A Low Complexity K-Best MIMO Detector with Adaptive Self-Adjusting Mechanism

Muh-Tian Shiue, Syu-Siang Long, Chin-Kuo Jao, and Kai-Chung Cheng

Abstract— In this paper, we propose a low computational complexity MIMO detector with K-best algorithm that can select K value adaptively. It combines distributed K-Best (DKB) and successive interference cancellation (SIC) algorithms to reduce visiting nodes. The proposed K-value adjusting mechanism is proposed to decide the number of possible candidates at each searching stage. Instead of calculating the SNR level, the proposed adaptively adjusting algorithm compares the similarity of minimum PED and next minimum PED to decide K-value. It can achieve much lower computational complexity than conventional K-best architecture without descending the BER performance. Applying with recursive pipelined circuit architecture, the proposed MIMO detector supports various antenna configurations and data modulations from 2x2 to 8x8 and 64-QAM to QPSK, respectively. The proposed circuit is fabricated in 90nm CMOS process with core area of 0.546mm². The maximum throughput is 126Mbps at operating clock of 127MHz, and the power consumption is 20.6mW under 1V power supply.

Index Term— Adaptive self-adjusting, Distributed K-Best (DKB), Multiple-input multiple-output (MIMO), Successive Interference Cancellation (SIC)

I. INTRODUCTION

In the recent years, multiple-input multiple-output (MIMO) technique has been considered as common specification in modern wireless communication standards to achieve higher bandwidth efficiency [1]-[3]. In MIMO detectors, maximum-likelihood (ML) algorithm is well-known as the optimum solution to obtain reference BER performance. However, ML detector has exponentially increased computing complexity with constellation size and number of antenna. Direct implementation of ML detector is not practical and has unreasonable hardware cost. Therefore, some simplifications from ML searching are proposed to reduce computational complexity. With the aid of QR decomposition on the channel matrix, the ML searching can be transformed into the levelled tree searching. Two kinds of tree searching algorithms called depth-first and breadth-first are proposed in the literatures. The sphere decoding algorithm (SDA) [4],[5] is a representative one of depth-first algorithms. The SDA detector can obtain exact ML solution and reducing computational complexity by pruning unpromising tree branches. However, the SDA decoder has irregular throughput rate because the amount of searching path varies with the channel signal-to-noise ratio (SNR). On the other side, K-best algorithm (KBA) is well-known as a typical type of breadth-first algorithm. The KBA searches all the possible branches at the same level and passes K branches which have minimum partial Euclidean distances (PED) to the next level. The KB detector has constant visiting nodes and top-down searching direction that can be easily implemented in pipelined circuit architecture [6]. However, the K-value has to be large enough to cover the possible branches especially when the SNR is low. Based on this condition, continuously increasing K value to keep BER performance will lead to massive computational complexity.

In the literatures, K-value in KBA is fixed through the decoding process to obtain a constant throughput rate. However, there will be redundant K-values causing unnecessary computing complexity when channel SNR is high. Ref. [7] proposes an algorithm to adjust K-values by calculating SNR. The accuracy of calculated SNR value can affect BER performance and the overhead of SNR calculating circuit is also expensive. In this paper, we propose an adaptively K-values adjusting algorithm which does not need to estimate the exact SNR value. In section II, the MIMO system model and distributed K-best (DKB) algorithm are going to be reviewed. The proposed algorithm and BER performance will be depicted and compared in section III. The circuit architecture and functional blocks are described in section IV. Finally, chip implementation results and comparison with other works will be shown in section V.

The authors would like to thank Chip Implementation Center of National Applied Research Laboratories (CIC/NARL) for providing chip implementation and measurement service. This work was supported in part by Ministry of Science and Technology, Taiwan, under Grant 103-2220-E-008-004.

