A Design of Fixed-Complexity Sphere Decoder Combined With Interference Mitigation Algorithm for Downlink MU-MIMO Systems

MINJOON KIM
Korea Electronics Technology Institute, Seongnam-si 13509, South Korea
e-mail: mjoon@keti.re.kr

This work was supported in part by the Technology Innovation Program under Grant 20016036; and in part by the In-Vehicle Infotainment Integrated AP and Application SW for Multi Tasking of Multi Users funded by the Ministry of Trade, Industry & Energy (MOTIE), South Korea.

ABSTRACT In this paper, a design and implementation of fixed-complexity sphere decoder (FSD) combined with interference mitigation algorithm for downlink (DL) multiuser (MU) multiple-input multiple-output (MIMO) system is presented. To overcome performance degradation from inter-user interference in DL MU-MIMO systems, a hardware friendly optimized interference mitigation algorithm is proposed. The proposed algorithm achieves 0.5 dB near-optimal performance, which is 4 dB better than existing interference whitening filter for the soft-output decision. In this paper, both hard and soft-output FSD detectors are implemented with the proposed interference mitigation algorithm to function 4 × 4 antenna MIMO detection and interference mitigation for 4 additional streams. A fully pipelined architecture is employed to support 3.6 Gbps at 150 MHz for signals modulated by 64-QAM. From the synthesis result, the designed symbol detector has a gate count of 887 K for the FSD without pre-processor and a gate count of 1220 K for interference mitigation. By comparing previously implemented MIMO detectors, the proposed design represents a promising architecture for DL MU-MIMO systems in terms of significant performance improvement relative to inter-user interference and the compatible hardware implementation maintaining high-throughput within acceptable gate count.

INDEX TERMS Fixed-complexity sphere decoder (FSD), inter-user interference, MIMO detection, multiuser multi-input multi-output (MU-MIMO), VLSI design.

I. INTRODUCTION

In wireless communications, increasing transmission speed in a limited frequency band has become progressively more important with the spread of broadband access services utilized by various handheld devices such as smart-phones, tablets, and laptops [1], [2]. Higher transmission rates have been achieved by various technologies, and studies to find methods to improve transmission rates are ongoing. For decades, multiple-input multiple-output (MIMO) communication has been considered a key technology for achieving high data rates [3], [4].

Recently, multiuser (MU)-MIMO antenna technology has been studied and applied to many consumer products. For such products, the representative standard is IEEE 802.11ac, which is a widely used wireless local area network standard in MU-MIMO downlink (DL) systems [5]. Standardization trends, such as IEEE 802.11n, have attracted considerable attention, and IEEE 802.11ac aims at a transmission speed of 1 Gbps or more [6]. For IEEE 802.11ac, DL MU-MIMO transmission technology in the 5 GHz band has been proposed [7]. The maximum transfer rate of the physical layer is approximately 7 Gbps with this technology.

Unlike conventional MIMO systems, in MU-MIMO systems, multiple users can communicate simultaneously using a single access point (AP). In other words, in a DL...
environment, a signal sent from the AP to a given user also reaches other users [8]. These signals are referred to as interfering signals, and users receive both the desired signal and the interfering signals simultaneously [9]. Consequently, performance is reduced by these interfering signals. Therefore, most DL MU-MIMO systems attempt to solve the interference problem by employing precoding technology in the AP [10], [11], [12], [13]. The pre-coding technique is a technique that the AP multiplies a precoding matrix on the channel in advance such that all users are not affected by the interference signals [10], [14], [15]. For example, in the representative zero-forcing beamforming technique, theoretically, an interfering signal can be zero, each user and the AP comprise of a MIMO system, and the entire MU-MIMO system is divided into MIMO [14]. However, in real-world environments, errors, referred to as practical errors, inevitably occur due to errors in channel estimation and incomplete calculation [16]. This practical errors result in an incomplete precoding matrix; thus, some users will still receive interfering signals [17]. Compared to the desired signal, the interference signal is relatively small; however, these signals are enough to affect system performance [18]. Therefore, an interference-aware receiver is required.

Various approaches to mitigate inter-user interference for DL MU-MIMO systems have been proposed. These methods can be categorized into two representative approaches: interference whitening (IW) and interference detection (ID) [19]. IW simply treats interference as Gaussian noise [20]; thus, it is less computationally complex than ID; however, it is unsatisfactory for strong interference. The IW approach has been studied by many interference rejection combiners and their extensions [22], [23], [24], [25], [26]. However, they still cannot achieve high performance with strong interference. In contrast, ID detects both desired signals and interference signals to mitigate interference [21]. ID can achieve high performance (regarded as optimal), however, it suffers from high computational complexity. There were several ways to reduce complexity using existing MIMO detector algorithms [27], [28], [29], [30], [31]. This approach can reduce complexity compared to identity, but is much more complex than IW. Thus ID is not considered for practical implementation. As mentioned previously, the performance of IW is not satisfactory under strong interference conditions, which is an issue with current and future of DL MU-MIMO systems.

In this paper, we propose a symbol detector architecture with an interference mitigation algorithm suitable for DL MU-MIMO systems to realize a suitable hardware implementation. To date, many hardware implementation studies related to the MIMO detection algorithms have been published [32], [33], [34], [35], [36], [37]. However, no studies have attempted to develop implementations for DL MU-MIMO systems despite their widespread usage. Therefore, we focus on improving existing MIMO detectors for DL MU-MIMO systems by combining them with an interference mitigation algorithm. In this process, the target is to minimize cost relative to compatibility with existing MIMO detectors, which we expect to simplify the adaptation of the MIMO detectors to DL MU-MIMO systems. The primary contributions of this paper are summarized as follows.

- An improved interference mitigation algorithm is proposed. The proposed algorithm can achieve better performance than existing interference mitigation algorithms in terms of bit error rate (BER) and computational complexity. Furthermore, the proposed algorithm is suitable for a hardware implementation, particularly with tree-search based MIMO detectors.
- An efficient method to combine the proposed interference mitigation algorithm and a fixed-complexity sphere decoder (FSD) algorithm is presented. The proposed interference mitigation algorithm is partially calculated according to the successive architecture of an FSD algorithm [38], [39].
- For high throughput, a fully-pipelined hardware architecture design is presented. The implemented design can support 3.6 Gbps at 150 MHz for 5 × [4, 4] with 64-QAM modulated signals in DL MU-MIMO systems.

