10, pp. 60-67, Oct 2004.

[6] X. Gao, O. Edfors, F. Rusek, and F. Tufvesson, "Linear pre-coding performance in measured very-large MIMO channels" in *2011 IEEE Vehicular Technology Conference (VTC Fall)*, pp. 1-5, Septemeber 2011.

[7] G. H. Golub, C. F. Van Load, *Matrix Computations*, 4th ed., Johns Hopkins Press, 2013.

[8] A. Burian, J. Takala, and M. Ylinen, "A fixed-point implementation of matrix inversion using Cholesky decomposition," *2003 46th Midwest Symposium on Circuits and Systems*, Vol. 3, pp. 1431-1434, 2003.

[9] M. Martel, A. Najahi, and G. Revy, "Toward the synthesis of fixed-point code for matrix inversion based on Cholesky decomposition," in *Proceedings of the 2014 Conference on Design and Architectures for Signal and Image Processing*, pp. 1-8, October 2014.

[10] C. Ingemarsson, and O. Gustafsson, "Hardware architecture for positive definite matrix inversion based on LDL decomposition and back-substitution," in *2016 50th Asilomar Conference on Signals, Systems and Computers*, pp. 859-863, November 2016.

[11] C. Ingemarsson, and O. Gustafsson, "On fixed-point implementation of symmetric matrix inversion," in *2015 European Conference on Circuit Theory and Design (ECCTD),* pp. 1-4, August 2015.

[12] D. Zhu, B. Li, and P. Liang, "On the matrix inversion approximation based on Neumann series in massive MIMO systems," *2015 IEEE international conference on communications (ICC)* , pp. 1763-1769, June 2015.

[13] H. Prabhu, J. Rodrigues, O. Edfors, and F. Rusek, "Approximative matrix inverse computations for very-large MIMO and applications to linear pre-coding systems," *2013 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 2710-2715, April 2013.

[14] F. L. Luo, and C. J. Zhang, *Signal processing for 5G: algorithms and implementations*, John Wiley & Sons, 2016.

[15] L. U. Choi, and R. D. Murch, "A transmit preprocessing technique for multiuser MIMO systems using a decomposition approach" in *IEEE Transactions on Wireless Communications*, vol. *3,* no. 1, pp. 20-24, Jan. 2004.

[16] A. Zaidi., F. Athley, J. Medbo, U. Gustavsson, G. Durisi, and X. Chen, *5G Physical Layer: principles, models and technology components*, Acaseminc Press, 2018.