RESEARCH

Low computational complexity methods for decoding of STBC in the uplink of a massive MIMO system

Check for updates

Seyed Hosein Mousavi and Jafar Pourrostam[*] iD

* Correspondence: j.pourrostam@
tabrizu.ac.ir
Department of Electrical &
Computer Engineering, University of
Tabriz, Tabriz, Iran

Abstract

Reducing the computational complexity of the modern wireless communication systems such as massive MIMO configurations is of utmost interest. In this paper, we propose algorithms which can be used to accelerate matrix inversion and reduce the complexity of common spatial multiplexing schemes in massive MIMO systems. Here, we specifically investigate the performance of the proposed methods in systems that utilize STBC (Space-Time Block Code) in the uplink of dynamic massive MIMO systems for different scenarios. A multi-user system in which the base station is equipped with a large number of antennas and each user has two antennas is considered. In addition, users can enter or exit the system dynamically. For a given space-time block coding/decoding scheme, the computational complexity of the receiver will be significantly reduced by employing the proposed methods. The first approach is utilizing Neumann series to approximate the inverse matrix for linear decoders. The second tactic is reducing the computational complexity of the STBC decoders when a user is added to system or removed from it. In the proposed schemes, the matrix inversion for ZF and MMSE decoding is derived from inversing a partitioned matrix and Woodbury matrix identity. Furthermore, the suggested techniques can be utilized when the number of users is fixed but the CSI changes for a particular user. The mathematical equations for both approaches are derived and the complexity of the suggested methods is compared to the direct computation of the inverse matrix. Moreover, the performance of the proposed algorithms is evaluated in terms of the system BER (bit error rate). Evaluations confirm the effectiveness of the proposed approaches.

Keywords: Fast update, Linear decoder, Massive MIMO, Neumann series, STBC, Woodbury formula

1 Introduction

Massive MIMO (multiple-input multiple-output) has been explored as one of the underlying technologies for the new generations of wireless communication systems in recent years [1]. In massive MIMO configuration for cellular communications the BS (base station) is equipped with a large number of antennas and simultaneously serves multiple users. In such formations high capacity, energy efficiency as well as high reliability can be achieved via relatively simple signal processing techniques [2].

Additionally, when the number of antennas at the BS is very large, uplink communication channels will be asymptotically orthogonal. Therefore, when multiple users transmit signal in the same frequency band and the same time slots, virtual point-to-point SIMO (single-input multiple-output) links are established in which each user has single antenna and the BS has multiple antennas. As a result, intra-cell/inter-cell interference can be largely eliminated utilizing simple linear signal processing methods such as ZF (zero forcing) or MMSE (minimum mean square error) decoders [3]. Moreover, because the capacity of the multiple antenna systems is proportional to the minimum number of transmit and receive antennas [4], using one antenna in the transmitter will lower the overall throughput of the system. Spatial multiplexing methods can be used to increase the total capacity of the system. For instance one solution to improve the diversity gain of each user in the uplink communication is using multiple antennas along with STBC (space-time block code) at the user side [4–8]. It has been shown that by using a good space-time block code with full diversity and linear receiver, the inter-cellular interference problem can be solved to a large extent [4]. For a massive MIMO system with two antennas at the user terminal, sufficient condition to design a good STBC with linear receivers is studied in [4]. Also, its performance in terms of attainable throughput is investigated.

It is worth mentioning that many benefits of various massive MIMO configurations come at the price of high computational complexity. For example, when the number of users increases, the linear STBC decoding methods such as ZF and MMSE algorithms require inverting a matrix with large dimensions. Therefore, computationally efficient methods must be developed to cope with this challenge and make the hardware implementation feasible.

In [9–14], researchers have explored ideas that aim to reduce computational complexity in different scenarios. Consider a cellular system with M users that are connected to the BS simultaneously and some of the users are moving with high speed. Complexity reduction has been investigated for the cases in which a user is added to the cell or removed from it as well as the case when a user's CSI (channel state information) is changed. In these circumstances, if we calculate the exact inverse of the decoder matrix using conventional methods such as Cholesky decomposition, high computational load will be imposed on the system. In this paper, we propose approaches to reduce the computational complexity at the receiver. One technique is to employ methods to approximate the inverse matrix such as Neumann series. Moreover, we propose calculating the exact inverse matrix by utilizing available information and matrix inversion identities to update the current inverse matrix.

2 Methods

In this work, the STBC scheme presented in [4] is adopted for a massive MIMO system and low complexity matrix inversion techniques are proposed and evaluated at the receiver of the uplink of the considered configuration. In other words, we will explore solutions to recover data from the received signal with lower computational complexity and without significant performance degradation.

One possible approach is approximating the inverse of the decoder matrix. For example, Neumann series has been used to calculate the inverse of a matrix at the receiver [10]. In the same work, it has been demonstrated that as long as the number of BS antennas is

much larger than the number of users, BLER (block error rate) is similar to the case when an exact inverse is calculated while the required computations is reduced by one order of magnitude. Here, we examine the complexity and the BER performance of this method for the considered system model for the different numbers of terms to be computed for the series that is referred to as the order of the Neumann series.

The next approach is proposed for a dynamic massive MIMO system. By dynamic we mean that the users are entering the system or exiting from it. In this situation, it is not necessary to recalculate the inverse of the linear decoder matrix and the existing inverse matrix is updated. For the selected STBC scheme, based on the matrix inversion lemmas such as the inverse of a partitioned matrix and the Woodbury formula [15], we propose and evaluate low-complexity methods to speed up STBC ZF and MMSE decoders. Update equations are derived for the cases that a user is added to or removed from the system as well as the case in which the channel estimate of a user is changed.

Algorithms are evaluated and compared in terms of BER performance and computational complexity. The proposed algorithms need fewer computations which naturally leads to reduction in the run time of a SDR (software-defined radio) program or the complexity of implemented hardware for the. Not only can these algorithms be used in a slow fading environment by switching active users, but also could be used in fast fading channels with frequent changes to the user channel estimates.

3 System model

Consider the uplink of a cellular multi-user massive MIMO system in which the BS is equipped with N antennas and serves M user ($M < N$) such that each independent user has two antennas, as illustrated in Fig. 1. The channel is supposed to follow Rayleigh small fading and large scale path loss and shadowing model.

