Efficient Compressed Landweber Detector for Massive MIMO

Yufeng Yang[1,2,3,4], Wence Zhang[5,*], Jiejun Jin[1,2,4], Zaichen Zhang[2,3,4], Xiaohu You[2,3] and Chuan Zhang[1,2,3,4,*]
[1]Lab of Efficient Architectures for Digital-communication and Signal-processing (LEADS)
[2]National Mobile Communications Research Laboratory, Southeast University
[3]School of Information Science and Engineering, Southeast University
[4]Quantum Information Center of Southeast University, Nanjing, China
[5]School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang, China
Email: {yfyang, chzhang}@seu.edu.cn, wencezhang@ujs.edu.cn

***Abstract*—Massive multiple-input multiple-output (M-MIMO) provides better efficiency and coverage than conventional MIMO. Efficient detector, being indispensable for M-MIMO, is the focus of nowadays research. For M-MIMO, MMSE detector with exact matrix inversion works well but suffers from complexity, not to mention the performance degradation in ill-posed cases. Thus, Landweber detector (LD) and improved Landweber detector (ILD) are considered. In this paper, the compressed Landweber detector (CLD) is proposed with much lower complexity but the same performance as ILD. Numerical results show that for 128×8 MIMO with 64-QAM, CLD approximates MMSE detector with lower complexity. When BER $= 10^{-4}$, the gap is only 0.24 dB with 2 iterations. When antenna configuration is 128×60, CLD can achieve 82% complexity reduction than ILD. Unified hardware architecture of ILD is proposed. Folding is considered for further reduction. With the same hardware as ILD, architecture of CLD is finally proposed.**

Keywords*—*Massive MIMO, detection, compressed Landweber, efficient hardware.

I. Introduction

Massive multiple-input multiple-output (M-MIMO) is a key technology for 5G wireless [1, 2]. By equipping hundreds of antennas at transmitters and serving tens of antennas of users [3], M-MIMO provides massive boost in interference reduction, spectral efficiency, link reliability, and transmit-power efficiency over small-scale MIMO [4], where point-to-point MIMO links are the focus.

With massive antennas in M-MIMO uplink, one major concern for data detection is the computational complexity. With the growth of number of antennas, the traditional detectors using zero forcing (ZF) [5] and minimum mean square error (MMSE) [6] will result in drastic complexity increase for matrix inversion [7]. Thus, iterative detection algorithms are proposed to balance the complexity and detection performance. For instance, methods like Gauss-Seidel [8], conjugate gradient (CG) [9, 10], and conjugate residual (CR) [11] are proposed. Meanwhile, precondition is also proposed to achieve better performance [12, 13], especially for ill-posed conditions. However, designers are expecting detectors which can directly take care of ill-posed problems without preconditioner.

To this end, in this paper, a linear algorithm named Landweber detection (LD) is considered, which is usually used in solving ill-posed problems [14] by using series of matrix polynomials. It is worth noting that LD is not a derivation or approximation of MMSE, but in a different way. However, LD converges slowly and is not feasible for M-MIMO detection. Thus, by optimizing the relax factor of LD, improved Landweber detection (ILD) is proposed in [15] for lower complexity and faster convergence. However, the complexity of ILD is high. In this paper, by substituting each matrix-matrix product in ILD with matrix-vector product, compressed version of ILD (CLD) is proposed to reduce the computational complexity. Numerical results under different antenna configurations and different iterations are given to demonstrate the advantages.

To provide a reference for hardware architecture, computational complexities of ILD and CLD are elaborated and compared with exact *Cholesky* decomposition scheme [7]. Then a unified design method proposed in [16] is adopted to help design the hardware architecture of ILD. In this paper, ILD is implemented by using iterative modules. To avoid the high cost of the direct implementation of ILD, folding transformation is utilized for simplification and efficiency. Furthermore, architecture of low-complexity CLD is proposed with the same hardware requirement as ILD.

The remainder of the paper is organized as follows. Section II goes over the system model and linear detection of M-MIMO systems. Section III presents the traditional Landweber detection and the improved version of it. The optimized ILD is proposed as CLD in the same section. Numerical comparison is given in Section IV. In Section V, computational complexity is elaborated and compared with different approaches. Then unified hardware architecture of ILD and its efficient version are proposed. Afterwards, architecture of CLD is proposed. Finally, Section VI concludes the entire paper.

