Lattice reduction-ordered successive interference cancellation detection algorithm for multiple-input–multiple-output system

Yunchao Song, Chen Liu✉, Feng Lu

School of Electronic Science and Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, People’s Republic of China
✉ E-mail: liuch@njupt.edu.cn

Abstract: Lattice reduction (LR) is a powerful technique for improving the performance of linear multiple-input–multiple-output detection methods. The efficient LR algorithms can largely improve the performance of the linear detectors (LDs). Note that the ordered successive interference cancellation (OSIC) system can decrease the interference between antennas and provide performance gain of the LDs. In this paper, a novel LR-aided algorithm called NLR-OSIC improving the performance of the OSIC system has been proposed. Most existing LR algorithms are designed to improve the orthogonality of channel matrices, which is not directly related to the error performance of the OSIC system. While the authors’ algorithm maximises the signal-to-interference-plus-noise ratio (SINR) of the detected symbol in each stage of the OSIC system, thus exhibiting improved error rate than the previous LR-aided LDs and their corresponding OSIC algorithms. In each stage, the authors verify that maximising the SINR of the detected symbol can be formulated as a shortest vector problem which is solved by a suboptimal algorithm in this study. In the end of this study, the error rate performance of the proposed algorithm as well as the required complexity has been demonstrated through extensive computer simulations.

1 Introduction

Multiple-input–multiple-output (MIMO) technology is maturing and is being incorporated into emerging wireless broadband standards like long-term evolution [1]. Actually, the more antennas the transmitter or the receiver is equipped with, the more degrees of freedom that the system can provide, and the better the performance, such as the data rate or the link reliability [2]. The price to pay for MIMO is the increased complexity of the hardware and the complexity and energy consumption of the signal processing at both the transmitters and the receivers [2]. In MIMO systems, the detectors are usually great concerns at the receiver. Designing reliable and computationally efficient detectors has become a critical challenge of the MIMO systems. For example, the well-known maximum-likelihood detector (MLD) [3, 4] provides optimal bit error performance, however, it has exponential complexity. The other performance-optimal signal detectors, such as the sphere detector also suffers from exponential complexity [3, 5, 6]. Moreover, the LD algorithms cannot provide excellent performance, i.e. the zero forcing (ZF) and the minimum mean-square error (MMSE) detectors have polynomial complexity, but they only collect poor diversity in MIMO systems [4, 7]. Many improved algorithms have been proposed with various complexity and performance tradeoff. In [8, 9], the MMSE-SIC (successive interference cancellation) and the block-iterative generalised decision feedback equaliser algorithms have been proposed to achieve high performance while the complexity is still very high. Two local neighbourhood search methods – likelihood ascent search (LAS) [10] and reactive tabu search [11] obtain near-optimal performance for lower-order quadrature amplitude modulation (QAM) but exhibit considerable performance degradation for higher-order QAM. To further improve performance for higher-order QAM, layered tabu search was proposed in [12]; however, this algorithm involves considerable high complexity when the problem size and/or the constellation size is large.

Recently, an efficient scheme called lattice reduction (LR) [13–15] has been applied in the MIMO detection. It is always used in conjunction with the LDs. The LR-aided technique can enhance the performance of MIMO digital communication systems. It attempts to find a more orthogonal basis for the channel matrix, as the orthogonality of the channel matrix largely affects the performance of the wireless system [16]. The Lenstra, Lenstra, and Lovasz (LLL) algorithm [17] has firstly been considered for LR-aided data detection. It allows suboptimum detectors to exploit all the available diversity, while it has polynomial complexity. As an alternative to the LLL algorithms, the dual LLL (D-LLL) algorithm, which reduces the lattice in the dual space, was studied in [18, 19]. Element-based LR (ELR) algorithm [20] was proposed to reduce the diagonal elements of the noise covariance matrix, which enhances the asymptotic performance of LDs. The ELR provides better bit-error-rate (BER) performance than the LLL algorithm, especially for large MIMO systems [20]. However, though the LR-aided detection algorithms can achieve the full diversity, their performance is still far away from the MLD. Most recent works, such as [18–21] aim to develop LR-aided detectors with better performance, while some of them still bring larger complexity. As the polynomial complexity of the LR-aided detection algorithms is acceptable in MIMO systems, then moderate increases in the complexity are affordable. The algorithm proposed in this paper also improves the performance of the LR-aided detectors with affordable complexity.

As in the OSIC system, the interference from already-detected components is subtracted out from the received signal vector, resulting in better performance than the LDs. In this paper, we propose a novel LR-aided algorithm called novel lattice reduction (NLR)-OSIC for OSIC system. Compared with the LLL and ELR algorithms, the proposed algorithm has a different designing goal: to maximise the SINR of the detected symbol in each stage of the OSIC system. Note that the SINR of the detected symbol in each stage is proportional to the diagonal elements of R where $\mathbf{QR} = \mathbf{H}$ is the QR factorisation of the channel matrix $\mathbf{H}$, and we aim to use the LR scheme to maximise the diagonal elements. We verify that maximising the diagonal elements can be formulated as solving the shortest vector problem (SVP). As the SVP is a well-known problem, many algorithms can solve the SVP. By taking the
2 Preliminaries

2.1 System model

Now consider a MIMO system with $N_t$ transmitted antennas and $N_r$ received antennas as

$$y^c = H^c x^c + z^c,$$  \hspace{1cm} (1)

where $x^c \in \mathbb{C}^{N_t \times 1}$ is the received signal, $H^c \in \mathbb{C}^{N_r \times N_t}$ is a Rayleigh fading channel, $x^c \in \mathbb{C}^{N_t \times 1}$ is the transmitted signal, and $z^c \in \mathbb{C}^{N_r \times 1}$ is the Gaussian noise. We assume that the number of the received antennas is no less than the number of transmitted antennas.

