electronics

MDPI

Article

Deep-Unfolded Tikhonov-Regularized Conjugate Gradient Algorithm for MIMO Detection

Sümeye Nur Karahan * and Aykut Kalaycıoğlu

Department of Electrical-Electronics Engineering, Ankara University, Ankara 06830, Turkey
* Correspondence: snkarahan@ankara.edu.tr

Abstract: In addressing the multifaceted problem of multiple-input multiple-output (MIMO) detection in wireless communication systems, this work highlights the pressing need for enhanced detection reliability under variable channel conditions and MIMO antenna configurations. We propose a novel method that sets a new standard for deep unfolding in MIMO detection by integrating the iterative conjugate gradient method with Tikhonov regularization, combining the adaptability of modern deep learning techniques with the robustness of classical regularization. Unlike conventional techniques, our strategy treats the Tikhonov regularization parameter, as well as the step size and search direction coefficients of the conjugate gradient (CG) method, as trainable parameters within the deep learning framework. This enables dynamic adjustments based on varying channel conditions and MIMO antenna configurations. Detection performance is significantly improved by the proposed approach across a range of MIMO configurations and channel conditions, consistently achieving lower bit error rate (BER) and normalized minimum mean square error (NMSE) compared to well-known techniques like DetNet and CG. The proposed method has superior performance over CG and other model-based methods, especially with a small number of iterations. Consequently, the simulation results demonstrate the flexibility of the proposed approach, making it a viable choice for MIMO systems with a range of antenna configurations, modulation orders, and different channel conditions.

Keywords: MIMO detection; Tikhonov regularization; conjugate gradient; deep learning; wireless communication

Citation: Karahan, S.N.; Kalaycıoğlu, A. Deep-Unfolded Tikhonov-Regularized Conjugate Gradient Algorithm for MIMO Detection. *Electronics* **2024**, *13*, 3945. https://doi.org/10.3390/electronics13193945

Academic Editors: Sherali Zeadally, Wenyu Zhang and Tianwei Hou

Received: 31 July 2024
Revised: 3 October 2024
Accepted: 4 October 2024
Published: 7 October 2024

1. Introduction

Multiple-input multiple-output (MIMO) systems are essential for enhancing spectral efficiency in modern wireless networks. Spatial multiplexing in MIMO systems allows for simultaneous transmission of multiple information streams across different antennas, setting it apart from diversity systems that focus on reliability by transmitting identical information. Achieving higher data rates through spatial multiplexing presents significant challenges at the receiver, particularly in detection complexity and efficiency, which have been the subject of research for over five decades, driving the evolution of MIMO detection methodologies [1,2]. The core of MIMO detection involves decoding transmitted symbols using known channel characteristics. While maximum likelihood (ML) detection minimizes bit error rate (BER) optimally, it is computationally impractical for physical implementations involving a large number of antennas. Therefore, alternative methods like sphere decoding (SD), zero forcing (ZF), and linear minimum mean squared error (LMMSE) have been developed for near-optimal performance with lower complexity [2]. There are also methods such as Neumann series expansion (NSE), Gauss–Seidel (GS), and conjugate gradient (CG) which utilize iterative matrix-vector multiplication to reduce system complexity [3–6]. Non-linear MIMO detectors are useful in reducing interference for subsequent signals, though errors in interference signals can degrade detection efficacy [7]. Advanced approaches, such as the Belief Propagation (BP) algorithm [8], are effective for high number of antennas and low inter-channel correlation, but may introduce delays and degrade performance in

fading channels due to their iterative nature. Therefore, developing a detection strategy that achieves high reliability without requiring excessive amounts of decoding time is one of the major challenges in MIMO systems [9].

In addition to conventional methods discussed above, recent studies have explored both model-driven and data-driven deep learning approaches [10]. Model-driven techniques enhance iterative algorithms like orthogonal approximate message passing (OAMP) [11], alternating direction method of multipliers (ADMM) [12], Viterbi [13], and expectation propagation [14]. Data-driven solutions use deep learning architectures such as autoencoders [15], deep neural networks (DNNs), and convolutional neural networks (CNNs) [16] for high detection accuracy. These deep learning (DL)-based MIMO detection methods outperform traditional detectors under various channel conditions. Although there are studies that discuss model-driven and data-driven approaches either separately or together, the increasing amount of data in new communication systems increasingly favors model-driven methods. Unsupervised deep learning techniques, such as autoencoders, can be used to learn the entire system for MIMO detection, as demonstrated in data-driven MIMO detection [15]. In addition, DetNet uses a model-driven approach to detection using iterative projected gradient descent [17]. Data-driven methods for MIMO detection in fixed-channel scenarios utilize CNNs and DNNs [16]. Another approach uses conventional deep learning network topologies for signal detection in MIMO systems with erroneous channels [18], while another study employs neural networks to identify decision zones for multi-user MIMO systems [19].

Deep unfolding (DU) algorithms, also known as model-driven deep learning methods, constitute a transformative approach that combines classical iterative methods with the adaptive capabilities of neural networks, and are a common solution for MIMO detection [20,21]. By structuring known iterative algorithms into neural network layers, each iteration treated is treated as a layer [22] that allows parameters to be trained via backpropagation rather than updated deterministically in a traditional way. This leads to improved solutions by incorporating additional or modified parameters to capture features that classical methods may miss [23]. Unlike traditional methods, the network can generalize to new inputs after training on different data sets, eliminating the need to recalculate parameters for each system change. This approach builds neural network layers over multiple iterations using advanced learning techniques to achieve unprecedented results [20,24–26]. Various deep unfolding-based algorithms for MIMO channel detection are reported, including trainable projected gradient detectors [27] and the conjugate gradient descent technique [28,29], with other alternative approaches in [11,12,30]. Despite these developments, there is still a significant research gap in improving these approaches, especially when it comes to dealing with the complexity and variability of harsh situations. This emphasizes the need for further advances in this area and the usefulness of the proposed approach in improving MIMO detection technology.

The deep unfolding approach also offers significant advantages in computational efficiency and hardware implementation [26]. This method is particularly beneficial for physical applications with hardware constraints and operational efficiency requirements. By predetermining the neural network's structure to mimic specific algorithm iterations, it reduces the need for extensive training data and computational resources, addressing major challenges faced by traditional DNNs.

This study presents a significant advance in the field of MIMO signal detection by introducing a unique detection strategy that combines Tikhonov regularization and CG method with deep unfolding. Detection of transmitted symbols over a multipath fading channel is considered as an ill-conditioned problem [31] that may result in slower or even a non-existent convergence. Using the matrix L as a regularization term enhances the detection process in CG-based detection methods by resolving the input signal while effectively suppressing the degrading effects. Thus, the regularization allows for significant improvements in detection performance over conventional methods for different channel conditions and antenna layouts.

The main contributions of this study are summarized below:

1. To the best knowledge of the authors, this is the first study where Tikhonov regularization is integrated with the conjugate gradient method for MIMO detection in a deep learning-based approach.
2. Performance of the proposed method has been compared with both iterative and model-driven techniques for different channel models such as Rayleigh, Kronecker, Tapped Delay Line A (TDL-A), and TDL-E.

The remaining sections of this study are organized as follows: Section 2 presents the relevant work and subjects. Section 3 provides a thorough explanation of the proposed approach. A comprehensive analysis of computational complexity is provided in Section 4. The simulation results are given in Sections 5 and 6, and conclusions are drawn and suggestions for further work are explored.

2. Materials and Methods

2.1. MIMO System Model

In this study, we investigate a MIMO system utilizing spatial multiplexing, wherein the receiver antennas concurrently receive symbols transmitted from the transmitter. The system comprises N_r receiving antennas and N_t transmitting antennas as shown in Figure 1. The received symbols, denoted as y_{N_r}, at the receiver side can be expressed as follows:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{N_r} \end{bmatrix} = \begin{bmatrix} h_{1,1} & \dots & h_{1,N_t} \\ \vdots & \dots & \vdots \\ h_{N_r,1} & \dots & h_{N_r,N_t} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{N_t} \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2 \\ \vdots \\ n_{N_r} \end{bmatrix} \tag{1}$$

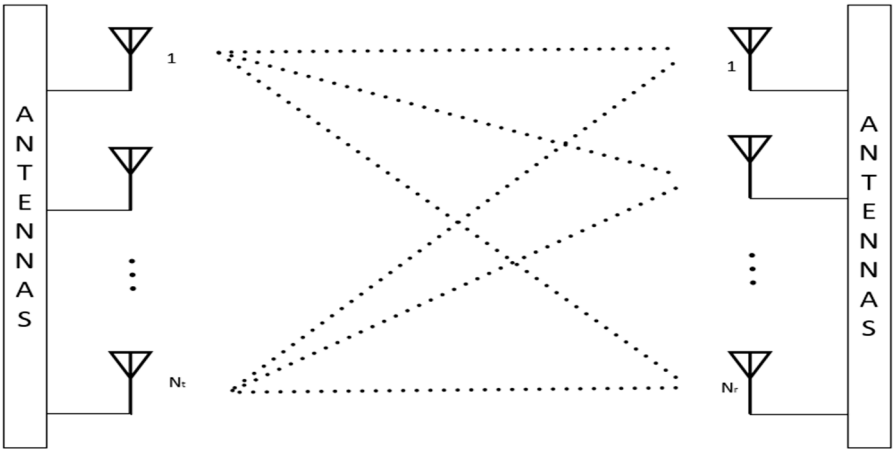

Figure 1. Simplified block diagram of a MIMO system.