M.-T. Shiue, S.-S. Long, C.-K. Jao, and K.-C. Cheng are with the Department of Electrical Engineering, National Central University, Chungli, 32001, Taiwan. (e-mail: mtxiue@ee.ncu.edu.tw, 945401026@cc.ncu.edu.tw)

Fig. 1. Channel matrix model of a typical MIMO system
II. SPATIAL MULTIPLEXING OF MIMO SYSTEM

A. MIMO System Model

Fig. 1 shows the antenna configuration of a typical MIMO system. Let us consider a spatial multiplexing MIMO system with \(N_t\) transmitter antennas and \(N_r\) receiver antennas. The complex baseband equivalent model can be rewritten as:

\[
\hat{\mathbf{y}} = \mathbf{H}\hat{\mathbf{s}} + \mathbf{n},
\]

where \(\hat{\mathbf{s}} = [s_1, s_2, \ldots, s_{N_t}]^T\) is the transmitted symbol vector, in which each element is independently chosen from a complex \(M\)-QAM constellation set \(C\); \(\hat{\mathbf{y}} = [y_1, y_2, \ldots, y_{N_r}]^T\) is the received symbol vector; \(\mathbf{H}\) denotes the \(N_r \times N_t\) channel matrix, whose entries are identically independent distributed (i.i.d) complex zero-mean Gaussian random variables; and \(\mathbf{n} = [n_1, n_2, \ldots, n_{N_r}]^T\) is a white complex zero-mean Gaussian noise vector with variance \(\sigma^2\). The optimum solution for (1) is maximum-likelihood (ML) decoding, which finds out the most likely input vector \(\hat{\mathbf{s}}\) according to minimum-distance criterion:

\[
\hat{\mathbf{s}} = \arg \min_{\mathbf{s} \in \Omega} \|\hat{\mathbf{y}} - \mathbf{Hs}\|^2.
\]

The exhaustive-search ML decoder should compute \(M^{N_t}\) possible input vectors. Its complexity grows exponentially with the number of antenna and constellation size. Thus, such ML decoder is not feasible for hardware implementation.

Then, the complex model in (1) can be expended to the real-valued equivalent model: \(\mathbf{y} = \mathbf{hs} + \mathbf{n}\), i.e.,

\[
\begin{bmatrix}
\Re\{\mathbf{y}\} \\
\Im\{\mathbf{y}\}
\end{bmatrix} =
\begin{bmatrix}
\Re\{\mathbf{h}\} & -\Im\{\mathbf{h}\} \\
\Im\{\mathbf{h}\} & \Re\{\mathbf{h}\}
\end{bmatrix}
\begin{bmatrix}
\Re\{\mathbf{s}\} \\
\Im\{\mathbf{s}\}
\end{bmatrix} +
\begin{bmatrix}
\Re\{\mathbf{n}\} \\
\Im\{\mathbf{n}\}
\end{bmatrix},
\]

where \(\Re\{\cdot\}\) and \(\Im\{\cdot\}\) denote the real and imaginary operation, respectively. In the literature, both complex and real models are widely used to design a spatial multiplexing MIMO detector. Our design chooses the real-valued model since the one-dimensional signal space \(\Omega\) is more suitable for a modulation configurable design.

In order to simplify the exhaustive-search problem by a tree-search structure, the channel matrix is decomposed by QR decomposition, i.e., \(\mathbf{H} = \mathbf{QR}\), where \(\mathbf{Q}\) is a \(2N_r \times 2N_r\) unitary matrix and \(\mathbf{R}\) is an upper triangular matrix of size \(2N_t \times 2N_t\). Applying \(\mathbf{Q}^H\) to \(\mathbf{y}\) yields \(\hat{\mathbf{y}} = \mathbf{Q}^H\mathbf{y} = \mathbf{R}\mathbf{s} + \mathbf{w}\), where \(\mathbf{w} = \mathbf{Q}^H\mathbf{n}\). Then the optimum solution of real model \(\mathbf{s}\) can be rewritten as follows:

\[
\hat{\mathbf{s}} = \arg \min_{\mathbf{s} \in \Omega} \|\hat{\mathbf{y}} - \mathbf{Rs}\|^2,
\]

where \(\Omega = \{-\sqrt{M} + 1, \ldots, -1, 1, \ldots, \sqrt{M} - 1\}\) is the set of real entries in the \(M\)-QAM constellation. Expanding the vector norm in (4) yields:

\[
T_i(s^{(i)}) = T_{i+1}(s^{(i+1)}) + |e_i(s^{(i)})|^2
\]

Owing to the upper triangular matrix \(\mathbf{R}\), finding the optimum solution in (5) is a tree-search process with \(2N_t\) layers. Beginning from layer \(i = 2N_t\), (5) can be accomplished in a recursive form as follows:

\[
\hat{\mathbf{s}} = \arg \min_{\mathbf{s} \in \Omega^{2N_t}} \|\hat{\mathbf{y}} - \mathbf{R}\mathbf{s}\|^2.
\]