The remainder of this paper is organized as follows. Section II reviews MIMO and MU-MIMO system models and briefly discusses MIMO detection and interference mitigation algorithms. Section III describes the proposed interference mitigation algorithm and its implementation. Performance and computational complexity comparisons with other algorithms are presented in Section IV. Section V describes the hardware design and implementation of the FSD with the proposed algorithm. Implementation results are compared to previously implemented MIMO detectors in Section VI, and conclusions are given in Section VII.

II. SYSTEM MODEL

This section introduces a basic review of MIMO and DL MU-MIMO systems. First, a mathematical description and MIMO detection algorithms for MIMO systems are presented. Then, the mathematical description and interference mitigation algorithms for DL MU-MIMO systems are discussed.

A. MIMO SYSTEM

1) MATHEMATICAL DESCRIPTION

We consider a MIMO system with $M$ transmit and $N$ receive antennas [4]. Here $H$ denotes an $M \times N$ channel matrix. For simplicity, symbol time in packet transmission is ignored. $x_m$ denotes symbols transmitted from the $m$-th transmit antenna and $y_n$ denotes signals received from the $n$-th receive antenna. Then, the signal vector received over the MIMO channel is given by

$$ y = Hx + n, \quad (1) $$

where $y = [y_1, y_2, \ldots, y_N]^T$, $x = [x_1, x_2, \ldots, x_M]^T$ and $n = [n_1, n_2, \ldots, n_N]_n^T$ denote a received signal vector, a transmitted signal vector, and a noise vector, respectively. $n$ is an independent and identically random distributed (i.i.d) complex additive white Gaussian noise vector with zero mean and
variance $\sigma^2$. Generally, MIMO channel capacity increases linearly as $\min(M, N)$. Note that controlling the trade-off between receive diversity and multiplexing gain is a fundamental problem in MIMO transmission.

2) MIMO DETECTION

A maximum likelihood (ML) detector can offer the best performance; however, it is unsuitable for hardware implementation due to its exponential complexity. This problem has led to the development of tree-search based detectors. The FSD algorithm provides near-ML performance in a predefined tree architecture with fixed complexity [38]. FSD can be considered a hybrid model that combines a full extension (FE) stage to perform ML detection and a single extension (SE) stage to perform linear detection as shown in Fig. 1 (a). The FSD algorithm can be implemented by regular successive stages and, similar to other tree-search algorithms, can be pipelined architecture. Thus, an FSD can be employed to implement a high-throughput MIMO detector. Note that categorizing FE stage and SE stages and ordering the streams in the SE stage are very important. The most popular stream ordering method is performed by estimating stream magnitude based on a channel matrix. In this paper, the tree-search unit is only considered for a hardware implementation.

The FSD algorithm begins with QR decomposition such that channel matrix $H$ is decomposed as $H = QR$, where $Q$ and $R$ are an $M \times N$ unitary matrix and an $M \times M$ upper triangular matrix, respectively. By multiplying $Q^H$ and the system model can be expressed as

$$ z = Rx + v, \quad (2) $$

where $z = Q^H y$ and $v = Q^H n$. For the first $p$ levels, all possible branches are extended. For the remaining $M - p$ levels, a single branch determined by successive interference cancellation (SIC) on each node is extended. These processes are defined by the FE and SE stages. The FSD generates a candidate list expressed as follows

$$ L = \{\tilde{x}_1, \ldots, \tilde{x}_{|Q|^p}\} \quad (3) $$

where $\tilde{x}_k = [\tilde{x}_{k,M}, \ldots, \tilde{x}_{k,1}]^T$, $\Omega$ denotes the constellation set. Here, the $i$-th level is expressed as

$$ \tilde{x}_{k,i} = \begin{cases} \text{one of symbols } \in \Omega, & i \in \text{FE} \\ Q((z - \sum_{j=1+i}^M R_{i,j} \tilde{x}_{j,i})/R_{i,i}), & i \in \text{SE} \end{cases} \quad (4) $$

where $Q(\cdot)$ denotes a slicing operation. The FSD solution is given by

$$ \tilde{x}_{\text{FSD}} = \arg \min_{\tilde{x} \in L} \|z - R\tilde{x}\|^2. \quad (5) $$

To extend the hard-output FSD to soft-output FSD (SFSD), a list-extended SD (LSD) algorithm has been proposed, as shown in Fig. 1 (b), such that additional tree searching is performed for $L$ candidates with the shortest Euclidean distance (ED) among $L$ [39]. For additional tree searching, only node expansion for its opposite bit in the SE stage is required. For example, for 16-QAM modulated signals from four transmit antennas, if $FE = 1$ and $L = 2$, additional visiting nodes for list expansion can be calculated as $2 \times 4 \times (3 + 2 + 1) = 48$.

Although the FSD can achieve ML performance and is suitable for a fully-pipelined hardware implementation with high throughput, it can have significant computational complexity depending on the total number of levels, particularly sensitive to the number of FE levels.

B. DL MU-MIMO SYSTEM

1) MATHEMATICAL DESCRIPTION

Here, a DL MU-MIMO system with a single AP with $M$ antennas for each user and $K$ users each having $N$ antennas is considered as shown in Fig. 2. The received signal vector for the $i$-th user, $y_i$, size of $N \times 1$, can be expressed as

$$ y_i = H_i \tilde{V}_i x_i + n_i, \quad (6) $$

where $n_i$ is an i.i.d complex zero-mean Gaussian noise with variance $\sigma^2$, $H_i$ is the $N \times M$ DL channel matrix corresponding to user $i$, $\tilde{V}_i$ is the precoding matrix, and $x_i$ is the transmitted data vector. Note that the transmit power is not considered for simplicity.

In this paper, a block diagonal precoding matrix design algorithm is assumed [14]. This scheme employs a precoding matrix that satisfies $H_i \tilde{V}_i = 0$ for $i \neq j$. Note that, under ideal conditions, the interference signal in (6) can be eliminated. However, in practice, the precoding matrix inevitably includes errors that occur due to channel estimation, feedback bit quantization, and subcarrier grouping. Such practical errors cannot be estimated; thus an imperfect precoding matrix $\tilde{V}_i$ can be represented by separating the error matrix as follows

$$ \tilde{V}_i = V_i + E_i, \quad (7) $$

where $E_i$ denotes the practical error matrix and $V_i$ denotes the ideal precoding matrix satisfying $H_i V_i = 0$ for $i \neq j$. With these terms, (6) can be represented in the following simplified form:

$$ y_i = H_D x_D + H_I x_I + n_i, \quad (8) $$

where $H_D = H_i (V_i + E_i)$, $H_I = [H_i E_i]_{j=1,j\neq i}^K$, $x_I = [x_j]^T_{j=1,j\neq i}$ and $x_D = x_i$.