Fig. 1 A cellular multiuser massive MIMO system. In this system, each independent user has two antennas and BS is equipped with N antennas that serve M users($M < N$)
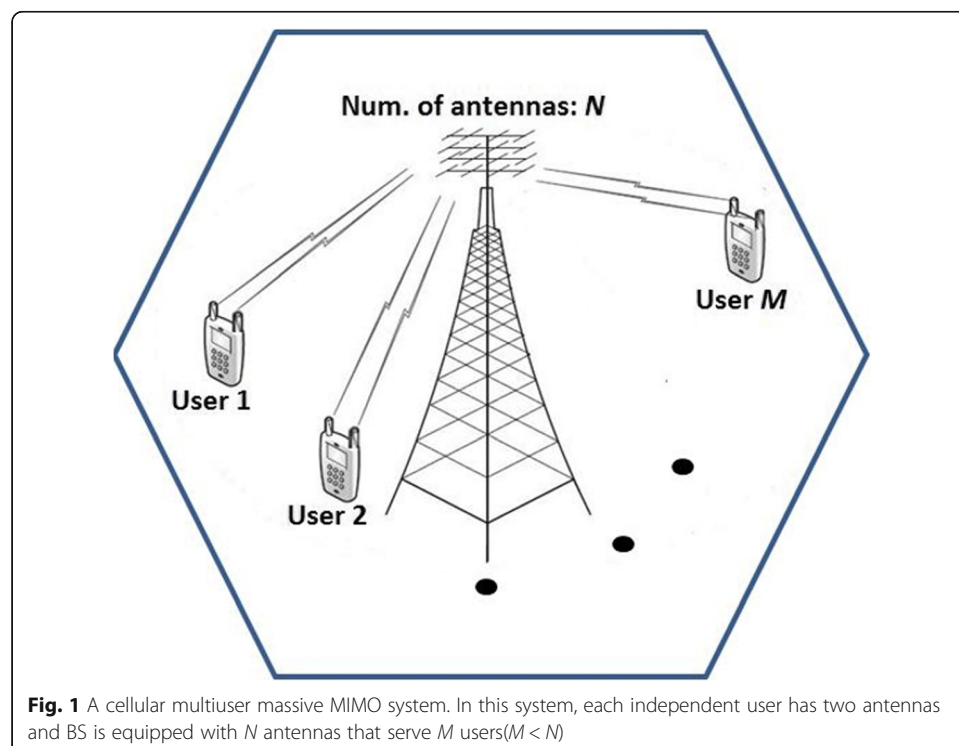
The channel gain between the jth antenna of the mth user and the nth antenna of the BS is formulated as $\beta_{nmj}h_{nmj}$ ($1 \leq n \leq N, 1 \leq m \leq M, \ j = 1, 2$), where β_{nmj} is related to the large scale path loss and shadowing and h_{nmj} denotes the small scale fading. It is assumed that $\beta_{nmj} = \beta_m$ for $n = 1, ..., N$ and $j = 1, 2$. In addition, to normalize the average power, we assume that $\beta_1 = 1$ and $\beta_1 \geq \beta_2 \geq ... \geq \beta_M$. Based on the Rayleigh fading model, h_{nmj} is assumed to be an i.i.d (independent and identically distributed) zero mean, circularly symmetric complex Gaussian random variable with unit variance. Furthermore, the fading coefficients and the large scale channel gains from the mth user to the BS are expressed as $\mathbf{H}_m = [h_{nmj}]_{N \times 2}$ and $\mathbf{L}_m = \beta_m \mathbf{I}_2$ respectively.

Suppose STBC is adopted by each subscriber in the cell, and the code of the mth user is expressed as $\mathbf{X}_m$ with the size of $2 \times S$. With these assumptions, the received signals in the base station over S time slots, $\mathbf{Y}_{N \times S}$, is written as follows:

$$\mathbf{Y} = \sum_{m=1}^{M} \sqrt{\frac{\rho}{2}} \mathbf{H}_m \mathbf{L}_m \tilde{\mathbf{X}}_m + \mathbf{W}$$
$$= \sqrt{\frac{\rho}{2}} \mathbf{H} \mathbf{L} \tilde{\mathbf{X}} + \mathbf{W}. \tag{1}$$

where $\mathbf{H} = [\mathbf{H}_1 \ \mathbf{H}_2 ... \mathbf{H}_M]$, $\mathbf{L} = diag(\mathbf{L}_1, \mathbf{L}_2, ..., \mathbf{L}_M)$ is a block diagonal matrix, and $\tilde{\mathbf{X}} = [\tilde{\mathbf{X}}_1^T, \tilde{\mathbf{X}}_2^T, ..., \tilde{\mathbf{X}}_M^T]^T$ with energy restriction $E\{tr(\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H)\} = 2S$, superscripts T and H represent the matrix Transpose and Hermitian operators, respectively. Also, ρ demonstrates the received SNR and $\sqrt{1/2}$ is used to normalize the transmitted signal energy to be "1" per time slot. $\mathbf{W}_{N \times S}$ represents the noise whose entries are i.i.d. taken from the zero-mean, circularly symmetric complex Gaussian random variables with unit variance. Next, we explain the STBC coding and decoding algorithms.

3.1 Coding matrix for each user

The transmitted signal matrix $\tilde{\mathbf{X}}$ is a STBC which is sent from two transmit antennas over S time slots. In this paper, we choose $S = 2$ and the corresponding STBC for the mth user $\tilde{\mathbf{X}}_m$ is designed as follows:

$$\tilde{\mathbf{X}}_m = \begin{bmatrix} a_m(x_{m1} + b_m x_{m2}) & \gamma_m a_m(x_{m3} + b_m x_{m4}) \\ c_m(x_{m3} + b_m x_{m4}) & c_m(x_{m1} + d_m x_{m2}) \end{bmatrix} \tag{2}$$

where $\mathbf{x}_m = [x_{m1}, x_{m2}, x_{m3}, x_{m4}]^T$ is the transmitted symbol vector of the mth user and $a_m, b_m, c_m, d_m, \gamma_m$ are constants that are determined to satisfy the orthogonality of the code. Denoting $\mathbf{H}_m = [\mathbf{h}_{m1} \ \mathbf{h}_{m2}]$ and substituting (2) into (1) and taking the vector form of $\mathbf{Y}$ we have

$$vec(\mathbf{Y}) = \sum_{m=1}^{M} \sqrt{\frac{\rho}{2}\beta_m} \tilde{\mathbf{H}}_m \mathbf{x}_m + vec(\mathbf{W}), \tag{3}$$

where

$$\tilde{\mathbf{H}}_m = \begin{bmatrix} a_m \mathbf{h}_{m1} & a_m b_m \mathbf{h}_{m1} & c_m \mathbf{h}_{m2} & c_m d_m \mathbf{h}_{m2} \\ c_m \mathbf{h}_{m1} & c_m d_m \mathbf{h}_{m2} & \gamma_m a_m \mathbf{h}_{m2} & \gamma_m a_m b_m \mathbf{h}_{m2} \end{bmatrix} \tag{4}$$

For linear decoders such as ZF and MMSE filters, when N is large enough, it is desired that the columns of $\tilde{\mathbf{H}}_m$ are asymptotically orthogonal. Applying the orthogonality criterion and energy constraint, i.e., $E\{tr(\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H)\} = 4$, it is shown that the coding constants are obtained as follows [4]:

$$
\begin{aligned}
&a_m = (1 + \mathbf{j}(1-b_m))/\sqrt{5}, \quad b_m = \left(1 + \sqrt{5}\right)/2 \\
&c_m = (1 + \mathbf{j}(1-d_m))/\sqrt{5}, \quad d_m = (1 + \mathbf{j}(1-d_m))/\sqrt{5} \\
&\gamma_m = \mathbf{j}
\end{aligned}
\tag{5}
$$

3.2 Linear decoding for each user

Let us define $\tilde{\mathbf{G}}$ as a matrix with dimensions of $2N \times 4M$ and $\mathbf{x}$ as a $4M$-dimensional vector:

$$
\begin{aligned}
\tilde{\mathbf{G}} &= [\beta_1 \tilde{\mathbf{H}}_1 \quad \beta_2 \tilde{\mathbf{H}}_2 \quad \dots \quad \beta_M \tilde{\mathbf{H}}_M] , \\
\mathbf{x} &= \left[\mathbf{x}_1^T, \quad \mathbf{x}_2^T, \quad \dots \quad , \mathbf{x}_M^T \right]^T .
\end{aligned}
\tag{6}
$$

Hence, we can rewrite Eq. (3) as

$$
vec(\mathbf{Y}) = \sqrt{\frac{\rho}{2}}\tilde{\mathbf{G}}\mathbf{x} + vec(\mathbf{W}).
\tag{7}
$$

Let $\mathbf{Q}_{ZF}$ and $\mathbf{Q}_{MMSE}$ be the ZF and MMSE decoder matrices, respectively, we will have

$$
\mathbf{Q}_{ZF} = \left(\tilde{\mathbf{G}}^H \tilde{\mathbf{G}}\right)^{-1} \tilde{\mathbf{G}}^H,
\tag{8}
$$

$$
\mathbf{Q}_{MMSE} = \left(\frac{2\mathbf{I}_{4M}}{\rho} + \tilde{\mathbf{G}}^H \tilde{\mathbf{G}}\right)^{-1} \tilde{\mathbf{G}}^H.
\tag{9}
$$

