Massive MIMO Detection based on Barzilai-Borwein Algorithm

Jiejun Jin[1,2,3], Zaichen Zhang[2,3], Xiaohu You[2], and Chuan Zhang[1,2,3,*]
[1]Lab of Efficient Architectures for Digital-communication and Signal-processing (LEADS)
[2]National Mobile Communications Research Laboratory, Southeast University
[3]Chien-Shiung Wu College, Southeast University
[4]Quantum Information Center of Southeast University, Nanjing, China
Email: {jjjin, chzhang}@seu.edu.cn

***Abstract*—For a massive multiple-input multiple-output (MIMO) system, how to cope with the detection difficulties brought by increasing antennas is intractable. Linear methods such as zero-forcing (ZF) and minimum mean square error (MMSE) can achieve sub-optimal performance while suffer from high complexity because of large-scale matrix inversion. Recently, some iterative detectors such as steepest descent (SD) and conjugate gradient (CG) have been proposed to balance the complexity and performance. However, their fast convergence would not maintain when the system loading factor increases. To address the aforementioned issues, this paper 1) introduces a new iterative algorithm called Barzilai-Borwein (BB) that outperforms SD with inexpensive operations and 2) proposes its improved form entitled SDBB to accelerate the convergence even in bad conditions. Both theoretical and numerical results have demonstrated its advantages over the state-of-the-art ones. More specifically, SDBB can surpass the existing split pre-conditioned conjugate gradient (SPCG) detector by more than 2 dB at the bit error rate (BER) of 10^{-3} when the number of users is relatively large, and reach a complexity reduction of 10%.**

Keywords*—Massive MIMO, linear detection, iterative method, steepest descent, Barzilai-Borwein, convergence analysis.

I. Introduction

Massive multiple-input multiple-output (MIMO) system has become one of the key techniques for the next generation mobile communication [1]. Unlike the small-scale MIMO used in the fourth generation communication system (4G-long term evolution, LTE), which allows the base station (BS) to equip with 8 antennas, massive MIMO is characterized by hundreds of antennas at the BS and tens of user terminals (UTs). The large-scale antennas array contributes to higher spectral efficiency, less transmitting power, faster data rates, and more stable connection [2]. Therefore, it is regarded as part of the basic infrastructure of future wireless system.

Although massive MIMO systems improve the capacity and reliability of wireless communication, it leads to a more complicated configuration, which is knotty for data detection. With the increased users and antennas, the complexity of the optimal detection methods relying on maximum-likelihood (ML) or maximum a posterior (MAP) criterion will grow exponentially, restricting their practical use. In order to lower the computational complexity, many linear detection schemes were proposed. Among them, zero forcing (ZF) method and minimum mean square error (MMSE) method are both near-optimal choices but still contain direct matrix inversion operations and have the complexity up to $\mathcal{O}(U^3)$ [3] where U is the number of users. As a result, they become unrealizable in large-scale systems.

To avoid matrix inversion, where the main complexity of basic linear detectors comes from, one can resort to approximate methods or iterative ones. The first group includes Neumann series expansion (NSE) method proposed in [4]. Although there is no need to compute inversion directly for this algorithm, when the term of NSE is more than 2, the computational complexity will return to $\mathcal{O}(U^3)$ [5]. What is more, when the number of users goes up, the performance gap between NSE method and accurate MMSE will be dramatically broadened. In view of its complexity and convergence problem, another inversion-free option is iterative methods, whose superiority in calculations is evident, for instance the steepest descent (SD) detector [6, 7] can reduce the complexity to $\mathcal{O}(U^2)$ with acceptable bit error rate (BER) when the number of users is 16 or less. But the robustness of such method cannot be guaranteed as the user number grows. Many literatures focused on this phenomenon and introduced different pre-conditioners [8] for better results. Whereas, even the simplest pre-conditioned process comes along with additional calculations and adds the difficulty for hardware design, which contradicts the original intention of this kind of methods.

Therefore, the aforementioned detectors fail to reach a tradeoff between detection performance and computational complexity, especially when the systems are intricate. Considering the low computation load and hardware-friendly features of iterative methods, one possible way for optimization is to speed up their convergence. Instead of trying line searches like SD scheme, quasi-Newton methods can achieve competitive performance with inexpensive operations. Here we propose a novel quasi-Newton detection method based on Barzilai-Borwein (BB) algorithm [9]. Its iterative scheme is similar to that of SD in appearance, and has similar computational complexity. But BB detector can overcome the disadvantage of slow convergence of SD in ill conditions. This advantage is obvious when the number of iteration is large.

Moreover, taken the similarity in iterative form of BB and SD into consideration [10], they can be combined with each other and induce the SDBB method. Mathematics derivation can prove that one iteration result of SDBB is equivalent to that of BB after one round through SD, which enhances the performance greatly. Furthermore, according to the SDBB algorithm, it does not need to store the previous iterative results like BB which also saves the storage space against the original algorithm. Both theoretical and numerical results confirm fast convergence of SDBB and show its advantages over previous iterative methods. More importantly, SDBB deals with the iterative form directly without adding extra procedure as pre-conditioned process. This optimization proves to be more effective than the split pre-conditioned conjugate gradient (SPCG) method [11], which could be demonstrated through simulation and analysis, providing a new idea for massive MIMO detection. All in all, SDBB method balances the performance as well as complexity in detection to a better degree, which surpasses the state-of-the-art ones.