In aforementioned Equation (1), h_{N_r,N_t} denotes the entries of the channel matrix corresponding to the communication link between the N_t-th transmitter antenna and the N_r-th receiver antenna. The term n_{N_r} signifies the additive white Gaussian noise (AWGN) present at the N_r-th receiver antenna, characterized by zero mean and variance σ^2. x_{N_t}, y_{N_r}, n_{N_r} represent complex-valued numbers, and h_{N_r,N_t} signifies a complex-valued channel which is assumed to exhibit flat Rayleigh fading, with the channel entries, h_{N_r,N_t}, being independently and identically distributed (i.i.d.) with zero mean and unit variance.

The matrix H and the vectors y, x, n have complex values, due to the necessity of using real numbers in the deep learning structure, and the MIMO channel model is expressed as follows for the simulation environment within the scope of the study:

$$\begin{cases} y_r = H_r x_r + n_r \\ y_r = \begin{bmatrix} Re(y) \\ Im(y) \end{bmatrix} \in R^{2N_r x 1} \\ x_r = \begin{bmatrix} Re(x) \\ Im(x) \end{bmatrix} \in R^{2N_t x 1} \\ H_r = \begin{bmatrix} Re(H) & -Im(H) \\ Im(H) & Re(H) \end{bmatrix} \in R^{2N_r x 2N_t} \end{cases} \quad (2)$$

The MIMO system's simplified block diagram is shown in Figure 1. Multiple antennas are used in this system, both at the transmitter and receiver ends.

The numerous signal routes between the antennas are depicted in Figure 1 by connecting each transmitting antenna to each receiving antenna. Through the use of spatial diversity and the ability to transmit many data streams at once, this arrangement improves the capacity and dependability of the system. The intricate interaction and signal propagation in a MIMO system are highlighted by the dotted lines.

2.2. MIMO Channel Model

MIMO channels are critical to today's modern communication systems. These systems can significantly improve transmission rate, reliability, and spectrum efficiency by using multiple antennas at both the transmitter and receiver. Each element in the matrix characterizing the MIMO channel represents the channel coefficient between a given pair of transmitting and receiving antennas. The effects of multipath propagation are well captured by this matrix, which is important for understanding and improving the functionality of advanced communication networks. A Rayleigh channel in MIMO systems is a model in which Rayleigh fading affects the channel coefficients. This phenomenon happens when there is no direct line-of-sight path and multipath propagation, causing changes in the signal's magnitude. Usually, the model for each element of the MIMO channel matrix H is an independent, identically distributed (i.i.d.) complex Gaussian random variable with unit variance and zero mean. A mathematical framework used in MIMO systems to make it easier to characterize spatial correlations between antennas at the transmitter and receiver is called the Kronecker channel model [32]. According to the model, transmitter and receiver correlation matrices at each end correspond to separable correlation structures that can be formed from the overall channel matrix. The mathematical expression for this model is as follows:

$$H = R_r^{1/2} H_w R_t^{1/2} \quad (3)$$

where $R_r^{1/2}$ and $R_t^{1/2}$ are the Cholesky decompositions of the receiver and transmitter correlation matrices, respectively, and H_w is an uncorrelated Rayleigh fading matrix.

In fifth-generation (5G) cellular systems, TDL (Tapped Delay Line) channel models—such as TDL-A and TDL-E—defined by 3GPP simulate multipath propagation [33]. The TDL-A channel model uses a Tapped Delay Line structure to simulate realistic time-varying and frequency-selective scenarios. The taps represent discrete paths with unique delays and power levels. The TDL-A model, which is widely used to simulate urban environments, precalculates the channel impulse response to facilitate the assessment of system performance in terms of signal fading and inter-symbol interference (ISI). The channel model holds great significance in the advancement and assessment of 5G technologies, including massive MIMO and beamforming. It offers valuable perspectives for enhancing communication protocols and algorithms, hence guaranteeing reliable performance in practical situations. As such, the TDL-A paradigm plays an important role in the design and implementation of high-performance 5G networks. On the other hand, the TDL-E channel model is particularly well known for its severe multipath conditions, which represent difficult time-varying

and frequency-selective scenarios. These conditions include extended delay spreads and large Doppler shifts. The TDL-E model provides a realistic simulation of harsh urban and suburban environments.

2.3. A Summary of MIMO Detection Methods

MIMO detection methods are simply classified into two categories as linear and non-linear methods, where non-linear methods consist of non-iterative, iterative, and deep learning approaches. Additionally, deep learning-based methods encompass both data-driven and deep-unfolded methodologies also called as model-driven. Within this section, we will only discuss the detection methods that are in the focus of this study, CG and learned CG methods, which are classified as iterative and model-driven deep learning methods, respectively. Details of the other well-known methods such as ML, MMSE, Sphere Decoder, and DetNet can be found in [16,34–36]. The conjugate gradient (CG) method, introduced by Hestenes and Stiefel in 1952 [37], is an iterative algorithm for solving systems of linear equations. CG as an iterative method is particularly useful in situations where direct methods are infeasible due to the large dimensions of the matrices involved. The CG method, summarized in Algorithm 1, provides an efficient and robust solution for signal detection in MIMO systems.

Algorithm 1: MIMO Detection with CG

Inputs: $\boldsymbol{y}, \boldsymbol{H}, \delta^2$
Output: Transmitted signal vector estimation $\hat{\mathbf{s}}$
1: *Initialization:*
$\boldsymbol{A} = \boldsymbol{H}^{\boldsymbol{H}}\boldsymbol{H} + \delta^2 \boldsymbol{I}_{N_T}$, $\boldsymbol{b} = \boldsymbol{H}^{\boldsymbol{H}}\boldsymbol{y}$
$\boldsymbol{x_0} = \mathbf{0}$, $\boldsymbol{r_0} = \boldsymbol{b}$, $\boldsymbol{p_0} = \boldsymbol{r_0}$
2: *for* $i = 0, \ldots, K$ *do*
3: $\alpha_i = \boldsymbol{r}_i^T \boldsymbol{r}_i / \boldsymbol{p}_i^T \boldsymbol{A} \boldsymbol{p}_i$
4: $\boldsymbol{x}_{i+1} = \boldsymbol{x}_i + \boldsymbol{\alpha}_i \boldsymbol{p}_i$
5: $\boldsymbol{r}_{i+1} = \boldsymbol{r}_i - \boldsymbol{\alpha}_i \boldsymbol{A} \boldsymbol{p}_i$
6: $\boldsymbol{\beta}_i = \boldsymbol{r}_{i+1}^T \boldsymbol{r}_{i+1} / \boldsymbol{r}_i^T \boldsymbol{r}_i$
7: $\boldsymbol{p}_{i+1} = \boldsymbol{r}_{i+1} - \boldsymbol{\beta}_i \boldsymbol{p}_i$
8: *end for*
9: *return* $\hat{\mathbf{s}} = \boldsymbol{x}_{i+1}$

The CG method starts with the provided inputs, the channel matrix H, the noise variance δ^2, and the received signal y, and the algorithm initializes the relevant matrix and vectors. The search direction vector p_i, the residual vector r_i, and the solution vector x_i are set first. At each iteration, the solution vector is updated by calculating the step size α, which indicates how far to go in the current search direction. The new solution estimate is then reflected in the residual vector. The algorithm determines a coefficient, β, which modifies the search direction to maintain efficiency in subsequent iterations. This iterative process continues until the desired number of iterations is reached or the solution is sufficiently accurate.

Deep learning can also be applied to MIMO detection methods. Thus, employing the deep unfolding technique on iterative detection procedures provides faster convergence and higher generalization within modeled behaviors and requires less training data in iterative detection procedures. Incorporating deep learning techniques to adaptively find the optimal parameters during iterations ensures that the learned conjugate gradient method (LCG) [28] has higher performance over the traditional CG method. The LCG method, shown in Figure 2, alters the conjugate gradient algorithm by making the step size and search direction coefficient parameters, denoted by α and β, respectively, modifiable through training instead of mathematically calculating them on the relevant data sets. The method can now adaptively adjust its step sizes and directions, which enhances detection performance in MIMO systems and facilitates more effective convergence. The limitations of fixed-parameter approaches can be overcome by LCG through the utilization of data-

driven insights, thereby providing a robust foundation for the management of the complex and variable conditions of wireless communication channels. As explained in detail in [28], two types of LCG approaches are introduced, namely scalar, LCG-S, and vector, LCG-V. Scalar trainable parameters α and β are used by LCG-S. By eliminating the requirement for matrix-vector multiplications and divisions, these scalar parameters simplify computations. Alternatively, vector trainable parameters α^i and β^i are used by LCG-V. These vector step sizes improve detection performance by allowing LCG-V to learn and adapt more effectively to the data characteristics. It is highlighted in the work [28] that LCG-V requires storing more parameters, even though its computational complexity is almost the same as that of LCG-S for its operations.