Multiplying (7) by these matrices from the left, we have

$$
\mathbf{Q}vec(\mathbf{Y}) = \sqrt{\frac{\rho}{2}}\mathbf{Q}\tilde{\mathbf{G}}\mathbf{x} + \mathbf{Q}vec(\mathbf{W}),
\tag{10}
$$

where $\mathbf{Q}=\mathbf{Q}_{ZF}$ or $\mathbf{Q}=\mathbf{Q}_{MMSE}$. The lth transmitted symbol of the mth user $\hat{x}_{ml}$ is estimated as

$$
\begin{aligned}
\hat{x}_{ml} = \ &\arg \min_{x} \left| \left[\mathbf{Q}vec(\mathbf{Y})\right]_{4(m-1)+l} - \right. \\
&\left. \sqrt{\frac{\rho}{2}} \tilde{q}_{4(m-1)+l,4(m-1)+l}\, \boldsymbol{x} \right|, \quad l = 1, 2, 3, 4.
\end{aligned}
\tag{11}
$$

where the minimum is over the signal constellation of the mth user, $[\mathbf{Q}vec(\mathbf{Y})]_p$ is the pth element of the vector $\mathbf{Q}vec(\mathbf{Y})$, and $\tilde{q}_{u,v} = [\mathbf{Q}\tilde{\mathbf{G}}]_{u,v}$.

4 Proposed algorithms

It is noted that the computational load of Eqs. (8) and (9) mostly lies within the inverse matrix calculation. Conventional methods to compute the inverse matrix, such as Cholesky decomposition, impose high computational complexity on the system and requires $O(M^3)$ operations which would be difficult to implement [16, 17]. Therefore, we

investigate matrix inverting methods which have less computational complexity and lead to feasible receiver for a massive MIMO system.

4.1 Inverse matrix approximation using Neumann series

From Eqs. (8) and (9), it can be seen that decoding of the received signal involves computing the inverse of the following matrix:

$$\mathbf{Z} = \begin{cases} \tilde{\mathbf{G}}^H \tilde{\mathbf{G}} & \textit{for } \text{ZF}, \\ \dfrac{2\mathbf{I}_{4M}}{\rho} + \tilde{\mathbf{G}}^H \tilde{\mathbf{G}} & \textit{for } \text{MMSE}. \end{cases} \tag{12}$$

where $\tilde{\mathbf{G}}$ is a $2N \times 4M$ dimensional matrix which contains the coding constants as well as the channel coefficients, however, from now on, for brevity it will be called the channel matrix. Considering that the matrix $\mathbf{Z}$, with dimensions of $4M \times 4M$, is almost diagonal, an efficient algorithm in terms of hardware constraints is used to approximate the inverse [18]. It is proven in [19] that if $\mathbf{Z}$ is decomposed as $\mathbf{Z} = \mathbf{D} + \mathbf{E}$ where $\mathbf{D}$ is a diagonal matrix with diagonal entries of $\mathbf{Z}$ and $\mathbf{E}$ is the corresponding hollow, then the Neumann series can be used to calculate its inverse as follows:

$$\tilde{\mathbf{Z}}_R^{-1} \approx \sum_{r=0}^{R-1} \left(-\mathbf{D}^{-1}\mathbf{E}\right)^r \mathbf{D}^{-1}, \tag{13}$$

where R is the number of terms to be computed in the series and $\tilde{\mathbf{Z}}_R^{-1}$ is the R-term approximation of $\mathbf{Z}^{-1}$. The convergence of (13) is only guaranteed if the maximum modulus of the eigenvalues of matrix $(\mathbf{I} - \mathbf{D}^{-1}\mathbf{Z})$ is less than 1 and the approximation approaches equality as $R \to \infty$ [19]. Moreover, the lower the eigenvalues, the faster the convergence; which holds true when the ratio $\alpha = N/M$ is high. The minimum value of this ratio for a high probability convergence of the method is 5.83 [9]. Here, given that each user is equipped with two antennas, the above ratio is half of the single antenna case. For example, if $N = 640$, the maximum of 55 double-antenna users can simultaneously communicate with the BS whereas for a system with single-antenna users the Neumann series converges for the number of users as high as 110.

Neumann series is a low complexity iterative method. Therefore, contrary to conventional inverse computation methods, it is hardware friendly [19]. As an example, for $R = 3$, we have the approximation as follows:

$$\tilde{\mathbf{Z}}_3^{-1} \approx \mathbf{D}^{-1} - \left(\mathbf{D}^{-1}\mathbf{E}\right)\mathbf{D}^{-1} + \left(\mathbf{D}^{-1}\mathbf{E}\right)\left(\mathbf{D}^{-1}\mathbf{E}\mathbf{D}^{-1}\right). \tag{14}$$

The number of computations for the first part is M divisions. While calculating the second and the third terms requires $3M^2 - 3M$ and $16M^3 - 2M$ real-valued multiplication, respectively. These values tapping out the existence of zeros in the diagonal and the fact that each part of (14) is Hermitian.

Now, let us define matrix $\mathbf{W} = \mathbf{D}^{-1}\mathbf{E}$ and rewrite (13) as

$$\tilde{\mathbf{Z}}_R^{-1} \approx \sum_{r=0}^{R-1} \left(-\mathbf{W}\right)^r \mathbf{D}^{-1}. \tag{15}$$

For Neumann series with $R = 3$, and substituting (8) and (9) in (10), we will have

$$\mathbf{Q}vec(\mathbf{Y}) \approx \tilde{\mathbf{Z}}_R^{-1}\mathbf{t}$$
$$= \left(\sum_{r=0}^{R-1}(-\mathbf{W})^r\mathbf{D}^{-1}\right)\mathbf{t} \qquad (16)$$
$$= \mathbf{D}^{-1}\mathbf{t} - \mathbf{W}\left(\mathbf{D}^{-1}\mathbf{t}\right) + \mathbf{W}\left(\mathbf{W}\left(\mathbf{D}^{-1}\mathbf{t}\right)\right).$$

Where $\mathbf{t} = \tilde{\mathbf{G}}^H vec(\mathbf{Y})$ is a $4M$-dimensional vector. As can be seen, the first term is obtained by multiplying diagonal matrix $\mathbf{D}^{-1}$ in vector $\mathbf{t}$ and the approximation is improved by adding each additional term while the computational complexity is slightly increased.

In Section 5, the efficiency of this approach will be examined in terms of system BER and its computational complexity.

4.2 Inverse matrix updating

In some situations, the decoding can be done without recalculating the inverse of the decoder matrix $\mathbf{Z}$. For example, a user may be added to or removed from the system or the channel estimate changes for a particular user. Under such conditions, the computational complexity can be greatly reduced by updating the inverse matrix instead of recalculating it. The proposed solutions are based on the inverse of a partitioned matrix and the Woodbury matrix identity. Suppose matrix $\mathbf{Z}$ is partitioned as

$$\mathbf{Z} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix}, \qquad (17)$$

where $\mathbf{A}$ and $\mathbf{D}$ are square matrices. The inverse of $\mathbf{Z}$ is given as

$$\mathbf{Z}^{-1} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{F}_{11} & \mathbf{F}_{12} \\ \mathbf{F}_{21} & \mathbf{F}_{22} \end{bmatrix}, \qquad (18)$$

where

$$\begin{aligned}
\mathbf{F}_{11} &= \left(\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{C}\right)^{-1}, \\
\mathbf{F}_{12} &= -\mathbf{F}_{11}\mathbf{B}\mathbf{D}^{-1}, \\
\mathbf{F}_{21} &= -\mathbf{D}^{-1}\mathbf{C}\mathbf{F}_{11}, \\
\mathbf{F}_{22} &= \left(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B}\right)^{-1}.
\end{aligned} \qquad (19)$$

In addition using the Woodbury formula, we have

$$\left(\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{C}\right)^{-1} = \mathbf{A}^{-1} + \mathbf{A}^{-1}\mathbf{B}\left(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B}\right)^{-1}\mathbf{C}\mathbf{A}^{-1}, \qquad (20)$$

Hence, equations given in (19) can be equivalently written as

$$\begin{aligned}
\mathbf{F}_{22} &= \left(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B}\right)^{-1}, \\
\mathbf{F}_{11} &= \mathbf{A}^{-1} + \mathbf{A}^{-1}\mathbf{B}\mathbf{F}_{22}\mathbf{C}\mathbf{A}^{-1}, \\
\mathbf{F}_{12} &= -\mathbf{A}^{-1}\mathbf{B}\mathbf{F}_{22}, \\
\mathbf{F}_{21} &= -\mathbf{F}_{22}\mathbf{C}\mathbf{A}^{-1}.
\end{aligned} \qquad (21)$$

Next, the algorithms for updating ZF and MMSE decoder matrices are described in different scenarios.