The remainder of this paper is organized as follows. Section II introduces the detection model of massive MIMO and lays the foundation for later analysis. Section III proposes the BB method and its improved form SDBB in detail. Numerical results are shown in Section IV. Finally, Section V concludes the entire paper.

Notation: In this paper, lower-case and upper-case boldface letters indicate column vector and matrix, respectively. $\mathbf{I_N}$ represents a $N \times N$ identity matrix. The operations $(\boldsymbol{.})^T$, $(\boldsymbol{.})^H$, $\mathbb{E}\{\boldsymbol{.}\}$, $\| \boldsymbol{.} \|_2$ and $(\cdot)$ stand for transpose, conjugate transpose, expectation, 2-norm and inner product, respectively. The vector $\mathbf{a}$ in the k-th iteration is $\mathbf{a}^{(k)}$.

II. Preliminaries

Consider a single cellular uplink massive MIMO system which is equipped with B antennas at the BS concurrently receiving data from U single-antenna UTs. Usually, B is bigger than U $(B > U)$. Loading factor $\rho = U/B$ is a parameter used to denote the characteristic of the system.

A. Massive MIMO System Model

For a typical uplink system, every user's information bit is encoded and mapped to a modulation constellation set $\mathcal{O}$ at the UTs first. Then the stream is transmitted through the channel. Eventually, the received symbol $\mathbf{y} \in \mathbb{C}^B$ at the BS is:

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

where $\mathbf{s} = [s_1, s_2, \ldots, s_U]^T \in \mathcal{O}^U$ is the $U \times 1$ transmitted vector. Its element $s_k \in \mathcal{O}$ $(k = 1, 2, \ldots, U)$ represents the symbol from the k-th user with normalized energy, i.e. $\mathbb{E}\{\|s_k\|^2\} = 1$, $\forall k$. $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, ...\mathbf{h}_N] \in \mathbb{C}^{B\times U}$ is the flat Rayleigh fading channel matrix whose entries are assumed to be independent identical distributed (i.i.d.) with zero mean and unit variance. The addictive white Gaussian noise (AWGN) is indicated by $\mathbf{n} \in \mathbb{C}^B$ with zero mean and σ^2 variance.

B. MMSE Detection Scheme

As interpreted in [12], linear detection method MMSE is able to achieve sub-optimal BER performance by minimizing the mean square error between the estimated transmitted symbol and the accurate one. Such estimation after the MMSE detector can be described as:

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

For the sake of convenience, we can use $\tilde{\mathbf{y}} = \mathbf{H}^H\mathbf{y}$ and $\mathbf{A} = \mathbf{H}^H\mathbf{H} + \sigma^2\mathbf{I}$ to replace the corresponding term and in the end, the estimated result is:

$$\hat{\mathbf{s}} = \mathbf{A}^{-1}\tilde{\mathbf{y}}. \tag{3}$$

As $\mathbf{H}$ and σ^2 are clearly known by the BS, and the average signal-to-noise ratio (SNR) can be calculated by $\zeta = \mathbb{E}\{\|s_k\|^2\}/\sigma^2 = 1/\sigma^2$ $(\forall s_k \in \mathbf{s})$, transmitted symbol can be solved through Eq. (3) in theory. But the computation load of calculating direct inversion of $\mathbf{A}$ is too heavy to afford. Even solved with Gauss Elimination approach like QR decomposition or Cholesky factorization, inversion complexity will turn overwhelming as user number increases, preventing MMSE detector from application in massive MIMO systems.

III. Proposed BB-Based Detection Method

In this section, the conventional SD algorithm for linear MMSE detection is reviewed first. Then a novel iterative method named BB is introduced. Details of this method including the derivation, advantages and the relationship with SD are also given. Based on the similarity between these two methods, the improved form SDBB is proposed. Convergence and complexity analysis of SDBB in mathematics are provided to show the enhancement.

A. Steepest Descent Algorithm

As has been discussed in Section II-B, the MMSE scheme transforms the detection problem into a linear equation form: $\mathbf{A}\hat{\mathbf{s}} = \tilde{\mathbf{y}}$, which is also the solution to $\hat{\mathbf{s}} = \arg\min_{\mathbf{s}\in\mathcal{O}^U} f(\mathbf{s})$ where $f(\mathbf{s}) = \frac{1}{2}\mathbf{s}^H\mathbf{A}\mathbf{s} - \tilde{\mathbf{y}}\mathbf{s} + \frac{1}{2}\mathbf{y}^H\mathbf{y}$. Traditional SD algorithm is an iterative method whose main idea is to select step size towards the negative gradient with line searches in every iteration period, so that the value of the objective function can be reduced to the minimum. When applied to the detection problem here, the estimated symbol $\hat{\mathbf{s}}$ can be attained through iterative process as follows:

$$\hat{\mathbf{s}}^{(k+1)} = \hat{\mathbf{s}}^{(k)} + \frac{\left(\tilde{\mathbf{y}}^{(k)} \cdot \tilde{\mathbf{y}}^{(k)}\right)}{\left(\mathbf{A}\tilde{\mathbf{y}}^{(k)} \cdot \tilde{\mathbf{y}}^{(k)}\right)}\tilde{\mathbf{y}}^{(k)}, \tag{4}$$

$$\tilde{\mathbf{y}}^{(k+1)} = \tilde{\mathbf{y}}^{(k)} - \frac{\left(\tilde{\mathbf{y}}^{(k)} \cdot \tilde{\mathbf{y}}^{(k)}\right)}{\left(\mathbf{A}\tilde{\mathbf{y}}^{(k)} \cdot \tilde{\mathbf{y}}^{(k)}\right)}\mathbf{A}\tilde{\mathbf{y}}^{(k)}. \tag{5}$$

The channel matrix $\mathbf{H}$ proves to be asymptotically orthogonal when the number of antennas at the BS is much greater than the users' amount and matrix $\mathbf{A}$ is assumed to be Hermitian positive definite (HPD) in a real scene [13]. In this case, the convergence of SD method can be ensured.

However, when the loading factor ρ of the system is comparatively large which leads to a poorly conditioned problem, the convergence rate of SD becomes relatively slow and may increasingly 'zigzag' as the gradient point approaches to the minimum value [10]. These barriers cause bad performance in BER and restrict the application scenarios of SD method.

B. Proposed Barzilai-Borwein Algorithm

Unlike the SD method, BB algorithm is derived as a quasi-Newton method. It seeks the approximation of Hessian matrix in primitive Newton method which satisfies the secant equation:

$$\mathbf{A}_k \mathbf{x}^{(k)} = \mathbf{z}^{(k)}, \tag{6}$$

where $\mathbf{x}^{(k)} = \hat{\mathbf{s}}^{(k+1)} - \hat{\mathbf{s}}^{(k)}$ and $\mathbf{z}^{(k)} = \nabla f(\hat{\mathbf{s}}^{(k+1)}) - \nabla f(\hat{\mathbf{s}}^{(k)})$.

To make the inversion of $\mathbf{A}_k$ easy to compute, BB method uses a scalar α_k as a substitution to solve $\alpha_k \mathbf{x}^{(k)} = \mathbf{z}^{(k)}$ in the least square sense. Accordingly, α_k is given as:

$$\alpha_k = \frac{\left(\mathbf{z}^{(k)} \cdot \mathbf{x}^{(k)}\right)}{\left(\mathbf{x}^{(k)} \cdot \mathbf{x}^{(k)}\right)} = \frac{\left(\mathbf{A}\mathbf{x}^{(k)} \cdot \mathbf{x}^{(k)}\right)}{\left(\mathbf{x}^{(k)} \cdot \mathbf{x}^{(k)}\right)}. \tag{7}$$

By updating the parameter in the quasi-Newton algorithm, the proposed BB algorithm can be obtained as below:

$$\begin{aligned}\hat{\mathbf{s}}^{(k+1)} &= \hat{\mathbf{s}}^{(k)} - \frac{1}{\alpha_{k-1}} \nabla f(\hat{\mathbf{s}}^{(k)}) \\ &= \hat{\mathbf{s}}^{(k)} - \frac{\left(\mathbf{x}^{(k-1)} \cdot \mathbf{x}^{(k-1)}\right)}{\left(\mathbf{A}\mathbf{x}^{(k-1)} \cdot \mathbf{x}^{(k-1)}\right)} \nabla f(\hat{\mathbf{s}}^{(k)}).\end{aligned} \tag{8}$$

It can be found that the search direction in the BB method is always negative gradient direction like the SD, but no line searches are required during the process. What is more, BB converges faster than SD with the same cost at each iteration [14], overcoming the weakness of SD in bad-conditioned problems (when the user number or the loading factor is relatively large), which could be illustrated by Fig. 1. Performance of Cholesky factorization is used as a benchmark.

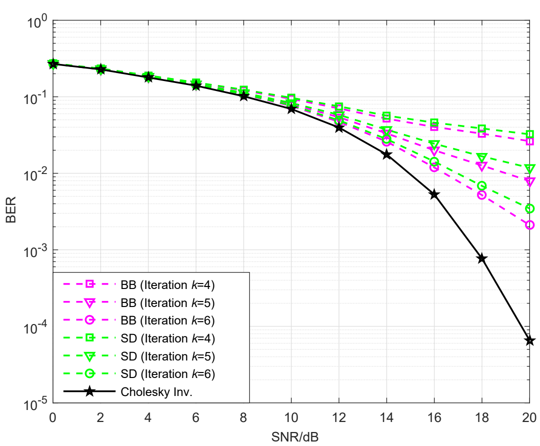

Fig. 1. BER performance comparison between BB and SD with $B = 128$ and $U = 32$.

Even though the performance of BB method outperforms that of SD in Fig. 1, there are some flaws left for perfection. First, although the number of iteration k is already up to 6, the convergence gap between BB and the ideal Cholesky factorization is still wide. In addition, since the kth iteration of BB needs the result of $\hat{\mathbf{s}}^{(k)}$ and $\hat{\mathbf{s}}^{(k-1)}$, more storage space will be taken up with the same computational complexity compared to SD. To this end, the improved method SDBB based on BB algorithm is proposed.