Notation: In this paper, the lowercase and upper bold face letters stand for column vector and matrix, respectively. The operations $(.)^T$ and $(.)^H$ denote transpose and conjugate transpose, respectively. The vector $\boldsymbol{\alpha}$ in the k-th iteration is $\boldsymbol{\alpha}_k$. $\mathbb{E}(.)$ denotes the expectation operation. $\rho_r(\mathbf{A})$ denotes the spectrum radius of matrix $\mathbf{A}$. $\nabla f(\mathbf{s})$ calculates the gradient of function $f(\mathbf{s})$. Computational complexity is denoted in terms of complexed-valued multiplication number of the algorithm.

II. Preliminaries

Consider an uplink of a M-MIMO system with N antennas at the base station (BS), which simultaneously serves M

single antenna users. Here, N is always much bigger than M ($N >> M$). The transmitted signal vector and received vector are denoted by $\mathbf{s} = [s_1, s_2, ..., s_M]^T$ and $\mathbf{y} = [y_1, y_2, ..., y_N]^T$, respectively, where $\mathbf{s} \in \mathbb{C}^M$, $\mathbf{y} \in \mathbb{C}^N$ and the transmitting power of each user antenna is $\mathbb{E}(|\mathbf{s}_i|^2) = E_s = 1$. Then the system model is described as

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{n}, \tag{1}$$

where $\mathbf{H}$ is an $N \times M$ uplink channel matrix, $\mathbf{n}$ is the vector representing circularly symmetric complex *Gaussian* distributed noise with zero-mean and variance σ^2. Also, the average signal-to-noise-ratio (SNR) per receive antenna is defined as $\mathrm{SNR} = \mathrm{ME_s}/\sigma^2$.

According to MMSE equalization scheme, at the BS side, the estimate of the transmitted symbol vector $\hat{\mathbf{s}}$ is

$$\hat{\mathbf{s}} = (\mathbf{H}^H\mathbf{H} + \sigma^2\mathbf{I}_M)^{-1}\mathbf{H}^H\mathbf{y} = \mathbf{A}\mathbf{y}, \tag{2}$$

where the matrix $\mathbf{I}$ is identity matrix with dimension M, and the detection matrix $\mathbf{A}$ is defined based on *Gram* matrix $\mathbf{G}$:

$$\mathbf{A} = (\mathbf{G} + \sigma^2\mathbf{I}_M)^{-1}\mathbf{H}^H, \tag{3}$$

where $\mathbf{G} = \mathbf{H}^H\mathbf{H}$.

For other linear methods like ZF, detection matrix $\mathbf{A}$ is

$$\mathbf{A} = (\mathbf{H}^H\mathbf{H})^{-1}\mathbf{H}^H. \tag{4}$$

Thus, computation of matrix $\mathbf{A}$ is very important. Nevertheless, computational complexity of exact matrix inversion in $\mathbf{A}$ is $\mathcal{O}(M^3)$. Methods such as *Cholesky* decomposition based method are not suitable for adoption in M-MIMO detection when the scale of it increases.

III. Compressed M-MIMO Landweber Detection

In this section, Landweber detection method for M-MIMO detection is introduced. Then by optimizing relax factor of LD, improved version of LD is introduced as ILD. Finally, CLD is proposed to reduce the computational complexity of ILD.

A. Landweber Detection

To specify LD, linear equation Eq. (1) is taken into consideration first. Suppose the noise vector is neglected, vector $\mathbf{n}$ in Eq. (1) becomes zero vector, which makes Eq. (1) become an ideal linear equation. Let $\mathbf{y}^*$ denote the exact received signal vector without noise, which is $\mathbf{y}^* = \mathbf{H}\mathbf{s}$. Assuming matrix $\mathbf{H}$ and received signal vector $\mathbf{y}$ are known exactly, then it can be deduced from Eq. (2) that

$$\mathbb{E}\|\mathbf{y} - \mathbf{y}^*\|_2^2 = N\sigma^2. \tag{5}$$

According to [17], a linear ill-posed problem is formed to get the optimal estimation of vector $\mathbf{s}$ from received signal vector $\mathbf{y}$ which is noise-polluted in Eq. (2). Landweber algorithm is perfect in solving ill-posed problems and thus is feasible in this application scheme.