4.2.1 Adding a user

Let us examine the case where a user is added to the cell covered by the BS. Suppose that the initial channel matrix is $[\tilde{\mathbf{G}}]_{2N \times 4M}$, and let the channel matrix of the user which enters the system be $[\mathbf{G}_a]_{2N \times 4}$. The new inflated matrix is denoted as $\mathbf{G}_e = \begin{bmatrix} \tilde{\mathbf{G}} & \mathbf{G}_a \end{bmatrix}$. Thus, the ZF decoding matrix defined in (12) is given as

$$
\begin{aligned}
\mathbf{Z}_{e(ZF)} = \mathbf{G}_e^H \mathbf{G}_e &= \begin{bmatrix} \tilde{\mathbf{G}}^H \\ \mathbf{G}_a^H \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{G}} & \mathbf{G}_a \end{bmatrix} \\
&= \begin{bmatrix} \tilde{\mathbf{G}}^H \tilde{\mathbf{G}} & \tilde{\mathbf{G}}^H \mathbf{G}_a \\ \mathbf{G}_a^H \tilde{\mathbf{G}} & \mathbf{G}_a^H \mathbf{G}_a \end{bmatrix}
\end{aligned}
\tag{22}
$$

Thus, the resulting matrix has a dimension of $4(M+1) \times 4(M+1)$. Using (21), the inverse of the decoding matrix is calculated as follows:

$$
\mathbf{Z}_{e(ZF)}^{-1} = \begin{bmatrix} \tilde{\mathbf{G}}^H \tilde{\mathbf{G}} & \tilde{\mathbf{G}}^H \mathbf{G}_a \\ \mathbf{G}_a^H \tilde{\mathbf{G}} & \mathbf{G}_a^H \mathbf{G}_a \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{F}_{11} & \mathbf{F}_{12} \\ \mathbf{F}_{21} & \mathbf{F}_{22} \end{bmatrix},
\tag{23}
$$

where

$$
\begin{aligned}
\mathbf{F}_{22} &= \left(\mathbf{G}_a^H \mathbf{G}_a - \mathbf{B}^H \mathbf{Z}_{o(ZF)}^{-1} \mathbf{B} \right)^{-1}, \\
\mathbf{F}_{11} &= \mathbf{Z}_{o(ZF)}^{-1} + \mathbf{Z}_{o(ZF)}^{-1} \mathbf{B} \mathbf{F}_{22} \mathbf{B}^H \mathbf{Z}_{o(ZF)}^{-1}, \\
\mathbf{F}_{12} &= -\mathbf{Z}_{o(ZF)}^{-1} \mathbf{B} \mathbf{F}_{22}, \\
\mathbf{F}_{21} &= -\mathbf{F}_{22} \mathbf{B}^H \mathbf{Z}_{o(ZF)}^{-1}.
\end{aligned}
\tag{24}
$$

where $\mathbf{Z}_{o(ZF)}^{-1} = [\tilde{\mathbf{G}}^H \tilde{\mathbf{G}}]^{-1}$ is the inverse matrix before updating and $\mathbf{B} = [\tilde{\mathbf{G}}^H \mathbf{G}_a]_{4M \times 4}$. As it is observed, in this algorithm we only need to calculate the inverse of $\mathbf{F}_{22}$ which is a 4×4 dimensional matrix and the rest of the computations are matrix multiplication. However, direct calculation of decoding matrix needs the inversion of a matrix with dimensions of $4(M+1) \times 4(M+1)$. Similarly, for the MMSE decoder we have

$$
\mathbf{Z}_{e(MMSE)} = \begin{bmatrix} \tilde{\mathbf{G}}^H \tilde{\mathbf{G}} + (2/\rho)\mathbf{I}_{4M} & \tilde{\mathbf{G}}^H \mathbf{G}_a \\ \mathbf{G}_a^H \tilde{\mathbf{G}} & \mathbf{G}_a^H \mathbf{G}_a + (2/\rho)\mathbf{I}_4 \end{bmatrix},
\tag{25}
$$

and

$$
\begin{aligned}
\mathbf{Z}_{e(MMSE)}^{-1} &= \begin{bmatrix} \mathbf{F}_{11} & \mathbf{F}_{12} \\ \mathbf{F}_{21} & \mathbf{F}_{22} \end{bmatrix}, \\
\mathbf{F}_{22} &= \left(\mathbf{D} - \mathbf{B}^H \mathbf{Z}_{o(MMSE)}^{-1} \mathbf{B} \right)^{-1}, \\
\mathbf{F}_{11} &= \mathbf{Z}_{o(MMSE)}^{-1} + \mathbf{Z}_{o(MMSE)}^{-1} \mathbf{B} \mathbf{F}_{22} \mathbf{B}^H \mathbf{Z}_{o(MMSE)}^{-1}, \\
\mathbf{F}_{12} &= -\mathbf{Z}_{o(MMSE)}^{-1} \mathbf{B} \mathbf{F}_{22}, \\
\mathbf{F}_{21} &= -\mathbf{F}_{22} \mathbf{B}^H \mathbf{Z}_{o(MMSE)}^{-1}.
\end{aligned}
\tag{26}
$$

where $\mathbf{Z}_{o(MMSE)}^{-1} = \left(\tilde{\mathbf{G}}^H \tilde{\mathbf{G}} + (2/\rho)\mathbf{I}_{4M} \right)^{-1}$ is the inverse matrix before updating, $\mathbf{B} = [\tilde{\mathbf{G}}^H \mathbf{G}_a]_{4M \times 4}$, and $\mathbf{D} = \mathbf{G}_a^H \mathbf{G}_a + (2/\rho)\mathbf{I}_4$.

4.2.2 Removing a user

Now we consider the scenario in which a user is removed from the cell. The current channel matrix is indicated as $\tilde{\mathbf{G}} = \begin{bmatrix} \mathbf{G}_f & \mathbf{G}_r \end{bmatrix}$ where $\mathbf{G}_r$ is the channel matrix of the

user to be removed and $\mathbf{G}_f$ is the channel matrix after the removal of the user. In this case, updating the ZF decoding matrix involves calculating $\mathbf{Z}_{f(ZF)}^{-1} = (\mathbf{G}_f^H \mathbf{G}_f)^{-1}$. Using the inverse of a partitioned matrix, before the user is removed we have

$$
\begin{aligned}
\mathbf{Z}_{o(ZF)}^{-1} &= \left[\tilde{\mathbf{G}}^H \tilde{\mathbf{G}} \right]^{-1} = \begin{bmatrix} \mathbf{G}_f^H \mathbf{G}_f & \mathbf{G}_f^H \mathbf{G}_r \\ \mathbf{G}_r^H \mathbf{G}_f & \mathbf{G}_r^H \mathbf{G}_r \end{bmatrix}^{-1} \\
&= \begin{bmatrix} \mathbf{F}_{11} & \mathbf{F}_{12} \\ \mathbf{F}_{21} & \mathbf{F}_{22} \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{T} & -\left(\mathbf{G}_f^H \mathbf{G}_f \right)^{-1} \mathbf{B} \mathbf{F}_{22} \\ -\mathbf{F}_{22} \mathbf{B}^H \left(\mathbf{G}_f^H \mathbf{G}_f \right)^{-1} & \mathbf{F}_{22} \end{bmatrix},
\end{aligned}
\tag{27}
$$