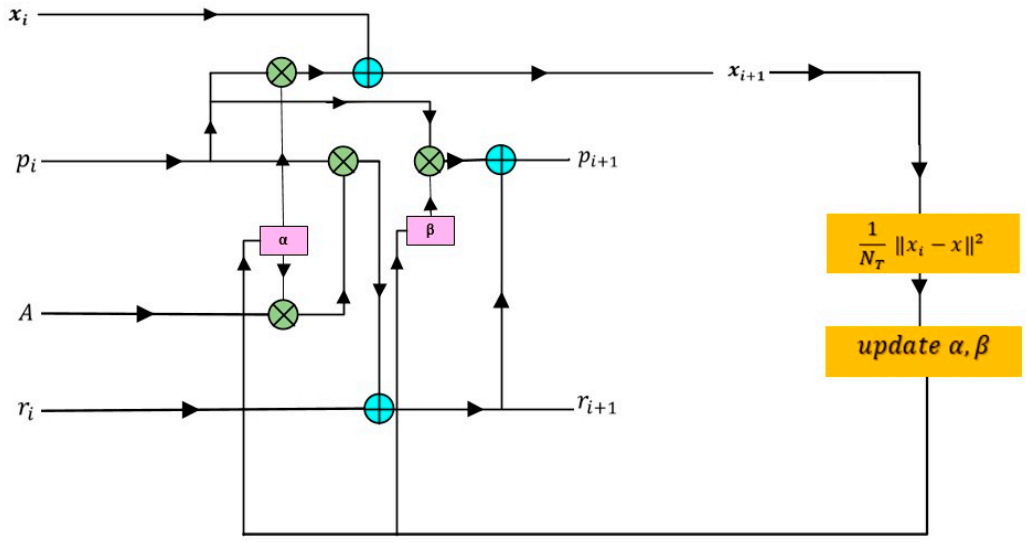

Figure 2. LCG algorithm in i^{th} layer.

3. Proposed Method

Tikhonov regularization, also known as ridge regularization, is a method which inserts a regularization term to the solution for solving ill-posed problems or preventing overfitting in linear regression. Penalizing the magnitude of the coefficients in the loss function we are attempting to minimize is the core idea behind Tikhonov regularization. In its simplest form, Tikhonov regularization penalizes solutions with large magnitudes by adding a regularization term to the objective function to be reduced [38]. The solution is more resilient to varying conditions thanks to this regularization term. Several applications in engineering and physics lead to the following types of linear least-squares problems:

$$min\ x_{i+1} \in R^{d_2}\ \|Ax_{i+1} - b^{\delta}\|, \quad A \epsilon R^{d_1 x d_2}, \quad b^{\delta} \epsilon R^{d_1} \tag{4}$$

Matrix A is of ill-determined rank, that is, its singular values gradually decay to zero without a noticeable gap, and the measured data, tainted by an unknown error $e\ \epsilon\ R^{d_1}$ of norm constrained by $\delta > 0$, are represented by b^{δ}. The matrix $A\ \epsilon\ R^{d_1 x d_2}$ has d_1 rows representing the number of measurements and d_2 columns corresponding to the number of unknowns or variables, which are the components of the vector x_{i+1}. Least-squares problems, also referred to as discrete ill-posed problems, require this kind of matrix. An exact approximation of the minimal norm solution $x^+ = A^+ b$ for the error-free least-squares problem associated with (4) is sought after. A^+ represents the pseudoinverse of Moore–Penrose in this case. Due to the error in b^{δ} and the clustering of A's singular values near the origin, the solution $A^+ b^{\delta}$ of (4) is usually not a reasonable approximation of x^+. Changing the minimization problem to a nearby problem whose solution is less vulnerable

to the error in b^{δ} is one method to overcome this problem (4). This substitution is sometimes referred to as regularization.

$$min\ x_{i+1} \in R^{d_2} \left\{ \|Ax_{i+1} - b^{\delta}\|^2 + \lambda \|x_{i+1} - x_i\|^2 \right\} \quad (5)$$

The regularization parameter $\lambda > 0$ in this case controls how sensitive the solution of (5) is to the error e in b^{δ} as well as how near the solution is to the target vector x^{+}. It is generally known that by substituting an appropriate regularization matrix for the Tikhonov minimization problem (5), it is frequently possible to increase the quality of the x^{+} approximation determined by Tikhonov regularization L.

$$min\ x_{i+1} \in R^{d_2} \left\{ \|Ax_{i+1} - b^{\delta}\|^2 + \lambda \|L(x_{i+1} - x_i)\|^2 \right\} \quad (6)$$

where the regularization matrix $L\ \epsilon\ R^{d_3 x d_2}$ typically has dimensions such that d_3 may vary depending on the specific regularization approach, although it is often equal to d_2 in the case of square matrices.

The regularization matrix L encodes the extra restrictions or previous knowledge about the solution x in (6), and λ is a regularization parameter that governs the trade-off between fitting the data and meeting the regularization term. The process of choosing the regularization matrix L and regularization parameter λ, which requires domain expertise and careful tuning, is critical to obtaining precise and reliable solutions to ill-posed inverse problems. The matrix L is typically an $N \times N$ matrix, where N represents the number of unknown variables, whereas parameter λ is usually a constant scalar. Basically, there are two main approaches for selecting the regularization matrix L. The first one is the non-derivative method which forms L as $L = DV^T$. Here, V^T is derived from the singular value decomposition (SVD) of matrix A, and D is a diagonal matrix of singular values. The latter one is the derivative-based method that uses first- or second-order derivative operators to construct L. An interested reader should refer to [38,39] for the details.

3.1. Deep-Unfolded Tikhonov-Regularized Conjugate Gradient Algorithm

We utilize Tikhonov regularization in the CG algorithm with deep unfolding to improve performance on different types of channels, addressing issues such as noise sensitivity and ensuring convergence in high-dimensional MIMO systems. The CG technique effectively tackles the complexity of high-dimensional MIMO systems, improving performance and signal estimates when paired with Tikhonov regularization. When incorporated into model-driven MIMO detection frameworks such as LCG, Tikhonov regularization appears to be an effective method for improving robustness and performance. This method involves adding a term L to the system matrix in the context of MIMO detection. In this work, the proposed method is dynamically adjusting the detection strength by the model during training thanks to the trainable parameter regularization matrix L, alpha, and beta. It is noteworthy that, unlike the original Tikhonov regularization approach shown in (6), which represents the parameters λ and L individually, our proposed method combines them into a single matrix L, which is the product of the scalar λ and the matrix L. By treating the multiplication of these elements as a single matrix, the network streamlines the computation and improves the flexibility of the model for different channel conditions. The pseudocode of the proposed method is shown in Algorithm 2.

The DU-TCG algorithm enhances the CG method with deep unfolding by incorporating Tikhonov regularization and an optimization step using trainable parameters α, β, and L. These parameters are optimized with an Adam optimizer [40,41] during each iteration to minimize the loss function, playing crucial roles in updating the residual, refining the search direction, and applying regularization. The algorithm begins by initializing matrices and vectors as in the standard CG method. The parameters α and β are initially set to 0, while the L matrix is initialized as an identity matrix of size $N_T \times N_T$. In each iteration, the residual r_i is updated with α, and the search direction p_i is refined with β. A

new solution estimate $\widetilde{x}$ is computed using Tikhonov regularization matrix L, with a loss function measuring the difference between $\widetilde{x}$ and x_i. The process iteratively refines the solution by updating trainable parameters until the final transmitted signal vector estimate $\hat{s}$ is obtained. The iterations of the Tikhonov-regularized CG algorithm are unrolled to create the proposed Deep-Unfolded Tikhonov-Regularized Conjugate Gradient (DU-TCG), a deep learning architecture in which each layer is associated with an algorithm iteration as shown in Figure 3.