2) INTERFERENCE MITIGATION

As mentioned previously, interference mitigation algorithms can be categorized into two representative approaches [19], i.e., IW and ID. The straightforward approach to inter-user interference is to treat interference as Gaussian noise [20]. To whiten the effective noise, which is assumed to be colored
Gaussian noise, the whitening filter is multiplied as follows

$$W_{yi} = WH_{D}x_{D} + WH_{I}x_{I} + Wn_{i},$$  \hspace{1cm} (9)$$

where the whitening filter and covariance matrix can be calculated as $W = R_{v_{i}}^{-\frac{1}{2}}$ and $R_{v} = H_{I}H_{I}^{\dagger} + N_{R}$, respectively. Note that $W$ is derived using Cholesky decomposition:

$$(W^{-1})^{\dagger}W^{-1} = H_{I}H_{I}^{\dagger} + N_{R}.$$  \hspace{1cm} (10)$$

The complexity of IW approach is relatively small because it is linear filter. Another approach to mitigating interference is to detect the interference signal together with the desired signal [21]. The ML solution of ID using joint ED can be defined as

$$\{\hat{x}_{D}, \hat{x}_{I}\} = \arg \min \|y_{i} - H_{D}\hat{x}_{D} - H_{I}\hat{x}_{I}\|^2.$$  \hspace{1cm} (11)$$

In addition, the FSD application of ID can be performed using a rank-deficient SD technique [28]. However, because ID approach detects interference symbols directly, complexity increases significantly as the number of interference signals increases. Furthermore, it must be modified by expanding existing MIMO detectors to detect more streams, which impacts compatibility with existing.

FIGURE 1. Tree structure for four streams with a 16-QAM modulation scheme. (a) hard-output FSD and (b) list-extended soft-output FSD.

FIGURE 2. Simplified DL MU-MIMO system structure.

III. PROPOSED ALGORITHM

This section introduces the proposed interference mitigation algorithm. First, the proposed algorithm is derived in consideration of trade off between computational complexity and performance. Then, a method to implement the proposed interference mitigation algorithm with the FSD algorithm is presented.

A. ALGORITHM DESCRIPTION

The goal of the proposed algorithm is to achieve an efficient interference mitigating technique to ensure high performance under the interference conditions without significantly increasing complexity compared to existing MIMO detectors. The proposed algorithm includes both IW and ID methods because the IW method alone does not achieve sufficiently high performance under strong interference conditions. The IW method involves a filtering process to whiten interference signals. Note that interference signals are detected directly as desired signals with the ID method.

Therefore, the key idea is to minimize the influence of an interference signal when detecting a desired signal using IW as a preprocessing step and to detect the interference signal for only the candidate desired symbol vectors detected by the FSD. The IW filter makes the effective channel matrix orthogonal, i.e., $(WH_{I})^{H}(WH_{I}) \simeq I$; thus, linear detection using a ZF detector is sufficient for adequate performance. Note that the effective channel matrix is not perfectly orthogonal in practice. However, note that the detection of interference signals are detected directly as desired signals with the ID method.

The overall flow of the proposed algorithm is divided into three stages.

1) IW. Prior to detecting the desired signal with the FSD, an IW filter is applied to minimize the effect of the interference signal. Here, the interference signal is considered the same as whited noise, and the detection of
the request signal does not have a colored effect on each stream.

2) FSD. A tree-search detector used in existing MIMO systems is utilized to detect a desired signal. Note that we assume an FSD algorithm for the hard-output detector and an SFSD algorithm for soft-output detector.

3) IC. The interference signal vector is detected based on the desired signal candidate vector obtained by the FSD, and the final output is generated. If a hard-output detector is assumed, the interference signal is calculated according to each candidate signal vector, and the final ED is recalculated to select the desired signal candidate vector that minimizes this value. With the soft-output detector, the ED, including the interference signal according to each desired signal candidate vector, is recalculated to generate the LLR value per bit.

### B. IMPLEMENTATION METHOD

The IW filter minimizes the influence of the interference signal and the FSD detects the desired signal without considering the interference signal. The IC process removes influence of the interference signal based on the candidate vectors. Here, the \( N_c \) candidate vectors of the desired signal obtained by the FSD are denoted \( \hat{x}_{D,1}, \cdots, \hat{x}_{D,N_c} \). These vectors are obtained by the FSD tree architecture. The ED can be obtained for each candidate vector. For the hard-output detector, the smallest ED value is used as the final detection vector. In the proposed algorithm, the IC is inserted to remove the influence of the interference signal prior to determining the result. Here, the interference signal corresponding to the candidate vector \( \hat{x}_{I,1}, \cdots, \hat{x}_{I,N_c} \) should be detected. This differs from joint detection in that detecting the interference signal does not affect the detection of the desired signal. As mentioned previously, it is considered that the detection of interference signals should target low complexity rather than accurate detection by minimizing the influence on the detection of the request signal. Therefore, in the proposed algorithm, interference signal detection is achieved with fast and low complexity using a ZF detector. First, QR decomposition in (2) is applied to the system model in (9), and is then passed to the IW filter as follows.

\[
Q^H W_{I} = R_{X_D} + Q^H W_{I} x_{I} + Q^H W_{I}
\]

where \( Q \) is an unitary matrix and \( R \) is an upper triangular matrix such that \( QR = WH_D \). The zero forcing detector of the interference signal \( \hat{x}_{I,i} \) for the \( i \)-th candidate vector \( \hat{x}_{I,i} \) is implemented as follows.

\[
Q^H W_{I} \hat{x}_{I,i,ZF} = Q^H W_{I} - \hat{R}_{D,i}
\]

By multiplying the pseudo inverse of \( Q^H W_{I} \) on both side, (13) can be rewritten in terms of \( \hat{x}_{I,i} \), as follows.

\[
\hat{x}_{I,i,ZF} = (Q^H W_{I})^{-1}(Q^H W_{I} - \hat{R}_{D,i})
\]

\[
= (Q^H W_{I})^{-1}(Q^H W_{I} - \hat{R}_{D,i})
\]

\[
= (H_I^H W^H Q^H W_{I} - \hat{R}_{D,i})
\]

\[
= (H_I^H W^H Q^H W_{I}) - (H_I^H W^H Q^H W_{I} - \hat{R}_{D,i})
\]

\[
= (H_I^H W^H Q^H W_{I}) - (H_I^H W^H Q^H W_{I} - \hat{R}_{D,i})
\]

\[
= (H_I^H W^H Q^H W_{I}) - (H_I^H W^H Q^H W_{I} - \hat{R}_{D,i})
\]

Through the assumption of (14), complex operations can be avoided, and the effect on actual performance is considered to be minimal. In addition, \( \hat{x}_{I,i} \) is finally obtained through the slicing function as follows.

\[
\hat{x}_{I,i} = \text{slice}(\hat{x}_{I,i,ZF})
\]

where \( \text{slice}(\cdot) \) denotes the slicing function to determine a symbol belong to one of the QAM constellations. The final output can be re-determined using (11) rather than the existing ED.