where

$$
\mathbf{T} = \left(\mathbf{G}_f^H \mathbf{G}_f \right)^{-1} + \left(\mathbf{G}_f^H \mathbf{G}_f \right)^{-1} \mathbf{B} \mathbf{F}_{22} \mathbf{B}^H \left(\mathbf{G}_f^H \mathbf{G}_f \right)^{-1},
$$

and $\mathbf{B} = \mathbf{G}_f^H \mathbf{G}_r$, hence we can write

$$
\begin{aligned}
\mathbf{F}_{11} &= \mathbf{Z}_{f(ZF)}^{-1} + \mathbf{Z}_{f(ZF)}^{-1} \mathbf{B} \mathbf{F}_{22} \mathbf{B}^H \mathbf{Z}_{f(ZF)}^{-1} \\
&= \mathbf{Z}_{f(ZF)}^{-1} + \mathbf{F}_{12} \mathbf{F}_{22}^{-1} \mathbf{F}_{21},
\end{aligned}
\tag{28}
$$

which means to update the inverse of the ZF decoding matrix, we partition the current inverse and find the updated inverse as

$$
\mathbf{Z}_{f(ZF)}^{-1} = \mathbf{F}_{11} - \mathbf{F}_{12} \mathbf{F}_{22}^{-1} \mathbf{F}_{21},
\tag{29}
$$

Also, for the MMSE decoder, we need to compute $\mathbf{Z}_{f(MMSE)}^{-1} = \left(\mathbf{G}_f^H \mathbf{G}_f + (2/\rho) \mathbf{I}_{4(M-1)} \right)^{-1}$. Before the user is removed, we hav:

$$
\begin{aligned}
\mathbf{Z}_{o(MMSE)}^{-1} &= \begin{bmatrix} \mathbf{G}_f^H \mathbf{G}_f + (2/\rho) \mathbf{I}_{4(M-1)} & \mathbf{G}_f^H \mathbf{G}_r \\ \mathbf{G}_r^H \mathbf{G}_f & \mathbf{G}_r^H \mathbf{G}_r + (2/\rho) \mathbf{I}_4 \end{bmatrix}^{-1} \\
&= \begin{bmatrix} \mathbf{F}_{11} & \mathbf{F}_{12} \\ \mathbf{F}_{21} & \mathbf{F}_{22} \end{bmatrix},
\end{aligned}
\tag{30}
$$

Therefore, similar to what we derived for the ZF decoder, we have

$$
\mathbf{Z}_{f(MMSE)}^{-1} = \mathbf{F}_{11} - \mathbf{F}_{12} \mathbf{F}_{22}^{-1} \mathbf{F}_{12}.
\tag{31}
$$

Where $\mathbf{F}_{11}$, $\mathbf{F}_{12}$, $\mathbf{F}_{21}$, and $\mathbf{F}_{22}$ are obtained from partitioning the current inverse matrix.

4.2.3 Updating a user

When a new channel estimate is obtained for a particular user, i.e., its CSI is updated, the number of rows and columns of the channel matrix remains the same. In this case a two-step approach is suggested. In the proposed method, first the rows and columns associated with the updated user are deleted by utilizing the proposed algorithm for removing a user. Then, using the proposed algorithm for adding a user, we apply the new channel coefficients and update the inverse matrix. In other words, in ZF decoding, we first use Eqs. (28) and (29) to remove the rows and columns of the specific user. Then, we use (23) and (24) for the final update of the inverse matrix. Clearly, for the MMSE

decoder equations (30) and (31) are used first, and then (26) is applied to find the inverse of the updated channel matrix.

In the next section, we evaluate the proposed techniques in different scenarios.

5 Numerical results and discussion

In this section, we evaluate and compare the computational complexity of the proposed algorithms as well as the BER performance of the system in the assumed configurations. The next two sub-sections include the computational complexity of the ZF STBC decoder in the uplink of a massive MIMO system and the system BER performance when the proposed algorithms are utilized. It should be noted that similar results can be obtained for the case of MMSE decoder.

5.1 Complexity analysis

For a massive MIMO system with $N = 320$ antennas at the BS and $M = 8, 16, 24, 32$ users, the computational complexity is studied in terms of the number of arithmetic operations. We consider scenarios in which a user is added to or removed from the system as well as the case that the channel estimate of a user has changed. Assuming that the matrix whose inverse needs to be updated is $4M \times 4M$ dimensional, and K is the number of rows and columns that are added to or removed from the matrix, the number of computations needed for the ZF decoder is summarized in Table 1. The second and the third row of this table shows the computational complexity of the decoder when the inverse matrix is approximated using Neumann series. Also, inflated channel matrix refers to the case that a user is added to the system ($M_{new} = M + 1$) and deflated matrix represents the case that a user is removed from the system ($M_{new} = M - 1$). It is clear that for the signal model and the STBC scheme used in this paper $K = 4$. The 6th row of Table 1 corresponds to the case in which a new channel estimate is obtained for a particular user. For the case that a user enters the system complexity of the decoder is compared for different number of users and different methods of inverse matrix calculation in Table 2. Moreover, in Table 3, complexity reduction is compared when a user exits the system. Table 4 compares computational complexity of the two-stage update algorithm with exact algorithm and inverse matrix approximation algorithm.

Table 1 Computational complexity of the proposed algorithms for the ZF STBC decoder

	Computational complexity
Direct inverse	$8M_{new}(4N + 4M_{new} - 1)$ $+ 2(4M_{new})^3/3 + 5(4M_{new})^2/6 - 7(4M_{new})/6$
Neumann series approximation with $R = 2$	$(4M_{new})(4N - 1) + 8M_{new} + 4M_{new}(8M_{new} - 1)$
Neumann series approximation with $R = 3$	$(4M_{new})(4N - 1) + 12M_{new} + 8M_{new}(8M_{new} - 1)$
Update inflated	$K^3 + K^2(12M + 1) + (4K + 1)(4M)^2$ $+ 4M_{new}(4N - 1)$ $+ 8M_{new}(8M_{new} - 1)$
Update deflated	$K^3 + K^2(4M) + (K + 1)(4M)^2$ $+ 4M_{new}(4N - 1)$ $+ 8M_{new}(8M_{new} - 1)$
Updating new CSI	$2K^3 + K^2(16M - 3)$ $+ (K + 1)(4(M - 1))^2 + (4K + 1)(4M)^2$ $+ 4M(4N - 1)$ $+ 8M(8M - 1)$

Table 2 Numbers of complex-valued operations required for the ZF STBC decoder: a user is added to the system

Number of users (M)	8	16	24	32
Direct inverse	126822	396587	950683	1920182
Proposed update	70180	178116	329060	523012
Complexity reduction (updating)	45%	55%	65%	72%
Inverse approximation $R = 3$	51264	105536	168000	238656
Complexity reduction (approximation)	60%	73%	82%	87%

As it can be seen, in all three scenarios applying the proposed update techniques for matrix inversion results in considerable reduction in the computational complexity. In addition, inverse matrix approximation method has lower computational complexity compared to the updating method. This complexity reduction is obtained at the cost of BER performance degradation which will be examined in the next section.

5.2 Simulation results

In this section, we present the simulation results to evaluate and compare the proposed algorithms in terms of decoder BER. For update scenarios, $\tilde{\mathbf{Z}}_3^{-1}$ is used as the initial inverse matrix. In the simulations, we use BPSK modulation and assume that the channel model is flat fading.