Algorithm 2: Deep-Unfolded Tikhonov-Regularized Conjugate Gradient Algorithm

Inputs: $\boldsymbol{y}$, $\boldsymbol{H}$, $\boldsymbol{\delta^2}$
Output: Transmitted signal vector estimation $\hat{\boldsymbol{s}}$
1: *Initialization:*
$\boldsymbol{A} = \boldsymbol{H^H H} + \delta^2 \boldsymbol{I}_{N_T}$, $\boldsymbol{b} = \boldsymbol{H^H y}$
$\boldsymbol{x_0} = \boldsymbol{0}$, $\boldsymbol{r_0} = \boldsymbol{b}$, $\boldsymbol{p_0} = \boldsymbol{r_0}$
2: *for* $i = 0, \ldots, K$ *do*
3: $\boldsymbol{r_{i+1}} = \boldsymbol{r_i} - \boldsymbol{\alpha_i A p_i}$
4: $\boldsymbol{p_{i+1}} = \boldsymbol{r_{i+1}} - \boldsymbol{\beta_i p_i}$
5: $\widetilde{\boldsymbol{x}} = \boldsymbol{x_i} + \left[\left(\boldsymbol{H^H H} + \boldsymbol{L} \right)^{-1} * \left(\boldsymbol{H^H} * \left(\boldsymbol{y} - \boldsymbol{H x_i} \right) \right) \right]$
6: $\boldsymbol{loss} = \frac{1}{N_T} \| \widetilde{\boldsymbol{x}} - \boldsymbol{x} \|^2$
7: *train:* ***AdamOptimizer(minimize(loss), parameters{α,β,L})***
8: $\boldsymbol{x_{i+1}} = \widetilde{\boldsymbol{x}} + \boldsymbol{\alpha_i p_i}$
9: *end for*
10: *return* $\hat{\boldsymbol{s}} = \boldsymbol{x_{i+1}}$

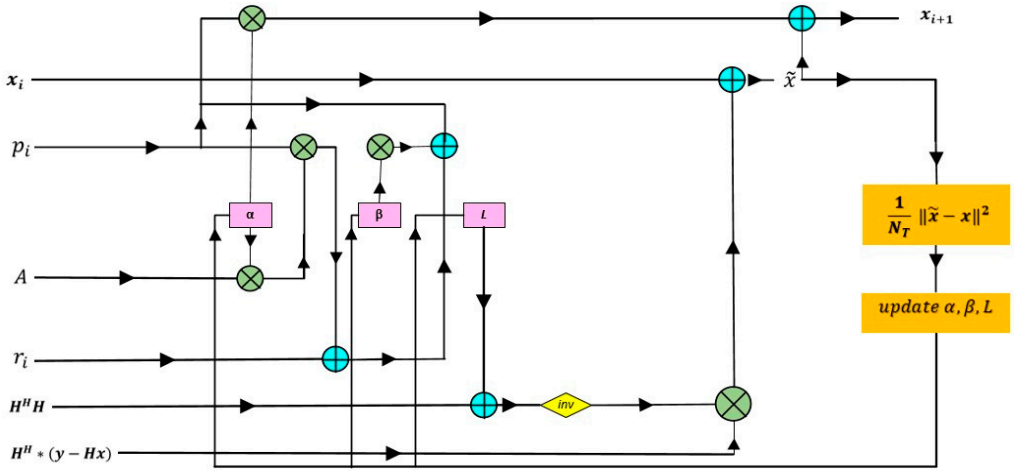

Figure 3. DU-TCG algorithm in i^{th} layer.

Using this method, the model can be trained to find the best parameters for improved performance in a range of channel situations. Along with the α and β parameters, the DU-TCG algorithm adds a new trainable parameter called the regularization, L, which is in matrix form. This combination allows for dynamic modification of the regularization strength during training and simplifies computation as it is processed by the network as a single matrix.

To guide the training process, DU-TCG's loss function is designed to assess the difference between the transmitted signal and the predicted output. For this purpose, the mean squared error (MSE) loss function is used as shown in Algorithm 2. The mean squared difference between the expected and actual values is quantified by the MSE loss function, giving the network a specific target to minimize during training. The network increases the resilience and accuracy of detection by iteratively updating the parameters α, β, and L to minimize this loss. The i^{th} iteration of the CG algorithm corresponds to the i^{th} layer

of DU-TCG detector. The layer-dependent trainable parameters of the DU-TCG detector is represented with $\theta^i = \{\alpha^i, \beta^i, L^i\}$ in the i-th layer of the network and α and β step size, search direction coefficient, and L regularization matrix are learnt from training samples $\{(y, H), x\}_{t=1}^{N_T}$ by minimizing mean square error as shown:

$$L_{DU-TCG}^{K}\left(\theta^1, \ldots, \theta^K\right) = \frac{1}{N_T}\sum_{t=1}^{N_t} \left\| x - \hat{x}^K\left(y, H; \theta^1, \ldots, \theta^K\right)\right\|^2 \quad (7)$$

K denotes the number of layers, and $\hat{x}^K\left(y, H; \theta^1, \ldots, \theta^K\right)$ denotes the output of DU-TCG with y and H inputs.

3.2. Training Details

The TensorFlow (version 2.6.2) library with the Adam optimizer was used to create the proposed DU-TCG network in Python (version 3.6.13), and channel matrices were generated using MATLAB R2021a. The test and training data sets, $\{(y_m, H_m), x_m\}_{m=1}^{train\_data}$ and $\{(y_n, H_n), x_n\}_{m=1}^{test\_data}$, were created randomly based on Equation (2) with different noise levels. The transmitted symbols, x, were selected from modulation schemes such as BPSK, 16-QAM, 64-QAM, and 256-QAM. Various channel models, including the Kronecker channel [28], Rayleigh fading channel, and TDL-A and TDL-E MIMO channels [29], employ different random generators for their channel matrices. The training process uses 5×10^4 samples with an SNR of 25 dB. In deep-unfolded MIMO detection, training with high SNR values is usually preferred to provide a better model tuning. For instance, refs. [28,42] have shown that higher SNR values provide a clearer signal, which allows the model to learn more effectively. Additionally, refs. [43,44] have stated that training at lower SNR levels can even degrade model performance. All trainable parameters were initially set to zero. Subsequently, 5×10^5 samples were used for training with SNR values ranging from 0 to 20 dB, incremented by 2 dB. This wider range of SNR values allows a comprehensive evaluation of the detector's performance. The learning rate was first set to 10^{-3} and was fine-tuned by halving it after each epoch, so that the detection is more robust. The average loss's point of discontinuity determines the stopping criterion. Also, to ensure a fair comparison under identical conditions, the number of layers used in deep-unfolded methods was chosen as the same. The number of layers used for Rayleigh channel was selected as 5, while 15 layers were used in other channel models to adapt to challenging channel conditions. The training method is computationally efficient, taking about two hours on a normal Intel i7-7500U processor, because the model has just three trainable parameters and works well with a small number of layers. It is anticipated that the training time will rise in proportion to larger MIMO systems or models with more trainable parameters.

4. Complexity Analysis

As in many other applications, the use of deep learning approaches in wireless communication systems presents significant challenges in terms of computational complexity. The CG and LCG methods have complexities of $\mathcal{O}\left(\mathrm{K}\left(8N_t^2 + 14N_t + 8\right)\right) + 2KRDiv$ and $\mathcal{O}\left(\mathrm{K}\left(4N_t^2 + 6N_t + 8\right)\right)$, respectively, where K is the number of layers, and $RDiv$ is one real division, assuming that one complex multiplication is four real multiplications and one complex division is four real multiplications plus one real division [28]. Computational complexity of the DU-TCG algorithm, $\mathcal{O}\left(\mathrm{K}\left(8N_t^3 + 12N_t^2 + 6N_t\right)\right)$, has a higher complexity than that of the CG and LCG. The proposed DU-TCG method exhibits a computational complexity of $\mathcal{O}\left(N_t^3\right)$ per iteration, primarily due to the matrix inversion operations involved in each iteration. However, the increased detection complexity improves the detection performance of DU-TCG compared to that of CG and LCG.

5. Simulation Results

Within this section, several MIMO layouts over different channel conditions and modulation orders are discussed to demonstrate the performance of the proposed DU-TCG detection method. In particular, the simulation results highlight the detection performance improvement of the DU-TCG over well-known approaches such as the MMSE, CG, and LCG. Unless otherwise stated, BPSK modulation is used for the sake of simplicity, as it allows a straightforward evaluation of the behavior of the algorithm and a demonstration of its core performance. Bit error rate (BER) and normalized mean square error (NMSE) metrics are employed to illustrate the superiority of the proposed method over the discussed other methods.

Figure 4 compares the BER performance of the proposed DU-TCG method with other methods, namely MMSE, CG, LCG, and ideal ML detector over a Rayleigh fading channel for a BPSK modulated 10 × 10 MIMO system.