C. Proposed SDBB Algorithm

If we reanalyze the expressions of BB in Eq. (8) while using $\tilde{\mathbf{y}}^{(k)}$ to replace $-\nabla f(\hat{\mathbf{s}}^{(k)})$ and denoting $\mathbf{x}^{(k-1)} = \hat{\mathbf{s}}^{(k)} - \hat{\mathbf{s}}^{(k-1)} = \frac{\tilde{\mathbf{y}}^{(k-1)}}{\alpha_{k-2}}$, the iterative process of BB method can be transformed into the following equations:

$$\hat{\mathbf{s}}^{(k+1)} = \hat{\mathbf{s}}^{(k)} + \frac{\left(\tilde{\mathbf{y}}^{(k-1)} \cdot \tilde{\mathbf{y}}^{(k-1)}\right)}{\left(\mathbf{A}\tilde{\mathbf{y}}^{(k-1)} \cdot \tilde{\mathbf{y}}^{(k-1)}\right)} \tilde{\mathbf{y}}^{(k)}, \tag{9}$$

$$\tilde{\mathbf{y}}^{(k+1)} = \tilde{\mathbf{y}}^{(k)} - \frac{\left(\tilde{\mathbf{y}}^{(k-1)} \cdot \tilde{\mathbf{y}}^{(k-1)}\right)}{\left(\mathbf{A}\tilde{\mathbf{y}}^{(k-1)} \cdot \tilde{\mathbf{y}}^{(k-1)}\right)} \mathbf{A}\tilde{\mathbf{y}}^{(k)}. \tag{10}$$

By this way, the relationship between BB and SD is more apparent through the comparison of Eq.s (4, 5) and Eq.s (9, 10). If t_k is used to represent the step length of SD in kth iteration, the corresponding result of BB can be written as t_{k-1}. To some extent, the iteration step size of BB is just one period behind.

In the light of this connection between these methods, we combine SD and BB, and entitle the detection method SDBB. As an improved form of BB, this method can speed up the convergence when the number of users is large and avoid the storage of previous data information, consequently achieving excellent performance.

For the detailed SDBB method, the first step is:

$$\mathbf{p}^{(k)} = \hat{\mathbf{s}}^{(k)} + t_k \tilde{\mathbf{y}}^{(k)} = \hat{\mathbf{s}}^{(k)} - t_k \nabla f(\hat{\mathbf{s}}^{(k)}). \tag{11}$$

If $\hat{\mathbf{s}}^{(k+1)}$ is used to take the place of $\mathbf{p}^{(k)}$, it will be the same as the classical SD method in Eq. (4). Then take $\mathbf{p}^{(k)}$ as the outcome of kth iteration $\hat{\mathbf{s}}^{(k+1)}$ and apply BB method:

$$\hat{\mathbf{s}}^{(k+2)} = \hat{\mathbf{s}}^{(k+1)} - t_k \nabla f(\hat{\mathbf{s}}^{(k+1)}) = \mathbf{p}^{(k)} - t_k \nabla f(\mathbf{p}^{(k)}). \tag{12}$$

At last, synthesize these two steps and substitute $\hat{\mathbf{s}}^{(k+2)}$ with $\hat{\mathbf{s}}^{(k+1)}$ to obtain the ultimate SDBB process :

$$\mathbf{p}^{(k)} = \hat{\mathbf{s}}^{(k)} - t_k \nabla f(\hat{\mathbf{s}}^{(k)}), \tag{13}$$

$$\hat{\mathbf{s}}^{(k+1)} = \mathbf{p}^{(k)} - t_k \nabla f(\mathbf{p}^{(k)}). \tag{14}$$

Organize and simplify these equations while eliminate the intermediate variable, the complete SDBB algorithm can be given in Algorithm 1.

Algorithm 1: Proposed SDBB Algorithm

Input: Matrix $\mathbf{A}$, vector $\tilde{\mathbf{y}}$
$\hat{\mathbf{s}}^{(0)} = \mathbf{0}$;
for $k = 0, \ldots, K-1$ **do**
  $\tilde{\mathbf{y}}^{(k)} = \tilde{\mathbf{y}} - \mathbf{A}\hat{\mathbf{s}}^{(k)}$;
  $t_k = \frac{\left(\tilde{\mathbf{y}}^{(k)} \cdot \tilde{\mathbf{y}}^{(k)}\right)}{\left(\mathbf{A}\tilde{\mathbf{y}}^{(k)} \cdot \tilde{\mathbf{y}}^{(k)}\right)}$;
  $\hat{\mathbf{s}}^{(k+1)} = \hat{\mathbf{s}}^{(k)} + 2t_k \tilde{\mathbf{y}}^{(k)} - t_k^2 \mathbf{A}\tilde{\mathbf{y}}^{(k)}$;
end
Output: $\hat{\mathbf{s}} = \hat{\mathbf{s}}^{(K)}$

In fact, the iteration process of the proposed algorithm presented here is so simple that there is no need to build the preprocessing unit for itself, which will lead to a hardware-friendly implementation compared with other pre-conditioned methods.