In the first simulation, we examine the efficiency of utilizing Neumann series in the given system. Here, the number of antennas in the BS is set to $N = 320$ and the number of users is $M = 16$. The BER performance of the system is evaluated by changing the order of the Neumann series, i.e., $R = 2, 3, 4$. As it can be seen in Fig. 2, utilizing higher order of Neumann series exhibits

better BER performance. However, it should be noted that each additional term in the series will increase the computational complexity by an order of $O(M^2)$.

For the simulation of the proposed updating algorithms one user is added to or removed from the system. The number of antennas at the BS is set to $N = 320$ and the number of current users is equal to $M = 8, 16$ resulting in the ratio $\alpha \gg 1$, which guarantees the convergence of (13) with very high probability. In all subsequent simulations, three decoding algorithms are used:

I. Calculating the exact inverse of the current matrix and then utilizing update algorithms,

II. Approximating the current matrix inverse with Neumann series and then applying update algorithms,

Table 3 Numbers of complex-valued operations required for the ZF STBC decoder: a user is removed from the system

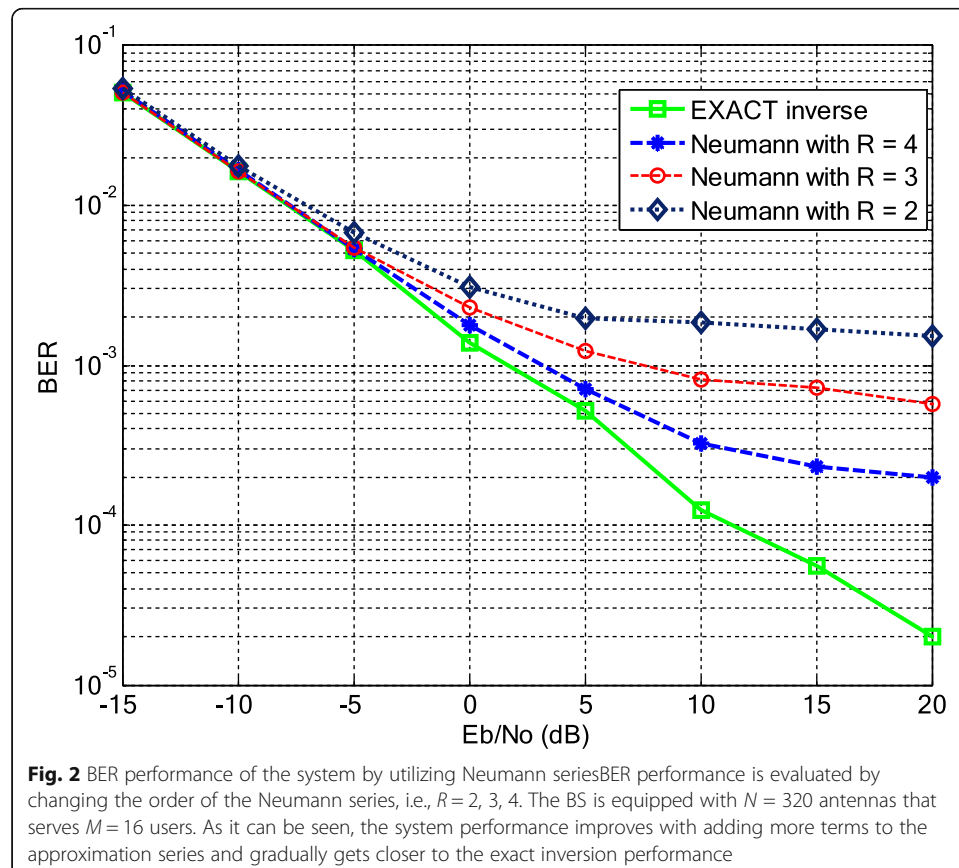| Number of users (M)	8	16	24	32
Direct inverse	88447	307610	778335	1631695
Proposed update	44588	112588	199020	303884
Complexity reduction (updating)	50%	63%	74%	81%
Inverse approximation $R = 3$	38976	91200	151616	220224
Complexity reduction (approximation)	55%	70%	80%	87%

Table 4 Numbers of complex-valued operations required for the ZF STBC decoder: new channel estimation for a user

Number of users (M)	8	16	24	32
Direct inverse	106565	350005	861392	1771797
Proposed update	68416	189920	364672	592672
Complexity reduction (updating)	36%	46%	58%	67%
Inverse approximation $R = 3$	45056	98304	159744	229376
Complexity reduction (approximation)	57%	71%	81%	87%

III. Using Neumann series without applying update algorithms.

Figure 3 shows BER performance after adding a user to the system. As it can be seen in Fig. 3a, utilizing Neumann series as well as the proposed algorithm for calculation of inverse matrix in the system with $M = 8$ users have a subtle performance loss compared to the exact case and they almost overlap. It is also seen in Fig 3-b, that when the number of simultaneous clients is increased, i.e., for $M = 16$, the performance loss of the approximation method will be more noticeable.

Simulation results for the case where a user is removed from the system is depicted in Fig. 4. As it can be seen in this figure, the number of users has a direct effect on the performance of the approximation method. This means that by comparing simulations (a) and (b), it is observed that BER performance of the system with $M = 8$ users is
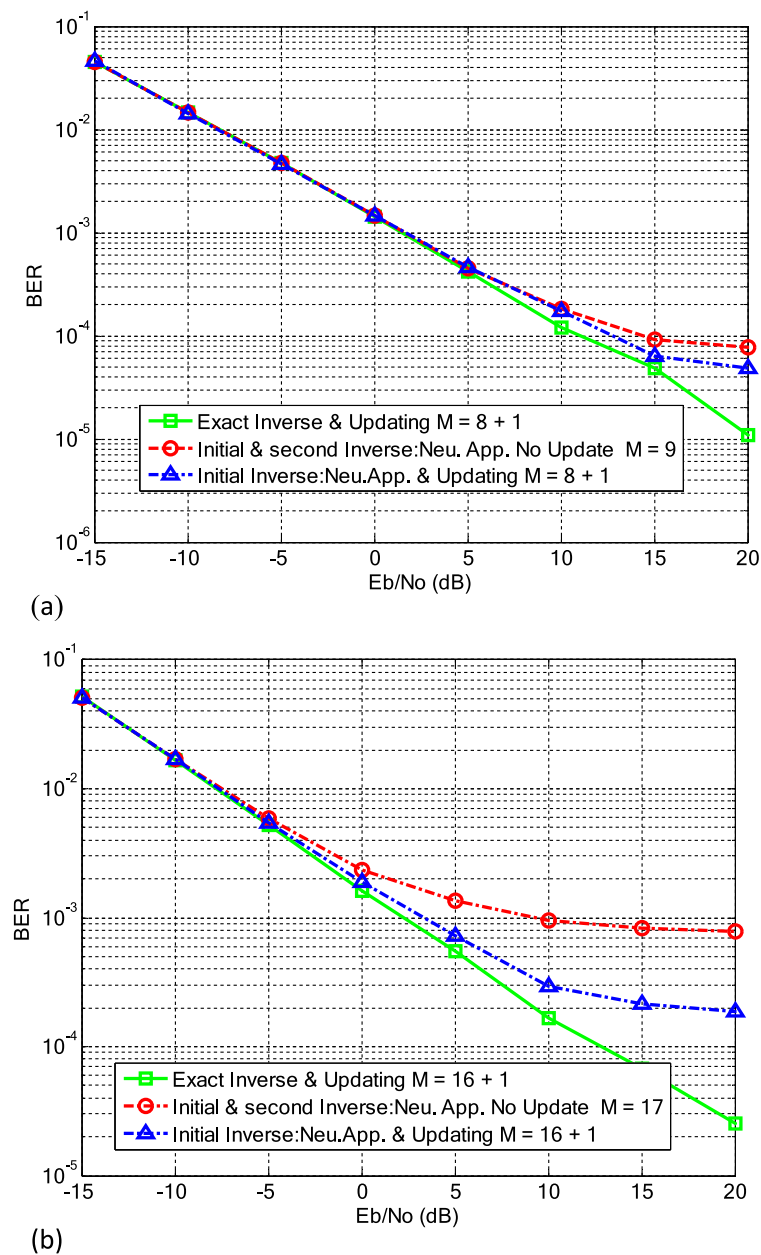