B. Distributed K-best Algorithm [6], [8]

In the KBA decoding process, the tree expansion and PED sorting are the most computationally complicated circuits [9]. Since each parent node expands to \(\sqrt{M}\) children, there are \(K\sqrt{M}\) PEDs needed to be calculated at each layer. The total number of visited nodes for real-modelled KBA is \(\sqrt{M} + (2N_t - 1)K\sqrt{M}\). As \(M\) becomes large, e.g., 64-QAM, the hardware design of conventional KBA is costly. Similarly, sorting a list of \(K\sqrt{M}\) PEDs requires a complexity of \(O(K^2M)\) in worst case [10], which is also proportional to \(M\).

In order to reduce the visited nodes at each layer, our design adopts DKB algorithm which can find the best \(K\) candidates by visiting only \(2K-1\) nodes at each layer. An example of real-modelled DKB at \(i\)-th layer for 16-QAM and \(K=4\) is depicted in Fig. 2. At \(i\)-th layer, the DKB divides the children nodes into \(K\) groups and each group corresponds to a unique parent node. Then we can find the first child (FC) with the local minimum PED of each group. Since one of these \(K\) FCs that has smallest PED (global minimum) must be the K-best candidate, this FC node is selected to the K-best list. Subsequently, the selected FC group immediately enumerates the sibling node with the next smallest PED to replace the selected FC. After iterating this procedure for \(K\) times, the total K-best candidates of \(i\)-th layer can be found. The detailed DKB decoding flow is described as follows:
1) Find the best \( K \) candidates at the first layer \((i = 2N_t)\).

2) Find the first child (local minimum PED) of each parent node at layer \( i+1 \). These \( K \) first children are denoted as contender set \( L_i \).

3) For \( k=1:K \)
   a) Choose the child in set \( L_i \) with the smallest PED.
   b) Announce this child as the \( K \)-best candidate at layer \( i \) and add it to the \( K \)-best list set \( K_i \).
   c) Replace this child in \( L_i \) with its next best sibling node.

The conventional \( K \)-best scheme can concurrently expand and sort the children nodes to find the candidates. However, the best \( K \) candidates of DKB scheme are one by one sequentially listed in \( K_i \) within \( K \) clock cycles. Owning to this feature, the DKB can avoid the complicated sorting circuit and only requires a minimum finder circuit to select the child in \( L_i \) which has minimum PED. It should be noted that the number of visited nodes at each layer is \( 2K-1 \), which is independent of the constellation size \( M \). Therefore, the DKB is more suitable to expand to a high-order constellation modulation. The total number of visited nodes for DKB is \( \sqrt{M} + (2N_t - 1)(2K - 1) \). Compared to the conventional KBA, the DKB scheme is an efficient procedure to choose the best \( K \) candidates.

III. PROPOSED ADAPTIVE K-ADJUSTING ALGORITHM

The conventional adaptively \( K \)-value adjusting scheme depends on SNR values provided by channel estimation. Its BER performance will degrade if the estimated SNR value is not accurate and thus choose improper \( K \)-value. Our proposed algorithm not only adaptively selects \( K \)-value without calculating SNR value, but also adopts an adaptive SIC scheme to further reducing redundant visiting nodes. The detailed decoding processes are shown in Fig. 3 and Fig.4 and described as follows:

A. Adaptive Successive Interference Cancellation (SIC) Scheme

When performing DKB algorithm, all \( K \) child nodes of parent nodes are visited and select \( K \)-best candidates to the next stage even at good SNR conditions. Therefore, we propose
adaptive SIC scheme in our algorithm to reduce unnecessary visiting nodes when channel noise level is low. In other words, the PED difference between the best FC and other FCs are distinct. Additional K candidates searching can be avoided to save power consumption. The timing when the proposed MIMO detector performing SIC scheme instead of DKB scheme depends on the similarity of K FCs that are found in step 1 in section III. If there are over half of K FCs having the same constellation with best FC, then the proposed MIMO detector will performing SIC scheme at this layer. Therefore, only K FCs are visited and considered as K candidates and directly pass to the next layer.