D. Proof of Convergence

As a lemma is used for many times in the convergence proof of SDBB, it is listed here first.

Lemma. *If matrix* $\mathbf{A} \in \mathbb{C}^{U\times U}$ *is positive definite and* $\mathbf{c} = \frac{(\mathbf{v}\cdot\mathbf{v})}{(\mathbf{A}\mathbf{v}\cdot\mathbf{v})}$ *(*$\mathbf{v} \in \mathbb{C}^U$ *and* $\mathbf{v} \neq \mathbf{0}$ *), then*

$$\|(\mathbf{I_U} - \mathbf{c}\mathbf{A})\mathbf{v}\|^2_{\mathbf{A}^{-1}} \leq \alpha \|\mathbf{v}\|^2_{\mathbf{A}^{-1}}, \tag{15}$$

and

$$\|(\mathbf{I_U} - \mathbf{c}\mathbf{A})(\mathbf{A}^{-1}\mathbf{v})\|^2_2 \leq \beta \|\mathbf{A}^{-1}\mathbf{v}\|_2, \tag{16}$$

where $\|\mathbf{x}\|^2_{\mathbf{A}^{-1}} = (\mathbf{A}^{-1}\mathbf{x}\cdot\mathbf{x})$*,* $\alpha = (\lambda_{max} - \lambda_{min})^2/(\lambda_{max} + \lambda_{min})^2$*,* $\beta = (\lambda^2_{max} - \lambda^2_{min})^2/(\lambda^2_{max} + \lambda^2_{min})^2$ *and* λ_{max} *and* λ_{min} *denote the largest and smallest eigenvalue of* $\mathbf{A}$ *respectively.*

Proof: Refer to [15].

After such preparation, the convergence contrast of SD and SDBB can easily be reached. For the SD method whose iteration process can be written as $\hat{\mathbf{s}}^{(k+1)} = \hat{\mathbf{s}}^{(k)} + t_k\tilde{\mathbf{y}}^{(k)}$, we have

$$\begin{aligned}\|\tilde{\mathbf{y}}^{(k+1)}\|^2_{\mathbf{A}^{-1}} &= \|\tilde{\mathbf{y}} - \mathbf{A}(\hat{\mathbf{s}}^{(k)} + t_k\tilde{\mathbf{y}}^{(k)})\|^2_{\mathbf{A}^{-1}} \\ &= \|\mathbf{y}^{(k)} - t_k\mathbf{A}\mathbf{y}^{(k)}\|^2_{\mathbf{A}^{-1}} \\ &= \|(\mathbf{I_U} - t_k\mathbf{A})\mathbf{y}^{(k)}\|^2_{\mathbf{A}^{-1}} \\ &\leq \alpha\|\mathbf{y}^{(k)}\|^2_{\mathbf{A}^{-1}}.\end{aligned} \tag{17}$$

Since the distance between the $(k-1)$th iteration result and the accurate one is $\hat{\mathbf{s}}^{(k)} - \hat{\mathbf{s}}^* = -\mathbf{A}^{-1}\mathbf{y}^{(k)}$, we have

$$\begin{aligned}\|\hat{\mathbf{s}}^{(k+1)} - \hat{\mathbf{s}}^*\|^2_2 &= \|(\mathbf{I_U} - t_k\mathbf{A})(\hat{\mathbf{s}}^{(k)} - \hat{\mathbf{s}}^*)\|^2_2 \\ &\leq \beta\|\hat{\mathbf{s}}^{(k)} - \hat{\mathbf{s}}^*\|^2_2.\end{aligned} \tag{18}$$

When it comes to the SDBB method, according to Algorithm 1, we have

$$\begin{aligned}\|\tilde{\mathbf{y}}^{(k+1)}\|^2_{\mathbf{A}^{-1}} &= \|\mathbf{A}(\hat{\mathbf{s}}^{(k)} + 2t_k\tilde{\mathbf{y}}^{(k)} - t_k^2\mathbf{A}\tilde{\mathbf{y}}^{(k)}) - \tilde{\mathbf{y}}\|^2_{\mathbf{A}^{-1}} \\ &= \|\tilde{\mathbf{y}}^{(k)} - 2t_k\mathbf{A}\tilde{\mathbf{y}}^{(k)} + t_k^2\mathbf{A}^2\tilde{\mathbf{y}}^{(k)}\|^2_{\mathbf{A}^{-1}} \\ &= \|(\mathbf{I_U} - t_k\mathbf{A})^2\tilde{\mathbf{y}}^{(k)}\|^2_{\mathbf{A}^{-1}} \\ &\leq \alpha\|(\mathbf{I_U} - t_k\mathbf{A})^2\tilde{\mathbf{y}}^{(k)}\|^2_{\mathbf{A}^{-1}} \\ &\leq \alpha^2\|\tilde{\mathbf{y}}^{(k)}\|^2_{\mathbf{A}^{-1}}.\end{aligned} \tag{19}$$

$$\begin{aligned}\|\hat{\mathbf{s}}^{(k+1)} - \hat{\mathbf{s}}^*\|^2_2 &= \|(\mathbf{I_U} - t_k\mathbf{A})^2(\hat{\mathbf{s}}^{(k)} - \hat{\mathbf{s}}^*)\|^2_2 \\ &\leq \beta^2\|\hat{\mathbf{s}}^{(k)} - \hat{\mathbf{s}}^*\|^2_2.\end{aligned} \tag{20}$$