### IV. ALGORITHM PERFORMANCE ANALYSIS

In this section, BER performance of multiple interference mitigation algorithms is presented for both hard-output and soft-output detection cases under two scenarios with different antenna configurations. In addition, a comparison of computational complexity approximated by the multiplier in the tree-search process is also presented for hard-output and soft-output detection cases.

#### A. BER COMPARISON

Computer simulations were conducted for hard-output and soft-output scenarios to evaluate multiple interference mitigation algorithms including the proposed algorithm. The simulations considered two scenarios in a DL MU-MIMO system comprising two users with three or four antennas each and an AP with six or eight antennas corresponding to the users, such that \( 6 \times [3, 3] \) and \( 8 \times [4, 4] \). For simplicity, all scenarios assumed that the desired and interfering user used 16-QAM mode. In addition, FSD and SFSD MIMO detection algorithms were used for each scenario. To verify the performance of the interference mitigation scheme, BER performance was evaluated realized to energy per bit to the noise-power-spectral density (Eb/N0) and relative to signal-to-interference ratio (SIR), respectively. Here, SIR is

| Parameter                  | Value                        |
|----------------------------|------------------------------|
| Scenario Name              | Uncoded                      | Coded                      |
| Communication              | DL MU-MIMO \[14\]            |                            |
| Antenna Configuration      | Scenario 1: 6 \times [3, 3]  |                            |
|                            | Scenario 2: 8 \times [4, 4]  |                            |
| Channel Model              | i.i.d Rayleigh Fading        | 16-QAM                     |
| Modulation                 |                               |                            |
| Eb/N0                      | \( 0 \sim 40 \text{ dB} @ \text{SIR} = 15 \text{ dB} \) | \( 0 \sim 25 \text{ dB} @ \text{Eb/N0} = 40 \text{ dB} \) |
| SIR                        |                               |                            |
| Channel Encoding           | none                         | PCCC (1/2)                 |
| Channel Decoding           | none                         | log-MAP (iter=8)           |
| Symbol Detection           | FSD \[38\]                  | SFSD \[39\]               |
| Interference Mitigation    | 1. No Scheme                 | 2. Interference Whitening \[20\] |
|                            | 3. Interference Joint Detection \[28\] | 4. Proposed Scheme         |
defined as $\delta = 10\log_{10}(\|H_D\|^2/\|H_I\|^2)$ to adjust the influence of the interference signal in (8) caused by an error in the precoding matrix in (7). The simulation scenarios and parameters are summarized in Table 1.

For the scenarios 1 and 2 with different antenna configurations, the uncoded and coded BER was evaluated and is presented from Fig. 3 to Fig. 8, respectively. Figure 3 and Fig. 6 show the BER versus Eb/N0 with the SIR fixed at 15 dB...
in order to determine the effect of traditional noise on performance due to channel estimation and synchronization errors. On the other hand, Fig. 4 and Fig. 7 show the BER versus SIR with the Eb/N0 fixed at 40 dB to show how interference affects performance. Finally, in Fig. 5 and Fig. 8, the coded BER performance using the channel codec is presented.

The uncoded BER performance was evaluated using hard-output detection with an FSD. As shown in Table 1, four interference mitigation algorithms were compared. Here, ID \((D)\) denotes \(D\) stream of interference signal is detected selectively according to a previously proposed method [28] because the original ID is too complex, and proposed \((F)\) denotes the proposed algorithm with the number of FEs of \(F\) in the FSD, which can generate more candidate outputs from the MIMO detector. Other interference mitigation algorithms do not depend on the number of candidate output; thus, \(F\) was set to 1. On the other hand, the coded BER performance using soft-output detection with the SFSD. All compared interference mitigation algorithms were set the same in the hard-output detection case. To generate more candidate outputs from the MIMO detector, the number of list extensions was set to \(L\) in the SFSD for proposed \((L)\). The number of FEs of \(F\) was set to 1 and \(L = 2\) for all cases (except for proposed \((L = 16)\)). For both the hard- and soft-output detection cases, the performance gap between ID and the proposed algorithm is controlled by parameter \(D, F\) and \(L\). However, the proposed algorithm is much simpler relative to computation and more compatible relative to hardware design than the ID algorithm. In other words, if similar performance can be achieved, we argue that the proposed algorithm is more suitable for DL MU-MIMO receivers.

Look at the scenario 1, with the antenna configuration of \(6 \times [3, 3]\). In Fig. 3, due to the influence of interference, it shows a performance convergence in which the BER does not decrease even if Eb/N0 increases. On the other hand, ideal \((\text{SIR} = 0\text{dB})\) does not show the performance convergence, and it explains why figure interference mitigation is required in a DL MU-MIMO system. Similarly, as shown in Fig. 4, for the increasing interference strength, performance gaps can be observed between none, IW, and ID. At a BER of \(10^{-5}\), IW can achieve 4 dB better than none, and ID \((D = 1)\) and ID \((D = 2)\) can achieve 2.5 dB and 7.5 dB better than IW. The BER convergence in Fig. 3 results this significant performance gap. For the proposed scheme, as going high SIR, proposed \((F = 1)\) and proposed \((F = 2)\) can achieve performance that is comparable to that of ID \((D = 1)\) and ID \((D = 2)\) within 0.5 dB and 2 dB gap, respectively. As shown in Fig. 5, relatively similar results were obtained in the hard-output detection case. IW achieved 3 dB better performance than none, and ID \((D = 1)\) and ID \((D = 2)\) achieved 5 dB and 9.5 dB better than IW, respectively. In addition, proposed \((L = 2)\) achieved 3 dB near performance to ID \((D = 1)\) and proposed \((L = 16)\) achieved 2 dB near performance to ID \((D = 2)\).