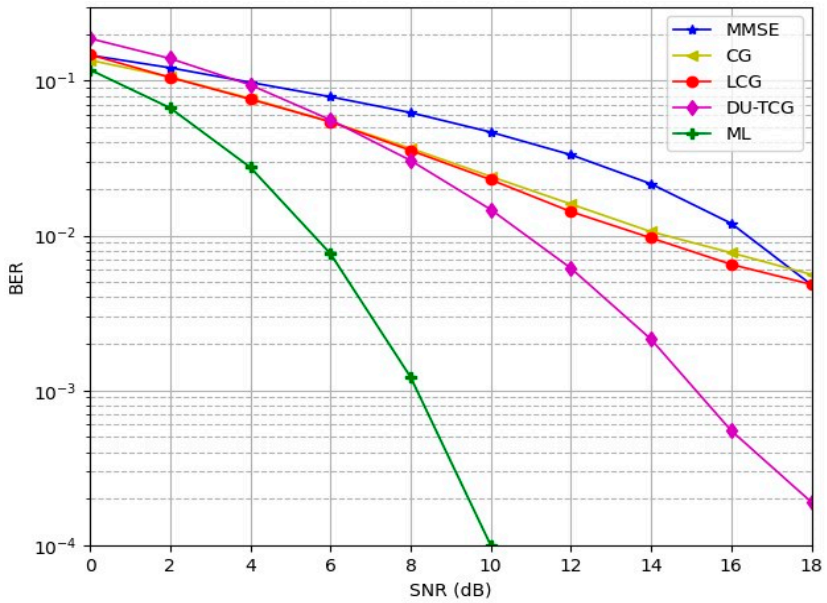

Figure 4. BER performance for a 10 × 10 MIMO system.

Figure 4 shows that although the ML detector offers the lowest bit error rates, the proposed DU-TCG can perform better than other gradient-based techniques such as CG, and LCG and traditional MMSE, which makes it a promising candidate for practical applications. The proposed DU-TCG method has approximately 4 dB SNR gain over CG and LCG methods at BER values of 10^{-3}. In Figure 4, all methods except ML have poor BER performance at low SNR values as expected. Unlike ML, iterative methods are more affected by low SNR values during the initial iterations, leading to suboptimal results.

Figure 5 shows that the proposed DU-TCG method outperforms CG and LCG for different number of layers in the 10 × 10 MIMO layout.

As shown in Figure 5, DU-TCG exhibits better BER performance with both 5 and 15 layers, compared to the CG and LCG methods. Although the performance of the CG and LCG improves when the number of layers is increased by 3 times, DU-TCG still outperforms these methods. However, as the complexity of the system is directly related by the number of layers, the results show that DU-TCG is a better candidate in terms of system cost.

Besides the conventional MIMO scheme and channel models discussed above, in Figure 6, we also compare the performance of DU-TCG with Kronecker, TDL-A, and TDL-E channel models for the 32 × 64 MIMO scheme with BPSK modulation. Since the channel conditions are challenging, 15 layers are used here in the training process.

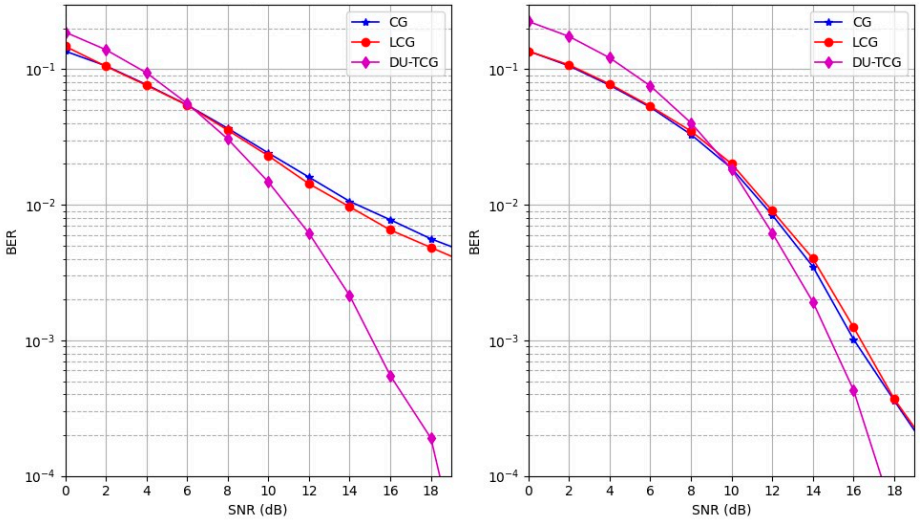

Figure 5. BER performance for different number of layers: (**a**) 5 layers; (**b**) 15 layers.

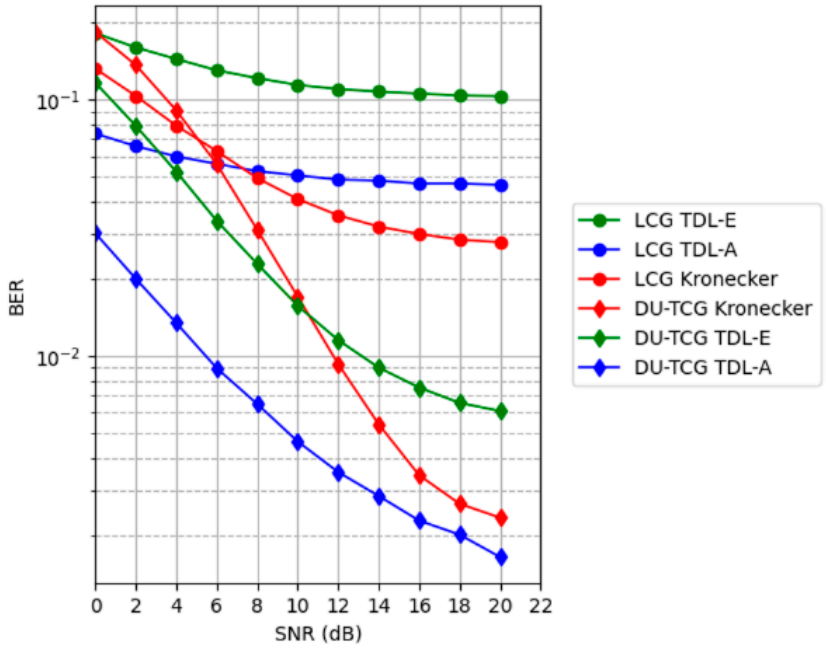

Figure 6. BER performance over different channel models: Kronecker, TDL-A, and TDL-E.

For Kronecker, TDL-A, and TDL-E channel models, both the proposed DU-TCG and existing LCG methods do not have a sufficient detection performance. However, DU-TCG still outperforms the LCG for all three channel models. DU-TCG has employed the benefit of Tikhonov regularization's superiority for the ill-posed problems, and DU-TCG's sophisticated regularization mechanism significantly improves the detection process compared to the LCG method.

Figure 7 presents the BER performance of CG, LCG, and the proposed DU-TCG methods for BPSK and QAM16 modulation types for a 32 $\times$ 128 MIMO layout in TDL-A channel.

It is well known that the performance of iterative detection methods increases significantly when the number of receiving antennas is much greater than the number of transmitting antennas, and this effect is reflected in our simulation results using the DU-TCG, LCG, and CG methods. Figure 7 shows the results of the system with such a MIMO structure. The detection performance of LCG is close to DU-TCG for lower-order modu-

lations in a difficult channel condition such as TDL-A, while DU-TCG outperforms LCG for a higher-order modulation. The figure demonstrates that the BER of DU-TCG is up to 1.2 times better than LCG when using BPSK modulation under TDL-A fading and up to 8.3 times better when using 16-QAM modulation. The superior performance of DU-TCG for difficult conditions such as high-order modulation systems suggests that integrating a regularization term into a deep-unfolded method will help convergence stability and hence improve detection performance.

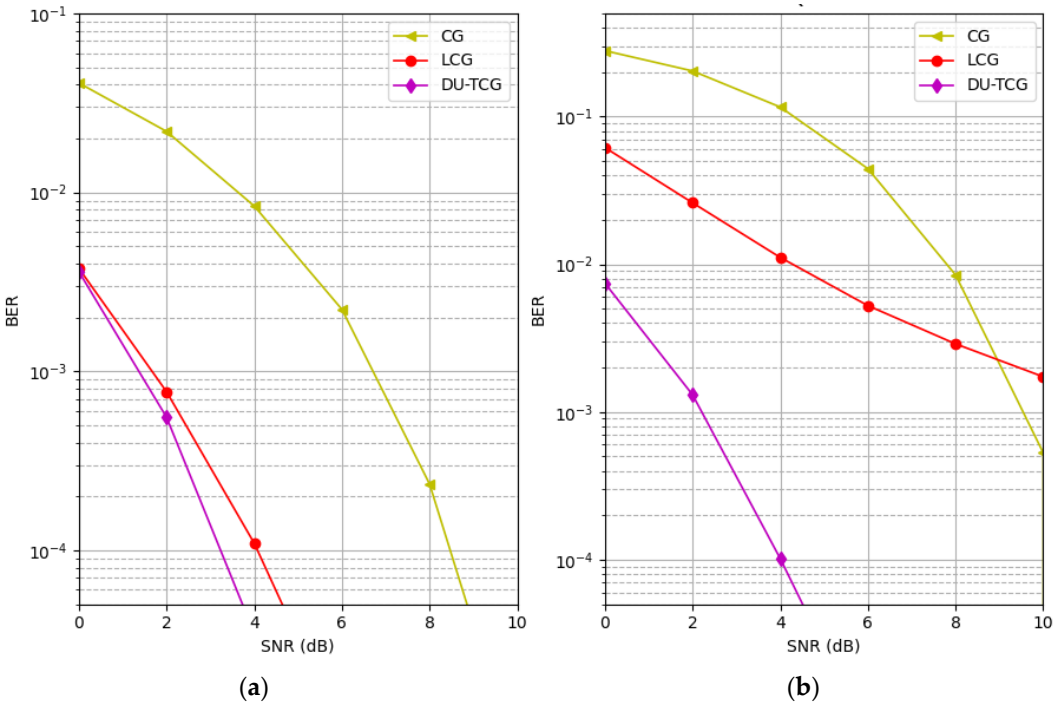

Figure 7. BER performance for different modulation types: (**a**) BPSK; (**b**) 16-QAM.