Fig. 2 BER performance of the system by utilizing Neumann seriesBER performance is evaluated by changing the order of the Neumann series, i.e., $R = 2, 3, 4$. The BS is equipped with $N = 320$ antennas that serves $M = 16$ users. As it can be seen, the system performance improves with adding more terms to the approximation series and gradually gets closer to the exact inversion performance

Fig. 3 BER performance comparison for inflation update and the number of current users $M = 8$, 16. BS is equipped with $N = 320$ antennas that serves $M = 8$, 16 users. In the event that a user is added to the system, the proposed algorithms are used to update the inverse matrix for decoding. The results for $M = 8$ is shown in simulation (**a**). Simulation **b** demonstrates the comparison results for $M = 16$ after a user is added. It is observed that using Neumann series to approximate the inverse matrix will degrade the BER performance compared to the exact matrix inverse update

nearly overlapped for all decoding methods, but for $M = 16$ users the BER performance of the approximation method is reduced compared to the exact inverse calculation.

In Fig. 5, which corresponds to matrix update without adding or removing a user, similar to previous simulations, by increasing the number of users, the BER performance of the approximation method is reduced compared to the exact algorithm.
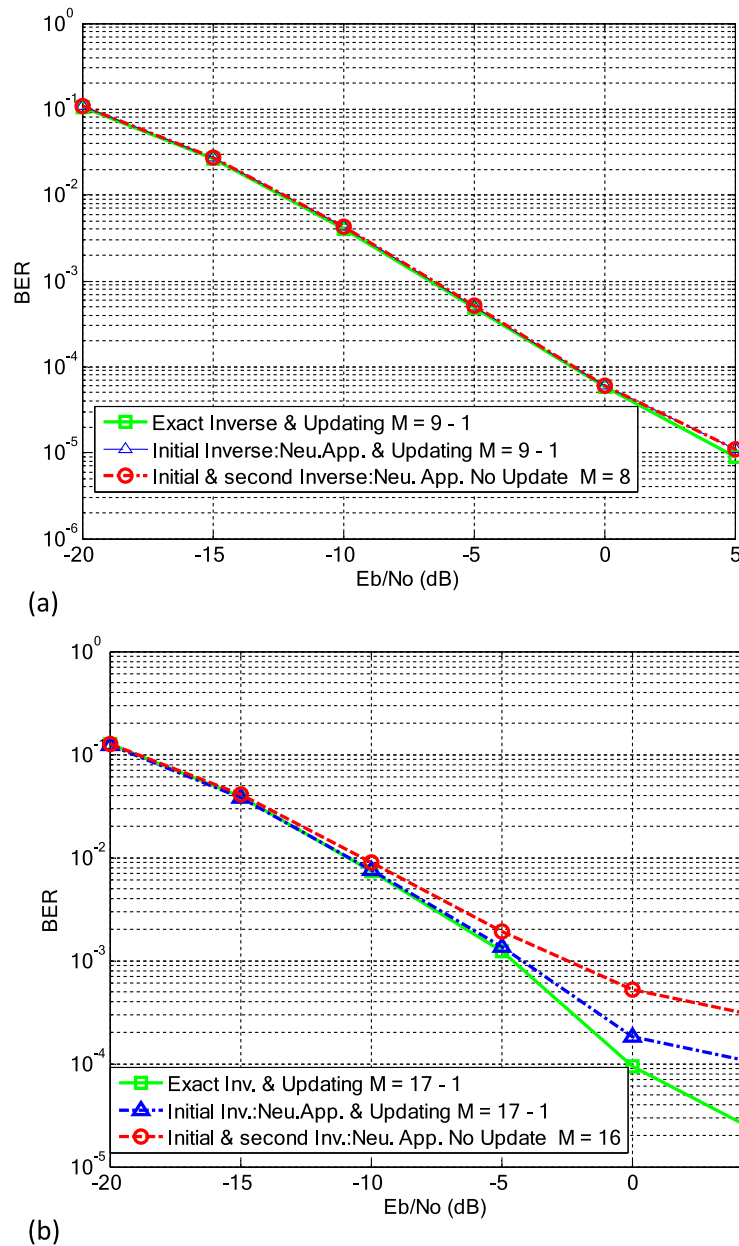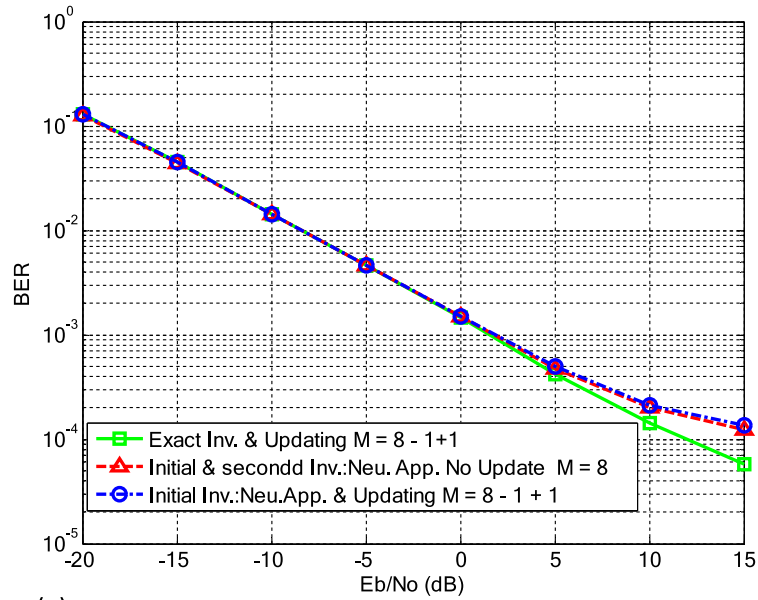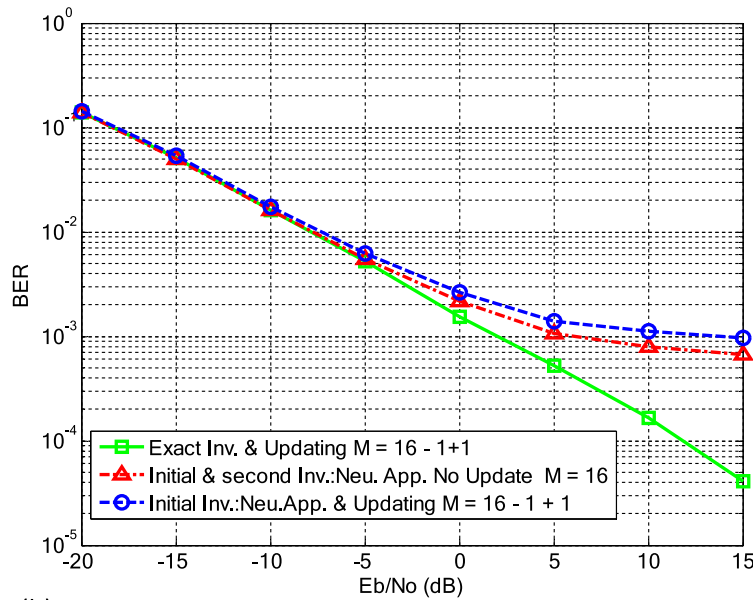
Fig. 4 BER performance comparison for deflation update and the number of current users $M = 9$, 17. BS is equipped with $N = 320$ antennas that serves $M = 9$, 17 users. Assuming that a user is removed from the system the proposed algorithm is used to update the inverse matrix for decoding. The results for $M = 9$ is shown in simulation **a** and simulation **b** demonstrates the comparison results for $M = 17$ after a user exits the cell. As it is predictable, the BER performance of the exact inverse matrix update algorithm is better than the approximation method

Notice that for the cases that a user is added or removed, the decoding algorithm II has a better BER performance than III whereas for updating the channel matrix of a specific user, algorithm III has a slightly better performance than II because the update algorithm is performed twice, and since the initial decoding in II is the Neumann series approximation, error propagation occurs and algorithm III in this case has a better performance.