The following assumptions are made throughout this paper.

\begin{align}
\text{(AS1)} & \quad \text{We write the channel matrix } H^c \text{ as } (h_1^c, h_2^c, \ldots, h_{N_t}^c), \text{ and the channel vectors } h_i^c, i = 1, 2, \ldots, N_t \text{ can be represented as}
\end{align}

$$h_i^c = [h_{i,1}^c, h_{i,2}^c, \ldots, h_{i,N_r}^c]^T,$$  \hspace{1cm} (2)

where $h_i^c$ are i.i.d. zero mean complex Gaussian variables with variance $(1/N_t)$ for $j = 1, 2, \ldots, N_r$.

\begin{align}
\text{(AS2)} & \quad \text{The transmitted symbols } x^c \text{ are the information symbols drawn from the QAM } \mathcal{S}, \text{ and the components } x_i^c \text{ of } x^c \text{ are i.i.d. variables with zero mean and average power } E(|x_i^c|^2) = \sigma_x^2, \text{ for } i = 1, 2, \ldots, N_t. \text{ Both the real and imaginary parts of } x_i^c \text{ are drawn from the set } \mathbb{D}, \text{ where } \mathbb{D} = [2m + 1 - \sqrt{M}, m = 0, 1, \ldots, \sqrt{M} - 1] \text{ with } M \text{ being the constellation size of } \mathcal{S}.
\end{align}

\begin{align}
\text{(AS3)} & \quad \text{The noise vector } z^c \text{ is an i.i.d. zero mean, circularly symmetric complex Gaussian vector and with covariance matrix}
\end{align}

$$E(z^c(z^c)^H) = \sigma_z^2 I.$$  \hspace{1cm} (3)

Then, the signal-to-noise ratio (SNR) is defined as \cite{7}

$$\rho = \frac{\sigma_x^2}{\sigma_z^2}. $$  \hspace{1cm} (4)

Note that (1) is equivalent to the real MIMO input–output linear model \cite{3}

$$y = Hx + z, $$  \hspace{1cm} (5)

where

$$y = \begin{pmatrix} R(y) \\ \triangle(y) \end{pmatrix}, \quad z = \begin{pmatrix} R(z) \\ \triangle(z) \end{pmatrix}, \quad x = \begin{pmatrix} R(x) \\ \triangle(x) \end{pmatrix}, \quad H = \begin{pmatrix} \triangle(H) & -\triangle(H) \\ \triangle(H) & \triangle(H) \end{pmatrix}.$$  \hspace{1cm} (6)

Then, $y \in \mathbb{R}^{K}, z \in \mathbb{R}^{K}, x \in \mathbb{R}^{K},$ and $H \in \mathbb{R}^{K \times N},$ where $K = 2N_t$ and $N = 2N_r$.

2.2 Review of the ZF and the MMSE OSIC systems

The ZF-OSIC system includes the ordering preprocess and the successive interference cancelation (SIC) scheme.

The SIC scheme can decrease the interference between antennas. Once we obtain some detection components of the transmitted signal, we can cancel these components from the received signal, so that we can eliminate the interference of these components. Here we introduce the detail of the SIC scheme which can be found in \cite{22}.

In case any specific detection ordering is used, the input–output relation of (4) along with the transmitted signal $x$, the received signal $y$, and the channel matrix $H$ need to be rearranged. Hence, without loss of generality, we assume that the transmit symbols $x_i$, $i = 1, 2, \ldots, N$ are sequentially detected at the receiver within $N$ stages in the order from the $N$th component to the first component given that $y$, $H$, and $x$ have been rearranged subject to the detection ordering method.

Assume the QR factorisation of the channel matrix $H$ is

$$H = QR,$$  \hspace{1cm} (7)

where $Q$ is a unitary matrix and $R$ is a upper triangular matrix. Substituting (5) into (4), we have

$$y = QRx + z.$$  \hspace{1cm} (8)

Multiplying (6) by the unitary matrix $Q^H$, (4) is equivalent to

$$\tilde{y} = Rx + \tilde{z}, $$  \hspace{1cm} (9)

where $\tilde{y} = Q^Hy$, $\tilde{z} = Q^Hz$.

Following the above discussion, the system detects the signal $x$ from the $N$th component to the first component. At the first stage, we have

$$x_N = \tilde{y}_N \frac{R_{N,N}}{R_{N,N}}, $$  \hspace{1cm} (10)

and the estimation of $x_N$ as $\hat{x}_N = Q(x_N')$, where $Q(\cdot)$ denotes the quantisation (slicing) operation appropriate to $\mathbb{D}$ which is defined above. Then, at stage $i (i \geq 2)$, we have

$$x'_{N-i+1} = \frac{\tilde{y}_{N-i+1} - \sum_{j=N-i+2}^{N} R_{N-i+1,N-j+1}\hat{x}_j}{R_{N-i+1,N-i+1}}, $$  \hspace{1cm} (11)

and the symbol $x_{N-i+1}$ is detected as $\tilde{x}_{N-i+1} = Q(x'_{N-i+1})$.

Now we briefly discuss the ordering process below. A detection order which maximises the instantaneous output SINR of the detected symbol at each stage of the SIC system substantially improves the error performance compared to any fixed detection order.

As a preprocessing and ordering approach, the Vertical Bell Labs Layered Space Time (VBLAST) system has been proved to have the optimal detection ordering \cite{22}. It maximises the signal-to-interference-plus-noise ratio (SINR) of the detected symbol at each stage substantially. We know that at stage $i$, the SINR of the detected symbol is proportional to the diagonal element $R_{N-i+1,N-i+1}$, $i = 1, 2, \ldots, N$ \cite{22}. The goal of this ordering is to find the permutation matrix $\Pi_I$ such that, keeping $R_k$, $k > N - i + 1$ unchanged, $R_{N-i+1,N-i+1}$ is maximised at the $i$th stage of the SIC column. The ordering algorithm is recursive and yields the optimal permutation $\pi$ in $N$ steps which means the components of $x$ is detected in the order of $\pi(1), \pi(2), \ldots, \pi(N)$.

The description of the ordering in the ZF-VBLAST system is indicated in Fig. 1.
The MMSE-VBLAST system has the equivalent channel matrix $H_{\text{MMSE}}$ and received signal $y_{\text{MMSE}}$ as

$$H_{\text{MMSE}} = \begin{bmatrix} H & \frac{1}{\sqrt{\rho}} \cdot \theta \end{bmatrix}, \quad y_{\text{MMSE}} = \begin{bmatrix} y \\ \theta \end{bmatrix},$$

where $I \in \mathbb{R}^{K \times N}$ is the unit matrix and $\theta \in \mathbb{R}^N$ is the zero matrix. Then replacing the $H$ and the $y$ in the ZF-VBLAST system by $H_{\text{MMSE}}$ and $y_{\text{MMSE}}$, we can get the MMSE-VBLAST system. It has been verified that the MMSE-VBLAST have better performance than the ZF-VBLAST system for it can provide larger SINR.