Therefore, Eq.s (17) and (19) can ensure that the sequence of $\tilde{\mathbf{y}}^{(k)}$ linearly converges to $\mathbf{0}$ while Eq.s (18) and (20) guarantee the convergence of $\hat{\mathbf{s}}^{(k)}$ to the optimal solution $\hat{\mathbf{s}}^*$. More importantly, we have $0 < \alpha \leq 1$ and $0 < \beta \leq 1$ by definition, which means as long as the condition number [16] of matrix $\mathbf{A}$ does not equal 1, $\alpha^2 < \alpha$ and $\beta^2 < \beta$ is always set up. Theoretically, the convergence rate of SDBB can surpass that of SD and achieve pleasant results. This can be verified vividly through the numerical results provided in Section IV-A as well.

E. Complexity Analysis

In this part, we analyze the computational consumption of SDBB method in terms of the required number of complex-valued multiplications. Although this method indeed costs more computation load than SD and BB alone, it can save the memory compared with BB since it does not have to store $\tilde{\mathbf{y}}^{(k-1)}$ when computing $\hat{\mathbf{s}}^{(k+1)}$, and it needs less calculation than SPCG method [11] owing to simple initial assignment as well as the intermediate process. The result below will give a clear demonstration.

1) Compute $\tilde{\mathbf{y}}^{(k)}$*:* The calculation of $\tilde{\mathbf{y}}^{(k)}$ only includes a multiplication of a $U \times U$ matrix $\mathbf{A}$ and a $U \times 1$ vector $\hat{\mathbf{s}}^{(k)}$, thus the total contribution to the complexity measured here in this step is U^2.

2) Compute t_k*:* The step length t_k of SDBB can be computed through a multiplication of a $U \times U$ matrix $\mathbf{A}$ and a $U \times 1$ vector $\tilde{\mathbf{y}}^{(k)}$ and 2 inner products of two $U \times 1$ vectors. As a result, the complexity in this step is $U^2 + 2U$.

3) Compute $\hat{\mathbf{s}}^{(k+1)}$*:* Since the result of $\mathbf{A}\tilde{\mathbf{y}}^{(k)}$ has been obtained in Section III-E2, according to the expression of $\hat{\mathbf{s}}^{(k+1)}$ in Algorithm 1, there are only 2 multiplications of a scalar and a $U \times 1$ vector totally, which contains the complexity of $2U$.

Therefore, the overall complexity of SDBB method is given by $K(2U^2 + 4U)$ where K is the iteration number.

IV. Numerical Results

Having analyzed the convergence and complexity properties of SDBB in Section III minutely, this section offers simulation results and comparison of the proposed method and other state-of-the-art ones for a better understanding. Both BER performance and complexity contrast reflect the superiority of SDBB method, showing its ability in balancing these two important factors.

A. BER Performance

As explained before, in order to improve the stability of SD method, BB is introduced to perfect SD. In order to solve the slow convergence rate problem when loading factor is relatively large, as well as to tackle with high storage issue caused by the memory used for last iteration and iteration before the last, SDBB is proposed to optimize BB.

Both Fig. 2 and Fig. 3 display the BER performance of SDBB method, classical SD algorithm and the SPCG scheme. Direct detection method using Cholesky factorization is still shown as a reference. The antennas at the BS are fixed at 128 while the number of users changes from 16 (Fig. 2) to 32 (Fig. 3). To reduce the calculation difficulties of iterative methods, the iteration number is chosen to equal 2, 3 and 4, respectively.

In Fig. 2, the iteration result of SDBB at $k = 2$ outperforms that of SD at $k = 4$ which fits the derivation of SDBB that one iteration of SDBB is equivalent to one in SD plus one in BB. The fast convergence rate also confirms the proof analysis in Section III-D. As for the contrast with SPCG method, the performance of SDBB is approximate to that of SPCG even with one less iteration. And the result of SDBB with 4

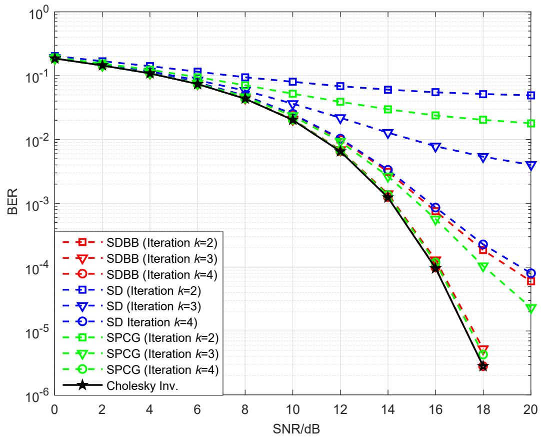

Fig. 2. BER performance of different detection methods with $B = 128$ and $U = 16$.