Additionally, in the simulation results presented above, the BER values are limited to 10^{-4}, as in similar studies utilizing deep unfolding methods in the literature [45,46]. Achieving lower BER values in the simulation environment, particularly for deep-unfolding-based methods, usually requires considerably longer simulation times and higher computational resources, which is impractical for our current computing environment. Nevertheless, we conducted an extensive simulation to achieve lower BER values, and the results are presented in Figure 8, which illustrates the BER performance of the LCG and DU-TCG techniques for 64-QAM and 256-QAM modulations in a 32 $\times$ 64 MIMO system under Rayleigh fading conditions.

The detection performance of higher-order modulation schemes, such as 64-QAM and 256-QAM, as shown in Figure 8, demonstrates that the proposed DU-TCG method retains its superior performance even with more complex modulation formats, which are widely used in sub-6 GHz 5G communication systems [45,47]. In contrast, other deep unfolding techniques, such as in [19,28,29,42], may show a degradation in detection performance as the modulation order increases.

In Figure 9, we compare the detection performance of the proposed DU-TCG method with DetNet over a 32 $\times$ 64 MIMO layout.

For the 32 $\times$ 64 configuration, DU-TCG's BER remains up to nine times lower than DetNet's, demonstrating its strong performance even at higher antenna counts. While the number of trainable parameters in DetNet is eight for each layer, $\theta_{DetNet} = \{W_{1l}, b_{1l}, W_{2l}, b_{2l}, W_{3l}, b_{1l}, \delta_{1l}, \delta_{2l}\}$, it is three for the proposed DU-TCG method, $\theta_{DU-TCG} = \{\alpha_l, \beta_l, L_l\}$. DetNet has a large number of trainable parameters, which can increase computational complexity and resource requirements. These results highlight the

usefulness of DU-TCG, providing better performance without excessive computational cost.

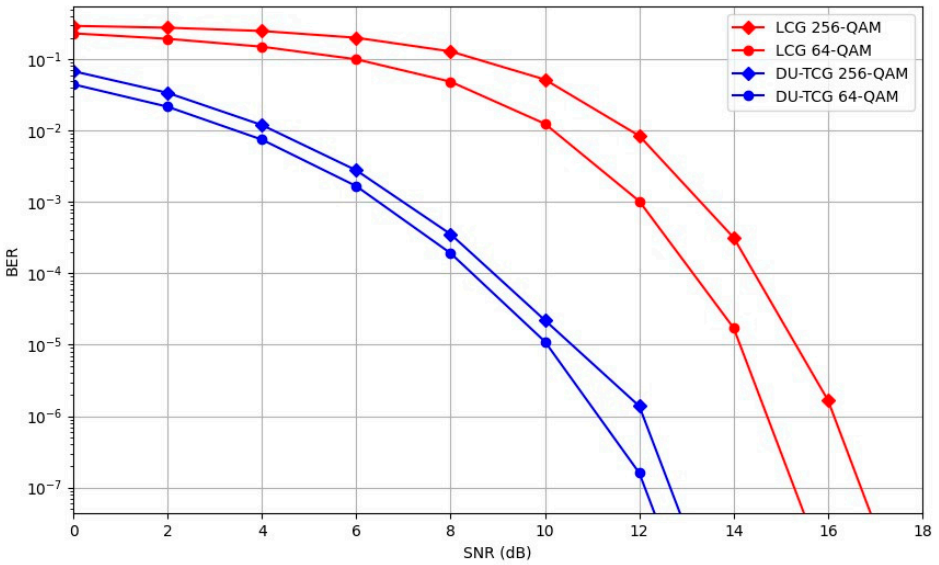

Figure 8. BER performance for different higher-order modulation types.

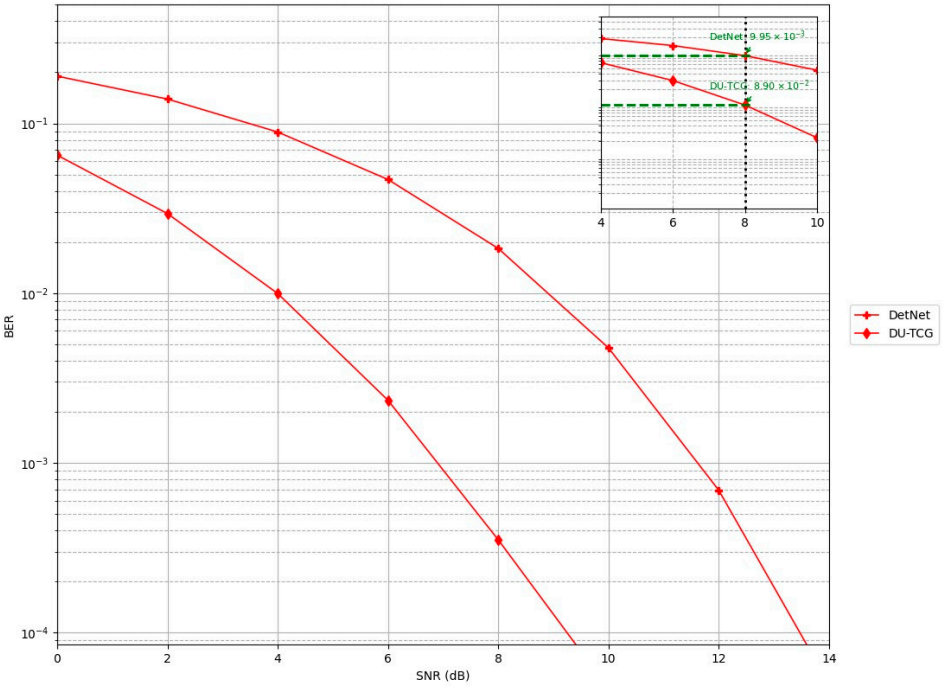

Figure 9. BER performance between DU-TCG and DetNet in a 32 × 64 MIMO system.

The NMSE performance of the proposed DU-TCG and LCG methods, shown in logarithmic scale, for scalar and vector parameterization, is illustrated in Figure 10 for a 32 × 64 MIMO layout.

Figure 10 is one of the most important pieces of evidence showing the superiority of using Tikhonov regularization as DU-TCG has lower NMSE values in both scalar and vector parameterization cases. As the simulation results show, in the case of scalar parameterization, the NMSE decreases with increasing SNR values for both DU-TCG and LCG methods, which indicates that the signal detection performance of the system increases. On the other hand, in the case of vector parameterization, the increase in SNR values for LCG does not affect the NMSE values after a point, while the NMSE value of the proposed

DU-TCG continues to decrease. This shows that the performance of the LCG in complex situations remains constant after a certain level and cannot learn effectively. In addition, the NMSE values for vector parameterization approaches are much smaller than those for scalar parametrization, indicating improved detection performance. Therefore, in case of vector parameterization, the effect of regularization on NMSE performance is higher, resulting in better BER performance.

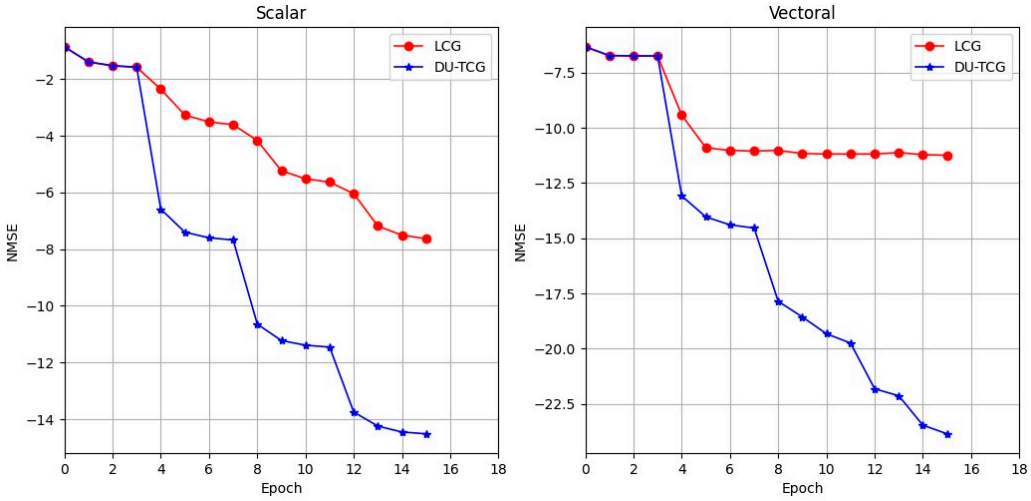

Figure 10. NMSE performance: (**a**) scalar parameterization; (**b**) vector parameterization.