### 2.3 Previous LR-OSIC algorithm

LR is one technique which can obtain a near-orthogonal matrix. The LR-aided LDs are based on the fact that if channel matrix is more orthogonal, the decision region of LDs will be closer to that of the MLD [15]. Therefore, to improve the error performance of LDs, we employ LR to find a near-orthogonal channel matrix. As a result, LR-aided LDs yield error performance ‘close’ to the MLD and have the same error performance as the MLD if the reduced matrix is orthogonal. The orthogonality of the matrix $H$ can be measured in terms of the orthogonality defect which is defined as [20, 21, 23]

$$\delta = \frac{1}{\det(H^TH)} \prod_{i=1}^{N} |h_i|^2$$

where $h_i$ is the $i$th column of $H$, and $\delta \geq 1$. Decreasing the orthogonality defect $\delta$ can improve the orthogonality of the matrix.

Given the basis matrix $H$, LR algorithms find a reduced basis matrix $H$ leaving the lattice unchanged. Accordingly, reducing basis $H$ is equivalent to find a unimodular transformation $H = HT$ [14], where $T$ is a unimodular matrix (i.e. all the entries of $T$ are integers, and the determinant of $T$ is $\pm 1$), such that $H$ is near orthogonal.

The previous LR-OSIC in ZF case based on the reduced basis $H$ is now performed as follows [20]. We first write (4) as

$$y = \tilde{H}\tilde{x} + z,$$

where $\tilde{H} = HT$ and $\tilde{x} = T^{-1}x$. Applying scaling and shifting on $x$, i.e. $(x - I_Nx/v_1)/2$ where $I_N \in \mathbb{R}^{N \times N}$ is the identity matrix with entries 1, we transfer the domain of $x$ to consecutive integer sets, which are further transferred to the lattice-reduction domain as $\tilde{x} = T^{-1}(x - I_Nx/v_1)/2$, whose elements are in consecutive integer sets as well. Hence, we firstly use the OSIC algorithm in last subsection to detect $\tilde{x}$. However, instead of using quantisation operation $Q(\tilde{x})$ ($i = 1, 2, \ldots, N$) to get the estimation of $\tilde{x}$, in last subsection where $\tilde{x}$ can be obtained from (8) and (9), we obtain the estimation $\tilde{x}$ of $\tilde{x}$, $i = 1, 2, \ldots, N$ as [20]

$$\tilde{x}_i = 2\hat{b}_i + c_i,$$

where $c_i$ is the $i$th element of $c = T^{-1}1_{N+1}, \hat{b}_i = \lfloor(\tilde{x}_i - c_i)/2\rfloor$, and $\lfloor \cdot \rfloor$ is a rounding function. Once we get the estimation of $\tilde{x}$, the final detection result can be obtained by

$$\hat{x} = Q(T\tilde{x}).$$

where $Q(\cdot)$ denotes componentwise quantisation as defined above, and $\hat{x}$ is the estimation of $x$. The LR-OSIC in MMSE case replaces the $H$ and $y$ above with $H_{\text{MMSE}}$ and $y_{\text{MMSE}}$, which is similar with the MMSE-VBLAST system.

### 3 NLR-OSIC algorithm

#### 3.1 Model of the NLR-OSIC algorithm

The previous LR detection work mainly aims to obtain better performance of the LDs. Note that the OSIC system can provide better performance than the LDs. Our algorithm aims to use the LR scheme to improve the performance of the OSIC system. We improve the performance by maximising the SINR of the detected signal at each stage in OSIC system.

Our algorithm maximises the SINR of the detected symbol at the $i$th stage of the OSIC system by maximising the diagonal element $U_{N-i+1,N-i+1}$, where $U_{i,k} = (k = 1, 2, \ldots, N)$ is the $k$th diagonal element of $U$, and $HT - QU$ is the QR factorisation of $HT$. It means that we need to find a unimodular matrix $T$ and get the reduced lattice $\tilde{H} = HT$ such that $U_{N-i+1,N-i+1}$ is maximised at the $i$th ($i = 1, 2, \ldots, N$) stage.

We give a theorem below which can derive an equivalent target of our NLR-OSIC algorithm.

**Theorem 1:** If the QR factorisation of the matrix $\tilde{H} \in \mathbb{R}^{K \times N}$ is $\tilde{H} = QU$, and the OQ factorisation of the matrix $\tilde{H}^H$ is $\tilde{H}^H = GL$, where $Q$ and $G$ are the unitary matrix, $U$ is the upper triangular matrix, and $L$ is the lower triangular matrix. Then, we have

$$U_{i,i} = \frac{1}{L_{K-N+i,i}}.$$
3.2 Algorithm in each stage

We have shown the target of NLR-OSIC algorithm in last subsection. Our NLR-OSIC algorithm have \( N \) stages, i.e. we find a unimodular matrix to minimise \( L_{k-1,N-k+1} \) at the \( i \)th stage, then multiply these unimodular matrix to get the matrix \( V_i \) and the detail can be found in the following subsection. Now we give an algorithm to find a unimodular matrix to minimise \( L_{k-1,N-k+1} \) at the \( i \)th stage. In the following subsection, it will be shown that, before each stage, the reduced matrix has the same form as \( M \in \mathbb{R}^{m \times n} / \mathbb{Q}^m \), and

\[
M = \begin{bmatrix}
M(1) & 0 \\
M(2) & M(3)
\end{bmatrix},
\]

where \( M(1) \in \mathbb{R}^{m \times m} (n \geq m \geq 1) \), \( M(2) \in \mathbb{R}^{m \times m} \), and \( M(3) \in \mathbb{R}^{l \times l} (l = 0, 1, \ldots) \). Set \( M = QL \) and \( MT = QL \) as the \( QL \) factorisations of \( M \) and \( MT \) where \( T \) is a unimodular matrix. In this stage, our target is to find a unimodular matrix \( T \) such that \( L_{i,m} \) is minimised and \( L_{j+m-1,j} = L'_{j+m-1,j} \) for \( m < j \leq l + m \), where \( L_{i,b} \) and \( L'_{i,b} \) is the \( i \)th row and the \( i \)th column vector of the reduced lattice basis of \( \mathbb{Z}^l \) and \( \mathbb{Z}^l \), respectively. Keeping \( L_{j+m-1,j} = L'_{j+m-1,j} \) for \( m < j \leq l + m \) means that it cannot change the diagonal elements we have minimised in the previous stages.