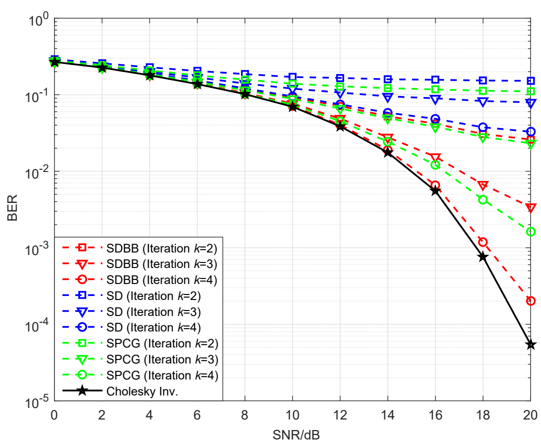

Fig. 3. BER performance of different detection methods with $B = 128$ and $U = 32$.

iteration shows negligible difference with the ideal Cholesky factorization. In other words, BER performance of SDBB in this condition is far beyond the others.

With an increase in the users' number, all methods in Fig. 3 suffer performance loss more or less due to a larger loading factor. However, unlike the SD method that hardly converges under such circumstances, SDBB owns quite stable performance. As can be seen, it surpasses SPCG with over 2 dB performance improvement at BER of 10^{-3}. Furthermore, the convergence gap between SDBB and the optimal method is greatly decreased compared to that between BB and SD indicated in Fig. 1.

More specifically, if the number of users is fixed at $U = 16$ while the antennas at BS are varied causing a dynamic loading factor ρ, Fig. 4 will provide a comprehensive simulation result for all methods. First, it is clear that all curves show upward trends with a growing ρ revealing less satisfactory performance. Whereas, the result of SDBB is always leading. Above all, it only needs half of the iteration time of SD to reach a better outcome and performs very closely to Cholesky factorization as long as ρ is smaller than 0.2.

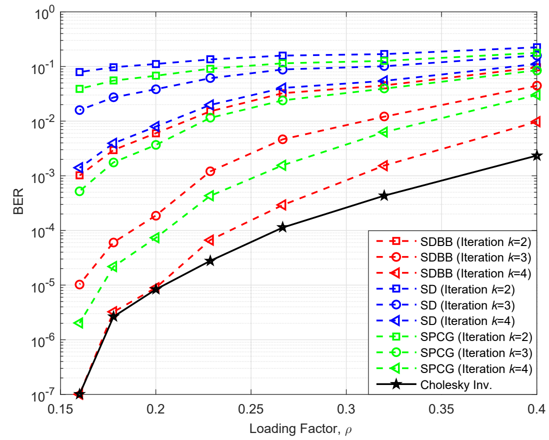

Fig. 4. BER performance of different detection methods with $U = 16$, SNR $= 20$ dB and various loading factors (ρ).

All in all, from all these three graphs (Fig.s 2, 3, and 4), we can easily conclude that SDBB method converges quickly and reliably. Whatever the system configuration is and the loading factor is, it always takes the lead in comparison with other methods, embodying its wider application.

B. *Computational Complexity*

In this part, the computational comparison between proposed SDBB method, SD method, SPCG method and Cholesky factorization is made. Such contrast will better highlight the effectiveness of SDBB method.

TABLE I. NUMBER OF COMPLEX-VALUED MULTIPLICATIONS.

Algorithm	**Computational Complexity**
Cholesky Inv.	$\frac{5}{6}U^3 + \frac{3}{4}U^2 + \frac{4}{3}U$
Classical SD	$K(U^2 + 4U)$
SPCG [11]	$2KU^2 + (7K + 1)U$
Proposed SDBB	$K(2U^2 + 4U)$

In Table I., the computational complexity of each method is listed in detail. Since the expression of SDBB does not include U^3, it can save a lot from the direct inversion method. The consumption of proposed method is smaller than that of SPCG by contrast as well. It can be deduced that SDBB will save nearly 10% complexity compared with SPCG when $U = 16$ and iteration number equals 4.

Fig. 5 exhibits the numerical results of computation cost of different detection methods except Cholesky factorization since the disparity between it and others is obvious. Here, the antennas number at BS $B = 128$ and SNR $= 10$ dB. It is manifested that SD costs the least calculations without extra operations while SPCG the most with the same iteration number. However, if the BER performance of figures in Section IV-A is taken into consideration, the performance of $k = 4$ for SD will be inferior to that of $k = 2$ for SDBB, and SD's corresponding complexity load will be heavier than SDBB's. In this way, SDBB method reaches the tradeoff between complexity and performance best among all these methods. This could be demonstrated more clearly by Fig. 6.

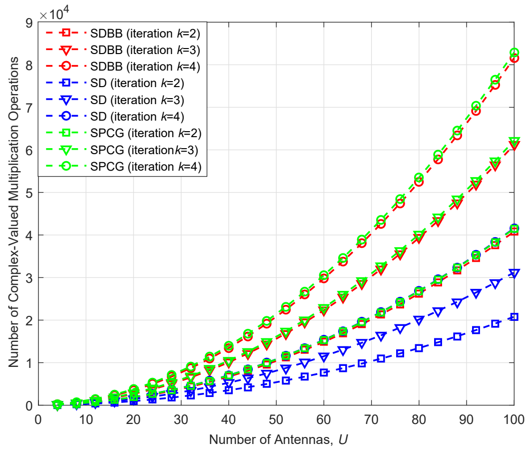

Fig. 5. Complexity comparison of different detection methods with $B = 128$ and SNR $= 10$ dB.