Fig. 5 BER performance comparison for updating the CSI of 1 user for the number of users $M = 8$, 16.BS is equipped with $N = 320$ antennas that serves $M = 8$, 16 users. This simulation studies the case in which the channel estimation in changed for a particular user while the total number of users remains constant. Simulations (**a**) and (**b**) corresponds to the BER performance of the system with 8 and 16 users, respectively

6 Conclusions

In this paper, methods for efficient calculating and updating inverse of a matrix in decoding of the space-time codes in large-scale MIMO systems were evaluated. At the receiver, Neumann series are used to approximate the inverse of a matrix with large dimensions. Moreover, by utilizing matrix inversion identities, efficient algorithms are proposed to update the inverse matrix for the ZF and MMSE STBC decoders when users are entering or exiting the system as well as the case in which a new channel

estimation is obtained for a user. For STBC ZF decoding, the proposed methods are investigated from two perspectives: reduction of the computational complexity and the BER performance of the system. Based on the complexity analysis and the simulation results, the update algorithms have better BER performance compared to the approximation method while approximation of the inverse matrix imposes less computational complexity on the system. It is worth mentioning that similar approach will also be applicable when more users are added to or removed from the system. However, as the number of users to be added to or removed from the system increases, the reduction in computational complexity decreases.

Last, but not least, it should be noted that although the proposed methods are investigated for the case of STBC, they can be generalized for common spatial multiplexing schemes in massive MIMO systems.

Abbreviations
BER: Bit error rate; BLER: Block error rate; BS: Base station; CSI: Channel state information; i.i.d: Independent and identically distributed; MIMO: Multiple-input multiple-output; MMSE: Minimum mean square error; SDR: Software-defined radio; STBC: Space-time block code; ZF: Zero-forcing

Authors' contributions
SHM is the main author of the paper. He proposes the main idea, derives the expression of the computational complexity for the proposed methods, and evaluates the algorithms. JP reviews and finalizes the ideas, equations, simulation results, and the write up of the final manuscript. Both authors read and approve the final manuscript.

Authors' information
Seyed Hosein Mousavi is working toward his PhD degree at the Department of Electrical & Computer Engineering, University of Tabriz, Iran. His research interest is wireless communications and signal processing.
Jafar Pourrostam received BSc degree in electronic engineering from AmirKabir University of Technology, Tehran, Iran, and MSc degree in communication systems engineering from University of Tehran, Iran, and Ph.D. degree in electrical engineering from Michigan Technological University, USA in 2000, 2003 and 2007, respectively. He is currently an assistant professor at the Department of Electrical & Computer Engineering, University of Tabriz, Iran. His research interest includes Mobile Communication Systems and Signal Processing.

Availability of data and materials
The authors declare that all the data and materials in this manuscript are available.

Competing interests
The authors declare that they have no competing interests.

References
1. E. G. Larsson, O. Edfors, F Tufvensson, and T. L. Marzetta, Massive MIMO for next generation wireless systems, IEEE Comm. Mag. Vol. 52, No.2, pp. 186–195, 2014.
2. E. Björnson, J. Hoydis, M. Kountouris, M. Debbah, Massive MIMO systems with non-ideal hardware: Energy efficiency, estimation, and capacity limits. IEEE Trans. Inform. Theory. **60**(11), 7112–7139 (2014)
3. H.Q. Ngo, E.G. Larsson, T.L. Marzetta, Energy and spectral efficiency of very large multiuser MIMO systems. IEEE Trans. Comm. **61**(4), 1436–1449 (2013)
4. H. Wang, X. Yue, D. Qiao, W. Zhang, A Massive MIMO System with Space-Time Block Codes, in Proc. IEEE Inter. Conf. Communications (ICCC), Chengdu, China, 2016.
5. J.C. Belfiore, G. Rekaya, E. Viterbo, The Golden code: A 2×2 full rate space time code with non-vanishing determinants. IEEE Trans. Inform. Theory. **51**(4), 1432–1436 (2005)
6. H. Wang, X.G. Xia, Optimal normalized diversity product of 2×2 lattice-based diagonal space-time codes from QAM signal constellations. IEEE Trans. Inform. Theory. **54**(4), 1814–1818 (2008)
7. W. Zhang, T. Xu, X.G. Xia, Two designs of space-time block codes achieving full diversity with partial interference cancellation group decoding. IEEE Trans. Inform. Theory. **58**(2), 747–764 (2012)
8. L. Shi, W. Zhang, X.G. Xia, Space-time block code designs for two-user MIMO X channels. IEEE Trans. Comm. **61**(9), 3806–3815 (2013)

9. D. Zhu, B. Li, and P. Liang, On the matrix inversion approximation based on Neumann series in massive MIMO systems , in Proc. IEEE Int. Conf. Communications (ICC), pp. 1763–1769, 2015.

10. F. Rosàrio, F.A. Monteiro, A. Rodrigues, Fast matrix inversion updates for massive MIMO detection and precoding. IEEE Signal Proc Lett **23**(1), 75–79 (2016)

11. Q. Deng, L. Guo, C. Dong, J. Lin, D. Meng, X. Chen, High-throughput signal detection based on fast matrix inversion updates for uplink massive multiuser multiple-input multi-output systems. IET Commun. **11**(14), 2228–2235 (2017)

12. Q.Deng, L.Guo, Ch.Dong, X. Liang and J. Lin Hybrid Iterative Updates Detection Scheme for Uplink Dynamic Multiuser Massive MIMO Systems, in Proc. IEEE. Conf. (GC Wkshps), Singapore, 2017.

13. T. Taniguchi and Y.K.N. Nakajima Partial update of antenna weight in multiuser MIMO for time-variant propagation channel, in Proc. Conf. Antennas and Propagation(EUCAP), Paris, Mar 2017.

14. C. Chuan Yeh, K. N. Hsu and Y. H. Huang A low-complexity partially updated beam tracking algorithm for mmWave MIMO systems, in Proc. IEEE Glob. Conf. Sig. & Inf. Proc. (GlobalSIP), Washington, DC, 2016.

15. J. Eilert, D. Wu, D. Liu, Efficient Complex Matrix Inversion for MIMO Software Defined Radio (Proc. IEEE International Symposium on Circuits and Systems, LA, USA, May, 2007)

16. Burian A, Takala J, and M. Ylinen, a fixed-point implementation of matrix inversion using cholesky decomposition, in proc. IEEE 46th Midwest Symp. Circuits and Systems (MWSCAS) 3, 1431–1434 (2003)

17. A. Rontogiannis, V. Kekatos, K. Berberidis, A square-root adaptive V-BLAST algorithm for fast time-varying MIMO channels. IEEE Signal Process. Lett. **13**(5), 265–268 (2006)

18. M. Wu, B. Yin, A. Vosoughi, C. Studer, J. R. Cavallaro, and C. Dick, Approximate matrix inversion for high-throughput data detection in the large-scale MIMO uplink, in Proc. IEEE Int. Symp. Circuits and Systems (ISCAS), 2013, pp. 2155–2158.

19. M. Wu, B. Yin, G. Wang, C. Dick, J.R. Cavallaro, C. Studer, Large-scale MIMO detection for 3GPP LTE: Algorithms and FPGA implementations0. IEEE J. Sel. Top. Signal Process. **8**, 916–929 (2014)

Publisher's Note