Before deriving our algorithm, Lemma 1 will give us a method to find the above unimodular matrix.

**Lemma 1:** If \( T(m) \in \mathbb{R}^{m \times m} \) is a unimodular matrix, and \( I_{i,l} \in \mathbb{R}^{l \times l} \) is a unit matrix, then the matrix

\[
T(m) = \begin{bmatrix}
T(m) & 0 \\
0 & I_{i,l}
\end{bmatrix}
\]

is also a unimodular matrix, where \( 0 \) is the zero matrix. Suppose we reduce the lattice matrix \( M = MT(m) \), and let the \( QL \) factorisation of \( M \) and \( MT \) be \( M = QL \) and \( MT = QL \) respectively, we get

\[
L_{j+m-1,j} = L'_{j+m-1,j}, \quad m < j \leq l + m.
\]

**Proof:** See Appendix 2. \( \square \)

Lemma 1 tells us that we can find one unimodular matrix \( T(m) \) to achieve our target above. Then, we need to obtain the matrix \( T(m) \) from (54), we know

\[
L = \begin{bmatrix}
L'(1) & 0 \\
L'(2) & L'(3)
\end{bmatrix},
\]

then \( L_{a,m} = L'(1)_{a,m} \). From (51), \( L'(1)_{a,m} \) (i.e. \( L_{a,m} \)) is equal to the 2-norm of the \( m \)th column of \( M(1)T(m) \). Note that if we can find a unimodular matrix to obtain the shortest vector of the lattice \( M(1) \), then using the column permutation, we can obtain the smallest \( \bar{L}_{a,m} \). Minimising the shortest vector of the lattice \( M(1) \) can be solved by the methods solving the SVP. The SVP asks to find the shortest non-zero vector in a given lattice. Here, it means given the lattice \( M(1) \), find a unimodular matrix \( T(m) \) such that the \( m \)th column vector of the reduced lattice basis \( M(1)T(m) \) is the shortest vector.

Since the SVP has been studied for a long time, there are many algorithms to get the shortest vector. However, the optimal solution of the SVP is shown to have very large complexity which is not acceptable in MIMO detection [23], and the suboptimal algorithms which have polynomial complexity can achieve good performance. Here we modify the algorithm in [20], so that it can be applied in our problem and has smaller computational complexity. Our algorithm is suboptimal, however, this algorithm has smaller complexity than other algorithms.

Note that the product of a series of column-addition matrices is the unimodular matrix [20]. Our algorithm factorises \( T(m) \) to some column-addition matrices, then solves the column-addition matrices iteratively. Each column-addition matrix can be obtained by updating the \( k \)th column of the unit matrix \( I \) as

\[
I_k \leftarrow I_k + \mu_k I_k,
\]

where \( I \) and \( I_k \) are the \( i \)th and the \( l \)th columns of the matrix \( I \), respectively and \( \mu_k \) is an integer. Set \( W = M(1) \), then multiplying the matrix by the above column-addition matrix, we update the \( k \)th column of \( W \) as

\[
W_k \leftarrow W_k + \mu_k W_k.
\]

then get the new matrix \( \bar{W} \). After one column-addition operation, we find that the square of the 2-norm of \( \bar{W} 's \) the \( k \)th column is

\[
\| \bar{W}_k \|_2^2 = \| W_k + \mu_k W_k \|_2^2
\]

\[
= \| W_k + \mu_k W_k \|_2^2 \left( W_k + \mu_k W_k \right)
\]

\[
= W_k^H W_k + 2 \mu_k W_k^H W_k + \mu_k^2 W_k^H W_k,
\]

while other vectors’ norm remain unchanged. The reduction \( \Delta_k \) is defined as

\[
\Delta_k = \| W_k \|_2^2 - \| \bar{W}_k \|_2^2
\]

\[
= -2 \mu_k W_k^H W_k - \mu_k^2 W_k^H W_k,
\]

Considering (22), the partial derivation of \( \Delta_k \) can be written as

\[
\frac{\partial \Delta_k}{\partial \mu_k} = \| W_k \|_2^2 - 2 \| W_k \|_2^2
\]

and \( \Delta_k \) is maximised when

\[
\mu_k = \left[ \frac{W_k^H W_k}{W_k^H W_k} \right],
\]

where \( [\cdot] \) is a rounding function.

From the above, we use \( \mu_k \) in (24) to update \( W \) as (20) given the pair \((i, k)\), so that we can maximise \( \Delta_k \) in the column-addition operation. Then the problem becomes how to choose the pair \((i, k)\) sequentially to get the shortest vector.

Zhou and Ma [20] give one efficient way to choose the pairs \((i, k)\) sequentially. We call \( W_k \) reducible if there exist one or more positive \( \Delta_{ik} (i \neq k) \) in (22). In each iteration, our algorithm finds the largest reducible \( \| W_k \|_2^2 \). Then, we calculate all \( \Delta_{ik}, i \neq k \) and choose \( i = \arg \max_{i \neq k} \Delta_{ik} \). We use \( \mu_k \) in (24) to update the matrix \( W \). Continue the iterations until all \( \| W_k \|_2^2 \) are irreducible. Finally, we need to find the shortest column vector, and exchange the shortest vector with the last column of \( W \) so that the last column is shortest and \( \bar{L}_{a,m} \) is minimised. Then according to Lemma 1, we can use \( T(m) \) to get \( T(m) \).

A detailed description of the algorithm is given in Fig. 2.

3.3 Description of the NLR-OSIC algorithm

Firstly, we introduce one technique which can be used in our algorithm.