C. Performance/Complexity Tradeoff

In Fig. 6, the complexity is represented by the required number of complex-valued multiplications while the BER is used to denote the performance. The number of antennas at the BS and UTs equal 128 and 32 respectively, and SNR is fixed as 20 dB. All curves are drawn with the changing iteration number. It is obvious that the closer to the left corner, the better the outcome with more accurate performance and less complexity. In this way, there is no doubt that SDBB is the best, reconfirming its significance.

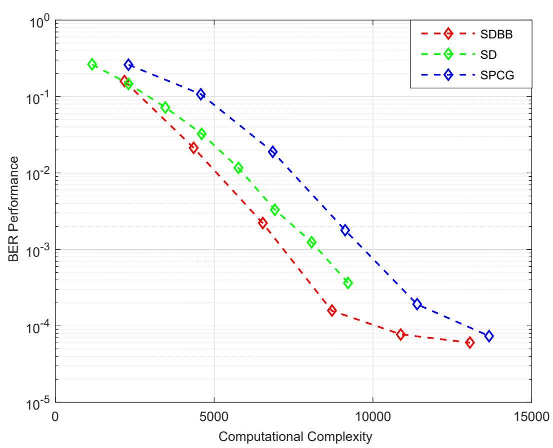

Fig. 6. Performance/complexity tradeoff with $B = 128$ and $U = 32$.

V. Conclusion

In this paper, an iterative detection method named S-DBB based on Barzilai-Borwein method has been proposed for massive MIMO uplink detection. Explanation about its derivation as well as the relationship with BB and SD are given in detail. Convergence advantages have been proved through mathematical approaches. Meanwhile, theoretical and numerical results have confirmed that this algorithm can reach the best tradeoff between BER performance and computational complexity. Further work will be directed towards the hardware architecture and implementation of SDBB method.

Acknowledgment

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. G. Larsson, O. Edfors, F. Tufvesson *et al.*, "Massive MIMO for next generation wireless systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 186–195, 2014.

[2] F. Rusek, D. Persson, B. K. Lau *et al.*, "Scaling up MIMO: Opportunities and challenges with very large arrays," *IEEE Signal Process. Mag.*, vol. 30, no. 1, pp. 40–60, 2013.

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

[4] H. Prabhu, J. Rodrigues, O. Edfors *et al.*, "Approximative matrix inverse computations for very-large MIMO and applications to linear pre-coding systems," in *Proc. IEEE Wireless Commun. and Networking Conf. (WCNC)*, 2013, pp. 2710–2715.

[5] Z. Wu, C. Zhang, Y. Xue *et al.*, "Efficient architecture for soft-output massive MIMO detection with Gauss-Seidel method," in *Proc. IEEE Intl. Symp. on Cir. and Syst. (ISCAS), 2016*, pp. 1886–1889.

[6] Y. Xue, C. Zhang, S. Zhang *et al.*, "Steepest descent method based soft-output detection for massive MIMO uplink," in *Proc. IEEE Intl. Workshop on Signal Process. Syst. (SiPS), 2016*, pp. 273–278.

[7] Y. Xue, Z. Wu, J. Yang *et al.*, "Adaptive Preconditioned Iterative Linear Detection and Architecture for Massive MU-MIMO Uplink," *J. Signal Process. Syst.*, pp. 1–15, 2017.

[8] M. Benzi, "Preconditioning techniques for large linear systems: a survey," *J. Comput. Phys.*, vol. 182, no. 2, pp. 418–477, 2002.

[9] R. Fletcher, "On the barzilai-borwein method," in *Optim. and Control with Appl.* Springer, 2005, pp. 235–256.

[10] M. Raydan and B. F. Svaiter, "Relaxed steepest descent and Cauchy-Barzilai-Borwein method," *Comput. Optim. and Appl.*, vol. 21, no. 2, pp. 155–167, 2002.

[11] J. Jin, Y. Xue, Y.-L. Ueng *et al.*, "A split pre-conditioned conjugate gradient method for massive MIMO detection," in *Proc. IEEE Intl. Workshop on Signal Process. Syst. (SiPS), 2017*, pp. 1–6.

[12] A. Sibille, C. Oestges, and A. Zanella, *MIMO: from theory to implementation*. Academic Press, 2010.

[13] E. Biglieri, R. Calderbank, A. Constantinides *et al.*, *MIMO Wireless Communications*. Cambridge University Press, 2007.

[14] M. Raydan *et al.*, "Convergence properties of the barzilai and borwein gradient method," Ph.D. dissertation, Rice University, 1991.

[15] J. Zhuang and Z. Peng, "A modified Cauchy-Barzilai-Borwein method," *J. on Numer. Methods and Comput. Appl.*, vol. 37, no. 3, pp. 186–198, 2016.

[16] Y. Saad, *Iterative methods for sparse linear systems*. SIAM, 2003.