To solve M-MIMO detection problems, LD aims at obtaining the optimal estimation of transmitted signal $\mathbf{s}$. Estimation of $\mathbf{s}$ is denoted by $\mathbf{A}_L\mathbf{y}$ in which

$$\mathbf{A}_L = \omega\sum_{t=1}^{T}(\mathbf{I}_M - \omega\mathbf{H}^H\mathbf{H})^t\mathbf{H}^H, \tag{6}$$

where T is the terminal factor to make a trade-off between performance and computational complexity, which is the iteration number K in real application and ω is the relax factor which controls the convergence of LD, which meets the restriction

$$0 < \omega < \frac{1}{\rho_r(\mathbf{H}^H\mathbf{H})}. \tag{7}$$

Key step of LD can be written as

$$\mathbf{s}_{k+1} = \mathbf{s}_k - \omega\mathbf{H}^H(\mathbf{H}\mathbf{s}_k - \mathbf{y}), \tag{8}$$

by supposing function $f(\mathbf{s}) = \|\mathbf{H}\mathbf{s} - \mathbf{y}\|_2^2/2$, Eq. (8) is

$$\mathbf{s}_{k+1} = \mathbf{s}_k - \omega\nabla f(\mathbf{s}), \tag{9}$$

which is a special case of gradient descent methods.

To better describe LD, LD for M-MIMO systems is elaborated in *Algorithm* 1 and is shown in the form of iteration.

Algorithm 1 Landweber Detection for M-MIMO Systems

Input: $\mathbf{H}$, ω and $\mathbf{y}$
1: $\mathbf{s}_0 = \omega\mathbf{H}^H\mathbf{y}$ and $\mathbf{r}_0 = \mathbf{0}$
2: **for** $k = 0, \ldots, K$ **do**
3: $\quad\mathbf{r}_{k+1} = \mathbf{y} - \mathbf{H}\mathbf{s}_k$
4: $\quad\mathbf{s}_{k+1} = \mathbf{s}_k + \omega\mathbf{H}^H\mathbf{r}_{k+1}$
5: **end for**
Output: $\hat{\mathbf{s}} = \mathbf{s}_{K+1}$

B. Optimization of Relax Factor

As can be seen in Section III-A, LD is explicitly described in *Algorithm* 1. However, the value of relax factor ω is pending, which can be set by users before. To achieve better performance, it can be optimized as shown in [15]. Taking advantage of Eq. (6), the matrix polynomial converges fast when the value of $\rho_r(\mathbf{I}_M - \omega\mathbf{H}^H\mathbf{H})$ is small. Thus relax factor ω can be optimized to ω^* by solving

$$\omega^* = \arg\min \rho_r(\mathbf{I}_M - \omega\mathbf{H}^H\mathbf{H}). \tag{10}$$

Seen in [15], solution to optimization function (10) is

$$\omega^* = \frac{1}{N + M}, \tag{11}$$

where N and M are antenna numbers of BS and user.

C. Improved Landweber Detection

By substituting ω of LD with optimized relax factor ω^*, convergence performance of LD can be improved. Meanwhile, computation structure of Eq. (6) can be derived into another form,

$$\mathbf{A}_{k+1} = (2\mathbf{I}_M - \mathbf{A}_k\mathbf{H})\mathbf{A}_k, \tag{12}$$

where detection matrix $\mathbf{A}$ is updated after each iteration. Meanwhile, by computing in this way, k times of iteration of ILD has the same performance with $2^k - 1$ times of iteration of LD, which can improve the convergence rate of ILD.

Synthesizing these two optimizations, improved Landweber detection [15] can achieve optimal performance compared with LD. Computation process of ILD is shown in *Algorithm* 2.

Algorithm 2 Improved Landweber Detection

Input: $\mathbf{H}$ and $\mathbf{y}$

$\omega = \frac{1}{N+M}$, $\mathbf{A}_0 = \omega \mathbf{H}^H$ and $\mathbf{R}_0 = \mathbf{0}$

for $k = 0, \ldots, K$ **do**

  $\mathbf{R}_{k+1} = 2\mathbf{I}_M - \mathbf{A}_k \mathbf{H}$

  $\mathbf{A}_{k+1} = \mathbf{R}_{k+1} \mathbf{A}_k$

end for

Output: $\hat{\mathbf{s}} = \mathbf{A}_{K+1} \mathbf{y}$

D. Proposed Compressed Landweber Detection

To avoid too much matrix-matrix product of ILD, a compressed version of ILD is proposed in this paper as compressed Landweber detection (CLD). Computation of ILD can be simplified by another way, in which every matrix-matrix product of ILD can be substituted by matrix-vector product.