Suppose \( x \in \mathbb{R}^N \), we define the Householder vector [24] as

\[
v = x - \| x \| e_N^H,
\]

where \( e_N \) is the \( N \)th column of the \( N \times N \) unit matrix. Householder
Input: H. Output: $T'(m)$

initializing: $W = M(1)$, $T' = I_{N \times N}$

While

Find the largest reducible $\|W_k\|_2$;

for $i = 1 : N, i \neq k$

$$\mu_{i,k} = -\langle W_i, W_k \rangle$$

$$\Delta_{i,k} = 2\mu_{i,k} W_i^H W_k + \mu_{i,k}^2 W_i^H W_i$$

end

$i = \arg \max_{i \neq k} \Delta_{i,k}$;

if $\Delta_{i,k} > 0$

$$T_k' \leftarrow T_k' + \mu_{i,k} T_i'$$

$$W_k' \leftarrow W_k + \mu_{i,k} W_i$$

else

break;

end

Find the shortest vector $W_s$;

Exchange $W_s$ and $T_s$ with $W_m$ and $T_m', T'(m) = T'$;

Fig. 2 Description of the algorithm in each stage of NLR-OSIC

reflection (HR) [24] is defined as

$$P = I - \frac{2}{\|x\|^2} x x^H$$

and $P$ is a unitary matrix. Then, we have

$$P x = [\|x\|^2] e_N$$

From the above analysis, we will derive the unimodular matrix $V$ in (17) below. Our algorithm has $N$ stages, and in the $i$th stage, we use the LR scheme to find one unimodular matrix to minimise $L_{K-1+1,N-1}$ once we get these unimodular matrices, then multiply these unimodular matrices, we can get the matrix $V$. The detail is as follows.

Note $B = H^{-1}$, and $GL$ is the $GL$ factorisation of $BV$. Considering $B$, we can use the algorithm in last subsection to find a matrix $T(N)$ to minimise $L_{K,N}$ at the first stage, then we set $V' = T(N)$ and define $B^{N-1} = BT(N)$. Take $B^{N-1}$ into consideration, where $B^{N-1}_i$ is the $i$th column of the matrix $B^N$, we get the HR of $B^{N-1}_i$ as $P(N)$. Then, we define

$$\hat{B}^N = P(N)B^N$$

and we know $\hat{B}^N = [B^N]_i e_N$.

As $\hat{B}^N$ has the same form as $M$ in the last subsection, the algorithm in the last subsection can derive $T(N - 1)$ to minimise $L_{K-1,N-1}$ in the second stage. Set $V = T(N)T(N - 1)$, and define

$$\hat{B}^{N-1} = \hat{B}^N T(N - 1) = P(N)B^N(N)T(N - 1)$$

where $\hat{B}^{N-1}_i = [B^{N-1}]_i e_N$, and $B^{N-1}_i$ is the $i$th column of $B^{N-1}$. Set $\hat{B}^{N-1}$ as the matrix deleting the last column and the last row of $\hat{B}^{N-1}$. Take $\hat{B}^{N-1}_i$ into account, and get HR of $\hat{B}^{N-1}_i$ as $P(N - 1)$, then we have

$$\hat{P}(N - 1)\hat{B}^{N-1}_i = [B^{N-1}_i] e_{N-1}$$. Let

$$\hat{P}(N - 1) = \begin{bmatrix} \hat{P}(N - 1) & 0 \\ 0 & 1 \end{bmatrix}$$

and define

$$\hat{B}^{N-1} = \hat{P}(N - 1)B^{N-1}$$

then, it is easy to verify that $\hat{B}^{N-1}_i = 0, k = N - 1, N$, $j < K - N + k$.

Considering $\hat{B}^{N-1}_i$, if $\hat{B}^{N-1}_i$ has the same form as $M$ in the last subsection, i.e.

$$\hat{B}^{N-1} = \begin{bmatrix} M(1) & 0 \\ M(2) & M(3) \end{bmatrix}$$

where $M(1) \in \mathbb{R}^{K-1+1\times N-1}$, $M(2) \in \mathbb{R}^{K-1+1\times N-1}$, and $M(3) \in \mathbb{R}^{K-1+1\times K-1+1}$. Then, we can use the algorithm in the last subsection to find the unimodular matrix $T(N - 1)$ to minimise $L_{K, i - 1}$. We define

$$\hat{B}^{N-1}_{i - 1} = \hat{B}^{N-1}_i T(N - i - 1)$$

If we set $\hat{B}^{N-1}_{i - 1}$ as the matrix deleting the last $N - i$ columns and the last $N - i$ rows of $B^{N-1}_{i - 1}$, and take $\hat{B}^{N-1}_{i - 1}$ into consideration, we can get the HR of $\hat{B}^{N-1}_{i - 1}$ as $P(N - i - 1)$. Let

$$P(N - i - 1) = \begin{bmatrix} \hat{P}(N - i - 1) & 0 \\ 0 & I_{(i+1)\times(i+1)} \end{bmatrix}$$

and define

$$\hat{B}^{N-1}_{i - 1} = P(N - i - 1)\hat{B}^{N-1}_{i - 1} = P(N - i - 1)\hat{B}^{N-1}_{i - 1} = P(N - i - 1)\hat{B}^{N-1}_{i - 1}$$

\label{eq:27}

then $\hat{B}^{N-1}_{i - 1} = 0, k = N - i - 1, \ldots, N; j < K - N + k$, and $\hat{B}^{N-1}_{i - 1}$ has the same form with $M$ in the last subsection.

Note that $\hat{B}^{N-1}_{i - 1}$ has the form of (25), using the method above, we can get $\hat{B}^{N-2}_{i - 1}$, and use $\hat{B}^{N-2}_{i - 1}$ to get $\hat{B}^{N-3}_{i - 1}$. This procedure continues until we get $\hat{B}^1$. Finally, we have

$$\hat{B}^1 = \prod_{i=1}^N P(i) \prod_{i=1}^N T(N - i + 1)$$(28)

Manipulating (28) as

$$\hat{B}^1 = \prod_{i=1}^N P(i) \prod_{i=1}^N T(N - i - 1)$$

\label{eq:29}

Then, $\prod_{i=1}^N P(i)$ is a unitary matrix, $\prod_{i=1}^N T(N - i + 1)$ is a unimodular matrix, and $\hat{B}^1$ is a lower triangular matrix, then comparing with (17), we have

$$V = \prod_{i=1}^N T(N - i + 1)$$

\label{eq:30}

and from $V = T^{-H}$ as shown above, we have $T = V^{-1}$H. We get the steps of our NLR-OSIC algorithm as shown Fig. 3.

Once we have $T$, substitute it to (12), we get

$$y = Hx + z$$
where $\hat{H} = HT$, and $\hat{x} = T^{-1}x$ is in $\mathbb{C}^N$ as described in Section 2. Then, use the OSIC algorithm in Section 2 to derive the estimation of $\hat{x}$, and we can obtain the final detected signal through (14). Our NLR-OSIC in MMSE case also replaces the $H$ and $y$ with $H_{\text{MMSE}}$ and $y_{\text{MMSE}}$.  