For the scenario 2, with the antenna configuration of \(8 \times [4, 4]\), the proposed scheme can show the more powerful performance because the number of interference signal to detect is increased. In Fig. 6, the similar result can be seen with scenario 1, although the performance gap between IW and ID is reduced. Similarly, as shown in Fig. 7, a BER of \(10^{-3}\), IW can achieve 5 dB better than none, and ID \((D = 1)\) and ID \((D = 2)\) can achieve 1 dB and 6 dB better than IW. Also, proposed \((F = 1)\) and proposed \((F = 2)\) can achieve performance that is comparable to that of ID \((D = 1)\) and ID \((D = 2)\) within 0.5 dB and 1 dB gap, respectively. As shown in Fig. 8, IW achieved 3 dB better performance than none, and ID \((D = 1)\) and ID \((D = 2)\) achieved 2.5 dB and 5 dB better than IW, respectively. In addition, proposed \((L = 2)\) and proposed \((L = 16)\) achieved 0.5 dB near performance to ID \((D = 1)\) and ID \((D = 2)\), respectively.

B. COMPLEXITY COMPARISON

It is difficult to estimate the design cost of the algorithm; thus, the approximated computational complexity was used to compare the complexities of the algorithms. The FSD algorithm is assumed in this paper; thus, we used the characteristic of the tree search algorithm. The tree architecture of the FSD comprises multiple nodes, and each node performs similar computations such that the number of visiting nodes is usually considered the total complexity of tree search algorithms. Therefore, for the compared interference mitigation algorithms, we first analyzed the number of visiting node and approximated the number of multiplications calculated per node. Then, the number of multiplications for ED calculation was added. Note that other processes, such as FSD channel ordering and QR decomposition, are not changed by the interference mitigation algorithms. In addition, for simplicity, pre-processing for IW with Cholesky decomposition was not considered in this analysis because it has much lower complexity than a tree search unit.

Table 2 shows the approximated complexity for hard-output detection with an FSD with the different interference mitigation algorithms. The preprocessing for IW is not included; thus, none and IW are the same in this analysis. First, ID and proposed scheme makes more number of visiting nodes in tree searching. Next, the proposed algorithm requires IC, which can be performed by each node; therefore, the number of multiplications per node is double that of the other algorithms. Finally, the number of ED calculations depends on the number of candidate outputs, and, compared to none and IW, ID and the proposed algorithm require double the number of multiplications for each calculation. At the upper part of the table, the calculation equation in each part is described using the parameter \(N, P, F\), and \(D\). At the lower part of the table, the calculation results of scenario 1 and scenario 2 are shown. Also for comparison, a normalized value based on none and IW is presented. As shown in BER performance results, proposed \((F = 1)\) and proposed \((F = 2)\)
show performance that is comparable to that of ID \( (D = 1) \) and ID \( (D = 2) \), respectively. However, as shown in Table 2, the normalized values of their approximated complexity are eight and nine times less than that of ID.

Table 3 shows the same analysis for the soft-output detection using the SFSD. Here, the number of visiting node was calculated, including additional nodes from the list extension technique. The proposed algorithm can achieve near performance to ID with 4 and 15 times lower complexity.

### V. ARCHITECTURE DESIGN

In this section, the architecture design of the FSD combined with the proposed interference mitigation algorithm for DL MU-MIMO systems is presented. First, the design target and overall architecture are presented. In addition, issues that should be considered when designing the symbol detector and overall architecture efficiency are discussed. Note that the architecture can be divided into two stages, i.e., the IW stage and the symbol detection stage. Design specifics are explained in the following.

#### A. DESIGN OVERVIEW

The designed symbol detector aims to improve communication performance of a DL MU-MIMO system by adding an interference mitigation algorithm to an existing MIMO detector. The considered environment is an \( 8 \times 4 \) DL MU-MIMO system comprising \( 4 \times 4 \) MIMO detector with interference mitigation of four streams. The modulation scheme is set to 16-QAM for both the desired and interference signals, and the MIMO detector is designed by the FSD and SFSD algorithms. Here, we focus on the meaningful hardware unit; thus, the tree search architecture, the preprocessing unit (e.g., FSD channel ordering and QR decomposition), and the calculator (e.g., minimum value finder and LLR calculator) were not implemented. The overall architecture for the hard-output detector is shown in Fig. 9, and the extra blocks for the soft-output detector are shown in Fig. 10.

A DL MU-MIMO was introduced to improve system throughput, such that the designed symbol detector can support high throughput. Generally, throughput \( \Phi \) can be calculated as

\[
\Phi = f_c \times \frac{B \times N_T}{N_{clk}}
\]

where \( f_c \) denotes the clock frequency, \( B \) is bit per symbol such that \( B = \log_2 M \) and \( N_{clk} \) is the number of clock cycles needed to detect one symbol. Note that it is common to use pipelining techniques to improve throughput because throughput is sensitive to \( N_{clk} \). Pipelining technique arranges the hardware such that more than one operation can be performed simultaneously. Our design achieves high throughput by applying a fully-pipelined architecture that makes each hardware unit work for a symbol in a clock. Therefore, \( N_{clk} = 1 \) in our design, which makes it possible to improve throughput significantly.
The pipelining technique increases throughput efficiently; however, it requires many hardware units, which can increase overall complexity. Therefore, an efficient algorithm that minimizes complexity should be used. As shown in Fig. 9 and 10, the proposed design can be divided into two stages, i.e., the IW and symbol detecting stages. In the IW stage, the IW filtering process is implemented according to an optimized technique. In the symbol detecting stage, the proposed interference mitigation algorithm is implemented efficiently using the existing FSD and SFSD algorithms.

**B. IW STAGE**

In the IW stage, finding $W$ according to (10) is the key part. In this equation, generally, the inverse matrix function is performed first and Cholesky decomposition is then performed; however, here, the reverse is performed such that the inverse operation is performed for the triangular matrix. Note that the final calculated matrix has the same characteristic used for IW.

The overall functional flow of the IW stage is shown in Fig. 11. First, the value of $H_I^{H}H_I + \sigma^2 I$ is computed for the given input $H_I$, passed through the Cholesky decomposition module, passed through the inversion module, and finally converted to $y$, $H_D$, $H_I$. Finally, $Wy_I$, $WH_I$, and $WH_I$ is generated as outputs.

First, we perform Cholesky decomposition on the given channel $H_I$. In the Cholesky decomposition module, the value of $H_I^{H}H_I + \sigma^2 I$ is given as input, and the Cholesky decomposition is designed to solve the $4 \times 4$ matrix using the divide and conquer method. The Cholesky decomposition module is implemented in 10 pipelining stages. To perform for the $4 \times 4$ matrix, four stages are required because the triangular matrix of the final output is obtained by the divide and conquer method ($4 \times 4, 3 \times 3, 2 \times 2, 1 \times 1$). As shown in Fig. 12, there are three sub-stages in each column stage. The four values in the first column are named $A$, $B$, $C$, and $D$. The first sub-stage passes $A$ through sqrt to obtain $\sqrt{A}$. In the second sub-stage, this value is divided by the remaining values of $B$, $C$, and $D$. This completes the value of the first...
column and updates the value of the remaining $3 \times 3$ matrix in the third sub-stage.