Post-multiplying $\mathbf{y}$ on both sides of Eq. (6) and denoting $\mathbf{I}_M - \omega \mathbf{H}^H \mathbf{H}$ by $\mathbf{R}$, CLD can lower the complexity from matrix-matrix product to matrix-vector product. By multiplying 2^k times of matrix $\mathbf{R}$, CLD can achieve the same performance with ILD while reduces the computational complexity of ILD to a large extent. Computation process of CLD is shown in *Algorithm* 3.

Algorithm 3 Compressed Landweber Detection

Input: $\mathbf{H}$ and $\mathbf{y}$

$\omega = \frac{1}{N+M}$, $\mathbf{R} = \mathbf{I}_M - \omega \mathbf{H}^H \mathbf{H}$ and $\mathbf{s}_0 = \omega \mathbf{H}^H \mathbf{y}$

for $k = 0, \ldots, K$ **do**

  $\mathbf{s}_{k+1} = \mathbf{s}_k + \mathbf{R}^{2^k} \mathbf{s}_s$

end for

Output: $\hat{\mathbf{s}} = \mathbf{s}_{K+1}$

IV. Numerical Results and Comparisons

As is mentioned in Section III-A, LD can be classified as a member of gradient descent detection methods. Thus in this section, ILD and CLD are compared with another gradient descent detection method CG [9] and exact *Cholesky* decomposition scheme. Numerical results for different M-MIMO configurations are given to validate CLD and ILD, meanwhile comparisons of each specific algorithm are given. Adopting 64-QAM scheme and using independent and identically distributed (i.i.d.) *Rayleigh* fading channel model, BER comparison between each method and three antenna configurations are considered. Here iteration time K is considered to be 2, 3 and 4, respectively.

It can be seen from three figures that as antenna ratio N/M increases, performance of CLD, ILD and CG increase as well. With optimized relax factor, ILD and CLD outperform CG under the same condition in each iteration. When iteration number $k \geq 3$, it can be seen in Fig. 2 and Fig. 3 that ILD and CLD can achieve nearly the same performance with *Cholesky* decomposition scheme and the performance gap is less than 0.1 dB after 3 iterations. ILD and CLD can still perform well after only 2 times of iteration. Take Fig. 2 for instance, after 2 times of iteration, ILD and CLD have more than 6 dB advantage over CG when BER=10^{-2} and has only 0.37 dB loss compared with CG which iterates 3 times when BER=10^{-3}. Fig. 3 can

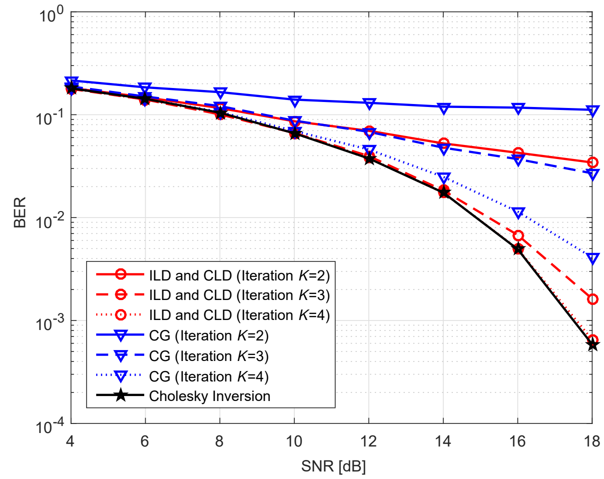

Fig. 1: BER performance comparison with $N \times M = 128 \times 32$.

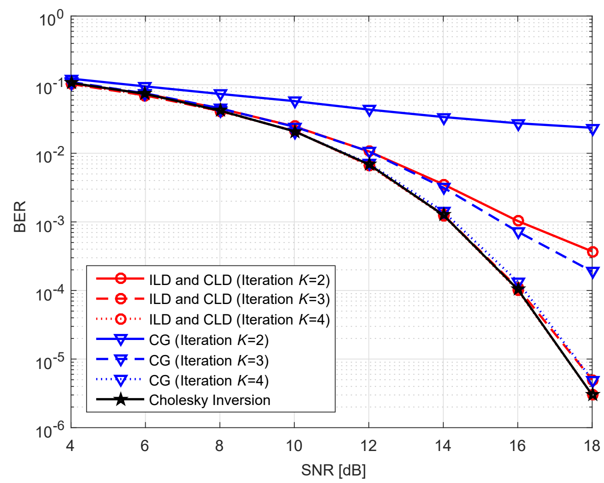

Fig. 2: BER performance comparison with $N \times M = 128 \times 16$.