### 4 Simulation and the analysis

#### 4.1 Performance analysis and simulations

In the OSIC system, we expand (7) as

$$\hat{y}_i = R_{i,i}x_i + \sum_{j=1}^{N} R_{i,j}x_j + \hat{z}_i.$$  

Suppose we detect $x_j$ ($j > i$) as $\hat{x}_j$, and $\Delta_j = x_j - \hat{x}_j$, then (31) can be written as

$$\tilde{y}_i - \sum_{j=1}^{N} R_{j,i}\hat{x}_j = R_{i,i}x_i + \sum_{j=1}^{N} R_{i,j}\Delta_j + \tilde{z}_i.$$  

As we know, the bit error probability of $x_i$ is affected by the SINR of $x_i$, where

$$\text{SINR}_i = \frac{E\left((R_{i,i}x_i)^2\right)}{E\left(\left(\sum_{j=1}^{N} R_{i,j}\Delta_j + \tilde{z}_i\right)^2\right)}.$$  

(33)

From (33), reducing the error propagation of $x_j$ ($j > i$) and enlarging $R_{i,i}$ can help us improve the SINR of $x_i$. In our algorithm, our target is to maximise $R_{i,i}$ from $R_{N,N}$ to $R_{1,1}$. In the first stage, maximising $R_{N,N}$ is equivalent to maximising the SINR of $x_N$, and the error probability of $x_N$ will be minimised, which means the error propagation of $x_N$ will be minimised. Then, in the second stage, as the error propagation of $x_N$ is minimised, maximising $R_{N-1,N-1}$ will result in maximising the SINR of $x_{N-1}$. Then, we can sequentially maximise the SINR of $x_{N-2}$, $x_{N-3}$ and so on. Compared with our algorithm, the previous LR algorithms did not aim to maximise the SINR of $x_j$ in each stage of the OSIC system, which will result in worse performance than our algorithm in the OSIC system.

Next, we validate the performance of the NLR-OSIC algorithm through the computer simulations over various MIMO systems. We compare the performance of the NLR-OSIC algorithm with the LLL-OSIC [17, 20], the ELR-OSIC [20], the ZF (MMSE)-OSIC [22], and the ML algorithm [3, 4]. The LLL-OSIC and the ELR-OSIC are briefly described in Section 2, and all the ordering process of LLL-OSIC, ELR-OSIC are the same as the VBLAST system. The ML algorithm has the optimal performance while it has exponential complexity. Fig. 4 shows that LR-aided algorithms have better performance than the VBLAST system while have worse performance than the ML algorithm. Fig. 4 represents that the LLL-OSIC algorithm has better performance than ELR-OSIC in the ZF case in $8 \times 8$ system, and $N_1 = N_2 = 3$. Fig. 5 is simulated in $8 \times 8$ system with 16QAM in MMSE case. Figs. 4 and 5 are simulated in uncorrelated fading channel described in Section 2. The performance of these algorithms in Figs. 4 and 5 are shown as a function of the SNR. These two figures show that LR-aided algorithms have better performance than the VBLAST system while have worse performance than the ML algorithm.}

![Fig. 4](http://example.com/fig4.png)

**Fig. 4** Comparison of the BER of different algorithms in ZF case in $8 \times 8$ system with 16QAM for uncorrelated fading channel

![Fig. 5](http://example.com/fig5.png)

**Fig. 5** Comparison of the BER of different algorithms in MMSE case in $8 \times 8$ system with 16QAM for uncorrelated fading channel
where \( H \) is the channel matrix defined in Section 2, \( C_r \) and \( C_t \) denote the channel covariance matrices at the receiver and transmitter, and \( A^{1/2}A^{1/2} = A \). We assume having uniform linear array antenna configuration [20, 25], and \( C_r \) and \( C_t \) have the Toeplitz structures of

\[
C_r = \begin{pmatrix}
1 & c_r & \cdots & c_r^{N_r-1} \\
(\cdot) & 1 & \cdots & c_r^{N_r-2} \\
\vdots & \vdots & \ddots & \vdots \\
(\cdot) & \vdots & \cdots & 1
\end{pmatrix}, \quad (35)
\]

\[
C_t = \begin{pmatrix}
1 & c_t & \cdots & c_t^{N_t-1} \\
(\cdot) & 1 & \cdots & c_t^{N_t-2} \\
\vdots & \vdots & \ddots & \vdots \\
(\cdot) & \vdots & \cdots & 1
\end{pmatrix}, \quad (36)
\]

where \( c_r \) and \( c_t \) denote the values of the correlation between the adjacent antennas and \((\cdot)^*\) denotes the conjugate of \((\cdot)\). Fig. 6 shows the attainable BER performance as a function of \( c_r/c_t \) for \( c_r = c_t = 0.3 \) in 8 x 8 system with 16QAM. We simulate the performance in the MMSE case, for the MMSE can provide better performance and is more widely used in MIMO system. From Fig. 6, we find that our proposed algorithm also can achieve better error performance than the previous LR-OSIC detectors in the MIMO systems in correlated fading channel.

### 4.2 Complexity analysis of NLR-OSIC algorithm

To a large extent, the complexity of the LR-aided algorithms will be affected by the required number of iterations, i.e. the number of the updates of the matrix \( H \) (which will be assessed by means of simulation results in Figs. 7 and 8), since across the iterations parallel or pipelined hardware structures are hardly possible.

Figs. 7 and 8 compare the number of updates of the NLR-OSIC with LLL-OSIC and ELR-OSIC algorithms, demonstrating the empirical CDFs of the total number of iterations. Fig. 7 describes the CDFs of the number of updates in the ZF case with 16QAM for uncorrelated fading channels, and \( N_r = N_t = 8 \) (as the MMSE case in uncorrelated fading channels has similar result with the ZF case in uncorrelated fading channels, and for reasons of space economy, we omit the figures in the MMSE case), while Fig. 8 describes the MMSE case with 16QAM for correlated fading.
channels, and \( N_t = N_r = 8 \). From Figs. 7 and 8, we easily find that the LLL-OSIC algorithm has the largest number of the iterations compared to the NLR-OSIC and the ELR-OSIC algorithms, while the ELR-OSIC has the smallest number of iterations.