Note that matrix inversion by Cholesky decomposition is simple due to the triangular matrix. The matrix inversion is also designed to enable full pipelining as shown in Fig. 12. In addition, matrix inversion can be performed with a delay in each column stage, which can make the design efficient with lower latency.

C. SYMBOL DETECTING STAGE

The FSD stage can be divided into three key functions, i.e., FSD pre-processing, FSD and LSD. Note that FSD preprocessing is excluded in the hardware implementation in this paper. The FSD and LSD are designed as a fully-pipelined architecture such that each has the same number of sub-stages as the number of tree levels.

A block diagram of the entire FSD stage (consisting of FSD-LSD) is shown in Fig. 9. First, hard-detection is performed in the FSD stage and the LSD stage is initiated to secure soft output. In this process, there is a minimum selector that determines the minimum candidate value based on the FSD output. Finally, there is an LLR calculator based on the SFSD output.

1) IMPLEMENTATION OF FSD UNIT

For a four-stream antenna, the number of full extensions is fixed to one, which is close to the near optimal performance. Figure 14 shows the hardware architecture of the FSD for $4 \times 4$-QAM. Here, each tree level has 64 nodes, and a total of $64 \times 4 = 256$ nodes are implemented for pipelining. Each node consists of an interference cancellation unit (ICU) that updates the matrix based on the detected symbols, and a node selection unit (NSU) that performs symbol slicing of the current stream. The ICU and NSU calculate (17) and (18), sequentially.

\[
\tilde{z}_i = z_i - \sum_{j=i+1}^{M} R_{i,j} \hat{x}_j, \quad (17)
\]
\[
\hat{x}_i = \text{slice}(\frac{\tilde{z}_i}{R_{i,i}}), \quad (18)
\]
In the first stage, which does not have to select a symbol because the first level is FE, only the ICU exists. In addition, in the last stage which does not have to update the matrix, only the NSU exists.

2) IMPLEMENTATION OF LSD UNIT

After passing through the four stages of the FSD, a bit-based tree-extension stage is initiated to generate soft-output for all bits. This tree extension is performed based on the LSD algorithm and generates the soft-output, and the list parameter $L$ is fixed to 2. In the SFSD stage, two of the 64 values from the FSD stage are selected to expand the opposite tree of each bit at each tree-level. The LSD block is designed as shown in Fig. 13. Since 64-QAM signals have six bits, six tree-extensions are performed on one node. Note that three stages are required to expand at the second tree level, and only two stages are required for expansion at the third and fourth tree levels; thus, $6 \times 6 = 36$ additional nodes are required because SFSD proceeds on two values in total. Thus, $36 \times 2 = 72$ nodes are created in total. As shown in Fig. 15, some nodes do not require the NSU because such nodes already have a pre-decided symbol. Note that such nodes in the first tree level also do not require the ISU likewise FSD.

3) IMPLEMENTATION OF IC STAGE

Note that the proposed interference mitigation scheme is implemented by hardware, and the proposed scheme is implemented to minimize additional hardware requirements by adding an IC unit to the FSD and LSD units as shown in Fig. 13. The interference signal is detected through the operation of (14), which can proceed sequentially in the tree-level structure. For calculation, after FSD preprocessing is performed, $H_I H_W Q$ is calculated by the IC preprocessing unit. As shown in Fig. 16, the IC unit is added and performs per FSD sub-stage. Each unit calculates a column of the matrix for the IC, $Q^H W_i - R_{3D,i}$. After four stages of the IC unit, the ZF matrix is completed and then goes through the slicing unit, which detects interference signals. Finally, the detected interference signal is used with the detected desired signal in the ED calculation unit.

Note that the IC unit for the LSD works in the same manner. Therefore, in total, $64 + 36 = 100$ candidates and those recalculated ED can be obtained. These values are used to calculate the LLR.

VI. IMPLEMENTATION RESULT

The proposed soft-output symbol detector design, including the proposed interference mitigation algorithm, was implemented for DL MU-MIMO system. Here, a $8 \times [4, 4]$ antenna configuration was considered, and a Verilog HDL and Synopsys Design Compiler were used for RTL design and synthesis, respectively. The synthesis was performed based on a 65-nm CMOS standard cell library with an operating frequency of 150 MHz, and the gate area was 2633 mm$^2$. Table 4 shows the gate count of each implemented block. As can be seen, the total gate count is 2107 K. The existing FSD and LSD units account for 887 K (42.1 %) of the total gate count, and the other units for interference mitigation account for 1220 K (57.9 %) of the total.

Table 5 compares the overall hardware implementation results of previous studies and the proposed design. [32] shows the implementation results of the SD algorithm with hard-output. [33], [34], and [35] show the implementation results of the K-best algorithm with hard and soft output, respectively. [36] and [37] show the implementation results of the FSD algorithm with hard and soft output, respectively. Proposed shows the implementation results of the proposed algorithm with only the FSD and with both the FSD and IC.

To compare the results fairly, under the same condition, throughput and power consumption were normalized to the 65-nm technology at a supply voltage ($V_{dd}$) of 1.2 V. The normalized throughput is calculated as follows:

$$\text{Normalized Throughput} = \left( \frac{\text{Tech.}}{65 \text{ nm}} \right) \times \text{Throughput}, \quad (19)$$

while the normalized power consumption is calculated as

$$\text{Normalized Power} = \left( \frac{1.2 \text{ V}}{V_{dd}} \right)^2 \times \left( \frac{65 \text{ nm}}{\text{Tech.}} \right) \times \text{Power}. \quad (20)$$

As shown in Table 5, [32] shows a gate count of 212 K and a normalized throughput of 132.9 Mbps at an operating clock rate of 193 MHz. Note that the SD algorithm is not friendly to the hardware architecture; thus the implementation is not efficient and cannot provide high throughput. [33] shows a gate count of 1760 K and a normalized throughput of 100 Mbps at an operating clock of 198 MHz. Note that this is the first implementation of the K-best algorithm with a sorter in each stage. Therefore, the implementation is not efficient, which results in a high gate count and low throughput. [34] attempts to optimize the existing K-best algorithm in terms of both throughput and gate count, and it shows a gate count of 298 K and a normalized throughput of 2 Gbps; however, it has a very high operating clock rate of 833 MHz, which is too high for use in a practical implementation. Recently, [35] was presented to achieve high-speed performance as normalized 3.28 Gbps at an operating clock rate of 137 MHz with a gate count 1753 K.