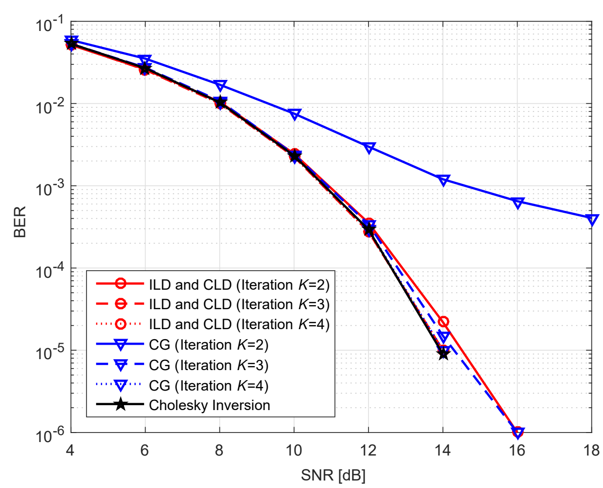

Fig. 3: BER performance comparison with $N \times M = 128 \times 8$.

validate the advantage of ILD and CLD over CG further. When iteration time $k = 2$, ILD and CLD have 5.13 dB advantage over CG when BER=4×10^{-4}. Besides, as antenna ratio N/M increases, performance gap between CLD and CG after 2 times iteration increases, which also showcases the competitive BER performance of ILD and CLD.

V. Hardware Architecture

In this section, computational complexities are compared to elaborate complexity of ILD and CLD. Then hardware architecture of ILD is proposed. To better implement ILD, an efficient way to design hardware architecture of ILD is adopted and then efficient architecture of ILD is proposed. Finally, architecture of low-complexity CLD is proposed.

A. Computational Complexity

In this section, computational complexity is analyzed in terms of numbers of complex multiplications of algorithms. Computational complexities of ILD and CLD are elaborated in contrast with CG and *Cholesky* decomposition scheme. The exact value of computational complexity is summarized in Table I to show the difference between three methods and *Cholesky* means the exact *Cholesky* decomposition scheme. K is the number of iterations.

TABLE I: Complexity Comparison of Different Algorithms.

Algorithm	Number of complex-multiplications
ILD	$2KNM^2 + N$
CLD	$NM^2 + NM + (2^K - 1)M^2$
CG	$NM^2 + K(2M^2 + 4M)$
Cholesky	$NM^2 + \frac{5}{6}M^3 + \frac{3}{4}M^2 + \frac{4}{3}M$

To make the comparison more explicit, complexity of three methods are shown in Fig. 4. In this scheme, number of antenna at BS is 128, SNR is set to be 20 dB. As is mentioned in Table I, complexity of ILD is higher than other algorithms while CLD lowers the complexity of ILD to a large extent. Meanwhile, main complexity of CLD is contributed by computing matrix $\mathbf{H}^H\mathbf{H}$, which is also needed in CG or *Cholesky* decomposition scheme. When number of user antenna M is 60, CLD can achieve nearly 72% complexity reduction compared with ILD after 2 iterations and nearly 82% reduction after 3 times of iteration compared with ILD after 3 times of iteration. It is also worth noting that CLD has similar complexity to CG and has lower complexity than *Cholesky* decomposition, which shows the feasibility of CLD in M-MIMO systems.

B. Proposed Unified Architecture of ILD

To help the implementation of ILD, unified hardware architecture of ILD is proposed in this paper. Firstly, a normalizing design method is adopted and then hardware architecture of ILD is designed under the guidance of this method.

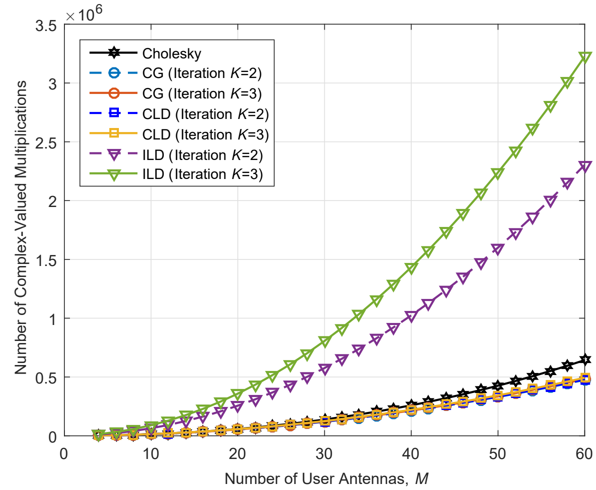

Fig. 4: Complexity comparison of different algorithms.