B. Adaptive K-value adjusting

The conventional K-value adjusting mechanism which is by SNR calculating cannot quickly response to the change of channel state during decoding procedure. According to observing the statistical characteristics of PED, we propose an approach which can adjust K-value in each decoding layer. Fig. 5 shows the cumulative distribution function (CDF) of PED difference between first child and next child nodes in various SNR channel. It is observed that high SNR has more distinguished PED difference than low SNR. The PED difference $\gamma = \text{PED}_{1 - \text{min}} - \text{PED}_{2 - \text{min}}$ in low SNR tends to centralize at low value while high SNR centralize at higher value. Based on this statistical characteristic, we can estimate SNR condition by simple subtraction and set up threshold values to decide which K-value we should apply. To simplify the circuit implementation, we apply three segments of adjustable K-values to accomplish adaptive turning as follows:

$$K = \begin{cases} 
10, & \gamma \leq \text{LTV} \\
5, & \text{LTV} \leq \gamma \leq \text{HTV} \\
2, & \text{HTV} \leq \gamma 
\end{cases}$$

(7)

where LTV and HTV are low and high threshold values, respectively. According to simulation results, we determine LTV=0.75 and HTV=3.

Fig. 6 shows the BER performance comparison between the proposed algorithm and conventional DKB scheme under 8x8, 64-QAM configuration. It is clear that either in low or high SNR, the BER performance loss of the proposed algorithm and conventional DKB can almost be ignored. Fig. 7 shows the computing complexity comparison of the proposed algorithm and the conventional DKB at different SNR. In high SNR condition, thanks to adaptively K-value adjusting and SIC scheme, there are much less visiting nodes need to be calculated. i.e., more multiplications $\sum_{j=1}^{2^N} R_{ij} S_j$ in (5) can be saved. The proposed algorithm can improve average 67% and 65% computational complexity of additions and multiplications, respectively.

IV. MIMO DETECTOR CIRCUIT ARCHITECTURE

Since K-values is fixed in all layers in the conventional DKB scheme, it is straightforward to implement in pipelined fashion. To accomplish adaptive K-values adjusting and flexible antenna configurations at the same time, we apply paralleled pipelined folding architecture in our proposed design.

A. Folding Circuit Architecture

Fig. 8 shows the pipelined recursive circuit architecture of the proposed MIMO detector. It contains first child unit (FCU), next child unit (NCU) and two banks of registers. This circuit architecture can flexibly support multi-antenna configurations.
and multiple data modulations. The received data vector $\hat{Y}$ and diagonal channel matrix $R$ are stored into the registers in the beginning and start to decode. In the first stage of tree searching, only FCU is needed to find initial $K$ FCs and goes into the next stage. Then, the NCU will take over the $K$ candidates stored in the register bank to find NCs and determine whether trigger SIC scheme or decide $K$-value. Meanwhile, FCU can continue to find FCs from previous stage. Therefore, in one stage cycle, FCU and NCU can calculate individual data at the same time to keep high hardware utility. Taking 8x8 antenna configuration for example, it needs recursive 16 stage cycles to complete total decoding procedure.

B. First Child Unit

Fig. 9 shows detailed circuit diagram of FCU. There are shift multipliers (SMs) at the input to calculate centers and PEDs. These SMs are composed of multiplexers, bits-shifters and adders to replace a power consumed real multiplier in our design. Since our MIMO system model is real-numbered, one dimensional SE enumeration is applied to be compatible with QPSK, 16-QAM and 64-QAM data modulations. After all $K$ FCs are found, these $K$-FCs and PED values will be stored into register and wait NCU to enumerate next stage $K$ candidates.

C. Next Child Unit

In the NCU circuit design, it needs not only to find $K$ candidates, but also to decide activating SIC or determine $K$-value of DKB scheme. Fig. 10 shows the block diagram of the proposed NCU circuit. When NCU circuit start to work, it receives $K$ FCs from FCU in the first place and compares the similarity between the FC which has minimum PED with other FCs to determine to enable SIC scheme or not. If SIC scheme is activated, the NCU circuit will directly consider these $K$ FCs as $K$ candidates and goes into low power mode by enabling clock gating. If $K$-value adjusting DKB is preferred, NCU will calculate PED difference of minimum PED and sub-minimum PED to choose proper $K$ value.