We now give a brief analysis about a picture of the per-iteration complexity. At the \( t \)-th stage of our algorithm, each iteration need to update a \( (K-i+1) \times (N-i+1) \) matrix. While both the LLL and ELR algorithms should update a \( K \times N \) matrix in each iteration [17, 20]. The computational complexity is compared in Figs. 9 and 10, showing the empirical CDFs of the total number of arithmetic operations. The simulation environment of Fig. 9 (Fig. 10) is the same as Fig. 7 (Fig. 8). The simulation results show that the LLL also has the largest computational complexity, whereas the ELR-OSIC has the lowest computational complexity, and our proposed algorithm has moderate computational complexity.

5 Conclusion

The LR-aided algorithms have become promising detection algorithms in MIMO system, for their excellent performance and complexity. In this paper, we propose a novel LR-aided algorithm which aims to use the LR scheme to improve the performance of the OSIC system. Like the VBLAST system, we improve the performance by maximising the SINR of the detected symbol at each stage of the OSIC system. In fact, maximising the SINR is equivalent to a shortest vector problem. In addition, through computer simulations, we also demonstrated that our proposed algorithm is capable of obtaining better BER performance than the previous LR-OSIC algorithms with moderate increase in complexity.

6 Acknowledgment

This work was supported by the Natural Science Foundation of China under Grants 61372126 and 61302101.

7 References

1. Dahman, E., Parkvall, S., Skold, J., Beming, P.: ‘3G Evolution HSPA and LTE for mobile broadband’ (Academic, 2008)
2. Rusek, F., Persson, D., Lau, B.K., et al.: ‘Opportunities and challenges with very large arrays’, IEEE Signal Process. Mag., 2005, 22, (1), pp. 40–60
3. Jalden, J., Ottersten, B.: ‘On the complexity of sphere decoding in digital communications’, IEEE Trans. Signal Process., 2005, 53, (4), pp. 1474–1484
4. Larsson, E.G.: ‘MIMO detection methods: How they work’, IEEE Signal Process. Mag., 2009, 26, (3), pp. 91–95
5. Hassibi, B., Vikalo, H.: ‘On sphere decoding algorithm I: expected complexity’, IEEE Trans. Signal Process., 2005, 53, (8), pp. 2808–2818
6. Hassibi, B., Vikalo, H.: ‘On the sphere-decoding algorithm II: generalizations, second-order statistics, and applications to communications’, IEEE Trans. Signal Process., 2005, 53, (8), pp. 2819–2834
7. Jiang, Y., Varanasi, M.K., Li, J.: ‘Performance analysis of ZF and MMSE equalizers for MIMO systems: in-depth study of the high SNR regime’, IEEE Trans. Inf. Theory, 2011, 57, (4), pp. 2088–2026
8. Liang, Y.C., Cheu, E.Y., Bai, L., Pan, G.: ‘On the relationship between MMSE-SIC and BI-GDFE receivers for large multiple-input multiple-output channels’, IEEE Trans. Signal Process., 2008, 56, (8), pp. 3627–3637
9. Jiang, Y.C., Sun, S., Ho, C.K.: ‘Block-Iterative generalized decision feedback equalizers for large MIMO systems: algorithm design and asymptotic performance analysis’, IEEE Trans. Signal Process., 2006, 54, (5), pp. 2035–2048
10. Vaidyanathan, G., Mohammadi, S.K., Chockalingam, A., Rajan, B.S.: ‘A low-complexity detector for large MIMO systems and multicarrier CDMA systems’, IEEE J. Sel. Areas Commun., 2008, 26, (3), pp. 473–485
11. Rajan, B.S., Mohammadi, S.K., Chockalingam, A., Sridhri, N.: ‘Low-complexity near-MMSE decoding of large non-orthogonal STBCs using reactive tabu search’ Proc. IEEE Int. Symp. Information Theory, June 2009, pp. 1993–1997
12. Sridhri, N., Datta, T., Chockalingam, A., Rajan, B.S.: ‘Layered tabu search algorithm for large-MIMO detection and a lower bound on ML performance’, IEEE Trans. Commun., 2011, 59, (11), pp. 2955–2963
13. Lenstra, A., Lenstra, H., Lovasz, L.: ‘Factoring polynomials with rational coefficients’, Math. Ann., 1982, 261, pp. 515–544
14. M. Windpassinger, C., Fischer, R.: ‘Low-complexity near-maximumlikelihood detection and precoding for MIMO systems using lattice reduction’. Proc. IEEE Inf. Theory Workshop, March 2003, pp. 345–348
15. Ling, C.: ‘On the proximity factors of lattice reduction-aided decoding’, IEEE Trans. Signal Process., 2011, 59, (6), pp. 2795–2808
16. Maurer, J., Matz, G., Seethaler, D.: ‘Low-complexity and full-diversity MIMO detection based on condition number threshold’. Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing, March 2003, pp. 345–348
17. Yao, H., Wuerf, G.W.: ‘Lattice-reduction-aided detectors for MIMO communication systems’. Proc. IEEE Global Telecommunications Conf., November 2002, pp. 42–46
18. Liang, Y.C., Cheu, E.Y., Bai, L., Pan, G.: ‘On the sphere decoding algorithm II: generalizations, second-order statistics, and applications to communications’ IEEE Trans. Inf. Theory, 2011, 57, (4), pp. 2088–2026
19. Vaidyanathan, G., Mohammadi, S.K., Chockalingam, A., Rajan, B.S.: ‘A low-complexity detector for large MIMO systems and multicarrier CDMA systems’, IEEE J. Sel. Areas Commun., 2008, 26, (3), pp. 473–485
20. Rajan, B.S., Mohammadi, S.K., Chockalingam, A., Sridhri, N.: ‘Low-complexity near-MMSE decoding of large non-orthogonal STBCs using reactive tabu search’ Proc. IEEE Int. Symp. Information Theory, June 2009, pp. 1993–1997
21. Zhou, Q., Ma, X.: ‘Element-Based lattice reduction algorithms for large MIMO detection’, IEEE J. Sel. Areas Commun., 2013, 31, (2), pp. 274–286
22. Chen, C., Shen, W.: ‘A new lattice reduction algorithm for LR-Aided MIMO linear detection’, IEEE Trans. Wirel. Commun., 2011, 10, (8), pp. 2417–2422
23. Wabnik, H., Foschini, G.J., Golden, G.D., Valenzuela, R.A.: ‘V-BLAST: an architecture for realizing very high data rates over the Rich-Scattering wireless channel’. Proc. Int. Symp. Signals, Systems and Electronics, September 1998, pp. 295–300
24. Wubben, D., Seethaler, D., Jalden, J., Matz, G.: ‘Lattice reduction’, IEEE Signal Process. Mag., 2011, 30, (1), pp. 70–91
25. Goldub, G.H., Loan, C.F.V.: ‘Matrix computations’ (Johns Hopkins University Press, 3rd edn., 1996)
26. Lee, K., Chun, J., Hanzo, L.: ‘Optimal lattice-reduction aided successive interference cancellation for MIMO systems’, IEEE Trans. Wireless Commun., 2007, 6, (7), pp. 2438–2443