1) Normalizing Design Method: To help the flexibility of hardware, for instance, users can change their detectors with existing resources as long as they want to, a normalizing design method is needed. With the application of such method, architecture of detectors can be reusable, which will save the hardware resources in some sense. Method in [16] introduces two modules, while in this paper, only iterative module is needed.

Iterative module is a module made up of two operating units, a multiplier and an accumulator, which performs a multiply and accumulation (MAC) operation.

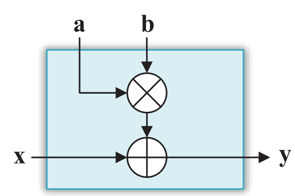

Fig. 5: Iterative module of normalizing design method.

Function of iterative module can be denoted by $\mathbf{y} = \mathbf{x}+\mathbf{ab}$. Besides this module, only some multipliers and delayers are needed, which improves the flexibility of architectures. By adopting this design method, hardware resources can be reserved for further usage.

2) Unified Architecture of ILD: With the optimized relax factor, ILD has optimized performance. By setting optimized relax factor before and optimize the architecture of computation process of LD, ILD only needs one iterative module. Unified hardware architecture of ILD is shown in Fig. 6.

In terms of this architecture, channel matrix $\mathbf{H}$ and received signal $\mathbf{y}$ are needed. Input from the left of the delayer is the initial value of detection matrix $\mathbf{A}$. Other modules have the same function with modules of the architecture of LD. It can be seen that by optimizing relax factor, architecture of ILD only needs one computation of ω, which is more efficient than the architecture of LD.

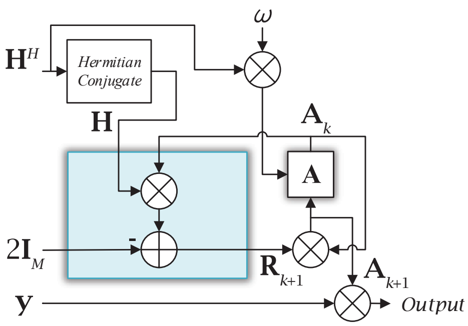

Fig. 6: Unified hardware architecture of ILD.

C. Proposed Efficient Architecture of ILD

Although unified architecture of ILD is efficient and need only one module, computation process of ILD contains two steps. Folding transformation [18] is an efficient way to save the hardware resource of ILD and by folding, an efficient architecture of ILD is proposed.

In the architecture design of digital signal processing (DSP), saving functional units like multipliers or accumulators is of great significance. Folding transformation can systematically control the circuit and by this, multiplexing of functional units can be achieved, thus improves the efficiency of the hardware. Generally, folding can save the number of functional units of the architecture. However, it may introduce more delayers, which will increase the complexity of the hardware. Thankfully, hardware architecture of ILD only involves one delayer, thus folding will not affect the complexity of hardware too much. Besides, folding sacrifices the latency for the space consumption. Fig. 7 shows the proposed efficient hardware architecture of ILD.

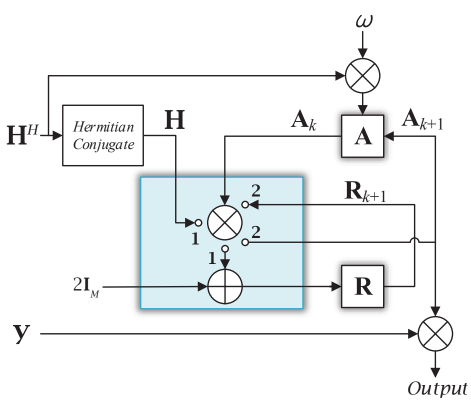

Fig. 7: Efficient hardware architecture of ILD.

Compared with the original unified architecture of ILD in Fig. 6, efficient architecture of ILD saves a multiplier and add a delayer for storage. Dividing the computation into two periods, multiplier in iterative module has two functions in two periods, respectively.