V. DESIGN SUMMARY AND CHIP COMPARISON

The proposed design has fabricated in 90nm CMOS process, chip photo and the measured results are showed in Table I. The maximum operating clock achieves 127MHz, and the throughput is 126Mbps at this clock. The proposed circuit applies clock gating technique to reduce static power loss and the core power consumption is 20.6mW. Fig. 11 shows the power saving ratios by applying clock gating technique. It can gain about 7% power consumption comparing with non-clock gating circuit. Table II lists the performance comparison of this chip and previous works. In order to have a fair comparison and eliminate the technology process factor, the normalized power is formulated as [13].
\[
\text{Normalized power} = \text{power} \times \left(\frac{1.0}{V_{dd}}\right)^2 \times \left(\frac{0.09}{\text{Technology}}\right) \times \left(\frac{1}{\text{Throughput}}\right)
\] (7)

Table I
Chip summary of the proposed MIMO detector

<table>
<thead>
<tr>
<th>Spec.</th>
<th>Chip measurement results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Antenna</td>
<td>8x8, 6x6, 4x4, 4x4, 2x2-3x2</td>
</tr>
<tr>
<td>Modulation</td>
<td>64-QAM, 32-QAM, 16-QAM</td>
</tr>
<tr>
<td>K-value</td>
<td>Adaptive (Maximum 10)</td>
</tr>
<tr>
<td>Throughput</td>
<td>120 Mbps</td>
</tr>
<tr>
<td>Clock freq.</td>
<td>127 MHz</td>
</tr>
<tr>
<td>Technology</td>
<td>0.18um CMOS Process</td>
</tr>
<tr>
<td>Supply voltage</td>
<td>1.0V</td>
</tr>
<tr>
<td>Power</td>
<td>20 mW</td>
</tr>
<tr>
<td>Chip area</td>
<td>0.75 x 0.65 mm</td>
</tr>
<tr>
<td>Gate count</td>
<td>115,000</td>
</tr>
<tr>
<td>Package</td>
<td>11 x 11D8</td>
</tr>
</tbody>
</table>

Table II
CHIP COMPARISON

<table>
<thead>
<tr>
<th>Reference</th>
<th>[10]</th>
<th>[11]</th>
<th>[12]</th>
<th>[13]</th>
<th>[14]</th>
<th>[15]</th>
<th>[16]</th>
<th>This work</th>
</tr>
</thead>
<tbody>
<tr>
<td>Antenna</td>
<td>4x4</td>
<td>4x4</td>
<td>4x4</td>
<td>2x2-3x2</td>
<td>2x2-3x2</td>
<td>4x4</td>
<td>4x4</td>
<td>2x2-3x2</td>
</tr>
<tr>
<td>Modulation on (QAM)</td>
<td>16</td>
<td>64</td>
<td>64</td>
<td>QPSK</td>
<td>64-QAM</td>
<td>16</td>
<td>QPSK</td>
<td>64-QAM</td>
</tr>
<tr>
<td>Algorithm</td>
<td>KB</td>
<td>KB</td>
<td>KB</td>
<td>Best-first</td>
<td>KB</td>
<td>KB</td>
<td>Depth-first</td>
<td>KB</td>
</tr>
<tr>
<td>K-value</td>
<td>10</td>
<td>10</td>
<td>5</td>
<td>5</td>
<td>1-16</td>
<td>Adaptive (Max:10)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tech. (um)</td>
<td>0.25</td>
<td>0.13</td>
<td>0.09</td>
<td>0.13</td>
<td>0.35</td>
<td>0.18</td>
<td>0.25</td>
<td>0.19</td>
</tr>
<tr>
<td>Area (mm²)</td>
<td>N/A</td>
<td>N/A</td>
<td>0.6749</td>
<td>1.77</td>
<td>5.76</td>
<td>N/A</td>
<td>N/A</td>
<td>0.54612</td>
</tr>
<tr>
<td>Gate count (Kg)</td>
<td>110</td>
<td>114</td>
<td>294</td>
<td>350</td>
<td>91</td>
<td>300</td>
<td>117</td>
<td>193</td>
</tr>
<tr>
<td>Soft/Hard decision</td>
<td>Hard</td>
<td>Hard</td>
<td>Hard</td>
<td>Soft</td>
<td>Hard</td>
<td>Hard</td>
<td>Hard</td>
<td>Hard</td>
</tr>
<tr>
<td>Mode/Re-al/Comp (x)</td>
<td>Real</td>
<td>Real</td>
<td>C</td>
<td>C</td>
<td>Real</td>
<td>C</td>
<td>Real</td>
<td>C</td>
</tr>
<tr>
<td>Clock freq. (MHz)</td>
<td>52</td>
<td>282</td>
<td>166</td>
<td>198</td>
<td>100</td>
<td>47</td>
<td>51</td>
<td>125</td>
</tr>
<tr>
<td>Throughput (MHz)</td>
<td>83</td>
<td>655</td>
<td>95</td>
<td>429</td>
<td>53.5</td>
<td>12-56</td>
<td>73.5</td>
<td>124</td>
</tr>
<tr>
<td>Power (mW)</td>
<td>N/A</td>
<td>135</td>
<td>N/A</td>
<td>74.8</td>
<td>626</td>
<td>N/A</td>
<td>360</td>
<td>22</td>
</tr>
<tr>
<td>Normalized power</td>
<td>N/A</td>
<td>0.099</td>
<td>N/A</td>
<td>0.084</td>
<td>0.236</td>
<td>N/A</td>
<td>0.282</td>
<td>0.177</td>
</tr>
</tbody>
</table>