After discussing the effect of scalar or parameterization techniques on the model-driven approaches above, we will show the impact of the same techniques on BER values in Figure 11.

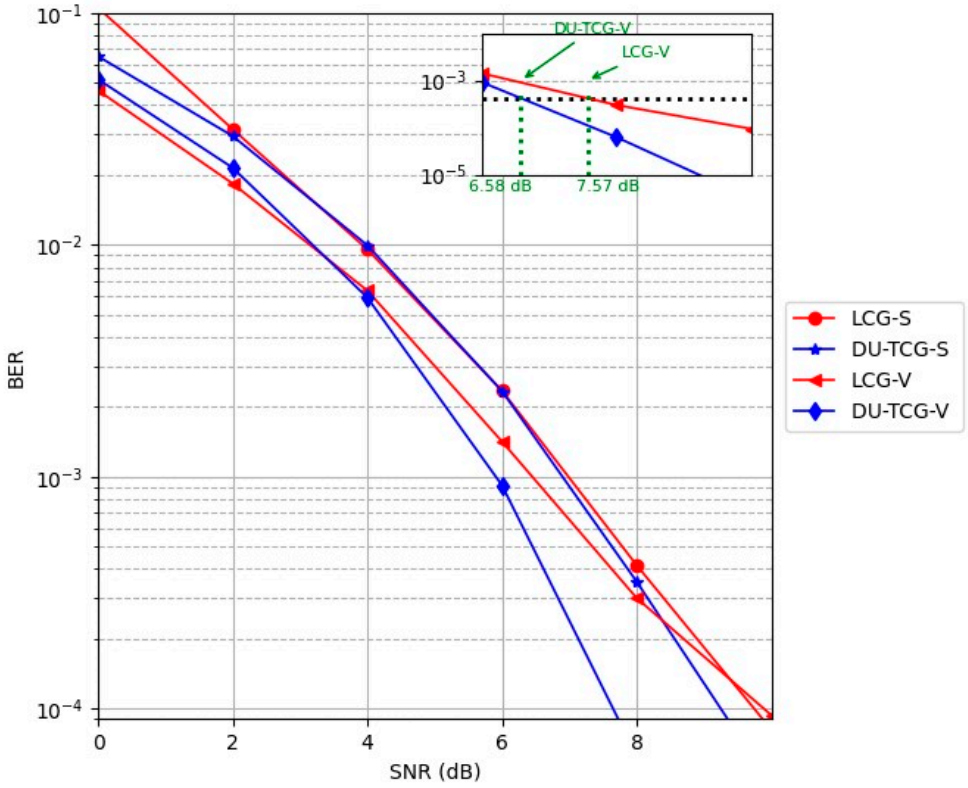

Figure 11. BER performance with scalar and vector parameterization in 32 × 64 Rayleigh channel.

In terms of BER values, in scalar parameterization, DU-TCG and LCG have similar detection performance. Additionally, LCG-V outperforms the DU-TCG-S. However, when we employ the vector parameterization with Tikhonov regularization in DU-TCG, more than 1 dB SNR gain is achieved at the 10^{-3} BER value compared to LCG-V.

6. Discussion

In this study, unlike traditional techniques, a method was proposed that integrates the iterative conjugate gradient method with Tikhonov regularization. The regularization matrix, step size, and search direction coefficients were used as trainable parameters in a deep unfolding approach. Extensive simulation results that demonstrate the superiority of the proposed method over existing methods were presented.

The results of this study clearly demonstrate that the proposed method, DU-TCG, is an improved detection strategy for MIMO systems, outperforming both state-of-the-art deep learning techniques such as DetNet and LCG, as well as traditional approaches such as MMSE and CG. Combining the conjugate gradient method with the Tikhonov regularization approach in a deep learning framework successfully reduces the degrading effects of channel conditions and higher-order modulation.

The scalability of DU-TCG is demonstrated by its consistent performance in both large (32×128) and small (10×10) MIMO systems, providing broad applicability to MIMO configurations of varying size and complexity. Additionally, the better performance of the DU-TCG is also shown under various channel conditions such as Kronecker, TDL-A, and TDL-E channel models, which are widely used in advanced communication systems. This research also reveals that DU-TCG surpasses DetNet in a 32×64 MIMO layout when system complexity increases. Simulation results also show that the proposed method provides up to 4 dB SNR gain compared to CG with considerably less iterations. In addition, DU-TCG stays ahead of CG and LCG as the number of layers increase, even surpassing LCG's high-layer performance with fewer layers. Ultimately, as compared to the vector parameterization of LCG, the NMSE and BER performances of DU-TCG are superior. For both scalar and vector parameterization, DU-TCG decreased the NMSE values.

This work combines Tikhonov regularization and the CG technique with deep unfolding, which significantly improves MIMO signal detection under various MIMO layouts, modulation orders, and channel conditions. The proposed method inserts a regularization term to the system, which improves the stability and generality of the solution. In particular, when there are noisy or imperfect data conditions, this regularization helps the CG converge more consistently. Thus, the approach efficiently handles the complexity present in high-dimensional MIMO systems while iteratively improving its signal estimates.

Consequently, this study highlights the advantages of the proposed DU-TCG method for MIMO detection over different scenarios. As a future work, combining Tikhonov regularization with other iterative detection algorithms for MIMO systems may be considered. Additionally, reducing computational complexity of model-driven approaches is also another challenge that needs to be discussed in the area of deep-unfolded algorithms.