1) Period 1: In this period, multiplier serves for the multiplication in step 3 of *Algorithm* 2, which computes the multiplication of detection matrix $\mathbf{A}_k$ and channel matrix $\mathbf{H}$. By connecting the nodes marked with "1", iterative module functions as usual.

2) Period 2: In this period, multiplier serves for the multiplication in step 4 of *Algorithm* 2, which updates the value of detection matrix by multiplying former detection matrix $\mathbf{A}_k$ and intermediate matrix $\mathbf{R}_{k+1}$. In period 2, multiplier disconnects nodes marked with "1" and connects nodes marked with "2". By switching nodes, multiplexing of the multiplier can be realized.

Hardware requirement of ILD is not large, however by folding transformation, hardware requirement of ILD can be reduced further. Dividing ILD into three stages, initialization stage, ILD algorithm stage and output stage, respectively. Hardware of unified architecture of ILD and efficient architecture of ILD in initialization stage and output stage both need one multiplier, which cannot be optimized. However, in ILD algorithm stage, architecture can be optimized by folding transformation, hardware requirement comparison of two architectures in ILD algorithm stage is shown in Table II.

TABLE II: Hardware Requirement of Different Architectures.

Architecture	**Accumulator**	**Multiplier**	**Delayer**
Unified	1	2	1
Efficient	1	1	2

In Table II, "Unified" means the unified architecture of ILD proposed in Section V-B2 and efficient architecture of ILD is denoted by "Efficient". As can be seen that efficient architecture of ILD only needs one accumulator and one multiplier, which is the most simple architecture and the need of this architecture is easy to meet. Compared with unified architecture of ILD, efficient architecture sacrifices time for space because it needs two periods to compute signals and realize the multiplexing of the multiplier. However, users can switch the hardware as they want to for the iterative module is easy to be reused.

D. Proposed Architecture of CLD

Because of the lower complexity compared with ILD, CLD is more feasible than ILD. Hardware architecture of CLD is designed as is shown in Fig. 8.

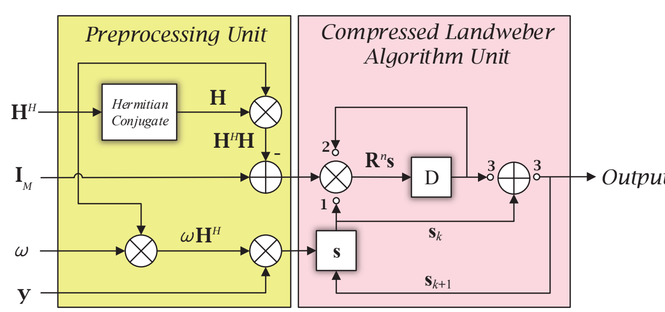

Fig. 8: Hardware architecture of CLD.

As is shown, hardware architecture of CLD contains two units, preprocessing unit and compressed Landweber algorithm unit. With the input of channel matrix and received signal vector, preprocessing unit functions as initialization of each matrix or vector. Then signals are sent to compressed Landweber algorithm unit, which functions as CLD algorithm and computes signals in three periods, respectively.

1) Period 1: In period 1, node with “1” connects and the estimation of transmitted signal $\mathbf{s}$ is multiplied with intermediate matrix $\mathbf{R}$. Left input of $\mathbf{s}$ is the initialization operation.

2) Period 2: In this period, delayer with “D” stores the multiplication of matrix $\mathbf{R}$ and $\mathbf{R}^n\mathbf{s}_k$ and in this period, node with “2” connects thus a loop forms. The loop terminates when $n = 2^k$ and the delayer stores the value of $\mathbf{R}^{2^k}\mathbf{s}$.

3) Period 3: In this period, nodes with “3” connect and with the value of the delayer with “D”, signal $\mathbf{s}$ can be updated by an accumulator.

It is worth noting that compressed Landweber algorithm unit requires one multiplier, one accumulator and two delayers, which is same with that of efficient architecture of ILD. Thus with the same hardware requirement and lower complexity, CLD is more suitable for real application.

VI. Conclusion

In this paper, LD is introduced as a low-complexity detection method for ill-posed problems. Then the improved LD is introduced as ILD, with optimized relax factor. Optimizing ILD further, CLD has lower complexity but maintains the same performance as ILD. Being special cases of gradient descent algorithm, ILD and CLD are compared with another gradient-based algorithm CG. Numerical results elaborate the comparison of ILD, CLD, CG and *Choleksy* decomposition scheme, which validate the feasibility of CLD in M-MIMO detection. Computational complexity analysis shows CLD can lower the complexity of ILD to a great extent in M-MIMO scheme. Finally by adopting a normalizing design method, unified hardware architecture of ILD is proposed. Then by folding transformation, efficient architecture of ILD is proposed. With the same cost, hardware architecture of low-complexity CLD is finally proposed.