As mentioned previously, the FSD algorithm is proposed as a hardware-friendly implementation. [36] shows much better performance with a gate count of 88.2 K and a normalized throughput of 1.98 Gbps at an operating clock rate of 165 MHz. Here, a potential disadvantage of the FSD
algorithm is how to obtain sufficient candidates for soft-output generation because it typically uses a much lower number of candidates compared to K-best algorithm. [37] shows the soft-output FSD with a gate count 555 K and a normalized throughput of 3.05 Gbps at an operating clock rate of 370 MHz.

Since there have been no previous implementations of the interference mitigation algorithm for an MIMO detector, the proposed algorithm was first implemented using only the FSD stage for comparison with existing MIMO detectors. In addition, it was implemented with the IC stage to evaluate the cost of extending the DL MU-MIMO system.

First, when comparing the results of [37] and the FSD component of this study, the FSD component shows a gate count of 887 K and a normalized throughput of 3.6 Gbps at an operating clock rate of 150 MHz. Recall that this study employs a fully-pipelined architecture; therefore, the throughput is much greater at the same operating clock rate although the gate count is greater than that of [37]. Similarly, power consumption can be lower due to the lower operating clock rate. In other words, the proposed implementation is comparable to the existing FSD implementation.

Next, the FSD+IC component is compared to the FSD component to determine how it is changed by extending the MIMO detector to a DL MU-MIMO detector. As shown in Table 3, the proposed FSD with IC requires approximately double the computational complexity of the existing FSD. This result can be observed in Table 4. If we consider FSD preprocessing, the total gate count is closer to the double value because the implementation includes preprocessing for only the IC because FSD preprocessing is usually excluded due to its variable implementation method. As shown in Table 5, with the interference mitigation algorithm, proposed algorithm can achieve a normalized throughput of 3.6 Gbps with a gate count of 2107 K, which is significantly better performance than existing MIMO detectors under DL MU-MIMO systems and when only IW is included.

**VII. CONCLUSION**

In this paper, we have proposed a design and implementation of an FSD combined with an interference mitigation algorithm for a DL MU-MIMO system. The simulation results show 0.5 dB near-optimal performance which is 4 dB better than the existing IW method for soft-output decisions. Both hard and soft output FSDs are implemented with the proposed interference mitigation algorithm for 8 × [4, 4] DL MU-MIMO systems. The proposed design can support 3.6 Gbps at 150 MHz using a pipelined implementation. In a comparison with the previous MIMO detectors, the proposed design represents a promising architecture for DL MU-MIMO systems in terms of significant performance improvement in inter-user interference situations and has a compatible hardware implementation that maintains high throughput at an acceptable gate count.

**REFERENCES**

[1] J. H. Winters, “On the capacity of radio communication systems with diversity in a Rayleigh fading environment,” IEEE J. Sel. Areas Commun., vol. SAC-5, no. 5, pp. 871–878, Jun. 1987.

[2] E. Björnson, J. Hoydis, and L. Sanguinetti, “Massive MIMO has unlimited capacity,” IEEE Trans. Wireless Commun., vol. 17, no. 1, pp. 574–590, Jan. 2018.

[3] A. J. Paulraj, D. A. Gore, R. U. Nabar, and H. Bolcskei, “An overview of MIMO communications—A key to gigabit wireless,” Proc. IEEE, vol. 92, no. 2, pp. 198–218, Feb. 2004.

[4] S. Yang and L. Hanzo, “Fifty years MIMO detection: The road to large-scale MIMOS,” IEEE Commun. Surveys Tuts., vol. 17, no. 4, pp. 1941–1988, 4th Quart., 2015.

[5] IEEE Standard for Information Technology Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Amendment 4: Enhancements for Very High Throughput for Operation in Bands Below 6 GHz, IEEE Standard 802.11acTM-2013, Dec. 2013.

[6] F. Chari, L. Chaari, and L. Kamoun, “PHY/MAC enhancements and QoS mechanisms for very high throughput WLANs: A survey,” IEEE Commun. Surveys Tuts., vol. 15, no. 4, pp. 1714–1735, 4th Quart., 2013.

[7] O. Bejarano, E. Knightly, and M. Park, “IEEE 802.11ac: From channelization to multi-user MIMO,” IEEE Commun. Mag., vol. 51, no. 10, pp. 84–90, Oct. 2013.

[8] D. Gesbert, M. Kountouris, R. W. Heath, Jr., C.-B. Chae, and T. S. Rappaport, “Shifting the MIMO paradigm,” IEEE Signal Process. Mag., vol. 24, no. 5, pp. 36–46, Sep. 2007.

[9] L. Liu, R. Chen, S. Geirhofer, K. Sayana, Z. Shi, and Y. Zhou, “Downlink MIMO in LTE-advanced: SU-MIMO vs MIMO-MIMO,” IEEE Commun. Mag., vol. 50, no. 2, pp. 140–147, Feb. 2012.

[10] L. U. Choi and R. D. Murch, “A transmit preprocessing technique for multiuser MIMO systems using a decomposition approach,” IEEE Trans. Wireless Commun., vol. 3, no. 1, pp. 20–24, Jan. 2004.

[11] I. Sohn, C. Park, and K. B. Lee, “Downlink multi-user MIMO systems with adaptive feedback rate,” IEEE Trans. Veh. Technol., vol. 61, no. 3, pp. 1445–1451, Mar. 2012.

[12] H. Shen, W. Xu, S. Jin, and C. Zhao, “Joint transmit and receive beamforming for multiuser MIMO downlinks with channel uncertainty,” IEEE Trans. Veh. Technol., vol. 63, no. 5, pp. 2319–2335, Jun. 2014.
[13] A. A. Esswie, M. El-Absi, O. A. Dobre, S. Ikki, and T. Kaiser, “A novel FDD massive MIMO system based on downlink spatial channel estimation without CSIT,” in Proc. IEEE ICC, May 2017, pp. 1–6.

[14] Q. H. Spencer, A. L. Swindlehurst, and M. Haardt, “Zero-forcing methods for downlink spatial multiplexing in multiuser MIMO channels,” IEEE Trans. Signal Process., vol. 52, no. 2, pp. 461–471, Feb. 2004.

[15] T. Ketseoglou and E. Ayanoglu, “Downlink precoding for massive MIMO systems exploiting virtual channel model sparsity,” IEEE Trans. Commun., vol. 66, no. 5, pp. 1925–1939, May 2018.