Author Contributions: Conceptualization, S.N.K.; methodology, S.N.K. and A.K.; software, S.N.K.; validation, S.N.K. and A.K.; formal analysis, S.N.K. and A.K.; investigation, S.N.K. and A.K.; resources, S.N.K. and A.K.; data curation, S.N.K.; writing—original draft preparation, S.N.K. and A.K.; writing—review and editing, A.K.; visualization, S.N.K.; supervision, A.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data set available on request from the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Yang, S.; Hanzo, L. Fifty years of MIMO detection: The road to large-scale MIMOs. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 1941–1988. [CrossRef]
2. Albreem, M.A.; Juntti, M.; Shahabuddin, S. Massive MIMO detection techniques: A survey. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 3109–3132. [CrossRef]
3. Wu, M.; Yin, B.; Vosoughi, A.; Studer, C.; Cavallaro, J.R.; Dick, C. Approximate matrix inversion for high-throughput data detection in the large-scale MIMO uplink. In Proceedings of the 2013 IEEE International Symposium on Circuits and Systems (ISCAS), Beijing, China, 19–23 May 2013; pp. 2155–2158.
4. Yin, B.; Wu, M.; Cavallaro, J.R.; Studer, C. Conjugate gradient-based soft-output detection and precoding in massive MIMO systems. In Proceedings of the 2014 IEEE Global Communications Conference, Austin, TX, USA, 8–12 December 2014; pp. 3696–3701.
5. Hu, Y.; Wang, Z.; Gaol, X.; Ning, J. Low-complexity signal detection using CG method for uplink large-scale MIMO systems. In Proceedings of the 2014 IEEE International Conference on Communication Systems, Macau, China, 19–21 November 2014; pp. 477–481.
6. Dai, L.; Gao, X.; Su, X.; Han, S.; Wang, Z. Low-complexity softoutput signal detection based on Gauss-Seidel method for uplinkmultiuser large-scale MIMO systems. *IEEE Trans. Veh. Technol.* **2015**, *64*, 4839–4845. [CrossRef]
7. Adeva, E.P.; Augustin, T.R.; Fettweis, G.P. Optimizing a pipelined MIMO sphere detector for energy efficiency. In Proceedings of the IEEE International Conference on Wireless Communications & Signal Processing (WCSP), Nanjing, China, 15–17 October 2015; pp. 1–6.
8. Fukuda, W.; Abiko, T.; Nishimura, T.; Ohgane, T.; Ogawa, Y.; Ohwatari, Y.; Kishiyama, Y. Low-Complexity Detection Based on Belief Propagation in a Massive MIMO System. In Proceedings of the IEEE 77th Vehicular Technology Conference (VTC Spring), Dresden, Germany, 2–5 June 2013; pp. 1–5.
9. Markus, M.; Ketonen, J. MIMO detector algorithms and their implementations for LTE/LTE-A. In Proceedings of the GIGA Seminar, Hamburg, Germany, 1 November 2010.
10. Hu, Q.; Gao, F.; Zhang, H.; Li, G.Y.; Xu, Z. Understanding deep MIMO detection. *IEEE Trans. Wirel. Commun.* **2023**, *22*, 9626–9639. [CrossRef]
11. He, H.; Wen, C.K.; Jin, S.; Li, G.Y. A model-driven deep learning network for MIMO detection. In Proceedings of the IEEE Global Conference on Signal and Information Processing (GlobalSIP), Anaheim, CA, USA, 26–29 November 2018; pp. 584–588.
12. Un, M.W.; Shao, M.; Ma, W.K.; Ching, P.C. Deep MIMO detection using ADMM unfolding. In Proceedings of the IEEE Data Science Workshop (DSW), Minneapolis, MN, USA, 2–5 June 2019; pp. 333–337.
13. Shlezinger, N.; Eldar, Y.C.; Farsad, N.; Goldsmith, A.J. Viterbinet: Symbol detection using a deep learning based viterbi algorithm. In Proceedings of the IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Cannes, France, 2–5 July 2019; pp. 1–5.
14. Ge, Y.; Tan, X.; Ji, Z.; Zhang, Z.; You, X.; Zhang, C. Improving approximate expectation propagation massive MIMO detector with deep learning. *IEEE Wirel. Commun. Lett.* **2021**, *10*, 2145–2149. [CrossRef]
15. O'Shea, T.J.; Erpek, T.; Clancy, T.C. Deep learning-based MIMO communications. *arXiv* **2017**, arXiv:1707.07980.
16. Baek, M.-S.; Kwak, S.; Jung, J.-Y.; Kim, H.M.; Choi, D.-J. Implementation methodologies of deep learning-based signal detection for conventional MIMO transmitters. *IEEE Trans. Broadcast.* **2019**, *65*, 636–642. [CrossRef]
17. Samuel, N.; Diskin, T.; Wiesel, A. Deep MIMO detection. In Proceedings of the IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Sapporo, Japan, 3–6 July 2017; pp. 1–5.
18. Chen, Q.; Zhang, S.; Xu, S.; Cao, S. Efficient MIMO detection with imperfect channel knowledge-a deep learning approach. In Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), Marrakesh, Morocco, 15–18 April 2019; pp. 1–6.
19. Faghani, T.; Shojaeifard, A.; Wong, K.K.; Aghvami, A.H. Deep learning-based decision region for MIMO detection. In Proceedings of the IEEE 30th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Istanbul, Turkey, 8–11 September 2019; pp. 1–5.
20. Hershey, J.R.; Roux, J.L.; Weninger, F. Deep unfolding: Model-based inspiration of novel deep architectures. *arXiv* **2014**, arXiv:1409.2574.
21. Balatsoukas-Stimming, A.; Studer, C. Deep unfolding for communications systems: A survey and some new directions. In Proceedings of the IEEE International Workshop on Signal Processing Systems (SiPS), Nanjing, China, 20–23 October 2019; pp. 266–271.
22. Yu, Y.; Ying, J.; Zhang, S.; Wang, J.; Guo, L.; Shang, J.; Wang, P. A deep learning approach based on Richardson and Gauss–Seidel for massive MIMO detection. *Trans. Emerg. Telecommun. Technol.* **2024**, *35*, e4947. [CrossRef]
23. Sirois, S.; Ahmed Ouameur, M.; Massicotte, D. Deep Unfolded Extended Conjugate Gradient Method for Massive MIMO Processing with Application to Reciprocity Calibration. *J. Signal Process. Syst.* **2021**, *93*, 965–975. [CrossRef]
24. Qin, Z.; Ye, H.; Li, G.Y.; Juang, B.H.F. Deep learning in physical layer communications. *IEEE Wirel. Commun.* **2019**, *26*, 93–99. [CrossRef]
25. Zappone, A.; Di Renzo, M.; Debbah, M. Wireless networks design in the era of deep learning: Model-based, AI-based, or both? *IEEE Trans. Commun.* **2019**, *67*, 7331–7376. [CrossRef]

26. Björnson, E.; Giselsson, P. Two applications of deep learning in the physical layer of communication systems. *arXiv* **2020**, arXiv:2001.03350.
27. Takabe, S.; Imanishi, M.; Wadayama, T.; Hayakawa, R.; Hayashi, K. Trainable projected gradient detector for massive overloaded MIMO channels: Data-driven tuning approach. *IEEE Access* **2019**, *7*, 93326–93338. [CrossRef]
28. Wei, Y.; Zhao, M.M.; Hong, M.; Zhao, M.J.; Lei, M. Learned conjugate gradient descent network for massive MIMO detection. *IEEE Trans. Signal Process.* **2020**, *68*, 6336–6349. [CrossRef]
29. Ahmed Ouameur, M.; Massicotte, D. Early results on deep unfolded conjugate gradient-based large-scale MIMO detection. *IET Commun.* **2021**, *15*, 435–444. [CrossRef]
30. Liu, X.; Li, Y. Deep MIMO detection based on belief propagation. In Proceedings of the IEEE Information Theory Workshop (ITW), Guangzhou, China, 25–29 November 2018; pp. 1–5.
31. Fan, X. The Constrained Total Least Squares with Regularization and Its Use in Ill-Conditioned Signal Restoration. Ph.D. Dissertation, Mississippi State University, Mississippi State, MS, USA, 1992.
32. Loyka, S.L. Channel capacity of MIMO architecture using the exponential correlation matrix. *IEEE Commun. Lett.* **2001**, *5*, 369–371. [CrossRef]
33. 3GPP TR 38.901: Technical Specification Group Radio Access Network; Study on Channel Model for Frequencies from 0.5 to 100 GHz. 2018. Available online: https://www.3gpp.org/ftp/Specs/archive/38_series/ (accessed on 20 May 2024).
34. van Etten, W. Maximum likelihood receiver for multiple channel transmission systems. *IEEE Trans. Commun.* **1976**, *24*, 276–283. [CrossRef]
35. Yu, Y.; Ying, J.; Wang, P.; Guo, L. A data-driven deep learning network for massive MIMO detection with high-order QAM. *J. Commun. Netw.* **2023**, *25*, 50–60. [CrossRef]
36. Albreem, M.A.; Alhabbash, A.H.; Shahabuddin, S.; Juntti, M. Deep learning for massive MIMO uplink detectors. *IEEE Commun. Surv. Tutor.* **2021**, *24*, 741–766. [CrossRef]
37. Hestenes, M.R.; Stiefel, E. Methods of Conjugate Gradients for Solving Linear Systems. *J. Res. Natl. Bur. Stand.* **1952**, *49*, 409–435. [CrossRef]
38. Buccini, A.; Donatelli, M.; Reichel, L. Iterated Tikhonov regularization with a general penalty term. *Numer. Linear Algebra Appl.* **2017**, *24*, e2089. [CrossRef]
39. Bianchi, D.; Buccini, A.; Donatelli, M.; Serra-Capizzano, S. Iterated fractional Tikhonov regularization. *Inverse Probl.* **2015**, *31*, 055005. [CrossRef]
40. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.37.
41. Bock, S.; Goppold, J.; Weiß, M. An improvement of the convergence proof of the ADAM-Optimizer. *arXiv* **2018**, arXiv:1804.10587.
42. He, H.; Wen, C.K.; Jin, S.; Li, G. Model-driven deep learning for joint MIMO channel estimation and signal detection. *arXiv* **2019**, arXiv:1907.09439.
43. Zheng, G.; Zhang, Q.; Li, S. Failure diagnosis of linear arrays based on deep residual shrinkage network. *Microw. Opt. Technol. Lett.* **2022**, *64*, 1627–1633. [CrossRef]
44. Xu, J.; Chen, E.; Chen, V. Energy-efficient data symbol detection via boosted learning for multi-actuator data storage systems. In Proceedings of the 2021 IEEE International Symposium on Circuits and Systems (ISCAS), Daegu, Republic of Korea, 22–28 May 2021. [CrossRef]
45. Samuel, N.; Diskin, T.; Wiesel, A. Learning to detect. *IEEE Trans. Signal Process.* **2019**, *67*, 2554–2564. [CrossRef]
46. Tan, X.; Xu, W.; Sun, K.; Xu, Y.; Be'ery, Y.; You, X.; Zhang, C. Improving massive mimo message passing detectors with deep neural network. *IEEE Trans. Veh. Technol.* **2020**, *69*, 1267–1280. [CrossRef]
47. Berra, S.; Chakraborty, S.; Dinis, R.; Shahabuddin, S. Deep unfolding of Chebyshev accelerated iterative method for massive MIMO detection. *IEEE Access* **2023**, *11*, 52555–52569. [CrossRef]