Acknowledgement

This work is supported in part by NSFC under grant 61501116, Jiangsu Provincial NSF for Excellent Young Scholars, Huawei HIRP Flagship under grant YB201504, the Fundamental Research Funds for the Central Universities, the SRTP of Southeast University, State Key Laboratory of ASIC & System under grant 2016KF007, ICRI for MNC, and the Project Sponsored by the SRF for the Returned Overseas Chinese Scholars of MoE.

References

[1] E. Larsson *et al.*, “Massive MIMO for next generation wireless systems,” *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 186–195, 2014.

[2] J. Lee, J.-K. Han, and J. C. Zhang, “MIMO technologies in 3GPP LTE and LTE-advanced,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2009, no. 1, pp. 1–10, 2009.

[3] H. Huh, G. Caire, H. C. Papadopoulos, and S. A. Ramprashad, “Achieving “massive MIMO” spectral efficiency with a not-so-large number of antennas,” *IEEE Trans. Wireless Commun.*, vol. 11, no. 9, pp. 3226–3239, 2012.

[4] L. Lu, G. Y. Li, A. L. Swindlehurst, A. Ashikhmin, and R. Zhang, “An overview of massive MIMO: Benefits and challenges,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 5, pp. 742–758, 2014.

[5] 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, 2004.

[6] E. G. Larsson, “MIMO detection methods: How they work [lecture notes],” *IEEE Signal Process. Mag.*, vol. 26, no. 3, 2009.

[7] A. Krishnamoorthy and D. Menon, “Matrix inversion using Cholesky decomposition,” in *Proc. IEEE Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*, 2013, pp. 70–72.

[8] Z. Wu, L. Ge, X. You, and C. Zhang, “Efficient near-MMSE detector for large-scale MIMO systems,” in *Proc. IEEE International Workshop on Signal Processing Systems (SiPS)*, 2017, pp. 1–6.

[9] B. Yin, M. Wu, J. R. Cavallaro, and C. Studer, “Conjugate gradient-based soft-output detection and precoding in massive MIMO systems,” in *Proc. IEEE International Workshop on Green Communications, parallel with IEEE GLOBECOM*, 2014, pp. 3696–3701.

[10] ——, “VLSI design of large-scale soft-output mimo detection using conjugate gradients,” in *Proc. IEEE Circuits and Systems (ISCAS)*, 2015, pp. 1498–1501.

[11] Y. Yang, Y. Xue, X. You, and C. Zhang, “An efficient conjugate residual detector for massive MIMO systems,” in *Proc. IEEE International Workshop on Signal Processing Systems (SiPS)*, 2017, pp. 1–6.

[12] J. Jin, Y. Xue, Y. L. Ueng, X. You, and C. Zhang, “A split pre-conditioned conjugate gradient method for massive MIMO detection,” in *Proc. IEEE International Workshop on Signal Processing Systems (SiPS)*, 2017, pp. 1–6.

[13] Y. Xue, C. Zhang, S. Zhang, and X. You, “A fast-convergent pre-conditioned conjugate gradient detection for massive MIMO uplink,” in *Proc. IEEE Digital Signal Processing (DSP)*, 2016, pp. 331–335.

[14] L. Landweber, “An iteration formula for Fredholm integral equations of the first kind,” *American journal of mathematics*, vol. 73, no. 3, pp. 615–624, 1951.

[15] W. Zhang, X. Bao, and J. Dai, “Low-complexity detection based on landweber method in the uplink of massive MIMO systems,” in *Proc. IEEE European Signal Processing Conference (EUSIPCO)*, 2017, pp. 873–877.

[16] C. Zhang, Y. Yang, S. Zhang, Z. Zhang, and X. You, “Residual-based detections and unified architecture for massive MIMO uplink,” *arXiv preprint arXiv:1802.05982*, 2018.

[17] Z. Jun, “An accelerated landweber iteration for solving ill-posed problems [J],” *Journal of Mathematics*, vol. 5, pp. 333–335, 2005.

[18] K. K. Parhi, *VLSI digital signal processing systems: design and implementation*. John Wiley & Sons, 2007.