[16] C. B. Ribeiro, K. Hugl, M. Lampinen, and M. Kuusela, “Performance of linear multi-user MIMO precoding in LTE system,” in Proc. 3rd Int. Symp. Wireless Persuasive Comput., May 2008, pp. 410–414.

[17] H. Wang, X. Xu, M. Zhao, W. Wu, and Y. Yao, “Robust transmission for multiuser MIMO downlink systems with imperfect CSIT,” in Proc. IEEE Wireless Commun. Netw. Conf., Apr. 2008, pp. 340–344.

[18] A. K. Hassan, M. Moinuddin, U. M. Al-Saggaf, and T. Y. Al-Naffouri, “Performance analysis of beamforming in MU-MIMO systems for Rayleigh fading channels,” IEEE Access, vol. 5, pp. 3709–3720, 2017.

[19] J. Lee, H. Kwon, and I. Kang, “Interference mitigation in MIMO interference channel via successive single-user soft decoding,” in Proc. Inf. Theory Appl. Workshop, Feb. 2012, pp. 180–185.

[20] J. H. Winters, “Optimum combining in digital mobile radio with cochannel interference,” IEEE J. Sel. Areas Commun., vol. SAC-2, no. 4, pp. 528–539, Jul. 1984.

[21] J. Lee, D. Toupakaris, and W. Yu, “Interference mitigation via joint detection,” IEEE J. Sel. Areas Commun., vol. 29, no. 6, pp. 1172–1184, Jun. 2011.

[22] J. Karlsson and J. Heinegard, “Interference rejection combining for GSM,” in Proc. 5th Int. Conf. Universal Pers. Commun. (ICUPC), 1996, pp. 433–437.

[23] Y. Ohwatari, N. Mikii, T. Asai, T. Abe, and H. Taoka, “Performance of interference rejection combining receiver to suppress inter-cell interference in LTE-advanced downlink,” IEICE Trans. Commun., vol. E94-B, no. 12, pp. 3362–3369, Jun. 2011.

[24] L. Zheng, K. Fukawa, H. Suzuki, and S. Suyama, “Near-optimal signal detection based on the MMSE detection using multi-dimensional search for correlated MIMO channels,” IEICE Trans. Commun., vol. E94-B, no. 8, pp. 2346–2356, May 2011.

[25] S. Iwelski, Z. Bai, G. H. Bruck, P. Jung, B. Badic, T. Scholand, and R. Balraj, “Analysis of interference-aware receivers in heterogeneous LTE networks,” in Proc. IEEE/CIC Int. Conf. Commun. China (ICCC), Aug. 2013, pp. 314–320.

[26] F. Chen, X. Li, S. Duan, L. Wang, and J. Wu, “Diffusion generalized maximum correntropy criterion algorithm for distributed estimation over multitask network,” Digit. Signal Process., vol. 81, pp. 16–25, Oct. 2018.

[27] M. Itagaki, T. Yamamoto, K. Takeda, and F. Adachi, “Uplink capacity of OFDM multi-user MIMO using near-ML detection in a cellular system,” IEICE Trans. Commun., vols. E95–B, no. 1, pp. 198–205, Jan. 2012.

[28] M. Kim, J. Park, H. Kim, and J. Kim, “Efficient near-optimal detection with generalized sphere decoder for blind MU-MIMO systems,” ETRI J., vol. 36, no. 4, pp. 682–685, Aug. 2014.

[29] J. Park, M. Kim, H. Kim, Y. Jung, and J. Kim, “Low-complexity MIMO detection algorithm with adaptive interference mitigation in DL MU-MIMO systems with quantization error,” J. Commun. Netw., vol. 18, no. 2, pp. 210–217, Apr. 2016.

[30] M. Kim and J. Kim, “Adaptive interference-aware receiver for multi-user MIMO downlink in IEEE 802.11ac,” in Proc. IEEE Int. Conf. Consum. Electron. (ICCE), Jan. 2015, pp. 645–646.

[31] M. Kim, Y. Jung, and J. Kim, “Adaptive interference-aware receiver for multiuser MIMO downlink transmission in IEEE 802.11ac wireless LAN systems,” IEEE Trans. Commun., vols. E99–B, no. 8, pp. 1884–1893, 2016.

[32] F. Borlenghi, E. M. Witte, G. Ascheid, H. Meyr, and A. Burg, “A 772 Mbit/s 8.81 bit/nJ 90 nm CMOS soft-input soft-output sphere decoder,” in Proc. IEEE Asian Solid-State Circuits Conf., Nov. 2011, pp. 297–300.

[33] S. Mondal, A. Eltawil, C.-A. Shen, and K. N. Salama, “Design and implementation of a soft-free K-Best sphere decoder,” IEEE Trans. Very Large Scale Integ. (VLSI) Syst., vol. 18, no. 10, pp. 1497–1501, Oct. 2010.

[34] D. Patel, V. Smolyakov, M. Shabany, and P. G. Gulak, “VLSI implementation of a WiMAX/LTE compliant low-complexity high-throughput soft-output K-best MIMO detector,” in Proc. IEEE Int. Symp. Circuits Syst., May 2010, pp. 593–596.

[35] I. A. Bello, B. Halak, M. El-Hajjar, and M. Zwolinski, “VLSI implementation of a fully-pipelined K-best MIMO detector with successive interference cancellation,” Circuits, Syst., Signal Process., vol. 38, no. 10, pp. 4739–4761, Mar. 2019.

[36] L. Liu, J. Loefgren, and P. Nilsson, “Area-efficient configurable high-throughput signal detector supporting multiple MIMO modes,” IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 59, no. 9, pp. 2085–2096, Sep. 2012.

[37] X. Chen, G. He, and J. Ma, “VLSI implementation of a high-throughput iterative fixed-complexity sphere decoder,” IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 60, no. 5, pp. 272–276, May 2013.

[38] L. G. Barbero and J. S. Thompson, “Fixing the complexity of the sphere decoder for MIMO detection,” IEEE Trans. Wireless Commun., vol. 7, no. 6, pp. 2131–2142, Jun. 2008.

[39] L. G. Barbero and J. S. Thompson, “Extending a fixed-complexity sphere decoder to obtain likelihood information for turbo-MIMO systems,” IEEE Trans. Veh. Technol., vol. 57, no. 5, pp. 2804–2814, Sep. 2008.

MINJOON KIM received the B.S. and Ph.D. degrees in electrical and electronic engineering from Yonsei University, Seoul, South Korea, in 2012 and 2018, respectively. He is currently a Senior Researcher with the Korea Electronics Technology Institute, South Korea. His research interests include digital signal processing algorithm and SoC/VLSI implementation for wireless communication systems.

* * *