VI. CONCLUSION

In this paper, we propose a MIMO detector combining two kinds of adaptively self-controlled mechanisms: adaptive SIC and K-value adjusting to reduce computational complexity without degrading BER performance. It also supports wide antenna configurations from 2x2 to 8x8 and data modulation from 64-QAM to QPSK. By the proposed algorithm, SNR calculation is unnecessary and channel noise level is recognized by PED difference. Total visiting nodes are greatly reduced and improves 65% computational complexity at high SNR; and an average improves 30% comparing to conventional DBK scheme in all conditions. Circuit architecture is implemented by pipelined folding architecture and clock gating technique contributes 7% power saving. By timing optimized circuit, the folding circuit can recursively processing continuous input data. The high circuit utilization is obtained under all antenna configurations.

Fig. 11. Power consumption of the proposed chip and the conventional DBK detector

REFERENCES
Muh-Tian Shiue was born in Taichung, Taiwan, in 1963. He received the B.S. degree from National Taiwan Normal University, Taipei, Taiwan, in 1986, the M.S. degree from National Chiao-Tung University, Hsinchu, Taiwan, in 1988, and the Ph.D. degree from National Central University, Chungli, Taiwan, in 1998.

From 1988 to 1996, he was with the Transmission Technology Laboratory of Telecommunication Laboratories, Ministry of Transportation Communications, Taiwan, where he was involved in digital subscriber loop technologies. From 1998 to 2000, he was with the Computer and Communications Research Laboratories, Industrial Technology Research Institute, Hsinchu, where was involved in the development of the Asymmetric Digital Subscriber Line (ADSL) chip set. In 2000, he joined IC Plus Corporation, where he was involved in the digital baseband chip design for the ADSL transceiver. From 2001 to 2003, he was the Chief Technology Officer of Trendchip Technologies Corporation. He is currently with the Department of Electrical Engineering, National Central University. His research interests include local access technologies, digital subscriber loop technologies, wireless communication systems, biomedical engineering, digital signal processing, very large scale integration techniques, and control systems.

Syu-Siang Long was born in Taoyuan, Taiwan, in 1981. He received the B.S. degree from National Chi-Nan University, Nantou, Taiwan, in 2003; the M.S. degree from National Central University, Chungli, Taiwan, in 2005, both in electrical engineering.

He is now working toward the Ph.D. degree. His research interests include wireless communication systems, digital signal processing, and very large scale integration techniques.

Chin-Kuo Jao was born in Mioli, Taiwan, in 1979. He received the B.S. degree from Chung-Hwa University, Hsinchu, Taiwan, in 2001; the M.S. degree from National Chi-Nan University, Nantou, Taiwan, in 2003; and the Ph.D. degree from National Central University, Chungli, Taiwan, in 2011, all in electrical engineering.

He is now with Industrial Technology Research Institute, Hsinchu, Taiwan. His research interests include wireless communication systems, digital signal processing, and very large scale integration techniques.

Kai-Chung Cheng was born in Banchao, Taiwan, in 1987. He received the B.S. degree from National Chi-Nan University, Nantou, Taiwan, in 2009; the M.S. degree from National Central University, Chungli, Taiwan, in 2011, both in electrical engineering.

His research interests include wireless communication systems, digital signal processing, and very large scale integration techniques.