8 Appendix

8.1 Appendix 1: Proof of Theorem 1

Proof: As \( QU \) is the QR factorisation of \( \tilde{H} \), we know

\[
\tilde{H}^{-1} = [(QU)^†]^H = [U^† Q^{-1}]^H. \tag{37}
\]

Note \( Q \) is the unitary matrix, and we have \( Q^{-1} = Q^H \), then (37) can be written as

\[
U^{-1} = U^† Q^{-1} = U^† Q^H. \tag{38}
\]

For \( U \) is an upper triangular matrix, suppose

\[
U = X_{K-N+1} \tag{39}
\]

where \( X \in \mathbb{R}^{N \times N} \) is the upper triangular matrix, and \( \theta_{(K-N) \times N} \) is a \((K-N) \times N \) zero matrix. If \( K = N \), it means that \( \theta_{(K-N) \times N} \) is...
empty, and \( U=X \). From [24], \( U^{-1} \) has the form as follows

\[
U^{-1} = \begin{bmatrix} Y_{11}^\prime \\ 0_{K-N\times N} \end{bmatrix}.
\] (40)

where \( Y = X^{-1} \) is also a upper triangular matrix. As \( XY = I \), we know

\[
\sum_{i=1}^{N} X_{i,i} Y_{i,i} = 1, \quad i = 1, 2, \ldots, N.
\] (41)

As both \( X \) and \( Y \) are upper triangular matrices, we know \( X_{i,i} = 0, i > k \) and \( Y_{k,k} = 0, k > i \). Then

\[
X_{i,k} Y_{k,i} = 0, \quad i \neq k.
\] (42)

Substituting (42) into (41), we have \( X_{i,i} Y_{i,i} = 1, \) \( i = 1, 2, \ldots, N \). So,

\[
X_{i,i} = \frac{1}{Y_{i,i}}, \quad i = 1, 2, \ldots, N.
\] (43)

Multiplying (40) by the row permutation matrix \( A \), we have

\[
AU^{-1} = \begin{bmatrix} 0_{(K-N)\times N} \\ Y_{11}^\prime \end{bmatrix}.
\] (44)

As \( Y \) is the upper triangular matrix, \( Y_{11}^\prime \) is the lower triangular matrix, then \( AU^{-1} \) is the lower triangular matrix. From (38), we know

\[
H^{-1} = QU^{-1} = QA^{-1}AU^{-1}A^{-1}. \] (45)

As the row permutation matrix is the unitary matrix \( A^{-1} \) and \( A \) are the unitary matrices, and we know \( QA^{-1} \) is a unitary matrix. Note that \( AU^{-1} \) is the lower triangular matrix, and \( (QA^{-1})(AU^{-1}) \) is the \( QL \) factorisation of \( H^{-1} \). As \( H^{-1} = GL \) is the \( QL \) factorisation, we have \( L = AU^{-1}A^{-1} \). From (39) and (44), we have

\[
U_{i,j} = X_{i,j}, \quad L_{K-N+j,i} = Y_{i,j}.
\] (46)

Substituting (46) into (43), we can get (15).

8.2 Appendix 2: Proof of Lemma 1

Proof: Firstly, as \( T(m) \) is a real unimodular matrix, we can figure that the determinant of \( T(m) \) is \( \pm 1 \), and all the elements of \( T(m) \) are integers. Then, it is easy to know that all the elements of \( T(m) \) are integers. Moreover, from [24], we know that

\[
|T(m)| = |T'(m)| \cdot |I_{1\times 1}|.
\] (47)

where \( |\cdot| \) is the determinant. Then

\[
|T(m)| = \pm 1 \cdot 1 = \pm 1.
\] (48)

So \( T(m) \) is also a unimodular matrix.

Then, we have

\[
MT(m) = \begin{bmatrix} M(1) & 0 \\ M(2) & M(3) \end{bmatrix} \begin{bmatrix} T(m) \\ 0 \end{bmatrix} = \begin{bmatrix} M(1)T(m) \\ M(2)T(m) \end{bmatrix}.
\] (49)

Suppose the \( QL \) factorisation of \( M(1), M(3) \), and \( M(1)T(m) \) are

\[
M(1) = Q(1)L(1), \quad M(3) = Q(3)L(3), \quad M(1)T(m) = Q(1)L(1),
\] (50)

then it is easy to verify that, the \( QL \) factorisation of \( M \) and \( MT(m) \) are

\[
M = \begin{bmatrix} Q(1) & 0 \\ Q(3) & 0 \end{bmatrix} \begin{bmatrix} L(1) \\ L(2) \end{bmatrix}, \quad M = \begin{bmatrix} Q(1) & 0 \\ Q(3) & 0 \end{bmatrix} \begin{bmatrix} L(1) \\ L(2) \end{bmatrix},
\] (52)

and

\[
MT(m) = \begin{bmatrix} Q(1) & 0 \\ Q(3) & 0 \end{bmatrix} \begin{bmatrix} L(1) \\ L(2) \end{bmatrix} = Q(1)L(1), \quad MT(m) = \begin{bmatrix} Q(1) & 0 \\ Q(3) & 0 \end{bmatrix} \begin{bmatrix} L(1) \\ L(2) \end{bmatrix}.
\] (53)

where \( L(2) = Q(3)^{-1}M(2), \quad L(2) = Q(3)^{-1}M(2)T(m) \). Note the \( QL \) factorisation of \( M \) and \( MT(m) \) is \( M = QL \) and \( MT(m) = QL \), respectively, then

\[
L' = \begin{bmatrix} L(1) \\ L(2) \end{bmatrix}, \quad \tilde{L} = \begin{bmatrix} L'(1) \\ L'(2) \end{bmatrix}.
\] (54)

Comparing the two terms in (54), we can get (18). □